TES-SAVi AWESUM™ Model-Based Systems Engineering (MBSE) for FACE™ Applications

Stephen M. Simi
Sean P. Mulholland
William G. Tanner

Tucson Embedded Systems
TES-SAVi AWESUM™
5620 N. Kolb Road, Suite 160
Tucson, AZ 85750
520-575-7283, 520.575-5563 (fax)

***Abstract*—This paper presents an industry-unique approach and improved model-based engineering practices for software-reliant aircraft systems aligned with the Future Airborne Capability Environment (FACE™) open system technical reference architecture.**

A lifecycle toolset is described which fuses systems and software modeling and simulation (M&S) capabilities, modular open system architectures (MOSA), and device and sensor integration techniques into a single package to enable rapid design, development, verification, certification, and deployment of interoperable, platform portable, embedded mission-critical safety-critical avionic systems.

TES' MBSE lifecycle toolset, AWESUM™, supports system architecture and virtual integration (SAVI) of cyber-physical system (CPS) assets, and evaluation of these platform assets through simulated mission rehearsals in a network-centric collaborative virtual environment. The result is improved systems product.

The toolset has been demonstrated to several aviation-focused communities including the American Helicopter Society International (AHS), the Association of Old Crows (AOC), the FACE Consortium, and the US Army's Aviation and Missile Research Development and Engineering Center (AMRDEC), with favorable response.

TES demonstrated a virtual simulation of realistically modeled and simulated multiple dissimilar aircraft platform systems conducting network centric operations (NCO). With the goal of improving flight-safety and enhancing the situational awareness (SA) of the battlespace, virtual missions were executed showing the interoperations of cyber-physical system (CPS) assets and capabilities from reusable FACE-aligned applications. The simulated battlespace was composed of device-configured simulated flight models of both manned and unmanned rotorcraft platforms operating within a virtual environment of real-world terrain data. The environment included simulated threats and obstacles used to stimulate platform devices that report detections and produce alerts for own-ship flight mishap avoidance. These detections were then packaged into communications messages (VMF) and shared off-platform via NCO to the common operating picture (COP) to enhance the SA operations of the battlespace flight group.

After a decade of development and use supporting US Army military common software programs for the PM-AME and AMRDEC, TES is now making its industry-unique approach and model-based airworthy systems/software lifecycle product line available to industry. The MBSE lifecycle toolset product suite, referred to as AWESUM™ for AirWorthy Engineering Systems Unified Modeling, is a comprehensive model-based environment aligned to DO-178C, DO-331, AC 20-148, AR 70-62, and FACE™.

The AWESUM™ tool suite is currently supporting the US Army's Joint Common Architecture project, a model-based avionics architecture for the Joint Multi-Role family of future vertical lift platforms, and two other FACE-aligned US Army Aviation programs within the military communications and aircraft survivability domains.

This is TES-SAVi's first publication and presentation to the IEEE. While we have applied our capabilities to the Military Aviation domain, these capabilities can be effectively applied to Aerospace and other domains.

978-1-4799-1622-1/14/$31.00 ©2014 IEEE

TABLE OF CONTENTS

1. INTRODUCTION

Let's assume you can tackle the biggest, meanest football player on the field 30% faster than anyone else. You can do this because you have studied him well. You know every aspect of your opponent and how he interacts alongside of his teammates. You have a model of him from the top-down and the bottom-up, you know his every move, his every intention to move, maybe as much or possibly better than he does. Moreover, if he travels to another ball field under different environmental conditions, you can reuse your knowledge of him and tackle him there 30% faster than you did before, collectively now 60% faster than anyone else.

These are the potential benefits that the TES-SAVi model-based system engineering toolset may have on the aviation community. TES' SAVi MBSE may well revolutionize the manner in which we develop and field improved product. In the systems engineering domain, where a measure of success is 10% process improvement for government programs, a tool suite capable of achieving these benefits would be more than groundbreaking, it would be AWESUM™.

Issues Impeding the Delivery of Aviation Systems

There are two primary issues impeding progress toward delivering military aviation systems solutions that have increased safety, system capacity, and efficiency: namely, system complexity and non-reusable stovepipe acquisition practices.

The first issue, systems complexity, is that space and military aviation flight-critical, mission-critical, and safety-critical systems are among the most complex cyber-physical systems (CPS) in industry. These systems have intensive development and verification requirements to ensure that missions are achieved without mishaps, perform as intended in austere conditions, and lives will not be risked.

The second issue impeding progress toward open reusable software systems is the myopic manner in which we procure military programs and Government's acquisition processes for these programs. That is that military programs and their funding lines favor stovepipe development practices focused on mission-specific requirements. Unfortunately stove-piped systems by design do not favor efficiencies of software reuse across the fleet, nor do they realize the potential benefits of common capability integration efforts. As a result, capability integration is cumbersome, costly, and those stove-piped capabilities do not interoperate well across the fleet within the same battlespace. Compounding these issues, current government development and business practices have limits, and government procurement policies and practices are unclear for reusable open systems software and capabilities.

Two Promising Movements

There are two promising movements that when combined can be used to improve product development and delivery. One is the Open Group's Future Aviation Capability Environment (FACE™). The second is adoption and acceptance of model-based tools to support lifecycle development and verification activities of reusable, platform-portable, interoperable aviation systems.

FACE™ [1] is a Consortium of Government and Industry focused on developing three fundamental aspects that may make modular open systems architecture (MOSA) components a reality for the aviation community.

- FACE™ is technical standard for developing software applications and capabilities that can be hosted on well-defined open system architectures. The FACE™ standard enables portability and reuse to enhance the interoperability of system of systems (SoS) in a network-centric operational battlespace. A tenant is that FACE™ conformant software written for one aviation platform can be ported and reused on a different platform. The FACE™ standard defines a model-based approach to software component interface definition. This approach is based on Zermelo-Fraenkel set theory and is developed with sufficient specificity to enable verification (to the standard "FACE conformance") and integration of components.

- FACE™ provides guidelines for the acquisition of FACE™ products. Significant effort and progress has been made to develop contract guidelines to assist the procurement of FACE™ applications.

- FACE™ will provide a registry where Government military platform integrators "shop" for FACE™ conformant products.

Collectively, these three FACE™ fundamental aspects establish a solid foundation for military aviation community to produce and procure products that are based on open systems architecture that can be ported for reuse across the military aviation community.

What is missing and is needed in the aviation community is an aviation-specific airworthy tool suite that can leverage these FACE™ standards and processes. Tooling that can develop open reusable FACE™ avionics applications. Tooling that can assist capability integrations and be used to optimize the airworthy release processes required by today's military aircraft. Tooling that can leverage model-based development practices, and leverage guidelines for certifications and reuse of certified airborne software systems, specifically the Army's AR 70-62 [2] and the FAA's AC 20-115C [3] which recognizes DO-178C [4], DO-331 [5], and reuse guidelines AC 20-148 [6] and DO-297 [7].

The second promising movement that will result in the improvement of product development and delivery is the adoption of MBSE techniques used to address development and verification activities of complex embedded space and military aviation systems. The adoption of MBSE techniques and MBSE tools very well may assist filling the capability gap of aviation-specific airworthy tools that will result in the development and fielding of improved capabilities to the aviation community and warfighter faster.

The authors propose that if one can use sufficient tools and develop to common reusable open standards that address the most-complex and most-stringent capability integration issue – the integration of military capabilities on dissimilar platforms – that solving the less stringent integration issues will follow using the same process methods with less stringent requirements applied. One result is forward movement toward the goal of less costly, less cumbersome integration efforts, and increased capability interoperability across a fleet of joint forces including airborne, sea, and ground assets.

The Adoption of MBSE Development and Verification

Twenty years has passed since the publication of the RTCA's DO-178B. Since then advances and experiences have been gained in model-based developments and verifications and their applications on airborne systems. In 2013, the FAA's AC 20-115C adopted the use of MBSE technologies as an acceptable means for showing compliance with applicable airworthiness regulations. This circular recognizes DO-178C and DO-331 model-based development and verification supplements. As such MBSE tooling is now recognized as acceptable support for the development and verification activities for airborne systems and equipment certifications.

This formal recognition may well be a "game changer" for the use of MBSE technologies within aviation. The TES-SAVi AWESUM™ for AirWorthy Engineering Systems Unified Modeling is one such game-changing toolset.

A MBSE Product Announcement

During the Open Group's FACE Member's meeting[1] in Tucson Arizona, TES announced the formation of a new division, TES-SAVi, and announced that the AWESUM™ product line suite of capabilities will be made available to industry.

This paper presents how the TES-SAVi AWESUM™ product-line suite is employing improved model-based engineering practices for software-reliant aircraft systems. It is aligned with the Future Airborne Capability Environment (FACE™) open system technical reference architecture and can be used to design and field capabilities for the future fleet[2].

In previous TES publications, TES described its patented [8] bottom-up top-down approach to abstracting application interfaces for opens systems architectures [9], using our approach for efficient automated verification activities [10], and applications of these process capabilities to the virtual simulation of a network centric interoperable battlespace [11].

