Model Based Systems Engineering for Simulator Sustainment

George Ayers, 1st Lt Chris Reed

AFLCMC/WNS

United States Air Force
Wright-Patterson AFB, OH

george.ayers@us.af.mil,
christopher.reed.62@us.af.mil

Austin Abraham, J. Aaron Doak, Marilyn Evans,
Carlton Jackson, Buford McCusker,
Patrick Reynolds, James Sermersheim

AFLCMC/WNS

Huntington Ingalls Industries,
Technical Solutions Division
Dayton OH

austin.abraham.4.ctr@us.af.mil,
joseph.doak.2.ctr@us.af.mil,
marilyn.evans.ctr@us.af.mil,
carlton.jackson.3.ctr@us.af.mil,
buford.mccusker.1.ctr@us.af.mil,
patrick.reynolds.5.ctr@us.af.mil,
james.sermersheim.ctr@us.af.mil

ABSTRACT

Model Based Systems Engineering (MBSE) is a powerful and effective tool for design purposes, but it also has utility in managing existing systems. The Air Force Life Cycle Management Center (AFLCMC), Agile Combat Support Directorate, Simulators Program Office (WNS) is responsible for thousands of simulators, of varying degrees of realism, distributed around the world. Additionally, every simulator helps to sharpen the warfighter's bite in a non-destructive environment so that they will be ready for any situation in a real aircraft. Many of these are large and complex Full Flight Simulators (FFS) that are expensive and have several functions separated into multiple, physically distinct subsystems. Each aircraft platform has many demanding FFS technical baselines to manage. It is vitally important to maintain awareness and management of all of these technical baselines. From the enterprise perspective, sustaining this system-of-systems is difficult with documents alone and will get more difficult.

In response to this need to manage the simulator portfolio more efficiently, WNS has decided to use MBSE to guide the creation of the Operational Training Infrastructure (OTI) Enterprise System Model (ESM). This effort has not been without obstacles. MBSE is best known for its usefulness in system development; as a result, there has been some discussion of the usefulness of MBSE for a system already in sustainment. The transition from the current Document Based Systems Engineering (DBSE) to MBSE has been met with resistance rooted in difficulty in changing established processes, unfamiliarity with the tools, and risk aversion. Finally, the experiences of failed efforts in the past to adopt digital engineering solutions have left some within the organization hesitant to embrace MBSE. This paper discusses how our team plans to use MBSE for simulator sustainment and how the technical and organizational challenges to adopting MBSE in AFLCMC/WNS are being addressed.

ABOUT THE AUTHORS

George Ayers and 1st Lt Chris Reed are engineers working for the Air Force Materiel Command, Life Cycle Management Center, Agile Combat Support Directorate, Simulators Program Office (AFLCMC/WNS). **Mr. Ayers** holds degrees from Arizona State University, Texas A&M University, and the Air Command and Staff College. He comes to AFLCMC/WNS after 13 years in the Air Force test community where he was a senior rocket sled test project manager. **1st Lt Reed** graduated from Embry-Riddle Aeronautical University with a B.S. in Aerospace Engineering in December 2017 and has been working in AFLCMC/WNS since January 2018. He was selected to work on a $900M project where he quickly learned MBSE to understand and effectively communicate the intricacy of new systems. In recognition, Chris was named the subject matter expert in MBSE for AFLCMC/WNS. The OTI ESM was started by Lt Reed to launch Digital Engineering (DE) in Air Force training systems acquisition.

Austin Abraham, J. Aaron Doak, Marilyn Evans, Carlton Jackson, Buford "Skip" McCusker, Patrick Reynolds, and James Sermersheim are engineers in the employ of Huntington Ingalls Industries, Technical Solutions Division at AFLCMC/WNS. They may be contacted through the primary author. **Mr. Abraham** is an alumnus of the University of Colorado at Boulder's Smead Department of Aerospace Engineering. He is a member of INCOSE whose professional interests lie in aeronautical design and systems engineering. **Mr. Doak** has a Master's degree in Industrial and Human Factors engineering from Wright State University. He has seven years of systems engineering experience within the automotive industry, has attained a green belt in Six Sigma, and is a member of INCOSE. **Ms. Evans** earned her Master's Degree in Business Administration (MBA) from Wright State University. **Mr. Jackson** has worked closely with the implementation of MBSE in the simulators division of the US Air Force dating back to 2018. Has completed Delligatti Associates - SysML Training Course, a member of INCOSE, and holds Lean Six Sigma Green Belt. **Mr. McCusker** has over 50 years of systems engineering and systems engineering management experience. **Mr. Reynolds** is an alumnus of the University of Dayton's Department of Computer Engineering. He is a member of INCOSE. **Mr. Sermersheim** has 30 years of experience in systems engineering, design, modeling, and development.

Model Based Systems Engineering for Simulator Sustainment

George Ayers, 1st Lt Chris Reed

AFLCMC/WNS

United States Air Force
Wright-Patterson AFB, OH

george.ayers@us.af.mil,
christopher.reed.62@us.af.mil

Austin Abraham, J. Aaron Doak, Marilyn Evans,
Carlton Jackson, Buford McCusker,
Patrick Reynolds, James Sermersheim

AFLCMC/WNS

Huntington Ingalls Industries,
Technical Solutions Division
Dayton OH

austin.abraham.4.ctr@us.af.mil,
joseph.doak.2.ctr@us.af.mil,
marilyn.evans.ctr@us.af.mil,
carlton.jackson.3.ctr@us.af.mil,
buford.mccusker.1.ctr@us.af.mil,
patrick.reynolds.5.ctr@us.af.mil,
james.sermersheim.ctr@us.af.mil

