# Model-based Tools designed for the FACE™ Technical Standard, Editions 3.0 & 2.1

**Stephen M. Simi**
**Tucson Embedded Systems (TES)**
**5620 N. Kolb Road, Suite 160**
**Tucson, AZ 85750**
**StephenS@TucsonEmbedded.com**

**Joe Uidenich**
**Raytheon Missile Systems**
**1151 E Hermans Rd. TU/M09/1**
**Tucson, AZ 85756**
**Joseph_M_Uidenich@raytheon.com**

**Sean P. Mulholland**
**Tucson Embedded Systems (TES)**
**5620 N. Kolb Road, Suite 160**
**Tucson, AZ 85750**
**Sean@TucsonEmbedded.com**

**Dr. James Head**
**Raytheon Missile Systems**
**1151 E Hermans Rd. TU/M09/1**
**Tucson, AZ 85756**
**James.N.Head@raytheon.com**

*Abstract* — **The promise of model-based systems engineering (MBSE), as described by DO-178C's supplement, DO-330 [1, 2] is with a sufficiently described system and software model, one should be able to auto-generate system's control software, testing, and lifecycle documentation. If aligned to a Modular Open Systems Approach (MOSA), like the Future Airborne Capability Environment (FACE) Technical Standard [3], and if aligned to Military Aircraft Airworthiness Qualification efforts, the lifecycle artifacts can be used and reused across a fleet of dissimilar aircraft systems, enhancing aircraft capabilities across the battlespace [4].**

**The Open Group FACE Consortium [ https://www.opengroup.org/face ] has long requested metrics regarding time savings and level of effort (LOE) using the Modular Open Systems Approach described by the FACE Approach. This paper presents three (3) working use cases of using the TES-SAVi AWESUM® MBSE tool suite converting FACE Technical Standard data models.**

**AWESUM® now has the capability to convert software developed to the FACE Technical Standard from Standard, Editions 2.x to 3, up to the interface validation process. Designed as a complete lifecycle tools suite, AWESUM® has the ability to address the complete lifecycle objectives described by DO-178C, support software aligned to the FACE Technical Standard, and support Military Airworthiness Qualification processes [5].**

**The use cases reported within this paper include the conversion of the BALSA (Basic Avionics Lightweight Source Archetype) User Supplied Model (USM) v2 with ~100 data elements to USM v3; secondly, the v2 to v3 conversion of a US Army Small Business Innovation Research (SBIR) topic requesting common reusable FACE development efforts, namely the Army Common Engine FADEC Interface (CEFI) FACE component, which was intentionally designed to leverage BALSA as its starting point for design; and thirdly a sizable real-world application, the conversion of a Raytheon Missile Systems' (RMS) program with ~15,000 data elements. This third product is a FACE Domain Specific Data Model (DSDM) awarded FACE Conformance Certification in April 2019 to FACE Technical Standard, Edition 2.1. This DSDM is based on the Unmanned Aircraft System (UAS) Control Segment (UCS) Version 3.4 [6]. This paper records the efficiencies of MBSE tools applied to FACE Technical Standard development efforts, lessons learned, and metrics on level of effort (LOE) saved. Should the products be ported and reused across a fleet of dissimilar aircraft platforms, the reuse efficiencies further increase.**

978-1-7821-2734-7/20/$31.00 ©2020 IEEE

## Table of Contents



## 1. Introduction

A typical and recurring programmatic question heard in the DoD military aviation circle is *why use model-based systems engineering (MBSE) tools and processes*? And *what are the benefits of MBSE*? The answers are summarized in the list below. *Because MBSE:*

- Speeds product development -- ties in contributory roles for all Stakeholders into the life cycle management command (LCMC) process
- Specifies data sufficient to auto-generate product artifacts; *i.e.*, software code, software tests, corresponding lifecycle documentation, and bi-directional tracing of high and low-level requirements to software tests and test results, used for conformance; *e.g.*, FACE

Certification, and DO-178C, and qualification processes; *e.g.*, Military AWR, AR-70-62 [5]

- Promotes cross-organizational developments of complex systems-of-systems
- Improves sustainment throughout the life cycle, development and post-deployment; *i.e.*, it aligns with OSA/MOSA approaches
- Opens vendor competition for best-of-breed with aviation 'plug-n-play' interfaces, and
- Is becoming DoD Directive Standard Practice in DoD acquisition programs

The bottom-line is MBSE is the preferred choice when planning to manage the complexity of next-generation systems-of-systems developments, integration, testing, qualification, and sustainment.

Defense Acquisition Systems defines a *System* as "a combination of two or more interrelated pieces of equipment (or sets) arranged in a functional package to perform an operational function or to satisfy a requirement. An open system uses modularity to provide 'plug-and-play' capabilities."

The TES-SAVi AWESUM® model-based tools suite, used to report results within this paper, is designed to support the complete lifecycle development and qualification of complex cyber physical systems (CPS) [4], systems that are aligned to:

- The FACE Technical Standard [3]
- Software reuse principles in accordance with AC 20-148 [7], and
- U.S. Army Military Airworthiness Qualification efforts (AR 70-62 [5]).

Tomorrow's military aircraft will be designed and operated as systems of systems operating on MOSA architectures. DoD Instruction, Information Technology Standards in the DoD [8] references DoD Directive 5000.1[9], which requires acquisition programs to employ a modular, open systems approach. The Open Systems Policy states,

> *"Acquisition programs shall be managed through the application of a systems engineering approach that optimizes total systems performance and minimized total ownership costs. A modular, open systems approach shall be employed, where feasible."*

The FACE Technical Standard describes how to develop and certify software for capability reuse within other FACE architectures.

## 2. Eco-System Tools for FACE Technical Standard, Editions 2.1 to 3.0 Data Model Translations and Code Generations

The TES-SAVi AWESUM® model-based tool suite [ https://tes-savi.com/awesum-product-suite/ ] was used to convert existing FACE data models, *e.g.*, BALSA, or develop and convert data models designed to the FACE Technical Standard [3]. The AWESUM conversion capabilities include the ability to:

- Export a Unit of Conformance (UoC) FACE data model or Domain Specific Data Model (DSDM) to FACE Standard, Editions 2.1 and/or 3.0 (soon 3.1) from one model
- Convert FACE data models from FACE Standard, Editions 2.1 to v3.0
- Upgrade the dependency of the UOC Supplied Model (USM) FACE Shared Data Model (SDM) from FACE Standard, Editions 2.1.x to 3.0.x
- Validate the Metamodel, the SDM, and Query & Template Languages
- And using the model, generate the software for FACE Technical Standard, Editions 2.1 and 3.0 using ecosystem tools, TES-SAVi AWESUM®[10] and RTI's Connext [11]

The data model software conversion process and results were demonstrated to the FACE Consortium during a FACE Consortium's Face-to-Face Member's meeting and the BITS event (BALSA Integration and Test Session) in St. Petersburg. Florida in April 2019; and at the US Air Force-sponsored FACE & SOSA Expo and Technical Interchange Meeting, in Dayton OH in September 2019. The results are described in the following three use cases:

1. BALSA, conversion of versions corresponding to FACE Technical Standard, Edition 2.1.3 to v3.0,
2. US Army SBIR Data Model to Common Controller, conversion from FACE Technical Standard, Editions 2.1.3 to 3.0. US Army Small Business Innovation Research (SBIR) topic A18-050 requested common reusable FACE development efforts, namely the Army Common Engine FADEC Interface (CEFI) FACE component, which was intentionally designed to include BALSA, and
3. Conversion of FACE Conformant Product, namely Raytheon CODE Domain Specific Data Model, from FACE Technical Standard, Editions 2.1.3 to 3.0. This product was certified FACE Conformant on 17 April 2019, Certificate #15555205.