This paper focuses on the model-based system engineering activities, and the benefits of such approaches. It focuses on the development of the Specification and Design models and how TES' MBSE toolset is used to develop aviation applications aligned with FACE™ that support the current and next generation of future vertical joint multi-role aircraft.

[1] FACE-Tucson - FACE™ Consortium Member Meeting, September 2013

[2] Architecture developments of the Army's Future Vertical Lift (FVL) and the NAVAIR's F/A-XX 6th generation fighter platforms

2. CURRENT STATE OF MBSE TOOLS

The current state of the art in MBSE is centered on several modeling technologies including Systems Modeling Language™ (SysML™), Unified Modeling Language (UML), MARTE, SAE Architecture Analysis and Design Language (AADL), and Domain Specific Languages (DSL) like the Future Airborne Capability Environment (FACE™) Data Model. Each of these technologies has benefits when applied to systems development in differing ways.

INCOSE's System Engineering Vision 2020 [12], published in 2007, states well the current ad hoc state of these systems (*abridged* for readability).

Currently, the MBSE process and methods are generally practiced in an ad hoc manner and not integrated into the overall system engineering processes. The MBSE tools support various modeling techniques, such as functional analysis and object-oriented analysis, but only partially support model and data interchange. The resulting lack of tool interoperability has been a significant inhibitor to widespread deployment of MBSE. The absence of convergent MBSE standards to date is further impediment to adoption. Systems modeling standards are beginning to emerge and that should have a significant impact on the application and use of MBSE. They include: Object Management Group (OMG) Systems Modeling Language (SysML™) and the ISO 10303-233 Application Protocol: Systems Engineering and Design.

INCOSE [12] also *predicts an anticipated evolution to replace the documentation-centric approach that has been practiced by systems engineers in the past, and to influence the future practice of systems engineering by being fully integrated into the definition of systems engineering processes. They anticipate that model-based systems definition practices will soon dominate and replace current documentation-centric approaches to address complex problems*.

A significant issue is that these existing model-based systems engineering tools rely on English for the core system specification. This reliance on a natural language, lends toward misunderstanding or ambiguities in the system model needed for the aviation community and its modeling needs.

Key Characteristics of MBSE needed to remove ambiguity

Of the five key characteristics that INCOSE identifies for future MBSE practices, the second key is *modeling standards based on a firm mathematical foundation that support high fidelity simulation and real-world representations* [12] is crucial. This fundamental characteristic will remove ambiguity and enable specificity.

Newer MBSE modeling languages such as AADL and the FACE™ Data Model language focus on defining models with a high-level of specificity that is sufficient for formal analytical methods.

AADL focuses on defining an unambiguous architecture-centric model of system software and hardware with the goal of providing analyzable models. A major benefit of AADL is that it unifies the models into one system architectural model. This allows identification of mismatches that occur when different models of timing, fault tolerance, security model, etc. are integrated, reducing the time of analysis to detect system failures that are typically realized later during system operation [13].

The FACE™ Data Model provides a standard method for data sharing between software components. Data modeling is required to enable information sharing and interoperability between software components. A common data model enhances reuse by establishing a standard communication data definition between software components.

The ultimate purpose of the FACE™ Data Model [14] is to define sufficiently the data semantics for messages (structures and fields) exchanged between two software components. The FACE™ v2.1 Data Model now provides for both a semantic description and an unambiguous Measurement System specification for each datum. By providing for an unambiguous non-arbitrary semantic and measurement description of message fields, developers can fully specify messages and can therefore increase interoperability and ease of integration of software components.

Both AADL and the FACE™ Data Model represent significant technological advances in MBSE. AADL defines an unambiguous architecture of data, and FACE™ Data Model defines unambiguous interaction between system software elements.

However, even if both approaches are applied, there are still significant gaps in defining a complete unambiguous model with sufficient specificity to provide for formal methods like the RTCA's DO-178C [4] of analysis, simulation, code generation, verification generation, validation, and artifact generation. This complete unambiguous model is the focus of the development and the essence of the TES-SAVi AWESUM™ modeling approach and toolset.

3. AWESUM™ MODELING APPROACH

The TES-SAVi AWESUM™ approach aligns avionics engineering activities to the Lifecycle V-Model (pronounced Vee-Model) process phases (Figure 1).

AWESUM™ is an industry-unique approach supporting the full range of airworthiness development, verification, qualification and certification efforts.

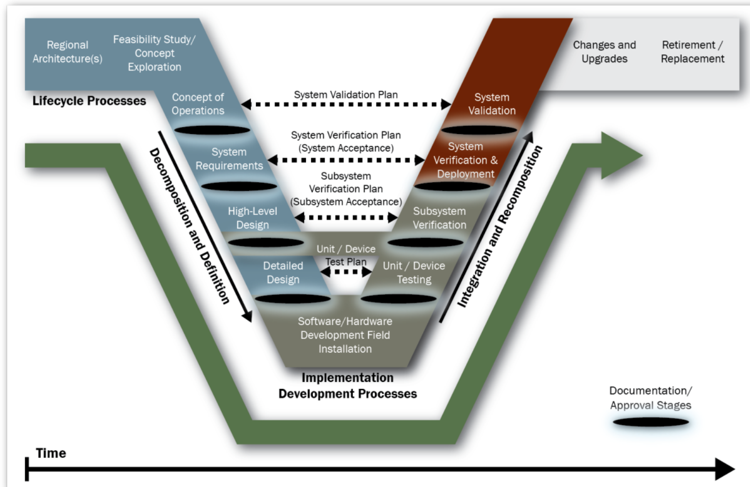


Figure 1 - TES-SAVi AWESUM™ Unified V-Model

Complete "Unified" Lifecycle for MBSE

A unified system and software model is central to AWESUM™. All aspects of the system lifecycle are modeled with sufficient specificity in the System Unified Model (SUM) to allow formal methods [4] to be applied including software code generation, and verification against both high-level and low-level requirements.[3] Developing aviation systems within a unified modeling system ensures compatibility from product requirements to coding to validation, and improves the product's maintainability throughout its sustainment. The SUM approach was developed by merging several modeling concepts of top-down bottom-up methods [8], DO-178C [4][5] modeling concepts and formal methods, and the FACE™ Standards and Data modeling [14].

[3] RTCA's DO-331 [5] describes two types of models: Specification and Design models (*abridged* for readability).

Specification Model—A Specification Model represents high-level requirements that provide an abstract representation of functional, interface, or safety characteristics of software components. This Model should express these characteristics unambiguously to support an understanding and does not prescribe a specific software implementation or architecture except for exceptional cases of justified design constraints.

Design Model—A Design Model prescribes the software component internal data structures, data flow, and/or control flow. It includes the low-level requirements and/or architecture. Design model expresses software design data, regardless of other content, and are used to produce code.

Top-down Bottom-up Modeling

The top-down approach of system engineering focuses on the system of interest or the system being developed. In this top-down approach the specification model precedes the design model efforts. Modelers begin with the concept of operations. In sequential order, modelers traverse the lifecycle through the system specification, system element specification, and then low-level development. Lastly, the verification and validation of the system is realized and derived from these early specifications.

This method is well suited to safety-critical cyber physical systems (CPS), however, it can "incur significant investment trying to keep documentation and plans up-to-date" when "unprecedented projects or projects with a high rate of unforeseeable change" occurs[4]. By employing a MBSE SUM approach, the changes to a system can be quickly implemented and necessary artifacts generated from the unified model.

One of the observed difficulties in developing a system with this top-down approach focusing on the system of interest, is the lack of information flowed down from the concept of operations. Often the workers on a lifecycle stage only have visibility to the previous stages output and not to the higher-level specifications. In order to mitigate this problem, the AWESUM™ approach utilizes a System of Systems (SoS[5]) hierarchy. The system of interest is modeled within the higher-level SoS model. This SoS model can then be used throughout the lifecycle to validate the lower-level models.

Another observed difficulty is the lack of detail specified for the lower level system elements. These system elements are often defined externally to the system of interest and are not modeled sufficiently for analysis or formal methods. The interface control document (ICD) is the lifecycle artifact that defines and details the external systems in military cyber physical systems (CPS). Unfortunately, ICDs are typically notoriously incomplete specifications for external devices and this incompleteness causes an incomplete system model unless the model itself augments the ICD in some formal manner.

The AWESUM™ approach also employs a bottom-up approach by modeling the low-level external systems. When this approach is combined with the top-down approach, it provides for the unambiguous specificity of both external and internal system elements within the system of interest needed for the development and fielding of high quality aviation systems [8]. This top-down bottom-up model-based approach to modeling cyber physical systems (CPS) gains the benefits of sufficient modeling of external systems and SoS elements to support the development and auto-generation of airworthy artifacts, and the validation of the system of interest.

[4] INCOSE SE handbook v3.2

[5] INCOSE SE handbook v3.2 section 2.5

System Hierarchy

The AWESUM™ MBSE approach implements a hierarchical SoS definition used to define a system as being constructed of other system elements. Each system element is described by both its behavioral model and by the systems data model. This hierarchical modeling approach is also applied to the concept of a System of Systems (SoS). SoS are system-of-interest whose system elements are themselves systems; typically these entail large-scale inter-disciplinary problems involving multiple, heterogeneous, distributed systems.