INTRODUCTION

The Air Force Materiel Command, Life Cycle Management Center, Agile Combat Support Directorate, Simulators Program Office (AFLCMC/WNS, or "WNS") is responsible for the acquisition, operation, and sustainment of thousands of simulators distributed around the world. Training systems, from the early Link Trainer in the 1930s to the most modern full flight simulators (FFS) used for fifth generation fighters, have generally been developed to fit the needs of specific platforms and programs. Connecting these different systems together through a wide area network is a relatively recent development meant to execute simultaneous mission training with different aircraft, and eventually support live, virtual, and constructive joint exercises. Managing this Operational Training Infrastructure (OTI) is complicated by the variety of devices, their capabilities, and the computers used to control them. The difficulty in managing this disparate enterprise is magnified by the balance of the demand for increased capabilities, the drive to reduce the amount of resources used to accomplish the training mission, and an ever-expanding number of training devices. The key to resolving these issues is making life cycle management processes more efficient and flexible. One means to that end is the use of digital engineering.

The National Defense Authorization Act of 2017 (United States, 2016) included language that required the use of Modular Open System Architecture in the development of major systems. In 2018, the Deputy Assistant Secretary of Defense for Systems Engineering issued a strategy which articulated five broad goals: to fully use models and modeling to inform decision-making, to use these models as an authoritative source of information for all stakeholders, to improve engineering practices across the lifecycle through advanced technologies enabled by digital engineering, to facilitate stakeholder collaboration and activity through improved infrastructure and tools, and to transform the organizational culture to implement digital engineering practices throughout system lifecycles (*Digital Engineering Strategy*, 2018).

Model Based Systems Engineering (MBSE) is best known for its usefulness in new system development and its role in sustainment has been overlooked by many, even among those that recognize it as an improved systems engineering approach. Indeed, there has been some discussion of how useful MBSE could be for a system in the sustainment phase of its life cycle. However, the capability of MBSE to manage system and enterprise documentation, to trace requirements from functions to components, and to collate information across the enterprise, makes it as useful a tool for systems sustainment as it is for development.

The current, electronic Document Based Systems Engineering (DBSE) has many pitfalls, including poor access to current information. While Contract Data Requirements Lists (CDRLs) mandate the format of the deliverables, different Integrated Product Team (IPT) members create working documents in a manner that suits their circumstances. This variation in documentation makes it difficult for the uninitiated to find information. This contributes to the relative difficulty in addressing inquiries on an enterprise level. Many of the training systems that WNS is responsible for were designed many years ago, and the chain of engineering change documentation may not be complete for a variety of reasons. Consequently, many IPTs lack full knowledge of their training system structure. MBSE can addresses these issues with a model by enforcing a standardized documentation structure for all programs, providing traceability between requirements and systems, as well as instantaneous propagation of updated information. Many CDRLs can be satisfied through an appropriately detailed model. Other benefits of using MBSE in training systems sustainment includes the opportunity to identify elements of a simulator common architecture and the ability to answer questions about the enterprise.

In complex systems, everything needs to be organized and separated into levels of abstraction. Maintaining traceability among these levels with DBSE is extremely difficult, while MBSE utilizes a relational database linking model elements, system components, and documents. Communication in a DBSE environment often involves several document revisions, sometimes with duplicate or contradicting information, with varying completion dates. This can become difficult to control, leading team members to act on outdated or incorrect information. An MBSE environment uses a single model that all team members reference and allows everyone to be confident that they are accessing the most updated and accurate information.

The use of MBSE, through an appropriately-constructed Operational Training Infrastructure (OTI) Enterprise System Model (ESM) that describes the architecture of the enterprise, enhances the understanding of training structure footprint (AFLCMC/EZS, 2019). Using the OTI ESM as the official source for trusted, up-to-date information will facilitate decision-making for all stakeholders. Sub-models may be used to efficiently communicate requirements and contractual deliverables. These sub-models will aid cyber resilience by providing information for planning systems audits and identifying the systems that need to be patched or upgraded – supporting greater Risk Management Framework (RMF) compliance on an enterprise level. The same model can be used to facilitate the identification of a simulator common architecture and identifying opportunities to more efficiently manage the sustainment of the simulator enterprise. The model can be used to answer queries on an enterprise level in a few hours that currently might require weeks, providing opportunities to minimize lifecycle costs across the enterprise.

By using MBSE in its processes, WNS will be satisfying directives to use digital engineering in a meaningful way by addressing each of the goals outlined in the National Defense Authorization Act of 2017 (United States, 2016) and the Digital Engineering Strategy (*Digital Engineering Strategy*, 2018). With this in mind, the WNS MBSE project was initiated in 2019.

THE WNS MBSE PROJECT

WNS leadership saw in 2017 the value of utilizing digital engineering concepts in a new program that was in development. This program team began to research the best way forward in the digital engineering world and determined that it was to use MBSE with Object Management Group (OMG) Systems Modeling Language (SysML) as the language. MBSE was selected because the program required a Modular Open Systems Approach and utilizes common architectures to the configuration item level. MBSE excels at visually representing complex systems in 2-dimensional models derived from an interconnected database. Selecting SysML for the language proved beneficial because it is already a well-documented industry standard and WNS contractors are either already using it, or willing to start. MBSE quickly expanded in interest to be included throughout the WNS enterprise when the benefits of MBSE for the new program became apparent.