*USE CASE 1 – BALSA (100 ELEMENTS)*

The US Army funded a FACE reference architecture and FACE software as an example for software developers to learn from and support more complex FACE development efforts. The suite of software is described in the Open Group Guide – FACE™ Software Supplier Getting Started Guide, Version 1.0 [12].

> [1]This Software Supplier Getting Started Guide (GSG) is written for Software Suppliers who are implementing the FACE Technical Standard. It is designed to be a navigational quick start guide providing access to sample conformant FACE software, developed FACE data models, and corresponding verification artifacts.
>
> The GSG includes a reference to a descriptive working example of BALSA. BALSA is the application being used as the "on-ramp" software example for the Getting Started Guide. BALSA serves as a working example for developers on how to implement the FACE Reference Architecture. The BALSA application is a collection of Units of Conformance (UoCs), which transform position information and aircraft identification to produce the Automatic Dependent Surveillance-Broadcast (ADS-B) messages required for all aircraft.
>
> BALSA is a working software example of applications aligned to the FACE Technical Standard executing in a FACE Reference Architecture (Figure1). It is a simplified version of an avionics suite comprised of basic avionics processes. BALSA consists of five separate FACE Portable Component Segment (PCS) and Platform Specific Services Segment (PSSS) Units of Portability (UoPs), which interact with the Transport Services Segment (TSS), Input/Output Services Segment (IOSS), and the Operating Systems Segment (OSS). These UoPs co-operate to combine position and altitude information with an aircraft ID and send it out "to the world" as ADS-B messages. The communication paths that connect the components in this example are all internal TSS connections, and an IOSS connection is used to write the ADS-B message to the "real world". [Note the TSS, IOSS, and OSS is denoted as TS, IO and OS in Figure 1].

The User Supplied Data Model (USM) for BALSA models the messages for ADS-B.

A FACE data model is the mechanism to describe all data into or out of a PCS or PSSS component in three key techniques: conceptual semantics, logical frames of reference, and message structure over the Transport Services layer. The distinction of each is fundamental to capturing the context of objects within the UoP domain and enabling interoperability of UoPs within the scope of a FACE solution.

[1] Text extracted from *FACE™ Software Supplier Getting Started Guide, Version 1.0, © 2017 The Open Group*. FACE™ is a trademark of The Open Group in the United States and other countries.

The FACE data model exists in an XMI format, with a ".face" extension. The XMI format is specified using the language notation prescribed by the FACE Metamodel. A Software Supplier must develop and document the message structure exchanged by a UoP. The model is intended to be a documented resource to aid integration efforts; also, through the use of tool sets, the data model can be used to generate the Transport Services interface and data type source code.

The BALSA Software, and corresponding FACE UoP Supplied Model (USM) for BALSA were used to test the software conversion from Editions 2.1.3 to 3.0 using the AWESUM® MBSE tools suite.

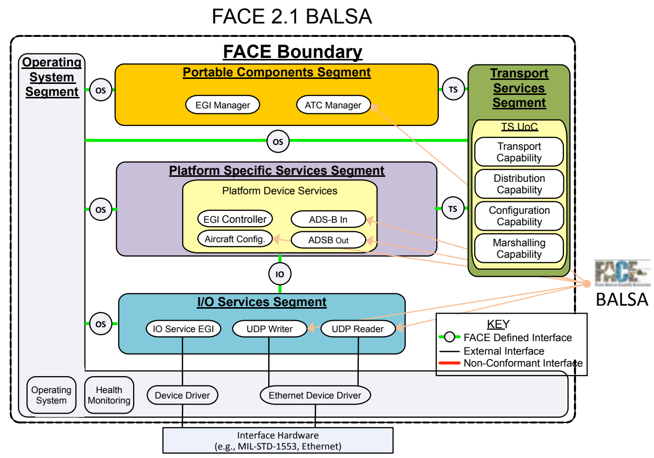


**Figure 1 - BALSA Architecture Diagram**

The data model standard conversion was performed live during a 25-minute demonstration at the FACE Consortium's member's meeting, BITS event, in front of 75 members out of 85 member organizations in St Petersburg, Florida at the FACE Face-to-Face member's meeting; and the demonstration was repeated during the US Air Force-sponsored FACE & SOSA Expo and Technical Interchange Meeting, in Dayton Ohio in September 2019. The conversion process took seconds. The resulting data model was shown to the Consortium (Figure 2)