Data Modeling

The AWESUM™ Data Model leverages the FACE™ data model specification [14] achieving significant benefits of describing all data elements with sufficient specificity in both semantics and measurements. This achievement is a result of the AWESUM™ MBSE top-down bottom-up approach [8].

4. AWESUM™ MBSE TOOLSET

TES-SAVi's AWESUM™ MBSE lifecycle toolset is positioned to revolutionize the manner in which aerospace and aviation industries develop, verify, and deploy mission-critical and safety-critical capabilities targeted for the next generation of future flight. As implied, these systems and capabilities can be designed and integrated to support airborne, sea, and ground assets.

The AWESUM™ approach and reuse model is aligned to support systems engineering activities for the development of open systems software intended for the operational use on multiple dissimilar aviation aircraft. Our model and approach (section 3) is intended to optimize model-based process benefits (~35%) and maximize the reuse of software and lifecycle activities within the complex boundaries of airborne certification and procurement policies and practices (~63% within a branch, and 43% across branches). These benefits are described in section 7 and are detailed in Appendix A of this paper.

Mindfully designed to align for demonstrating compliance with airworthy process described in DO-178C/DO-331 [4][5], AWESUM™ is used to analyze and develop solutions to the technical issues of complex cyber-physical systems that the aviation community faces as it continues to pursue increased safety, systems capacity, and efficiency.

AWESUM™, for AirWorthy Engineering Systems Unified Modeling, is an industry-unique approach and model-based airworthy systems/software lifecycle product line [8]. The product suite is aligned to the qualifications requirements of US military software-reliant aircraft systems considering model-based practices, FACE, and reuse guidelines for reusable software components [references 1-7].

In addition to Army aviation qualification processes, TES has studied and is mindful of the Air Forces 516B [15] and the Navy's JSSSC SSSH [16] processes and how they cross-compare to the Army's AR 70-62 processes (section 7).

MBSE Tool Usage in Army AMRDEC

The US Army AMRDEC utilizes many different modeling tools within the organization. The AMRDEC's JCA[6], MIS[7], and R2C2[8] programs are using and evaluating the following tools: Sparx Systems Enterprise Architect (EA), Vanderbilt's ISIS FACE™ Toolset, IBM Rational products including Rhapsody, and TES-SAVi AWESUM™ toolset.

The JCA project's tool of choice is Enterprise Architect (EA). EA is being utilized to develop the JCA platform independent models (PIMs). These include the JCA data model, behavior model, and system model. AWESUM™ was chosen to augment EA for its ability to rapidly populate the JCA data model from existing Army supplied documents, database, and ICDs. AWESUM™ provides an integrated EA export directly inserting a FACE™ 2.0 conformant data model into the EA tool.

The MIS project is utilizing a broader set of MBSE tools since the project covers a larger swath of the system lifecycle. The modeling tools utilized by MIS include Rhapsody for systems modeling, DOORS for requirements management, and AWESUM™ for data modeling, external system modeling, and device handler code generation. AWESUM™ exports the data model in the standard XMI format, which is then imported into Rhapsody.

R2C2 is utilizing the TES-SAVi AWESUM™ toolset for data and interface modeling, external system modeling, artifact generation, and software verification.

5. Application of Modeling Processes

This section details the application of the AWESUM™ MBSE Modeling process. Actual applications of the process and toolset will be described. In addition to these real-world applications of the modeling processes, a simplified non-proprietary example is used to providing a detailed examples of the data modeling process. The example system is a ground vehicle with Vetronics (vehicle electronics) focusing on an integrated device called the general purpose (GP) radio. The GP Radio itself is a simple fictitious radio that implements ten (10) Mil-Std-1553 A/B messages.

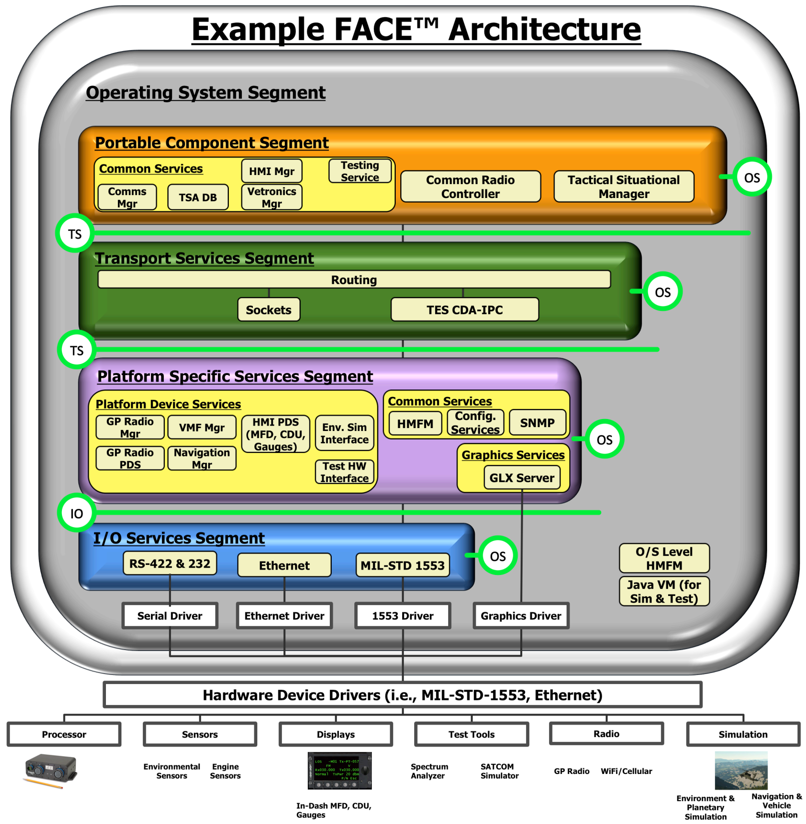


Figure 2 - Example Vetronics system onto a FACE™ Architecture

Figure 2 shows the example Vetronics system overlaid onto the FACE™ technical reference architecture diagram. For more information about the diagram and the details of the FACE™ technical reference architecture diagram and FACE™ segments, please see the "Technical Standard for Future Airborne Capability Environment (FACE™), Edition v2.0 [14].

6 The JCA data model product will be built upon open systems modular approaches, align itself with the FACE™ technical reference architecture, support software reuse across the Joint Multi-Role (JMR) fleet, and embrace automation to support lifecycle and airworthiness efforts.

7 MIS – Modular Integrated Survivability – is a FACE™ candidate situational awareness domain product and simulation. Our simulation suite models the aircraft platforms, simulates flight, and controls the operations of actual or simulated aircraft survivability technologies (AST) to illustrate enhanced situational awareness (SA) of the platform in flight in DVE conditions. ROSAS – Route Optimization for Survivability Against Sensors, is the follow on program of the MIS S&T effort.

8 R2C2 - Reusable Radio Control Component – is a FACE™ candidate communications domain application. R2C2 is written to FAA's DO-178B Level-C Design Assurance Level (DAL), and is aligned to the FACE™ reference architecture standard and to the FAA's AC-20-148 guideline for reusable software components [6].

Data Modeling: Conceptual & Logical Models

The data modeling practices described in this section are being developed over the course of several avionics projects, each using the TES-SAVi AWESUM™ tool suite for all or part of the project deliverables. Given both the flexibility of the tool suite and its support for the unambiguous, concise, complete, and traceable specification of model elements throughout the lifecycle, these practices can be categorized into roughly two related types, each aimed at the precise definition of systems, subsystems, components, and their interactions, required to support mission-critical and safety-critical solutions that offer no room for error.

Discovery Type Activities—The first set of data modeling practices is geared toward the loose-form, *discovery* type activities that are employed when there is uncertainly in subject matter, or domain, definitions. This usually occurs at the beginning of a project, or when refactoring is required.

During this stage, formal specifications are often unneeded and undesirable due to the frequency at which the proposed elements change, along with the freedom the engineers have in defining the parts of a system. The TES-SAVi AWESUM™ tool suite supports the importing of data[9] from many definition types and formats including Interface Control Documents (ICD), OMG IDL, UML/XMI, XSD, Requirements Specifications (SRS), Data Dictionaries, Mind Maps, and databases.

Specification Activities—The second set of data modeling practices support rigorous *specification* activities that provide a level of detail required to completely define, verify, document, integrate, and provide traceability[10] for, the interoperating parts of a system.

Following the discovery phase of a project, these practices, aligned with those required by FACE™ data modeling activities, are used to model the constituent parts of the system at the *conceptual*, *logical*, and *platform* levels.

These model elements consist of 1) foundational elements upon which all subsequent elements are defined, e.g., measurement systems, quantities, and units; 2) entities and associations between entities which provide meaning and context of the data in the system; and 3) the subsystems, capabilities, interfaces required to build a fully interoperable system that provides the desired functionality adhering to the required performance characteristics.