As a result, the WNS MBSE Implementation Project was formally initiated in 2019 using the program team's initial research and selected language. The project began with the task of developing a high-level model of the OTI ESM to be used as a means of understanding the scope of the enterprise and as a basis for a more detailed model. The benefits of using a nested model to answer queries on an enterprise level were obvious. Past experience in determining if a readily available document was the most recent (essential for configuration management) highlighted the need for an authoritative data source. Access to data of sufficient depth and breadth is essential to making good decisions that affect the entire enterprise, as well as supporting efforts to determine a common simulator architecture. Appropriate constituent models have been used to trace requirements to components and can be used in contracting actions to communicate between government and industry. It is anticipated that these constituent models will be used by integrated product team (IPT) members to collaboratively manage training systems programs more efficiently.

Successful execution of the WNS MBSE project will serve as an excellent starting point for future implementation of digital engineering practices.

When the MBSE implementation project is complete, WNS engineers and IPT support contractors will have access to the OTI ESM from their workstations. The model will include system information down to the component level for all training devices. More detailed constituent models could be built to suit the needs of individual IPTs. Access to the model will be appropriate to the role of each user. The project is developing the OTI ESM with five project level lines of effort (LOE): 1) providing a collaborative modeling environment, 2) developing a style guide to standardize constituent models, 3) developing a training program for WNS personnel, 4) modeling each program within the enterprise, and 5) updating organizational systems engineering plans and processes. Much progress has been seen along these lines of effort since project initiation.

LOE 1: COLLABORATIVE MODELING ENVIRONMENT

A good modeling environment requires suitable hardware to run the computationally intensive modeling software. A challenge to using digital engineering within the Air Force involves the hardware, software, and networks required to create modeling environments. The Air Force and the DoD at large have strict security requirements and procedures for every piece needed in the creation of a modeling environment. Due to the time required for each approval, it is usually best to use existing resources to complete objectives quickly. Fortunately, there are resources available in the Air Force which allow this to happen. One of these resources is the Air Force Systems Engineering Resource Center (AFSERC). AFSERC utilizes the network-based modeling environment outlined above to create a virtual desktop on a standard government computer via a third party program and remote government server. The AFSERC environment clearly demonstrated the positives and negatives to the network-based environment and gave clarity for future efforts to the MBSE team. The modeling environment was provided for very low cost and was expanded for operational use rapidly. However, the large latency involved with launching the modeling tool and transferring data over the network could not be resolved. The latency was caused by a combination of the hosting method and the firewalls on the networks between the computer and the remote server. The AFSERC environment allowed the team to successfully complete a small project, but a much better environment is needed for future collaboration on the large OTI ESM.

The current modeling environment implemented by the team uses standalone computers and manual data transfer. This method is robust because the team is not reliant on a network connection to run the tool. However, this method inhibits collaboration because manual data transfer is slow and proper version control is critical to prevent work loss. This environment is a stop-gap that allows the team to continue modeling while a better collaboration environment is researched, approved, and created. Fortunately, there are organizations in the Air Force, like the Digital Engineering Enterprise Office (DEEO), that were recently created with the objective of creating better digital environments. Within a few months of its creation, the DEEO was able to aid the WNS MBSE team. One recommended environment was the Hanscom mil Cloud (HmC). The MBSE team was able to gain access to a trial run in the HmC network-based modeling environment and tested latency for both tool launch and data transfer. Since the HmC environment runs a virtual desktop in a web browser over the network to a cloud-based host with fewer layers of obstacles, the latency was demonstrated to be negligible for both tool launch and data transfer. Using the direction from the DEEO, the team decided to procure and utilize this environment for future modeling efforts. This environment allows the team to make effective use of the simulator model library in order to create common models across the enterprise. The library had to be updated manually with each version release of the style guide with the stand-alone computers currently used. The collaborative advantage of the model saves a significant amount of rework. This new environment will facilitate creation of each program model within the OTI ESM by teams of government and support contractor engineers. The benefits of a collaborative environment will increase as each program adds more parts to the library.

Looking into the far future, the team will have more options for better modeling environments as Air Force digital initiatives reach maturity and as the Simulators Program Office gains capability outlined in new contracts. The key to take advantage of the future environment is to remain flexible while delivering the best simulator capabilities possible to sharpen the warfighter's bite.

LOE 2: STYLE GUIDE DEVELOPMENT

During early MBSE implementation, in order to develop a reference architecture model in support of request for proposals (RFP), a modeling Style Guide had to be established. This is due to each modeler having their own preferred modeling style and further complicated by the flexibility of SysML. The initial draft of the Style Guide was captured in a Word document, but soon transitioned to a SysML model. This became the template for the initial build of training systems models. The Style Guide model template contains a preliminary structure for each program model, a simulator library, and a set of style guidelines. These contents standardize the diagrams in each model, save time through requirement and component reuse, and defines a common structure for all models.

The Style Guide currently contains the Code of Federal Regulations requirements for flight simulators and Aerial Refueling Airplane Simulator Qualifications which are governing documents for simulator level classifications; additional requirement documents will be included as the Style Guide matures. The Style Guide model template also contains tip sheets, port types, and a component library. This component library contains items such as components for hardware, software, visual systems, computing, and communications which are available for reuse throughout the enterprise.

The MBSE team focused the Style Guide development on achieving the primary goal of modeling training systems that are in sustainment, so it is oriented towards capturing the current “as-is” configurations. Future work is likely to require further development of the style guide to accommodate an end-to-end systems engineering process and other useful content to support IPTs.