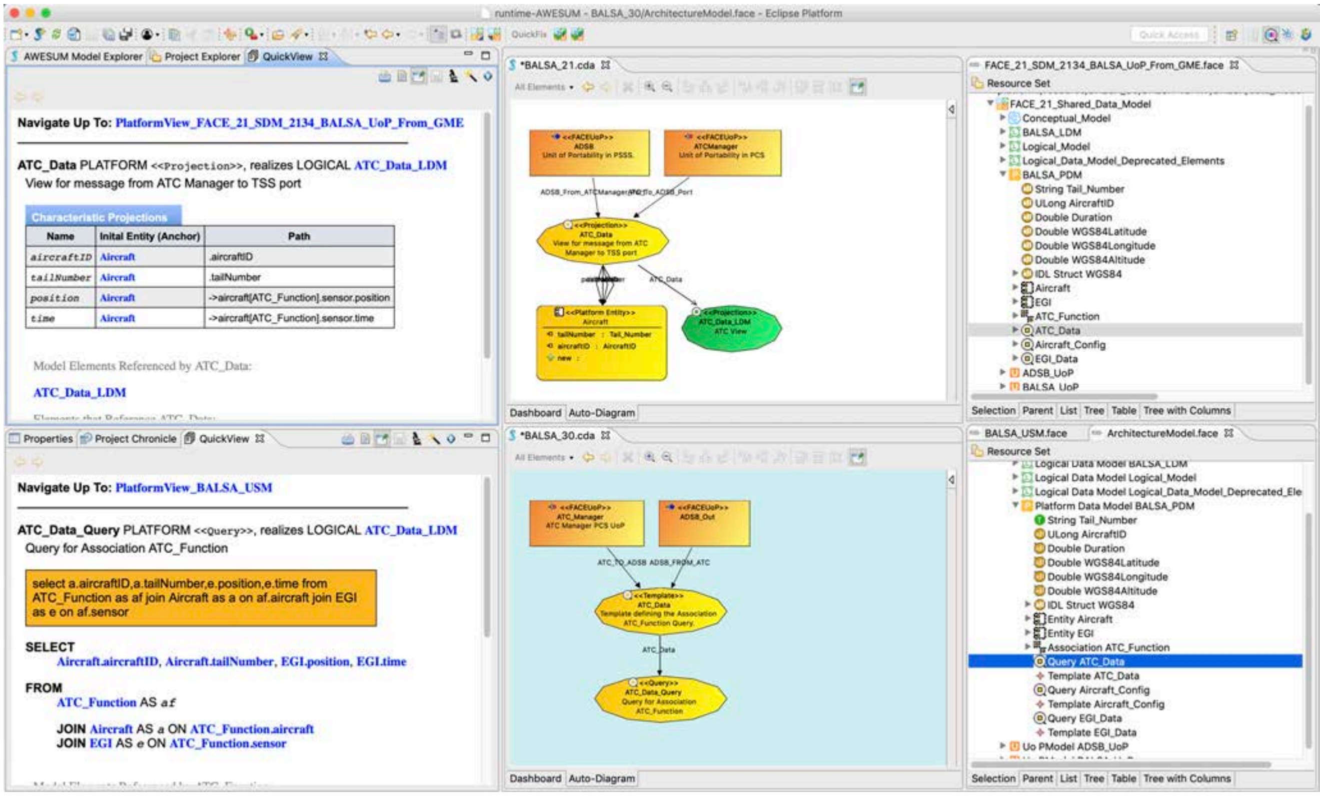


**Figure 2 – BALSA 100 element model converted form edition 2.1.3 to 3.0 using TES-SAVi AWESUM®**

***BALSA MBSE LEVEL OF EFFORT USING AWESUM® MBSE TOOL SUITE***

The level of efforts savings estimated for conversion using AWESUM® is:

- Converted BALSA UoPs, Views & DM from FACE Technical Standard, Editions 2.1 to 3.0
- 4 UoPs, 3 Views, < 100 DM Elements
- 8 – 40 hours saved

The BALSA software baseline was re-used as a starting point for the second use case development efforts. More specifically, Tucson Embedded Systems, Inc. (TES) was able to reuse most of previous BITS demonstrations effort [13]. We removed the FACE Conformant Honeywell EGI software, and added in new FACE data model and new FACE UoC software components specifically designed for US Army SBIR efforts. We ran the software on a Linux-based Raspberry PI.

***USE CASE 2 – SBIR FADEC (with a demonstration of operations)***

The second use case involves a SBIR topic A18-080, Common Engine Controller, delivered March 2019. The SBIR is based on modular open systems design and alignment to the FACE Technical Standard. Although the SBIR objective requested FACE Technical Standard Edition 3.0, the 3.0 eco-system wasn't available during the SBIR's 6-month period of performance (PoP). The SBIR's objective was to design an open architecture, construct a data model of the architecture messages, and achieve alignment to the FACE Technical Standard. TES sub-contracted to Management Sciences, Inc (MSI) in the last five (5) weeks of the Phase I PoP. Using AWESUM®, TES resources designed, developed, and used the FACE Conformance Test Suite (CTS) (FACE CTS) to verify the SBIR FADEC data model and software. TES then presented the software developed to the FACE Technical Standard to the Army Improved Turbine Engines Program Office on schedule.

Subsequent to the Army's SBIR presentation, TES then used AWESUM® to convert the data model from FACE Technical Standard, Edition 2.1 to 3.0 and auto-generated the FACE Transport Services Segment software [11]. The conversion and demonstration was performed and presented live during the FACE members meeting at the BITS event in April 2019 in St Petersburg, FL., and the demonstration was repeated during the US Air Force-sponsored FACE & SOSA Expo and Technical Interchange Meeting, in Dayton OH in September 2019.

The design and FACE development efforts involved:

- Designing the Architecture (Figure 3a) with multiple FACE units of conformance (UoC) software models (UoCs).
- Modeling the messages using AWESUM® (Figure 3b)
- Using FACE CTS to verify the model is aligned to the FACE Technical Standard (Figure 4).

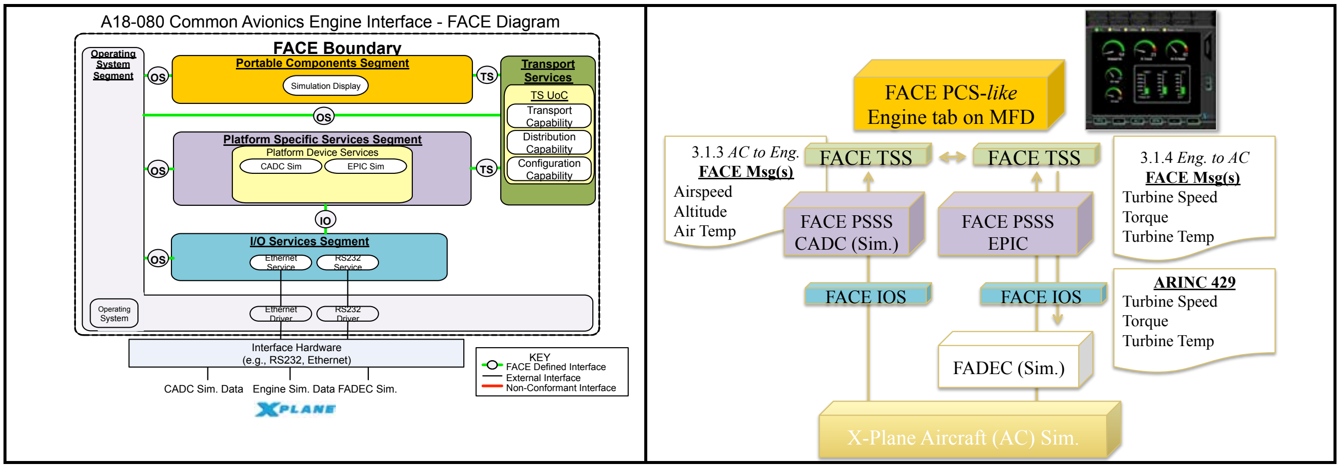


**Figure 3 - (a) FACE Architecture Diagram for Army SBIR, and (b) model of messages and FACE UoC software block diagram**