[9] Although AWESUM™ provides many importers off the shelf, customized importers can be created for virtually any definition and format given the openness of the product and the Eclipse platform upon which it is developed.

[10] The AWESUM™ tool suite allows bi-directional tracing of HLR to LLR to code to verification results. This ensures sufficiency of development and verification of the modeled system and products. If tracing has been imported into the tool suite, it can be traced to, or traced from any other model element.

The following sections describe these data modeling practices in further detail. They are shown in light of the modeling and modeling processes currently in progress to support the US Army's Joint Common Architecture effort for the Future Vertical Lift Joint Multi-Role (JMR) aircraft program, and other US Army FACE-aligned programs namely MIS and R2C2. The TES-SAVi AWESUM™ tool suite is supporting all three programs development and verification efforts.

The JCA Project is defining the Army's candidate Open Avionics Systems Architecture for the Future Vertical Lift (FVL) Family of Systems (FOS). Part of the JCA scope is defining a software product line enabled by the FACE™ technical standard. As such a common set of semantics and interfaces for reusable, interoperable software components, are sufficiently modeled to demonstrate FACE™ Conformance and corresponding lifecycle documentation is developed to demonstrate support of Airworthy qualification efforts across a suite of dissimilar target military Aviation platforms.

The goals of the JCA Project and JCA Data Model product are to align with the FACE™ technical reference architecture and align to the FACE™ shared data model [1]. In light of these goals, the discovery and specification phases described below support the analyses of many disparate sources of avionic subject matter domains with modeling tasks aligned to the stated requirements. In support of a JCA verification effort scheduled to commence in 2014, a "slice" of JCA focusing on situational awareness has been built-out.

It is noteworthy that all of the data imported and/or entered during each of the phases is stored in the System Unified Model, and leveraged by the AWESUM™ tool suite, allowing traceability and refinement as the model progresses. The overall process is shown in the Figure 3 below. This figure illustrates TES patented process [8].

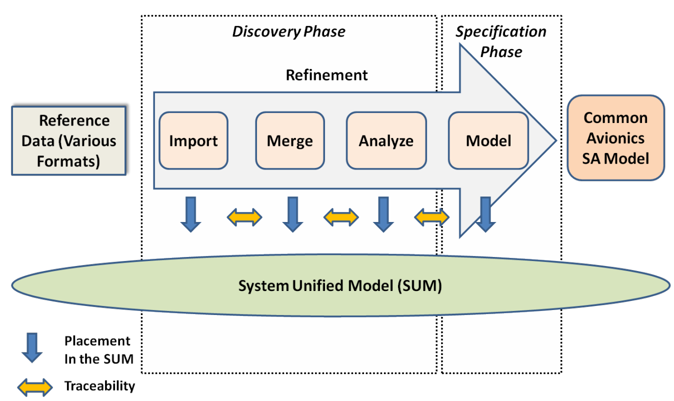


Figure 3 - Overall Modeling Process

Discovery

The goal of the discovery phase, as implied, is to discover the relevant SA objects common in avionics systems along with their characteristics and relationships. The discovery phase is broken up into three sub-phases: Import, Merge, and Analyze.

Import—At the beginning of the project, requirements, consisting of DoDAF Operational Views (OV-1: High-level Concepts, OV-2: Node Connectivity, and OV-6c: Event Trace), are imported into the Requirements View of the tool suite. By placing the requirements into the SUM, each is available for reference and traceability to the developed model elements. The hierarchical nature of the Requirements View allows nesting of the high-level and subsequent derived requirements. The tool suite provides reporting features that are run throughout the effort, ensuring that the modeling is progressing according to the defined requirements.

In order to have a realistic and complete source of avionics domain knowledge, existing Interface Control Documents (ICDs) for avionics Line-Replaceable Units (LRUs), high-level reusable application programming interfaces (APIs) such as those described by the US Army's Reusable Radio Control Component (R2C2), military standards such as MIL-STD 2525C[11], Variable Message Format (VMF)[12], VICTORY[13], avionics data dictionaries, field manual contents, glossaries, and several other situational awareness and data fusion model sources [17][18][19] and AMRDEC Aviation Vision concepts for future battlespace operations [20] were imported into AWESUM™ tool suite using a mix of existing and custom developed ICD and specification importers.

Merge & Analyze—Even though the Merge and Analyze sub-processes can be viewed as separate activities, in practice they are closely inter-related. It is during these phases that the overall abstractions of the avionics SA elements are seen, understood, and realized into a high-level model ready for the formal specification activities.

Once imported into their respective views[14], the imported data is analyzed for commonality in SA subject matter areas. This analysis, along with referencing several industry and scholarly articles [17][18][19] on aircraft situational awareness, major knowledge areas became apparent. These knowledge areas then become the groupings for all SA information, named for the elements, which they contain, primarily: airspace, battlespace, environment, geographical, identification, man-made, mission, platform vehicle, and system.

With its broad cut/copy/paste abilities, between both similar and dissimilar model elements (e.g., types, functional parameters, and data dictionary entries), the tool suite's database organizational capabilities make it easy to group like concepts together, compare their characteristics and associations, and produce properly attributed situational awareness abstractions in the appropriate groupings.

At this stage of the process, a formal specification of the concepts is not required, nor is desired. Of the considerable number of specification attributes provided by the tool suite for SUM elements, most of the effort is concentrated on specifying the names of the domain, knowledge areas, conceptual objects, associations, their descriptions, and their characteristics. Traceability to the requirements is employed to ensure that each of the concepts is needed.

Specification

Once the discovery model is developed, it is time to formally specify the domain, knowledge areas, conceptual objects, and their characteristics in terms of their foundational elements, entities and associations, and capabilities/interfaces (Figure 4).

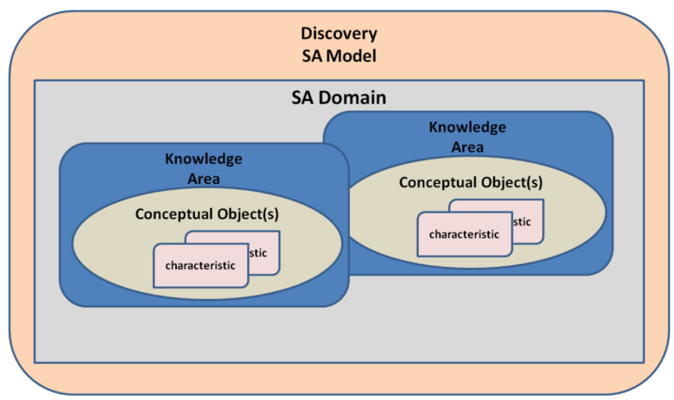


Figure 4 - Discovery SA Model Elements

For Army's Joint Common Architecture (JCA) effort, this is performed at the conceptual and logical levels. This is shown in Figure 5 below with respect to the overall AWESUM™ process and modeling artifacts.

[11] MIL-STD 2525C is a NATO standard for military map marking symbols for aviation and land-based systems.

[12] Variable Message Format (VMF) MIL-STD-6017 is a communications protocol used in communicating tactical military information.

[13] Vehicle Integration for C4ISR/EW Interoperability.

[14] The AWESUM™ tool suite provides several views that can "house" the imported data. The JCA project made use of the Requirements, Type, ICD, and API views to contain the imported data. For the merge and analysis sub-phases, the Visualize, Trace, and Statistics views were used extensively.

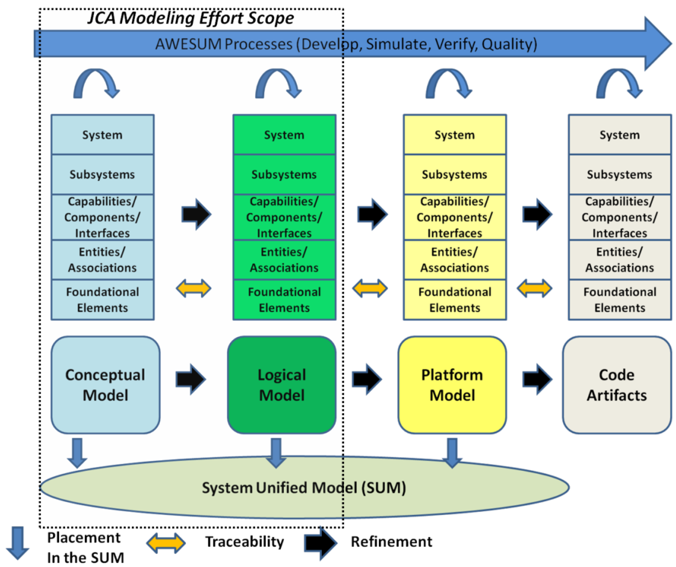


Figure 5 - JCA Modeling Effort Scope in light of the AWESUM™ processes and modeling artifacts

The AWESUM™ tool suite views the specification of a system as the refinement of the foundational and subject matter elements (systems, subsystems, capabilities, and entities/associations) from the conceptual through the generation of code artifacts[15]. Note that this differs slightly from the FACE™ Data Model Architecture in that the *entire set* of allowable model elements are specified *at each level of abstraction* and carried forward to the next.