LOE 3: TRAINING PROGRAM

The Style Guide has also been used to help focus the content of the courses developed in LOE 3 and introduce students to the preferred modeling conventions. The course provides initial exposure to the concept of MBSE modeling for simulator sustainment. Studying *SysML Distilled* (Delligatti, 2014) and *A Practical Guide to SysML* (Friedenthal, Moore, & Steiner, 2015) proved to be an excellent starting point for the MBSE team. Because the current workforce within WNS had few people who were trained to model, it would be difficult to build and use models to begin the change to MBSE. The MBSE team began working with the Air Force Institute of Technology (AFIT) in summer 2019 to set up formalized MBSE classes for WNS. The purpose of these classes was primarily a way to get the engineers of the different simulator IPTs to learn the basics of modeling. It was expected at the time that these engineers would at least need to be able to use the model to obtain information from it and possibly model themselves to make small modifications when needed. On top of the four day class, a two hour session was added specifically for those in more managerial roles who would have less direct hands-on with the models but might need to understand its outputs or usages in the future. The first class taught SysML and pure Object Oriented Systems Engineering Methodology (OOSEM) since the WNS modeling style guide was still in early development. OOSEM is intended for modeling systems through the entirety of their lifecycle, from beginning to end, so its material for modeling systems that are already in the sustainment stage of their lifecycle is lacking. However, the first round of classes still proved useful because it gave the MBSE modeling team a good foundation on which to begin building a custom version of OOSEM for modeling training systems in sustainment.

Although the preliminary methods of training WNS employees worked to some degree, those methods could be improved. It was evident that people were losing their modeling skills because the tools were not available for use on their workstations. The MBSE team has planned efforts to combat this skill loss. One effort is to institute regular consulting time to allow for WNS employees to discuss anything related to MBSE, taking advantage of the smaller conversation audience to allow discussion better focused on individual issues. Another idea involved building an Air Force MBSE community of practice (COP). This COP would be hosted on a website and serve as a knowledge hub for anything pertaining to modeling. An additional idea for future implementation is to make modeling tutorial videos. This would supplement training classes and individual tutoring, and would allow engineers to see the process as many times as they need to. The MBSE training program will continue to be adapted to suit the needs of WNS engineers. As more people use systems models and try learning MBSE, the training processes will only continue to improve.

LOE 4: WNS PROGRAM MODELING

The WNS MBSE team is continuing to build on the foundation of its past efforts by developing best modeling practices for future WNS program models. The availability of information and personnel have constrained modeling progress; however, many innovative modeling practices have been developed that have led to an elegant enterprise modeling solution. The initial model describes a topology that helps to improve comprehension of the general system. As Figure 1 shows, each type of training system is broken down into its subsystems; more complicated subsystems can be broken down further, as needed. The consistent breakdown is embedded in a model architecture that is easy to navigate and understand. This comprehensive view assists current IPT members to better understand simulator structures, and will reduce the time spent orienting new team members. Table 1 presents a summary of a few examples of how MBSE can be used by IPT members and other organizations in training systems acquisition and sustainment processes. While not all of the envisioned capabilities are in wide use, most have either been planned or have been demonstrated on a small scale.

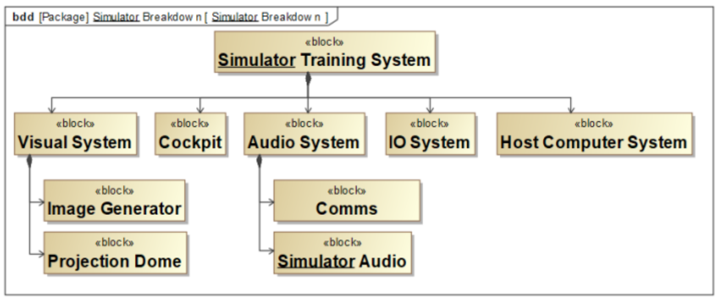

Figure 1: Subsystem Breakdown Example

Table 1: Summary of OTI ESM Capability Development for IPT Members

Discipline/Sector	RFP	ATO	VSP	ECPs	PCA	POE	Program Understanding	Requirements Tracking
Program Management	3			1	1	1	1	2
Engineering	3	3	2	1	1	2	3	3
Configuration Management	1			1	1		1	
Logistics	2			1	1		1	
Financial Management	1			1		1	1	
Contracting	1			1		1	1	
Cybersecurity	2	3	2	1			2	
Other US Government	3	3					3	
Industry	3						2	

RFP – Request for Proposal, ATO – Authority to Operate, VSP – Vulnerability Scanning Program,
ECP – Engineering Change Proposal, PCA – Physical Configuration Audit, POE – Program Office Estimate;
(1) Planned Capabilities; (2) Demonstrated Capabilities; (3) Capabilities in use

ENHANCING ENGINEERING DRAWINGS

With traditional engineering drawings, tracking connections between components and across pages can be difficult. Internal Block Diagrams (IBDs) can dive a level deeper than what is shown in Figure 1. In SysML, an IBD is a lower level diagram used to show the components within a block, as well as their connections and properties. This means that for the structure of the training system models, IBDs are essentially the engineering drawings for the system. Transferring the drawings into the models allows for easier navigation between drawings, and data flow information between components can also be added. This improved navigation and easily accessible information in these diagrams creates an environment that increases the ease of detailed system comprehension. Subsequently, tasks like evaluation of RFPs and Engineering Change Proposals (ECP), troubleshooting, and audits will be less daunting, leading to faster results. Each subsystem shown in Figure 1 has at least one corresponding IBD with component level information; Figure 2 is a generic example of the component level information found in the Host Computer System Block from Figure 1.