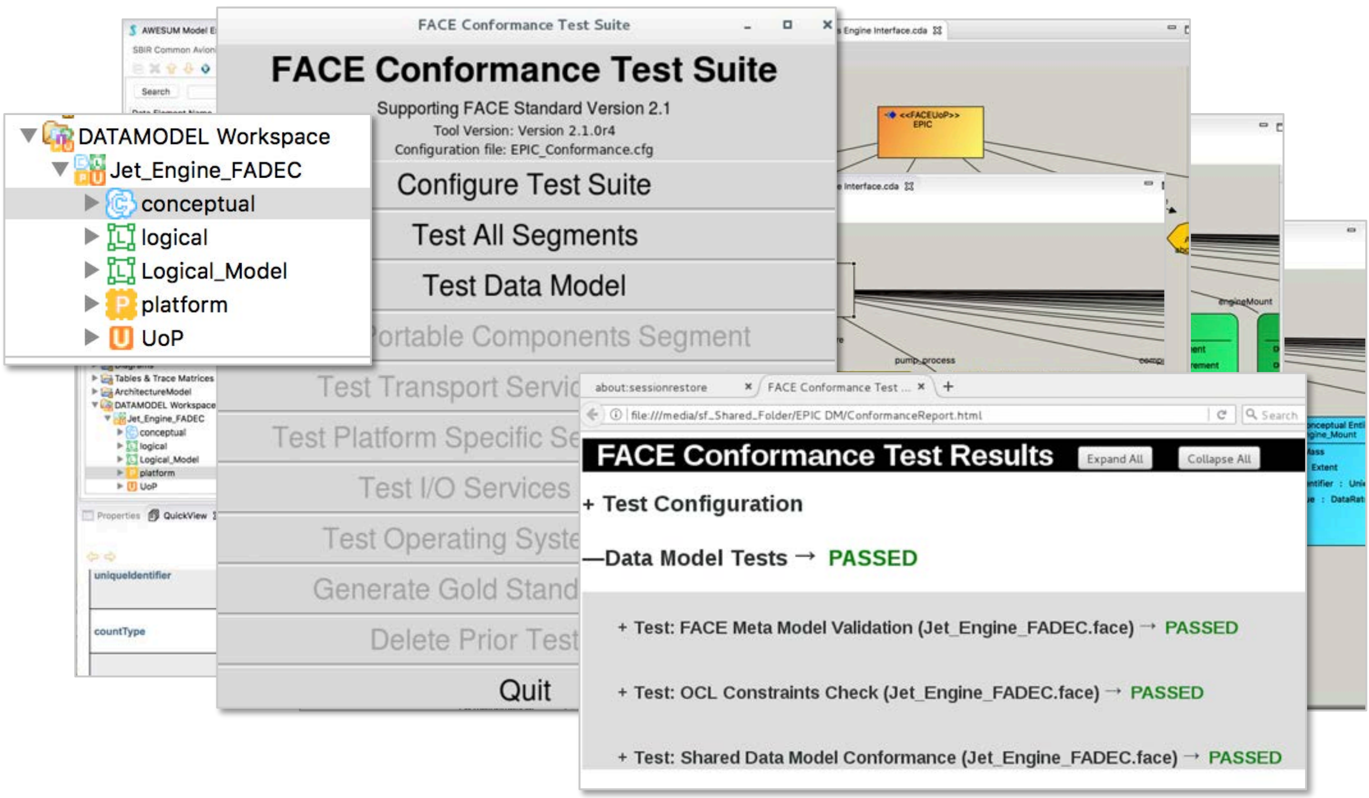


**Figure 4 - FACE Conformance Test Suite results on SBIR FADEC Jet Engine FACE Data Model**

The data model was designed as a jet engine domain specific data model (DSDM) with six (6) messages to interface with a common FADEC (Figure 3b). The six messages were altitude, air speed, air temperature, turbine speed, turbine temperature, and torque. TES developed the conceptual, logical, and measurement models. Once the model passed FACE CTS (Figure 4), we also used AWESUM® and auto-generated the transport services segment (TSS) software from the data model [11].

X-Plane flight X-simulation [ https://flight-simmer.com/xplane ] was integrated to drive aircraft position, altitude, air speed, and air temperature. The system was remotely demonstrated first as a formal development demonstration to the US Army SBIR program, then second as a capability demonstration to the FACE member's meeting BITS event. In both cases remote demonstrations were performed using WebEx teleconference software tied back to Tucson Embedded Systems, Inc. laboratory facilities [10]. Using AWESUM®, TES resources developed the four FACE data models, and verified the software using the

FACE CTS. The models and CTS results are illustrated (Figure 4).

***SBIR MBSE LEVEL OF EFFORT USING AWESUM® MBSE TOOL SUITE***

TES was able to perform the complete design (that is aligning data models and software to the FACE Technical Standard Edition 2.1.3) through demonstration in five weeks (two experienced systems engineers for a total of 7 person-weeks) using TES-SAVi AWESUM® model based tool suite and the TES Capability Driven Architecture (TES CDA) process [14] embedded within the AWESUM® tools suite.

Subsequent to the SBIR Phase I final report and Army demonstration, the AWESUM® model based tool suite introduced the conversion capability for FACE Technical Standard, Editions 2.1.3 to 3.0. During the FACE Consortium's Member's Meeting in St Petersburg, Florida during the scheduled BITS event (BALSA Integration and Test Session), TES performed the (this second) conversion of the SBIR data model live to the Consortium within seconds. The same demonstration was repeated during the US Air Force-sponsored FACE & SOSA Expo and Technical Interchange Meeting, in Dayton OH in September 2019.

It is estimated the tool suite saved approximately 15 person-weeks required to develop a FACE Technical Standard Edition 3.0 data model from scratch. It should be qualified that the TES resources are well versed with the FACE Technical Standard (all Editions), FACE data modeling, and FACE software development requirements, Figure 5b.

*USE CASE 3 – RMS CODE (FACE STANDARD Edition 2.1.3, being translated to 3.0 – and Lessons Learned Developing to the FACE Technical Standard in a Collaborative Environment*

The third use case originated from the Defense Advance Research Project Agency (DARPA) Collaborative Operations in Denied Environment (CODE) program, with a purpose to align the program's products to a Modular Open Systems Approach (MOSA) referred to as "Conformance to Standard". Raytheon Missile Systems (RMS) was provided the UCS data model as a starting point to this DARPA effort. RMS and TES began with conversion of the model, the Domain Specific Data Model (DSDM) based on the Unmanned System (UxS) Control Segment Version 3.4 (UCS 3.4) [6].

The purpose of the RMS MOSA "Conformance to Standard" demonstration was to take an RMS software product and put it through the process of adhering to an open systems standard. RMS selected the FACE Technical Standard as the open standard for conformance. The DARPA CODE program was already working towards creating an open architecture aligned to the FACE Technical Standard, making it a natural choice for taking a software component through the FACE Conformance Program. RMS performed model-based open systems design efforts on CODE modules.

The UCS model came with its own set of unique challenges, specifically, the UCS model preceded the FACE Technical Standard and hence is not necessarily aligned to FACE, and the CODE Domain Specific Data Model (DSDM) size exceeds 15,000 elements, which is extremely large when compared to other FACE Conformant data models.

Using this baseline and with the aid of the TES-SAVi AWESUM® tool suite, an integrated team of Raytheon and TES resources developed and obtained a FACE Conformance Certificate for the CODE Auto-Router Service Unit of Portability (UoP), September 2018, and a FACE Conformance Certificate for a DSDM, April 2019.