In other words, across the same model element, the element is refined, having different sets of meta-data appropriate for the level of abstraction. This allows simulation, verification, and qualification activities to occur (if desired) during all phases of the development.

Figure 6 shows the mapping between the discovery model elements to those required at the conceptual and logical levels. Each of the sub-sections below describes in greater detail the processes used to create these conceptual and logical model elements.

[15] As of this writing, the platform and code artifacts are not currently part of the JCA effort that being the yellow and beige colored columns of Figure 5.

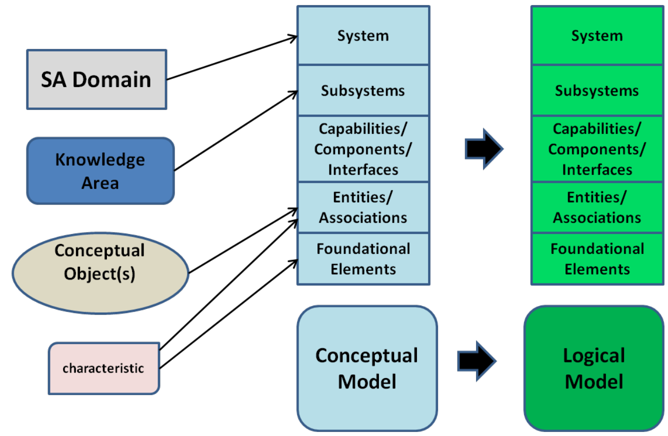


Figure 6 - Mapping of Discovery Element to Formal Specification Elements

Foundational Elements—Foundational elements, or basis elements as they are called in the FACE™ specification and shared data model, are those elements that provide a foundation upon which all other elements are constructed.

At the Conceptual level, there is one foundation element called the Observable. An Observable is roughly akin to an SI[16] quantity and is simply a property that can be measured, such as length, with no further detail added such as unit or reference frame. For the JCA effort, Observables were created for each of the SI Quantities, such as length and time duration, plus those derived from two or more quantities, such as for position and orientation.

As shown in the Figure 6, the characteristics of the Conceptual objects from the Discovery phase are used as input to defining the Observable foundational elements. This was especially helpful for those not directly related to SI quantities.

Figure 7 illustrates a simple example of a general Radio Conceptual Entity with three characteristics of the radio: volume, frequency, and time.

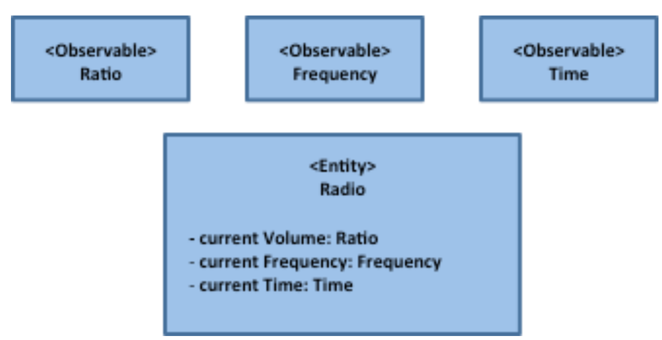


Figure 7 - Example Radio Conceptual Elements

At the logical level, the foundational elements are refined into Measurement Systems, Measurements, and Units. Measurement Systems provide a reference, orientation, and

[16] International System of Units (SI)

a specification for the Measurements they describe – a Measurement being the realization of an Observable.

For example, a local horizontal measurement system provides the measurements of elevation and azimuth angle relative to a local reference point. Units represent magnitudes of quantities, and are taken from the set of units defined by the SI as well as other derived units required in avionics systems.

For the JCA effort, only those measurement systems needed to describe Observables, in the context of the conceptual entities, were modeled. One of the fundamental aspects of the JCA modeling architecture is that every attribute of an entity is described by a foundational element. At the conceptual level, this means all entities are described in terms of Observables, at the Logical Level in terms of Measurements, defined within a measurement system, with a given reference point and unit. In this way, a complete foundational specification for remaining model elements, including data conversions, can be created. Figure 8 illustrates this clearly.

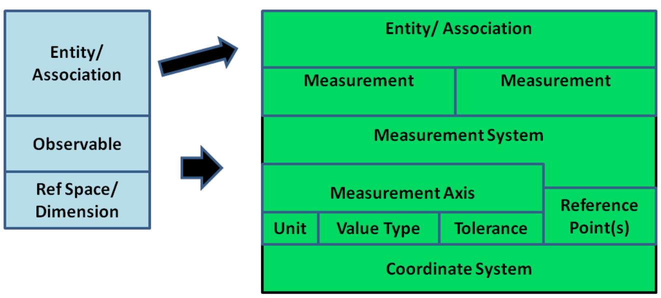


Figure 8 - Foundation Elements from Conceptual to Logical

Domain Modeling

Entities, Associations & Attributes—Within a subject matter domain, entities, association types, and their attributes are the elements modeled and refined at each of the levels of abstraction.

Entities and associations provide context for the foundational elements that describe them. For example, two entities, one named **Aircraft**, the other **WeatherEvent**, might each have an attribute called **altitude**. These **altitude** attributes are different because the context of each is different, even though they both represent **altitude**. The context of an attribute is important when specifying the content of messages between components. It is undesirable for one component to specify the sending of the **altitude** of the Aircraft while the other specifies the receipt of the **altitude** of a storm system that is in the flight path of Aircraft.

In addition to entities and association types, relationships between these model elements are also modeled. With respect to an entity or an association type, a reference to, composition of, and specialization of another entity or association type is made. This differs slightly from the FACE™ Data Model Architecture in that it only allows the former two and not the latter.

Given these modeling constructs: entities, association types, and relationships, the following describes the level of detail at each level of abstraction.

At the conceptual level, entities and association types are modeled as named objects with one or more named attributes whose types are Observables. At the logical level, conceptual entities and association types are refined into their logical realizations. This realization includes the refinement of each conceptual attribute into a logical attribute.

The Observable attribute definition is refined into a measurement defined within a measurement system. For JCA, after having spent a considerable amount of time on the definition of measurement systems and measurements, the realization of entities and association types from conceptual to logical was straightforward – a simple refinement of each from attributes based on observables to attributes based on measurements defined within a measurement system. This was especially true due to the fact that JCA, being a reference architecture, was aimed at reuse, and therefore chose a single unit for each realized Observable, e.g., meters for length, kilograms for mass, etc.

Figure 9 illustrates the realization of the Conceptual Observables as Logical Measurements. For example, **Frequency** is realized as **RS_Frequency_Mhz**, which refines the Observable and provides sufficient detail for unambiguous data transformations, processing and potential verification.

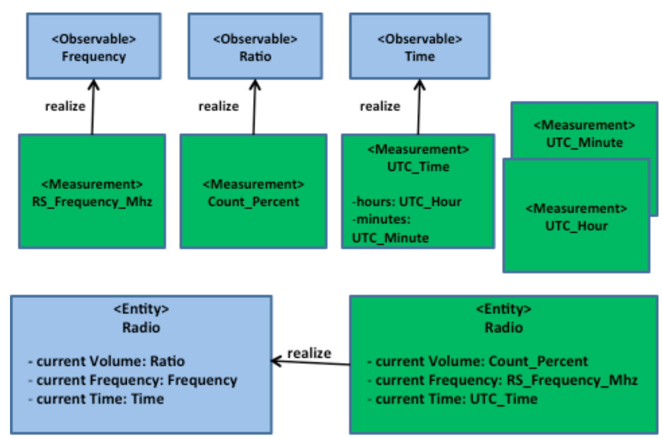


Figure 9 - Example GP Radio Logical Elements Realizing Conceptual Elements

In addition, this example also shows the Logical Radio Entity realizing the Conceptual Radio Entity. The Logical Radio also refines each characteristic, such as **current Volume**, by adding additional information about the measurement. This is seen in that the **current Volume**, is represented logical as a **Count_Percent**. Not shown in this figure is the definition of the measurement system, which provides all of the detail about the Measurement such as the **Count_Percent** range of values, precision, and Units.

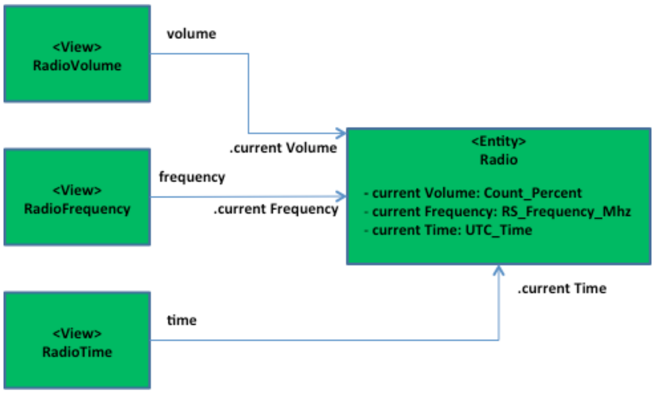


Figure 10 - Example GP Radio Logical Elements Realizing Conceptual Elements

The Platform Model described in previous sections, not shown, provides the mapping to IDL data types for each Logical attribute. This provides a consistent definition for transformations of the data model into various programming languages such as C/C++, Ada, and Java.