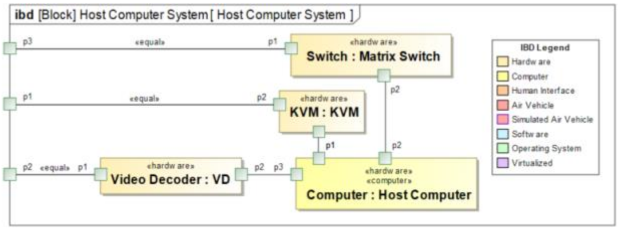

Figure 2: IBD Example

Using an MBSE systems model will reduce the possibility for confusion due to different team members accessing data from multiple documents. This is possible because the MBSE model is a database containing source data, and all of the model elements can be easily and efficiently searched (as opposed to needing to know where a multitude of files are located). This also will allow for quick and easy table and report creations, which saves the IPT analysis time. Additionally, this means that engineering data for every part in a system, such as form, fit, function, and obsolescence, can be linked to the technical baseline, thus making important part information readily available. The models provide increased understanding that enable IPTs to make better informed decisions. Informed decisions lead to reduced administrative burden.

IMPROVING PROGRAM COMPREHENSION

When developing an RFP, the IPT program offices must transfer program information to potential bidders as efficiently as possible. The current method of doing this is through a bidder's library, which contains the program's tech data package where hundreds of supplementary documents are stored. Even with all of the information in the bidders' library, full understanding of the program is still difficult to achieve. Utilizing MBSE models for contractor review facilitates understanding more efficiently than documents alone. While these documents relay the relevant information about the program, the reader is often compelled to cross-reference several documents in order to understand the specific aspects of the program related to the RFP requirements. Furthermore, retaining information as text can be difficult. This is often aided with appropriate tables and figures; but these depend on the time and effort available from the IPT. Alternatively, if an MBSE model exists for the program, understanding of the program will be greatly enhanced. Rather than having to decipher descriptive paragraphs about the program, the IPT can refer bidders to the model provided with the bidder's library. Figure 3 illustrates how a model diagram can clearly and succinctly contain a large amount of information, providing a logical structure to follow, thus making the program breakdown easy to understand. This structure also acts as a single container for relevant program information, making it readily available and easy to find. When the government is creating the RFP, having an MBSE model will allow them a quick reference to confirm the accuracy of documents. If desired, tables and reports can be easily produced from the model to aid in the RFP as well.

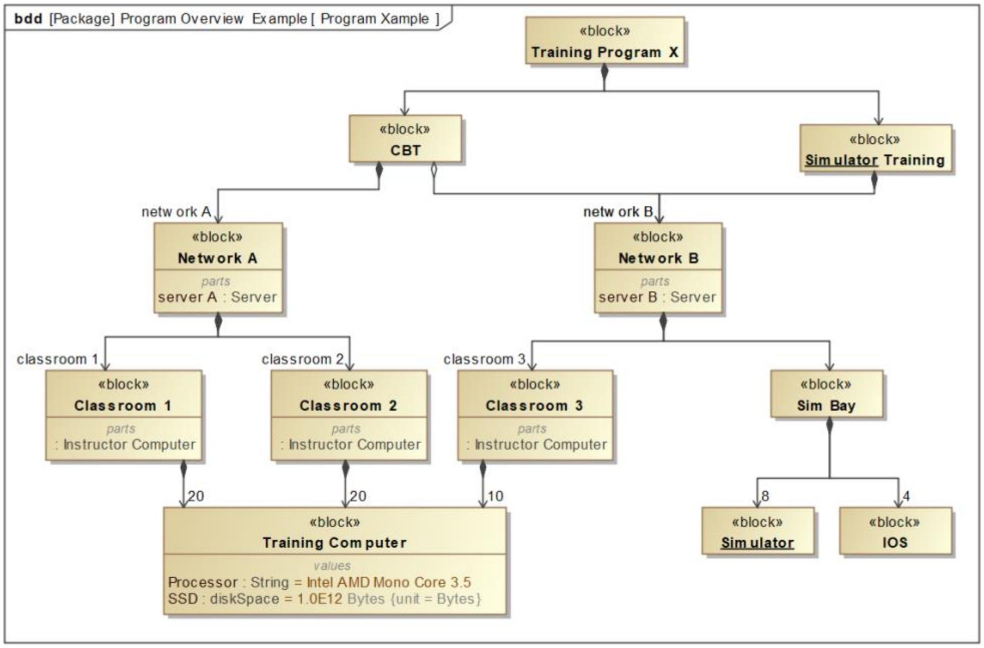

Figure 3: Overview Example

FACILITATING CYBERSECURITY EFFORTS

Cybersecurity processes can also benefit from MBSE. Cybersecurity is an ever-increasing concern for much of the DoD, and WNS is no exception to this. One process that has been focused on by the MBSE team is that of a simulator system obtaining its ATO. The MBSE team is working to make each program model output a nearly-complete ATO documentation set using data from the model. For now, the MBSE team is focusing on getting the model to output three main artifacts for obtaining ATO: authorization boundary diagrams, hardware/software lists, and implemented security controls.

Another challenge for obtaining an ATO is having an accurate network topology. Network topologies are used as a reference for cybersecurity personnel attempting to locate vulnerabilities within a training system, to obtain an ATO, or to run and assess the results of Vulnerability Scanning Program (VSP) scans on the system. Once fully completed, MBSE models will have important network information embedded within them such as port types, Virtual Local Area Network (VLAN) information, and partial IP addresses for all components. This information is necessary for the VSP scan process since it allows the cybersecurity personnel to plan the scan before it is executed. Incorrect topology information can result in either partial or failed scans and flawed cybersecurity verification.

These examples illustrate some of the benefits MBSE can bring to WNS cybersecurity personnel. It would allow them to complete their daily work and other recurring deliverables in a much more efficient way, while also opening up future opportunities to continue the improvement of cybersecurity activities across all Air Force training systems. For example, when enough information is in the OTI ESM, the entire enterprise of simulators can be quickly searched for defective network equipment. If a switch was found to be insecure then instead of individually contacting each of the hundreds of Air Force installations, one can simply search the OTI ESM for occurrences of this faulty equipment to address the issue. On top of this, the requirements for that class of equipment can be quickly traced, making selection of a suitable replacement easier.