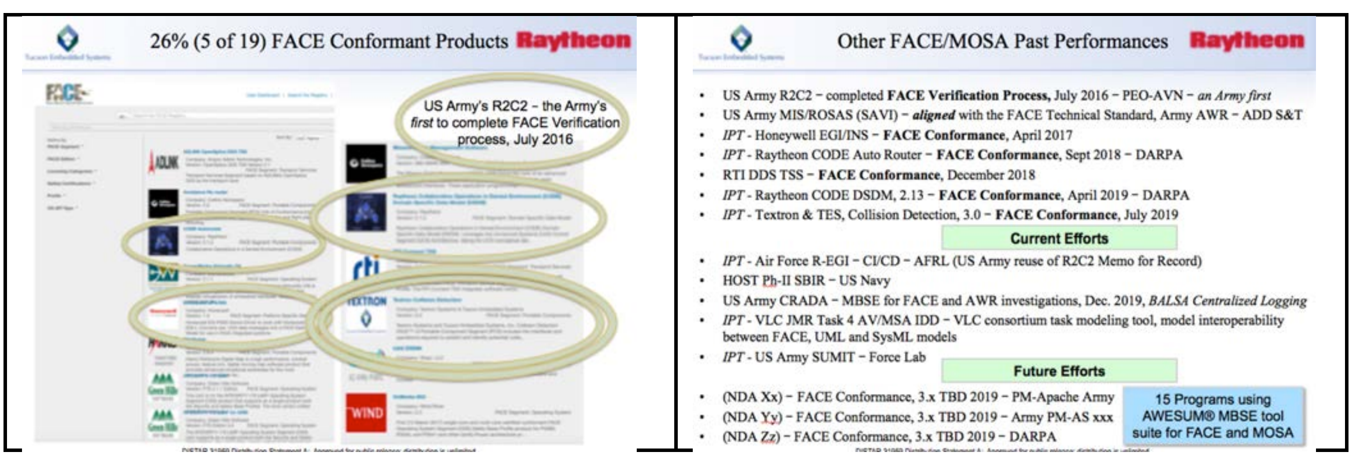


**Figure 5 – (a) 4(5) of 19 FACE Conformant Products, and (b) other FACE/MOSA Past Performances**

The sheer size and complexity highlights the benefits of using MBSE tools to manage complexity and save development efforts.

The lessons from these efforts are categorized into experiences using FACE Technical Standard, issues with USC data model, and issues and experiences with the FACE Verification and Conformance processes.

***Collaborative Operations in a Denied Environment (CODE)***

CODE is a series of 50+ software modules being developed as a DARPA program. DARPA requested that the contractors (Raytheon and Lockheed Martin) define and propose a particular open systems approach for the program and to support UCS as the communication protocol to the supervisory node. Raytheon chose to use the FACE Technical Standard and to start with an incorporation of the Unmanned Systems (UxS) Control Segment (UCS) model with the intent to reuse existing services and to create an autonomy domain for UCS [6]. DARPA and Raytheon CODE program both thought initially that a newly defined autonomy domain could be officially added to the UCS Standard. As will be described, unforeseen difficulties in aligning UCS to FACE may diminish the importance of UCS to all concerned.

The CODE Auto-Router Service Unit of Portability (UoP) was selected by the CODE team as a standalone service with a small number of interfaces that would serve as a simple example for demonstration purposes. The CODE Auto-Router Service supports generation of air vehicle routes around a supplied set of obstacles and evaluates routes for validity.

The CODE Auto-Router Service UoP was developed to be a FACE Portable Components Segment (PCS), which leverages its own set of FACE Technical Standards. PCS UoPs will only interface to other system components through the Transport Services (TS) interface to exchange data. The CODE Auto-Router Service UoP provides the following public interfaces:

- [Inbound Message] Generate Route Plan
- [Outbound Message] Route Plan Response
- [Inbound Message] Evaluate Route Plan
- [Outbound Message] Route Evaluation Response

The CODE Auto-Router Service UoP also requires the capability to request obstacle and planning region data from a Common Operating Picture (COP) Manager in order to plan and evaluate routes:

- [Outbound Message] Request COP Data
- [Inbound Message] COP Data Response

Internal to the CODE Auto-Router Service, Digital Terrain Elevation Data (DTED) is used to perform collision avoidance with terrain, directly reading from DTED files to obtain terrain data. This interface must also be conformant to the FACE Technical Standard, only using approved Operating System (OS) interfaces to read these data files.

***Tucson Embedded Systems (TES)***

TES was subcontracted to RMS as part of a MOSA Internal Research and Development (IRAD) demonstration to support RMS through the FACE Conformance Program. TES provided invaluable FACE expertise and guidance, supporting development of FACE conformant data models, conformance documentation artifacts, and performing conformance verification testing. The TES subsidiary, TES-SAVi, as a FACE approved Verification Authorities was subcontracted to perform FACE Verification Services to obtain certification.

TES performed a gap analysis for their products against the FACE Conformance Program on the CODE Auto-Router Service software. TES reported on gaps for coding, tests, and documentation artifacts to reach conformance. Next RMS contracted TES to support development efforts through the large learning curve associated with the product development efforts. Collaboratively, RMS and TES resources developed products prepared for FACE Conformance Program. Experiences gained can be reused on follow-on efforts when applied to Raytheon's product line of complex composable systems software modules.

***Issues with UCS and Data Model Requirements of the FACE Technical Standard***

The CODE Conceptual, Logical and Platform Data Models (CDM, LDM, and PDM respectively) are based upon the UCS Specification, v3.4. Seeded with the UCS data model – which is a message model – it was observed that UCS is not aligned well to data model guidance described by the FACE Technical Standard. The whole UCS specification model was converted into a FACE CDM, utilizing the FACE CDM Shared Data Model (SDM) observables. The CDM was then transformed into an LDM, again utilizing the FACE LDM SDM logical measurements, axes, respective measurement systems, and value type units. Finally, the LDM was transformed into a PDM with CODE defining the necessary PDM IDL structures and primitives for the associated LDM measurements and axes.

Conversion of the entire UCS Specification to align with the FACE Technical Standard was arduous and time consuming. The entire UCS specification was comprised of thousands of entity types and inheritance/generalization was used extensively. Fortunately, it was possible to create scripts to replace UCS observables with FACE CDM SDM observables, and remove inheritance/generalization by composing inherited attributes on the derived entities. This initial conversion effort required a month to accomplish with many iterations as errors were discovered and corrected.

For comparison, the US Army's Reusable Radio Control Component (R2C2) has completed the FACE Verification process, with assistance from the U.S Army AMRDEC FACE Verification Authority (VA), a FACE Consortium approved VA on 7/25/2016. A FACE Verification Statement is available upon request. This R2C2 data model has 800 entities and took ~6 man-months or ~1,080 hours LOE to develop with resources having significant FACE data model experience.