Capabilities, Components & Interfaces

As part of the various verification events/demonstrations needed for JCA, and other efforts that are based on the JCA data model, a single component– the **JCA SA Data Manager** – was created to store and provide data to the various system components.

A simple Create, Read, Update, Delete (CRUD) interface was designed to perform the most common operations for access to the SA data. Due to the openness of the AWESUM™ tool suite, to address this requirement[17], the set of capabilities and CRUD interfaces are generated into the Systems Unified Model (SUM) from model elements already defined in the SUM.

Figure 11 further expands on the GP Radio example by showing the GP Radio Handler Component and its interface to the Communications (Comms) Manager.

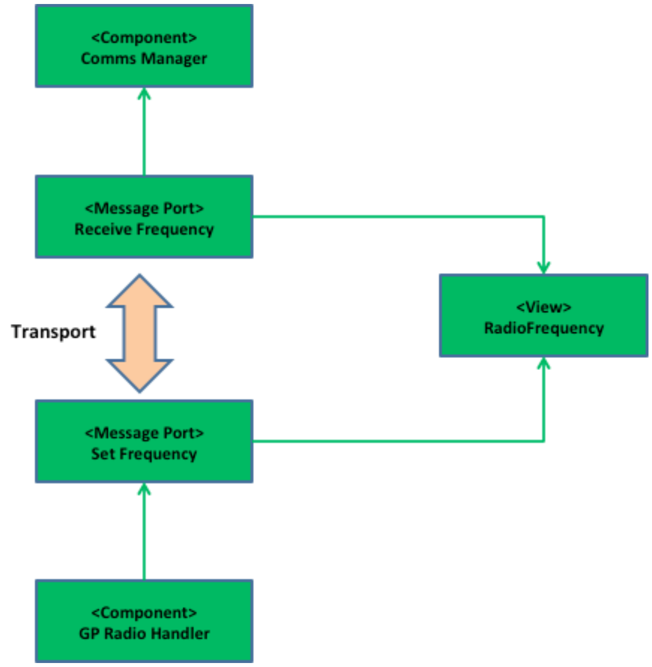


Figure 11 - Example Radio Comms Mgr to GP Radio Mgr Handler Interface

In this example interface, the **Comms Mgr** is setting the operational frequency of the GPR Radio. The radio frequency was earlier defined in Figure 10 Logical Model and maps to the radios **current Frequency**. This mapping identifies the semantic meaning of the field. This semantic meaning, when combined with the full measurement specification of **RS_Frequency_Mhz** sufficiently defines the message field, and eliminates any mismatch or ambiguity between the **Comms Mgr** and the **GP Radio Handler** to what is the frequency.

While this is a trivial example, it illustrates that by sufficiently defining entities, characteristics, and their measurements; we can achieve a significant reduction in the potential mismatch between components.

Further, the SUM provides the data required for the execution of the system at the conceptual, logical, platform, and implementation levels. This illustrates the heart of TES-SAVi AWESUM™ robust unambiguous modeling process.

[17] Required for the demonstration.

6. Initial Adopters and Applications

A cooperative research and development agreement (CRADA) between the US Army AMRDEC[18] and TES-SAVi was executed for purpose of use of the AWESUM™ tool chain. This allows AMRDEC early adopters to evaluate the potential benefits and return on investment of the unified model-based tooling environment for reusable opens systems software targeted for use on a fleet of dissimilar aviation aircraft.

The fundamental interest is toward the development and verification activities of FACE™ applications targeted for reuse across the fleet of US Army aviation platforms.

Additionally TES-SAVi is providing tooling to, and working with, platform system integration laboratories (SILs) so they can evaluate the efficacy of developing reusable capabilities across US Military and Foreign Military Sales (FMS) variant platforms. The expected benefits are improved capability fielding with increased verification activities for reduced schedule and costs.

These early adopters will use the TES-SAVi AWESUM™ tool suite so that AMRDEC personnel may develop and test common reusable embedded aviation software systems and products aligned with the FACE™ technical reference standard and other open architecture standardization efforts.

TES-SAVi believes that with its products and services, AMRDEC users can develop, deliver and support high quality systems, and software engineering support libraries that will significantly reduce costs, schedules, and risk of systems development while substantially increasing the quality of the software system.

7. AWESUM™ MBSE Benefits

The TES-SAVi AWESUM™ toolset manages the busy work of aviation engineering, so that the engineers can focus more on engineering the product, not the process.

Numerous studies have shown that a product's total cost of ownership (TCO) is reduced when software issues are discovered and resolved earlier in the lifecycle. Our goal was to produce a product that can manage design, development, and verification activities, optimize automation and reuse.

[18] U.S. Army Aviation and Missile Research Development and Engineering Center (AMRDEC) Software Engineering Directorate (SED) is a recognized leader in supporting the acquisition, research, development, and sustainment of some of our Nation's sophisticated weapon systems. The Aviation Engineering Directorate (AED) is the Airworthiness authority for Army- developed aircraft.

The AWESUM™ MBSE tooling has been specifically designed to host data with sufficient specificity to support the auto-generation of lifecycle artifacts required by the Army's airworthiness process (see Appendix B, **Qualify™**) and support the reuse of these lifecycle artifacts across the US Army Aviation fleet following the guidelines of the FAA's AC 20-148 (Figure 12) and demonstrating DO-178C objectives.

Our model and approach is intended to optimize model-based process benefits (~35%) and maximize the reuse of software and lifecycle activities within the complex boundaries of airborne certification and procurement policies and practices (~60% within a branch, and 40% across branches).

These are significant process and cost savings projections. This section qualifies these numbers. Caveat: it will be only through the usage of our lifecycle products on actually fielded systems, that actual data can be quantified. Few programs have the S&T funds and resources available to dual path a "business as usual" case parallel to a process improvement test case. As such in this section we provide range estimates for projected development activities, such as document generation, and percentages of what can be essentially auto-generated using the fidelity of the data within the MBSE environment of our tool chain.

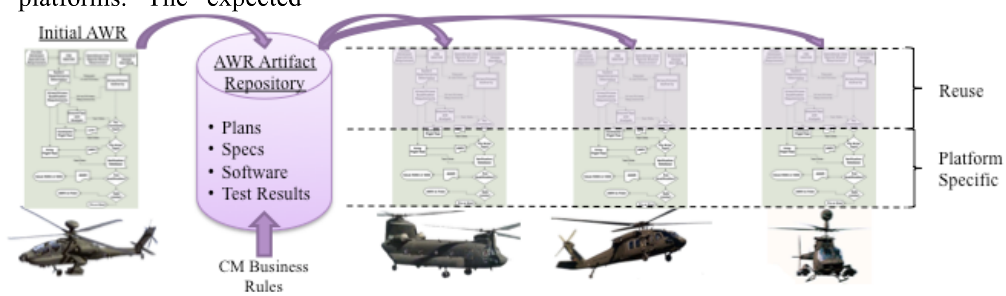


Figure 12 - FAA's AC 20-148 Reuse Process [6] applied to Army Aviation

Model-Based Artifact Engineering ~35% automation gains

There are 35 software lifecycle artifacts required to support the certification and qualification of US Army Aviation Systems [21]. The refactoring of these 35 lifecycle documents required for airworthiness qualification efforts into a model capable of auto-generating and managing the aspects of the product that is to be qualified is unique [TES-SAVi patent pending].

This model-based artifact engineering technology uses the AWESUM™ tool suite data base, models the format and contents of lifecycle Contract Data Requirements List (CDRL) per the data item description (DID) suggested format, and allows AWESUM™ system users to manage product-to-artifact as well as artifact-to-product, and produces lifecycle artifacts prepared to support product certification and qualification efforts.

Current technology and practices require a set of lifecycle artifacts that are typically produced after (not concurrent to) the product (e.g., software). The artifact suite is required for certification/qualification efforts; to ensure that proper work practices for the design, development, and verification of the product was used. Based on years of observations of program management and overseeing software product developments, these artifacts are typically produced after the product (software code). Often the artifacts are not aligned with the current state of the product. Contractually, suites of documentation are required in multiple states (draft, final, controlled) as the product progresses through the lifecycle. Product changes and requirements churn often dictate the reconstruction and re-delivery of previously delivered documentation. The process is cumbersome and expensive, compounded by the fact that the documentation may not actually represent the current state of the product. Stakeholders often question the usefulness of disconnected lifecycle documentation.

Model-based artifact engineering is unique in that it connects the product to its artifact. At any point, a customer, a Designated Engineering Representative (DER), or any stakeholder has the ability to request a document of the current state of the product. Unique is that producing an artifact is equivalent to producing the product. All aspects of artifacts are modeled. This model-based structure leverages the format as described by the data item descriptions (DID) for the lifecycle artifacts required for the qualifications efforts for airborne software systems.