RESOLVING MODELING ISSUES

Many issues were encountered when the team started to build the OTI ESM. Some of the more pertinent challenges which were overcome are discussed below.

Logical v Physical Structure

In OOSEM, logical diagrams are created in early system development, primarily in order to convey concept and function, while physical diagrams are created to represent actual subsystems and components deployed in the field. For the WNS MBSE team's implementation of MBSE for sustainment, the physical systems already exist and therefore the logical diagrams take on a modified role. Rather than convey the general function of each subsystem, the logical diagrams are used to show the baseline variations of each training device. These logical baseline diagrams are still developed to the component level, as the physical diagrams are, but detailed information is omitted. For example, where there may be a generic host computer component depicted in the logical baseline diagram, the physical diagram will have the same appearance, but the host computer will be identified and appropriate hardware and software specifications will be included. While the logical baseline diagram could house more detail and be configured as a physical diagram, the models were set up like this for three reasons: there was a need for a clear distinction between the models of the actual simulators and their baselines, the actual simulators belonging to the same baseline have minor differences that will need to be captured, and omitting data from the baseline diagrams allows for a more rapid development of the models.

Model Element Properties

In order to capitalize on the search capabilities in the MBSE tool, model elements need to have identifiers assigned to them so that they can be included in any table or report generated using the proper criteria. One of these identifiers is the stereotype, which serves as an extension to Unified Modeling Language elements. In addition to the standard SysML stereotypes, custom stereotypes such as "hardware," "software," and "computer" were created to better classify the various components that make up a training device. Stereotypes can be given their own properties in the form of tag definitions, which function similarly to block properties. For instance, the "software" stereotype may be given a "version" tag, which could then be defined on any model element with the "software" stereotype applied. Using stereotypes and tags makes made it easier to interpret the model. As the modeling process developed, the visual aspect of the models needed improvement to support the expected variation of user expertise. One potential solution to this problem came in the form of the MBSE tool's "Symbol Properties" window, wherein any element's visual depiction on the model could be altered in several ways (e.g. fill color, font size/color, shown values). However, this alone would be insufficient, as changing the default symbol properties for each stereotype causes other graphical issues down the road. The next suggested solution was to use legends. Legends allow the user to visually group diagram symbols with custom styles, similar to the previous solution's custom visual properties. Legend items are applied directly to model elements, are not tied to specific stereotypes, and automatically apply the defined style to the diagram symbol. While they lack the individual properties and functionality of custom stereotypes, legend items serve the purpose of easy identification for any model element, including relationships. Beyond this, legends can be added to different diagram types such as tables, and will modify specific table elements according to the legend item's custom visual style. Using both stereotypes and legend items together meets the needs of the modeling team, and they have been incorporated into the style guide and modeling process to enhance understanding of the OTI ESM.

Diagram Boundaries Connection

One modeling error that was caught and corrected early involved boundary connections. Consider the situation shown in Figure 4, where a component from one subsystem is connected to that of another subsystem. In order to model this, three separate diagrams are being used: one that shows component A connecting to the boundary of subsystem A, one that shows component B connecting to the boundary of subsystem B, and one diagram showing how the boundary ports of subsystem A connects to those of subsystem B. The problem with this is that the model would indicate that there are three wires and four ports connecting component A and component B. In reality, there is a single connecting cable running from the port in component A to the port in component B (Figure 5).

To resolve this issue, binding connectors are utilized in most instances where a connector meets a boundary port. A binding connector symbolizes that the ports on either side of the connector are the same port, represented in two places. This is the same as if the edge of the component block was shared with the subsystem boundary and only a single port was shown. The binding connector allows for the component block to be moved away from the subsystem boundary, permitting a more efficiently modeled diagram. Figure 6 shows the usage of binding connectors for this situation.

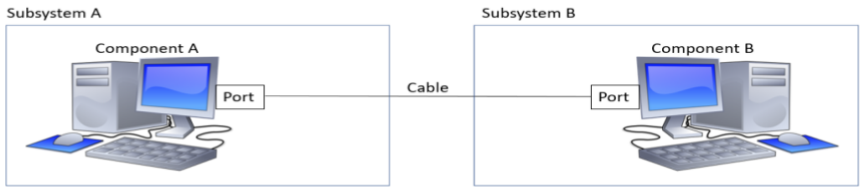

Figure 4: Actual situation

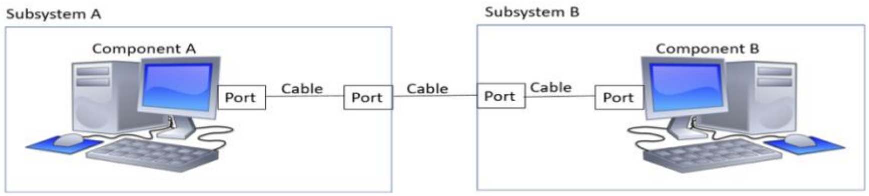

Figure 5: Incorrect Representation of Actual Situation

Reference Properties

A reference property is a method for alluding to a block owned by another system. The modeling team had hoped to use these in order to simplify the interconnect diagrams. Rather than having these connectors track across three diagrams, as shown in Figure 6, a reference property of the component in subsystem B could be placed in subsystem A with all connections going directly to the reference property. However, issues with the tool used by the MBSE team inhibit association of part properties and reference properties in a straightforward manner. Due to these issues, it was decided that reference properties could not be used in the main structures of our models; however, they could still be used in other packages and views.