RMS observed that the UCS Specification and resulting CODE data model are so large, greater than 15,000 model entities, that other existing modeling tools did not accommodate this model well. Specifically RMS observed:

- The Vanderbilt ISIS: Sparx Enterprise Architect (EA) Tools for FACE™ Data Modeling tool required greater than four hours to export the entire CODE Data Model (CDM, LDM, PDM and UoP Model)
- EA has issues performing large-scale operations with larger data models -- DAO database errors occur. Several CODE scripts had to be refactored to sidestep issues with EA handling of the large CODE model.
- The TES-SAVi AWESUM® product suite was built from the ground-up to support data modeling aligned to the FACE Technical Standard and proved very helpful, especially for providing design and development guidance by enforcing the necessary modeling rules.

In converting the UCS Specification data model to be a FACE CDM, the UCS observable types were inadvertently converted into FACE entity types, none of which were unique from a CDM perspective. Each entity type contained a single "UniqueID elementID" attribute, meaning none of these CDM types were unique when compared to each other. Also, UCS contains several other observable extended types for Observable Specs, Requirements, Capabilities, and Errors among others, which, while converted correctly to a FACE entity type on the surface, did not meet the CDM uniqueness requirement.

Additionally, the CODE development team had created many "placeholder" entity types that had never been updated to be unique CDM entity types. These custom CODE UCS types did not meet the uniqueness requirement and were deleted as a result. TES attempted to remedy the CDM conflicts regarding uniqueness by writing scripts to auto-generate uniqueness of entity types by randomly adding unique observables until a unique composition of characteristics was achieved. This sometimes required multiple passes, populating several "junk" unique attributes per entity type.

Near the very end of the CODE Auto-Router UoP conformance effort, in order to reduce the amount of CODE data model messages, the RMS CODE team produced a greatly simplified data model. This reduced the number of entities from 3,000+ to just the required ~20 for the CODE Auto-Router Service. This eased TES's burden of correcting errors in the exported CODE data model and ultimately allowed us to complete the FACE Conformance Program successfully prior to the set deadline. This experience aided the team when addressing the 15,000 element DSDM.

***Resolution of Non-Unique CDM Entity Types for Full CODE DSDM***

The CODE CDM contained 138 categories on non-unique entity types, which had to be resolved in order to pass the FACE CTS. Given that CODE was starting from the existing UCS CDM, it was generally not possible to redesign these non-unique entity types to a single, more comprehensive or appropriate entity type whose characteristics could be projected to view types corresponding to each of the non-unique entity types. This would be the most appropriate course of action to define a robust FACE data model, but the CODE DSDM was coupled to the UCS data model which was intended to be kept unmodified as much as possible. Due to the utilization of the UCS data model as the starting point, other means for achieving CDM entity types uniqueness had to be developed.

For those uniqueness categories with eight or more characteristics, similarities in content and purpose between those non-unique entity types mostly allowed the entity types to be reduced to a single generic entity type that could be reused in place of the other types. An example of this would be a configuration state of a subsystem, where the configuration state entity could be used to both command and receive status for a given subsystem. In this case, two separate CDM entity types are not necessary with both able to be reduced to a single type. The single entity type can then either be realized by separate LDM entities (command and status types), or the single entity type can be realized as-is to the PDM level and then projected as two different view types or message ports on the UoP model.

Figure 6a shows a contrived example of two non-unique CDM entity types, Subsystem Configuration Command and Subsystem Configuration Status. Due to the related nature of these entity types and because their characteristic signatures are the same, the two entity types can be condensed into a single CDM entity type, Subsystem Configuration (as shown in Figure 6b). Once condensed into a single unique CDM entity type, it can be realized into a single unique LDM and PDM entity type (Figure 7a) or it can be realized into two separate LDM types, in turn each realized by a separate PDM type (Figure 7b).

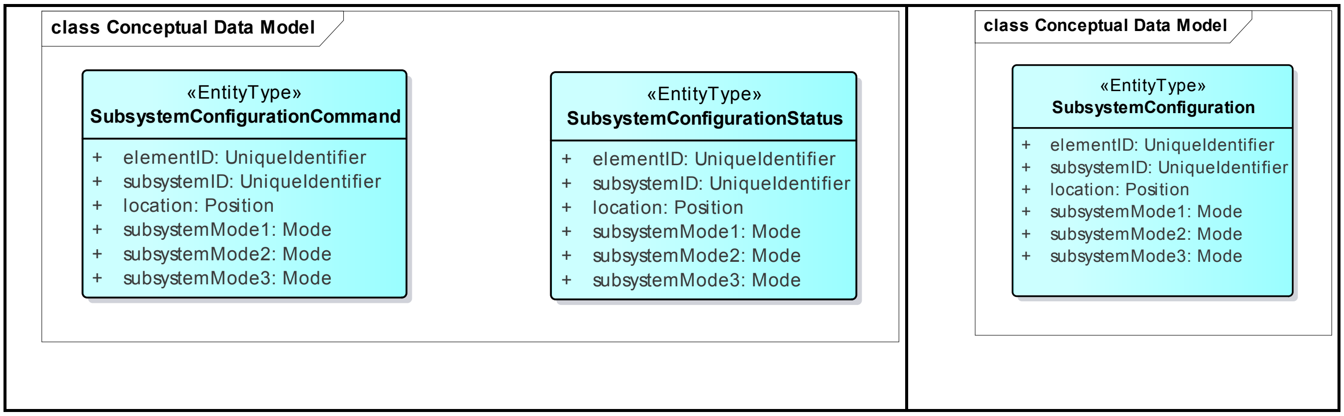


**Figure 6 – (a) Non-Unique CDM Entity Types for Subsystems Configuration Command and Status, and (b) Condensed to Single Entity Type**