Reuse of Artifacts–alignment to AC 20-148 ~43-63% reuse

The Authors have conducted significant studies of how lifecycle artifacts can be used to leverage DO-178C objective credit, and the reuse of these credits using the FAA's AC 20-148 guidelines [6] to support qualification efforts of a reusable software component (RSC) on multiple aviation platforms to improve capability integration efforts, reduce time, costs, and program risks of follow-on platform integrations.

Figure 12 illustrates a process of how lifecycle artifacts, if designed for reuse, can be reused to support follow-on integration and aircraft qualification efforts with significant savings.

If we categorize the 35 lifecycle documents required for airworthiness qualification efforts into Planning (7), Requirements and Design (10), Coding (4), Verification (6), and Engineering Process (8), and intentionally write the Plan for Software Aspects of Certification (PSAC) for multiple target platforms (e.g., Apache, Chinook, Blackhawk, and Kiowa as illustrated above), the AC 20-148 guidance permits the reuse of lifecycle artifacts. For example Full, Partial or no-Credit can be specified in the PSAC for DO-178C approval objectives. For example, Planning documents, SEMP, SDP, SCMP, SQAP, SSPP can be written such they are non-specific to an individual platform and therefore Full credit can be sought for the Software Planning Process objectives listed in DO-178C Table A-1 [4], thereby adding to the reuse %. However, verification artifacts like a Test Report is specific to a target platform, and as such no-Credit can be sought for DO-178C Table A-6 [4].

The Authors have compared the Army's AR 70-62 [2] qualification process to the Air Forces 516B [15] and the Navy's JSSSC SSSH [16] process, and there is a potential of reuse of up to 63% reuse efficiency for follow-on integration efforts of Army and Air Force platforms, and up to a 43% reuse potential if credit is sought for Navy platforms (see Appendix A).

TES' automation and reuse paper [10] quantified that for "sized" program, the lifecycle documentation required for qualification could consume up to 3 years of effort.

With the TES-SAVi AWESUM™ tool suite and model-based artifact engineering integration capability in Qualify™ users could achieve up-to 1-year (35%) reduction in cost/schedule on initial integration efforts by auto-generating artifacts to support qualification efforts, and up-to 2-years (63%) reductions cost/schedule on follow-on integration efforts following the FAA's AC 20-148 guidelines.

8. Summary

As wider experience is gained with MBSE tools like TES-SAVi AWESUM™, and as simulation products are used for obtaining certification under DO-178C/DO-331, it is believed that the acceptance of the use of model-derived tools and simulation results for specifications and design acceptance will be a function of the fidelity of the models(s) (e.g., the system, it operating host/target operating system, flight models, and environment). Then significant process benefits will be achieved, and it is expected that the issues that have been impeding progress toward the objective of the delivering aviation systems solutions that have increased safety, system capacity, and efficiency will be addressed.

As such, the TES-SAVi AWESUM™ toolset is a comprehensive end-to-end lifecycle for airworthiness products. It should speed capability development, integration, and reduce the total cost of ownership (TCO) by employing improved model-based system engineering practices in a unified modeling environment.

TES-SAVi's AWESUM™ MBSE lifecycle toolset is positioned to revolutionize the manner in which aerospace and aviation industries develop, verify, and deploy mission-critical and safety-critical capabilities targeted for the next generation of future flight. These systems and capabilities can be designed and integrated to support airborne, sea, and ground assets.

Questions & Comments

Questions and comments can be sent to StephenS@TucsonEmbedded.com.

REFERENCES

[1]"Technical Standard for Future Airborne Capability Environment (FACE™) Edition 2.0," The Open Group, February 2013.

[2] "Airworthiness Qualification of Aircraft Systems," US Army Regulation AR 70-62, Research, Development, Acquisition, HQ Department of the Army, 21 May 2007.

[3] "Advisory Circular AC 20-115C – Airborne Software Assurance," US Department of Transportation, Federal Aviation Administration, July 2013.

[4] "RTCA DO-178C – Software Considerations in Airborne Systems and Equipment Certification," RTCA Dec. 2011.

[5] "RTCA DO-331 – Model-Based Development and Verification Supplement to DO-178C and DO-278A," RTCA Dec. 2011.

[6] "Advisory Circular AC 20-148 – Reusable Software Components," US Department of Transportation, Federal Aviation Administration, December 2004.

[7] "RTCA DO-297 – Integrated Modular Avionic (IMA) Guidance and Certification Considerations," RTCA Nov. 2005.

[8]"Method and Apparatus for Interfacing with Multiple Objects using an Object Independent Interface Protocol," US Pat No 8,239,586, Tucson Embedded Systems' Capability Driven Architecture (TES' CDA).

[9]"Capability Driven Architecture: An Approach to Airworthy Reusable Software," Tucson Embedded Systems, American Helicopter Society 63rd Annual AHS International conference, Virginia Beach, Virginia, May 2007.

[10] "Reusable Automated Platform SIL Testing -- A Cost-Effective Risk-Reduced Process for Airworthy Reusable Software," Tucson Embedded Systems, American Helicopter Society 66th Annual Forum, Phoenix, AZ., May 2010.

[11] "Enabling Situational Awareness and Network Centric Operations for Systems utilizing FACE™ Open Systems Architectures," Tucson Embedded Systems, American Helicopter Society International Specialists' Meeting on Unmanned Rotorcraft and Network Centric Operations, Scottsdale, Arizona, January, 2013

[12] "Systems Engineering Vision 2020," International Council on Systems Engineering (INCOSE), TP-2004-004-02, 2007.

[13] "Model-Based Engineering with AADL," Feiler & Gluch, Pearson Education Inc., 2013.

[14] "FACE™ v2.0 & v2.1 Reference Implementation Guide," Open Group, 2014.

[15] "MIL-HDBK-516B – Airworthiness Certification Criteria US Department of Defense (DOD), Sept 2005.

[16] "Joint Software System Safety Committee (JSSSC) Software System Safety Handbook," 1999.

[17] "A Survey of Situational Awareness Requirements in Air-toAir Combat Fighters," Endsley, 1993, Department of Industrial Engineering Texas Tech University. Journal of Aviation Psychology.

[18] "Situational Awareness Information Dominance & Information Warfare," Endsley & Jones, 1997, U.S. Air Force Armstrong Laboratory, AL/CF-TR-1997-0156.

[19] "Revisions to the JDL Data Fusion Model," Steinberg, Bowman, White & Jones, 1999, ERIM International Inc.

[20] "Aviation 2050 Vision - Technology for Tactics," 2013, Dr. Bill Lewis, Director of the AMRDEC's Aviation Development Directorate.

[21] "Developer's Handbook for Airworthy, Reusable FACE Units of Conformance," Carter, Simi, Tompkins; 2014, US Army AMRDEC-SED.

[22] "System Architecture Virtual Integration (SAVI)," Aerospace Vehicle Systems Institute, Texas A&M University System.

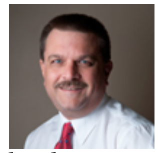
Biography

Sean P. Mulholland, serves as TES-SAVi's Senior Vice President, General Manager, and Chief Technological Officer. Sean has 25 years of experience and is recognized in the industry as one of the top embedded systems engineer in the country. Sean is a contributing author to the FACE™ technical reference architecture and has been active serving as a key FACE Data Model technical leader, using TES' work on the JCA DM. Sean is a co-founder and vice president of TES. Continuing, as he is doing today, Sean will serve to guide all technological decisions for SAVi's products and services.

Sean has a B.S in Computer Science and Systems Design from the University of Texas at San Antonio and has over 25 years of experience in software development, design, integration and testing, especially as it relates to mission critical and safety critical systems. Sean is one of the four co-founders of TES.

Stephen M. Simi, serves as TES-SAVI's Vice President. Stephen has 28 years of experience design and developing engineering and scientific applications, and managing multiple programs. Stephen is also very active in the FACE Consortium's Conformance, Airworthiness, and Outreach subcommittees. He is recognized as an industry innovator of agile technologies that can be applied to Joint forces across the common operating picture/battlespace of C4ISR assets, and an industry expert in lifecycle development of reusable software systems. He has several technical publications and presented to the AHS, AOC, AIAA/IEEE societies, to FACE and MITRE on areas of software development, reusable systems, and advanced modeling and simulations of those systems. Stephen currently manages 4 US army programs, JCA, MIS, R2C2, and MICD for TES.

Stephen has a B.S. in Physical Sciences (Math Computer Sciences, and Engineering) and a M.S. in Engineering from the University of Maryland. Before working for TES, Stephen served as the Director of Software Development, and Director of Software Business Development at world-renown optics company Breault Research. He also served as a technical fellow at the MITRE Corporation for the US Army, and served various other organizations designing, developing, and testing engineering and scientific applications over his 30-year technical career.



William G. Tanner has a B.S. in Computer Science Engineering from Northern Arizona University and over 20 years of embedded software and embedded software application development experience, 14 of which are in software engineering project and product management. Bill is a contributing author to the FACE™ technical reference architecture, has been active serving in the FACE Data Model Working Group, and is the lead data modeler for the Joint Common Architecture (JCA) Data Model. Prior to joining TES in 2006, Bill worked for IBM as a software engineer in the disk storage system division, and Project Technology and Mentor Graphics as a data modeler and project manager for the division's UML modeling, verification, and code generation tools.