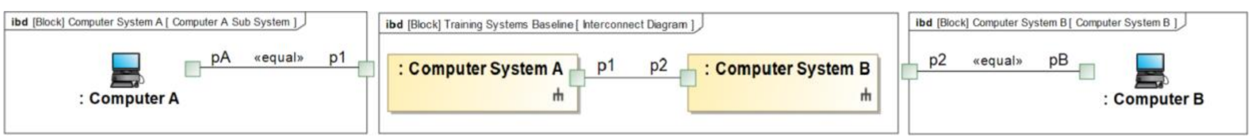

Figure 6: Binding Connector Example

Virtualized Components

The training systems modeling process involves capturing virtualized components in addition to the real components. Among these components are virtual machines (VMs) and VLANs, which proved difficult to adequately represent in the models. Since virtual machines emulate computer systems without additional hardware, they must be captured in such a way that they can be distinguished from tangible model elements and tied to the real hardware that they are running on. In order to accomplish this, the team first developed a better understanding of how virtualized components are currently represented and explored the options for representing them in the MBSE tool. The team determined that VMs are best represented as the computer system they emulate with the “software” and “virtualized” stereotypes applied, with a meaningful name that indicates that it is a virtual machine. Directed aggregation relationships are used between the VM and the hardware hosting it, as well as between the VM and any additional programs running on it. In this way, there is clear traceability of software components existing both on physical and emulated computer systems.

Determining how to model VLANs was not so simple. Early in the modeling process, available documentation depicted VLANs through color-coded connections between training device components. In an effort to preserve this presentation, the modeling team deemed it best to use the previously-described legend method to depict VLANs within the system. By creating a “VLAN Legend” and applying custom styles to its items, connections between model elements were color-coded similarly to how they are depicted in the source documentation. Initial reviews suggested that this method would be sufficient for capturing VLANs within the models; however, other issues were encountered.

Applying a specific VLAN configuration to the connections within a particular system (e.g. a logical “ethernet switch” block within a computational system) defined that configuration for all iterations of that system in the model. This is not an accurate description of reality. As more of the training program is modeled, there are more deviations from this baseline configuration. The team determined that VLANs would be modeled as virtualized components of the owning system and would serve as an intermediate step in describing the model's physical connections. Within a VLAN block, virtual switches receive connections from nested ports on the block's boundary, then connect outside the block to nested ports on the boundary of the other end of the physical connection according to their particular configuration (see Figure 7). This process allows the modeled physical connections between networked components to be preserved, while all VLAN configuration information is contained within a block unique to that system's network. Further implementation will determine how successful this method is in representing VLANs within device networks.

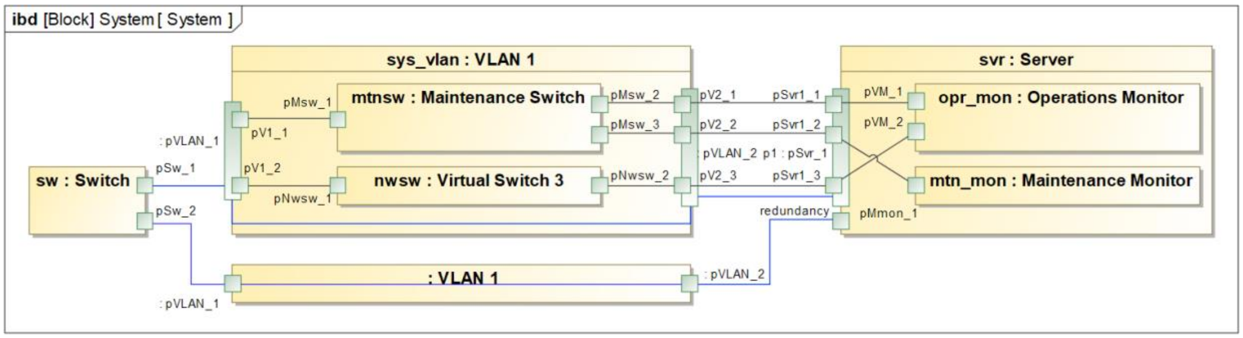

Figure 7: VLAN Example

LOE 5: SYSTEMS ENGINEERING PROCESSES

While the technical efforts required to implement MBSE for sustainment are significant, other efforts are necessary to assure a successful transition to the new processes. Addressing technical obstacles is only part of a successful transition – overcoming organizational and personnel issues is essential (c.f. Eggers and Bellman, 2015, Kane et al., 2015, Gupta, 2018). Developing a change management strategy that addresses organizational and personnel issues, and dedicating resources to it, will provide assurance of a successful transition and an earlier realization of the benefit to the organization.

Organizational issues can largely be addressed by developing a plan for the transition from DBSE to MBSE processes, communicating the reasons and vision for making this change, and supporting the transition in meaningful way (Kotter, 2006). Efforts are underway to develop procedures and practices to use the OTI ESM to support the work of the IPTs. MBSE project team members and IPT members have developed report templates and processes to automate some of the tasks required to support simulator programs. As these processes mature, IPT members will be able to see how using the OTI ESM will improve their work. However, WNS and IPT systems engineering plans will need to be modified to either permit or require these model-based processes. Contract language should be reviewed and adopted to permit MBSE as a means of communication between industry and government. Using MBSE should be part of the way that WNS operates.

The people that make up the organization must be willing to support the new processes (Baggio, Digentiki, & Varma, 2019). Resistance to change among the people expected to execute the transition can be addressed by engaging with the people – identifying specific benefits, addressing concerns, providing training and tools, and recognizing successes. (Hiatt, 2006). Training classes have gone a long way to making the proposed processes easier to use for IPT members. These classes have exposed students to both the practical uses of MBSE as well as the potential benefits.