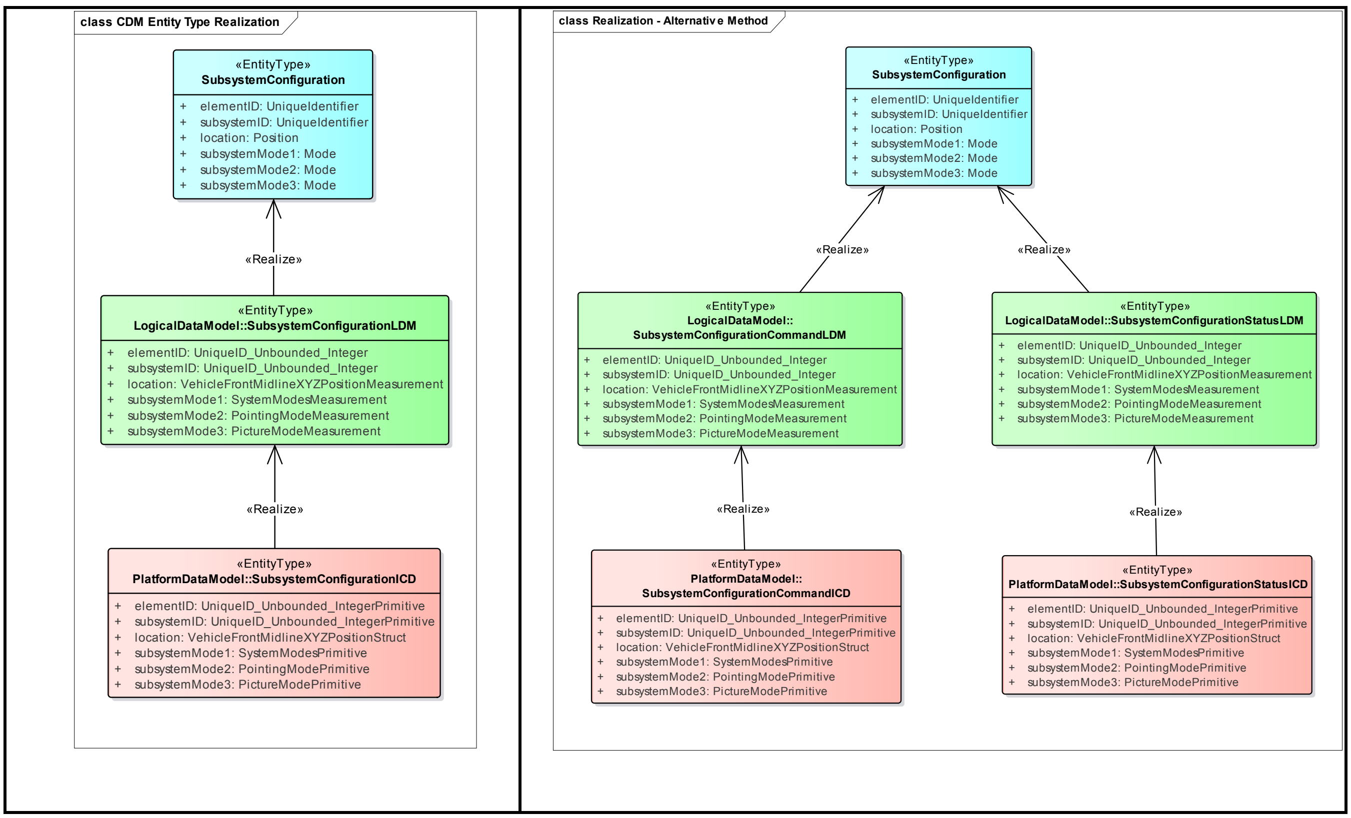


**Figure 7 – (a) Single LDM and PDM Realization Method: and (b) One-to-Many Realization of CDM to Separate LDM and PDM Entity Types**

For the UoC that may use these PDM entity types as view types for its associated message ports, the two view types (one for the "Command" and another for the "Status" message) can either both project the same common PDM entity type or each project the specific "Command" or "Status" PDM entity. These two view type options are shown in Figure 8.

The one-to-one realizations with single common CDM-LDM-PDM entity types (shown on left side Figures 7a and 8a) is the preferred method of reducing CDM entities to achieve uniqueness because of the reduced duplication of nearly identical entity types in the LDM and PDM. The one-to-many realization method is only preferable when there is an existing data model, UoP model and source code base that may be too extensive to change so significantly. Realizing the existing LDM and PDM entity types to a common CDM type avoids large changes to the model and source code while still achieving FACE conformance.

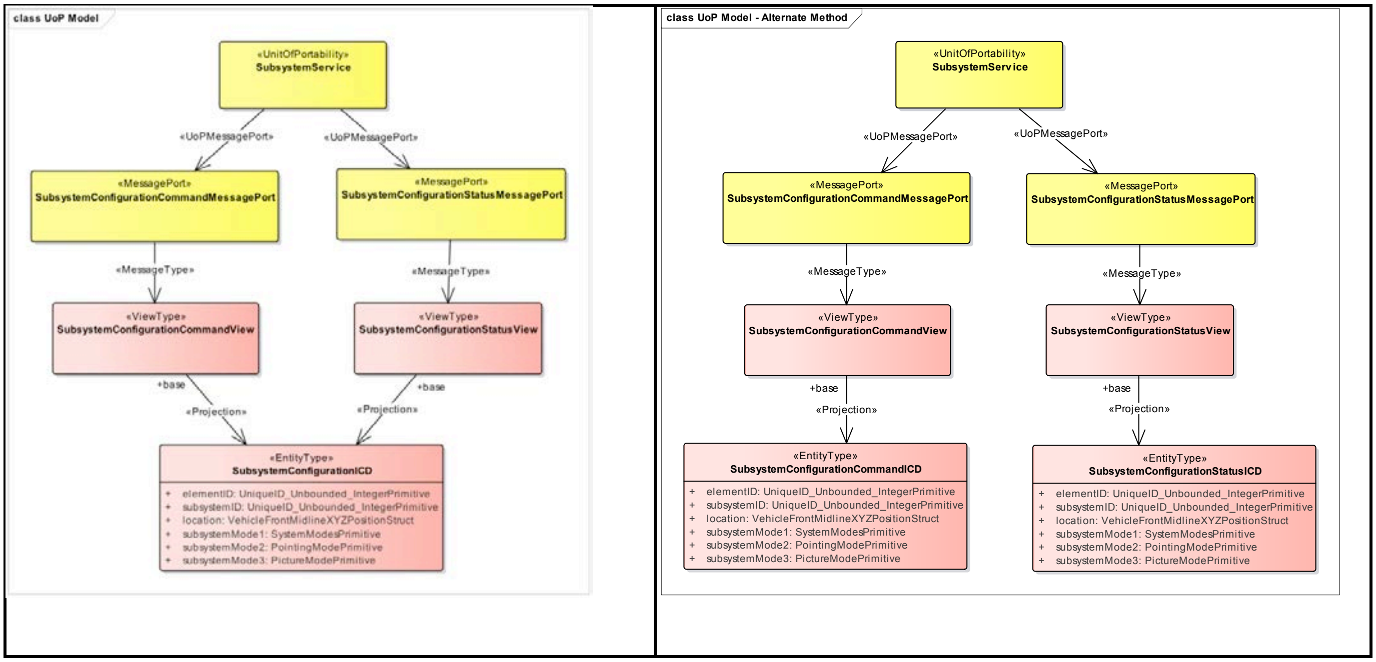


**Figure 8 – (a) Common PDM Entity Type Projected by both View Types: and (b) Each View Type Projects own PDM Entity Type**

***Observations***

The LDM SDM Measurements, Measurement Axes, Measurement Systems, and Measurement System Axes provided with the FACE shared data model contain a significant set of reference logical entities, but there are some oversights. Error and uncertainty types seem to be missing in general, particularly covariance measurements. Field of View (FoV) angle-based measurements are missing as well as simple course/heading direction measurements. The software developer end user cannot create measurement Systems and System Axes; the FACE Consortium must approve any proposed additions.

Suggestions for implementing missing measurements and measurement systems include:

- Bringing new measurement system requirements to the FACE Consortium Change Control Board (CCB) for review and adjudication. This PR/CR process may take months for incorporation into the FACE Technical Standard [3]. The recommended new measurement systems include: Error/uncertainty types, particularly covariance for position, velocity, orientation, position-velocity.
- Field of View (FOV) angles, horizontal and vertical axes.
- Vehicle course/heading measurements.
- Reusing existing measurement systems for creating new measurements by looking for systems that are "close enough" in number and type of measurement system axes. This is a work around rather than a recommended approach, but it will permit the developer to achieve a conformant data model.