APPENDIX A – LIFECYCLE COMPARISON

Comparison of Lifecycle Artifacts for Software targeted for Airborne Systems US Army's AR 70-62 [2], Air Force's 516B [15], Navy's JSSSC SSSH [16]

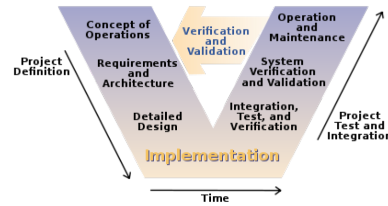
Software Lifecycle Artifacts for Military Airworthy Systems								
						Working Copy dated: Jan. 2014 sms		
Airworthiness Artifacts				Platform	Across			
Required by				Reusable	Branches			
Air Force (Note 1) (516B)	Army (AR 70-62)	Navy (JSSSC SSSH)	MBSE auto-generated	AC 20-148 Platform Reusable Army Aviation	Potential Reuse Across Branches			
Plans			0%	100%	67%	***Plans***	Abr.	**DO-178C and DID mappings**
Y	Y			Y		PSAC/AQS	PSAC	DO-178C 11.1 DI-MISC-80508B
Y	Y			Y		Systems Engineering Management Plan	SEMP	DI-IPSC-81785
Y	Y	Y		Y	Y	Software Development Plan	SDP	DO-178C 11.2 DI-IPSC-81427A
Y	Y	Y		Y	Y	Software Validation Plan/Test Plan	SVP/STP	DO-178C 11.3 DI-IPSC-81438A
Y	Y	Y		Y	Y	Software Configuration Management Plan	SCMP	DO-178C 11.4 DI-CMAN-80858B
Y	Y	Y		Y	Y	Software Quality Assurance Plan	SQAP	DO-178C 11.5 DI-MISC-80508B
Design and Specifications			75%	63%	63%	***Design and Specifications***	*	
Y	Y	Y	A	Y	Y	System/Subsystem Design Document	SSDD	
Y	Y	Y	A	Y	Y	System/Software Requirements Specification	SRS	DO-178C 11.6, 11.9 DI-IPSC-81433A DI-IPSC-81434A SAE-ARP-4754 5
Y	Y	Y	A	Y	Y	API Specification (Tool generated document that is a combination of SRS and ICD)	ICD	
Y	Y		A	Y		Requirements Trace and Verification Matrix	TVM	
Y	Y	Y	A	Y	Y	Software Design Document	SDD	DO-178C 11.10 DI-IPSC-81435A DI-IPSC-81436A
Y	Y	Y	A	N	Y	Software Version Description	SVD	DO-178C 11.16 DI-IPSC-81442A
Y	Y			N		SCM Records	SwCMR	DO-178C 11.18
Y	Y			N		SQA Records	SwQAR	DO-178C 11.19
Safety Analysis (ARP-4761, et al)			0%	17%	17%	**Safety Analysis (ARP-4761, et al)**	*	
Y	Y			Y	Y	System Safety Program Plan	SSPP	DI-SAFT-81626B, MIL-STD-882
Y	Y	Y		N	N	Preliminary Hazard Assessment	PHA	
Y	Y	Y		N	N	Failure Mode Effects and Criticality Analysis	FMECA	
Y	Y			N	N	Functional Hazard Assessment		DI-SAFT-80101B SAE-ARP-4761 App A
Y	Y	Y		N	N	System (Software) Safety Assessment		DI-SAFT-80101B SAE-ARP-4761 App B, D, G
Y	Y	Y		N	N	Safety Assessment Report (Software)		DI-SAFT-80102B SAE-ARP-4761 App C
Reviews			0%	100%	20%	***Reviews***	*	
Y	Y			Y		Architecture Peer Review Results		
Y	Y			Y		PDR/CDR/TRR/FRR Milestone Review Exit Criteria Achieved		
Y	Y	Y		Y	Y	Source Code Peer Reviews		
Y	Y			Y		SIL Configuration Audit		DO-178C 11.15, 11.16, Table A-8
Y	Y			Y		Software Test Procedures Peer Review Results		
Test Artifacts			71%	64%	36%	***Test Artifacts***	*	
Y	Y		A	Y		Unit Test Results - developer document	UNIT TST	
Y	Y	Y	A	Y	Y	Software Verification Cases and Procedures SVCP	STD	DO-178C 11.13 DI-IPSC-81439A
Y	Y	Y	A	N	Y	Software Verification Results	STR	DO-178C 11.14 DI-IPSC-81440A
Y	Y		A	N		Software Robustness Test Results	STR*	depends on DAL
Y	Y		A	N		Performance Test Results	PTR*	
Y	Y			Y		Statement Coverage Analysis Report	CODE COV	depends on DAL
Y	Y			Y		Static Code Analysis Report	CODE ANA	depends on DAL
Y	Y		A	Y		Integration Test Plan		Table A-5
Y	Y		A	N		Integration Test Results	SIT R	Table A-6
Y	Y	Y	A	N	Y	Software Problems Reports	SPRs	DO-178C 11.17
Y	Y	Y		Y	Y	Software Change Requests	SCRs	
Y	Y			Y		CCB Minutes	RRM	DO-178C 11.18, DI-ADMN-81505
Y	Y			Y		Action Items	AI	
Y	Y	Y		Y	Y	Software Accomplishment Summary	SAS	DO-178C 11.20 DI-MISC-80508B
		Y	N	Y	Y	***Training Documentation***		
			35%	63%	43%	lifecycle AWR		
			auto-gen	reusable	reusable			

APPENDIX B – AWESUM™ PRODUCT

TES-SAVi AWESUM™ Model-Based Toolset

The AWESUM™ product line is purposely named and composed of a suite of integrated model-based tools that align with the lifecycle V model and components described by the RTCA's DO 331 [5].

Unlike other model-base market toolsets, TES-SAVi AWESUM™ unifies the lifecycle activities. Specifically, Systems and Software Design phase, the leading edge of the V Model, is handled by the **Develop™** module. The **Verify™** module handles the Software Implementation phase, the bottom side, and Verification phase, the following edge. Additionally, AWESUM™ model-based product suite is especially designed to support the lifecycle activities required for most-stringent airworthiness qualification efforts with our **Simulate™** and **Qualify™** modules. These product modules are further described.



Develop™ is an airworthiness software development environment that accelerates development and ensures traceability of high-level systems requirements through the lower-level requirements of software development phases. **Develop** imports, models, and automatically structures development projects from target device-level ICDs. System Engineers focus on core development tasks while **Develop** automatically generates device control code for common and modeled functions. All code in **Develop** directly reflects design requirements, establishing complete documentation traceability. **Develop** is used to create and manage the specification and design models throughout these important requirements and development lifecycle phases, i.e., the leading edge side of the V-model. These specification and design models are described further in section 3.

Verify™ is a software testing and verification system that accelerates and intensifies robustness testing of cyber-physical systems (CPS). **Verify** manages the verification lifecycle activities and assists in determining documentation and verification process completion, generates device design tests, executes tests that help ensure system functionality, and manages test results. **Verify** deepens testing by orders of magnitude compared to other industry tools with robustness testing, while dramatically reducing testing time with automating verification activities. By maintaining bi-directional tracing links between requirements, modeled capabilities, and software code, **Verify** enables full bi-directional traceability and enables changes to the requirements documentation to generate code changes on the fly. As such, models or code artifacts are no longer "throw-away".

Simulate™ brings systems level inputs into the AWESUM tool chain, by adding a high fidelity virtual environment with outside stimulation from physical or simulated hardware. **Simulate** enables virtual operation of aircraft devices and platforms within their concept of operations, with real-time feedback through military line replace unit (LRU) equipment e.g., military communications. By basing all modeled factors on the target interfaces (i.e., ICD) and a platform concept of operations, Simulate validates and verifies intended operations of cyber-physical systems before expensive, real-world flight-testing occurs.

While wide experience with the use of simulation products used for obtaining certification under DO-178C/DO-331, remains unclear, it is believed that the acceptance of the use of model-derived simulation results for specifications and design acceptance will be a function of the fidelity of the models(s) (e.g., the system, its operating host/target operating system, flight models, and environment). **Simulate™** brings these components together to represent an as-built system and model, and simulates operational characteristics of platforms, devices and their interactions with an environment. This area of study, SAVI – Systems Architecture Virtual Integration [22], remains to become an exciting development in systems engineering, as the objectives of SAVI become a reality.

Qualify™ is an integrated system for producing and managing airworthiness lifecycle documentation (CDRLs and DIDs) required to satisfy all DO-178C applicant objectives. **Qualify™** automatically generates and allows for artifact engineering [TES Patent pending] of the software aspects of airborne system documentation required for airworthiness qualifications efforts applicable to airworthiness regulations, AR 70-62, AC 20-115C, DO-178C/DO-331, DO-278A, and the reuse guidelines of AC 20-148. In addition **Qualify™** is designed to the produce the lifecycle documentation required for FACE™ conformance and verification efforts.