Giving employees opportunities to use constituent models of the OTI ESM and demonstrating their usefulness is critical to generating the desire to use it. Early MBSE adopters within WNS include the B-1 Training Systems, KC-10 Aircrew Training Systems (ATS), and cybersecurity personnel assigned to IPTs. The willingness of the lead engineer for the KC-10 ATS allowed the modeling team to use the KC-10 Training System model as their first exposure to a simulator structure and gave a way for the team to learn how to properly model a training system. With this cooperation, the team demonstrated the capabilities of MBSE to other simulator programs. The KC-10 training system engineer continues to be supportive of the efforts and has gradually introduced more detail into the KC-10 model.

WNS cybersecurity personnel saw how the system model could greatly improve the efficiency of obtaining an Authority to Operate (ATO) for their systems. Obtaining an ATO is reliant upon independent artifacts that did not easily communicate interrelationships and interdependencies within an integrated system. Obtaining an ATO is a high priority for all simulator programs and this early success provided an opportunity to demonstrate a needed process improvement. In turn, the interest shown by several programs served to reinforce the utility of MBSE to WNS leadership.

Building on these successes, the B-52 training systems program was selected as a flagship model to be used to demonstrate the utility of MBSE to all disciplines within other IPTs. This is a large, complex program with a long history and significant remaining life. The completion of this flagship model and subsequent satisfaction of the program team will be a huge step in the direction of instilling a desire to use MBSE in other WNS simulator programs. This model will be used to further develop and demonstrate MBSE based processes for IPT members (for example, creating ATO exhibits, satisfying CDRLs, or cross-checking requirements and RFP documentation).

CONCLUSIONS

The pressures to do more with fewer resources will continue to shape how the sustainment of training systems is conducted. Adopting the proven practices of digital engineering in DoD processes will enable WNS to more efficiently manage the systems already within the OTI Enterprise. The WNS MBSE project is ambitious but it needs to be done to better support the warfighter through sustaining the legacy systems and products they use to execute their mission (Waugh, 2020). It is not enough to tell people how MBSE can improve processes, it must be demonstrated to improve sustainment efficiency and lower costs in order to the open the door to accepting the process change and changing the organizational culture regarding digital engineering.

The continued development of the flagship model will offer many opportunities to refine the modeling process and develop new processes that more efficiently manage the vast OTI enterprise. Several advances have been made while developing the OTI ESM. Efficiently producing artifacts for the ATO process and more efficiently communicating requirements between training system programs and their industrial partners are just two examples. The full potential of MBSE to dramatically improve processes can be demonstrated through the use of a well-developed flagship model, reducing the barriers to accepting MBSE within WNS.

There are several plans for furthering the WNS modeling efforts past the limits of this paper. Some obvious items include refining the style guide and modeling the remaining programs within the Air Force. Eventually a substantial amount of the modeling work will likely be transitioned to contractors that manage the simulators themselves in order to eliminate any data access issues. Working with these contractors will be key to the future success of the WNS OTI ESM, in order to maintain the desired style and organization of the models. Depending on the individual simulator programs' needs, additional technical diagrams will need to be developed, such as activity and parametric diagrams. WNS modeler's functions in the program will evolve into a role of evaluating and validating contractor models, as well as integrating them into a cohesive OTI ESM. The WNS modelers will continue to provide additional aid to WNS IPTs to maximize their ability to use the new models and increase daily efficiencies. Working with other Air Force and DoD modeling groups to unify modeling standards will also be high priority moving forward, as well as furthering the development of the SysML modeling practices within the modeling community as a whole.

ACKNOWLEDGMENTS

The authors would like to thank Mr. Tony DalSasso, Mr. Mike Baker, Mr. Mark Adducchio, Mr. Thomas Bridgman and Mr. Heath Morton for taking time to review this paper.

WORKS CITED

AFLCMC/EZS. (2019). *Managing the Technical Baseline.* Standard Process, Wright-Patterson AFB.

Baggio, A., Digentiki, E., and Varma, R. (2019, October 7). *Organizations do not change. People change!* Retrieved from https://www.mckinsey.com/business-functions/organization/our-insights/the-organization-blog/organizations-do-not-change-people-change

Delligatti, L. (2014). *SysML Distilled: A Brief Guide to the Systems Modeling Language.* Addison-Wesley.

(2018). *Digital Engineering Strategy.* Office of the Deputy Assistant Secretary of Defense for Systems Engineering.

Eggers, W. D., and Bellman, J. (2015). *The Journey to Government's Digital Transformation*. Deloitte University Press.

Friedenthal, S., Moore, A., and Steiner, R. (2015). *A Practical Guide to SysML: The Systems Modeling Language*. Morgan Kaufmann Object Management Group/Elsevier.

Hiatt, J. M. (2006). *ADKAR: A Model for Change in Business, Government and Our Community.* Loveland, Colorado: Prosci Learning Center Publications.

Kane, G. C., Palmer, D., Phillips, A. N., Kiron, D., and Buckley, N. (2015). *Strategy, Not Technology, Drives Digital Transformation: Becoming a Digitally Mature Enterprise*. MIT Sloan Management Review and Deloitte University Press.

Kotter, J. (2006). *Leading Change*. Harvard Business School Press.

United States. (2016). *National Defense Authorization Act for Fiscal Year 2017*. U.S. Government Publishing Office.

Waugh, P. C. (2020, January 6). *The Agile Combat Support Directorate We Need*. Wright-Patterson AFB, OH.