The UCS FACE Incompatibilities and adaptation recommendation are described in the table below.

**Table 1 - UCS to FACE Incompatibility and Adaptation**

| UCS-FACE Incompatibility | FACE 2.1 Adaptation, Recommendation |
|---|---|
| Significant usage of inheritance / generalizations within UCS, contrary to recommended FACE guidance. | Composed generalized characteristics into sibling entities, thereby removing generalizations and inheritance. |
| Abundant non-unique conceptual entity types in UCS data model. | Simplified non-unique entity types to common/generic type whenever possible.<br><br>For overtly simple entities (e.g. containing only 1-2 unique identifiers), that were either composed or inherited/generalized, directly compose non-unique entity's characteristics on the encompassing entity type. |
| UCS logical measurement systems not available in FACE. | For error covariance measurements, developed generic covariance CDM entity type, realizing it as the various specialized LDM entity types (e.g. position covariance, velocity, orientation).<br>For remainder of missing FACE measurements, "close enough" measurements (i.e. same number of measurement axes, similar value type units) chosen as work-around.<br>Recommendation is to request addition of new measurements via the FACE Consortium Change Control Board (CCB). |
| Circular dependencies, references and links within UCS data model. | Replace circular dependencies with unique identifiers linking to the referenced instance. |

***Summary RMS' Lesson Learned***

There was a steep learning curve to understand FACE data model design and development. We found it extremely useful to have a FACE data model expert on hand for guidance. The FACE Approach has great depth in requirements regarding the FACE Data Model and UoP Model that require attention to fine detail, which could be easily missed when modeling in other FACE Eco-system tools. The FACE Reference Implementation Guide was helpful to a limited extent, providing examples for simple implementations but lacking more complex concepts. Without the guidance of FACE data model experts it would have been difficult to determine the necessary FACE data model implementation nuances required.

Inheritance/Generalizations are not supported in FACE Technical Standard, Edition 2.1. The FACE Domain Interoperability Working Group (DIOG) Guidance Subcommittee issued a White Paper in 2016 that recommended that the FACE v2.1 Generalization mechanism not be used, or used minimally for four primary reasons:

- Generalizations in Edition 2.1 are ambiguous and can cause issues during Characteristic Projection and code generation
- Generalizations in Edition 2.1 did not support Inheritance and caused additional modeling at the CDM, LDM and PDM
- FACE data models are based on Set Theory and therefore Generalizations are not necessary for FACE modeling, and
- FACE Technical Standard Edition 3.0 utilizes Specialization instead of Generalization.

The DIOG recommends the user compose Generalized Characteristics into Sibling Entities, as it is fully compatible with Editions 2.1, 3.0, and supports the basic tenets of Set Theory. For the CODE data model, Generalized Characteristics were composed into Sibling Entities per the FACE DIOG recommendation.

TES reported issues with our CODE data model realizations, particularly directionality, were not readily apparent in other FACE modeling tool suites, nor were these errors caught by XMI Export tool. These errors were not caught until TES imported the CODE data model XMI into their AWESUM® product suite.

The CODE team has since written EA-base CODE SDK plug-in verification tools for finding and reporting bad realization relationships between CDM, LDM and PDM entity types within the models. These tools verified the existence and direction of association link for all realization relationships between CDM, LDM, and PDM entity types. These verification tools were instrumental in finding missing, broken or reversed realization associations within the CODE data model in preparation for certifying the CODE DSDM.

We observed that the CODE data model was more ontology than a description of concepts within the system. TES suggested development of a more "real" CODE CDM to grow from going forward. We noted that the UCS specification is more of a message model and less a description of the concept of the various systems. TES examined the UCS specification and found it very difficult to leverage in a manner that aligns with the FACE Technical Standard.

***RMS CODE DSDM MBSE Level of Effort***

The level of efforts savings estimated for conversion using AWESUM® is:

- Converted (Conformant) RMS CODE DSDM from FACE Technical Standard, Edition 2.1.3 to 3.0

➢ 15,000 Data Model Elements

- 1,200 – 3,000 hours saved, i.e., 0.5 – 1.5 person-years saved.

***Next Steps***

The roadmap for our tooling is to apply MOSA principals, and develop reusable software capabilities aligned with open systems standards and airworthiness guidelines for applications on Future Vertical Lift (FVL) Family of Systems (FoS) next-generation aircraft systems both manned and unmanned and teaming. Working within the airworthy community and addressing fight-critical and safety-critical constraints positions these products to integrate and interoperate with other fielded-systems (*e.g.*, land vehicles, ships), which serves to integrate sensor capabilities across a battlespace, improve situational awareness within the aircraft and across a common operating picture (COP) for enhanced safety of flight, safely of operations, and improved mission success.

TES and RMS collaborations are likely to include developing systems capabilities for seven domains approximately 50 software modules to DO-178C Design Assurance Level B, and FACE Technical Standard edition 3.x Safety-extended.

## 3. SUMMARY – FACE ECOSYSTEMS - MODEL-BASED TOOLS APPLIED TO REAL-WORLD USE CASE DEVELOPMENT EFFORTS

This paper described and illustrated the benefits of model-based tools applied to the FACE Ecosystem with three (3) use cases. The theoretical promise of auto-generation from a sufficiently described Single Source of Truth (SSoT) model, described by DO-187C supplement DO-331, is observed, and the level of effort time and resource savings are qualified and quantified. Small to large data models are converted from one FACE Technical Standard, Edition 2.1 to 3.0, and savings or resources are quantified. The promise of reusability, maintainability, and lifecycle sustainability are realized using MOSA and model-based systems engineering tools and processes.

The day nears when model-based tools will help us design, develop, test, integrate, and qualify the next-generation of complex cyber physical systems and capabilities, those that will comprise our next-generation aircraft systems. Visit https://tes-savi.com/ for additional information on these TES-SAVi AWESUM® Product line MBSE products, and to obtain a list of related technical publications.

***Disclaimer***

The views and opinions expressed in this article are those of the authors and do not necessarily reflect the official policy or position of any agency of the U.S. government.


## ACKNOWLEDGEMENTS

Thanks to our counterparts and coworkers, the integrated product team resources that helped develop the products described in these three uses case examples, the developers of the TES-SAVi AWESUM® product line suite of MBSE tools. Namely Mr. William Tanner, TES, Mr. Ken Erickson, TES, Mr. Edward LeBouthillier, TES, Mr. Chi Dang, TES, Mr. Robert Kuan, TES; Dr. David Bossert, Raytheon, Mr. Joe Uidenich, Raytheon, Dr. James Head, Raytheon, Mr. Steve Thelin, Raytheon, Mr. David Vega Raytheon, Mr. Al Coit, Raytheon; Mr. Todd Peterson, MSI, Mr. Ken Blemel, MSI; Ms. Alicia Taylor, U.S. Army PEO Aviation, Mr. Christopher Crook, U.S. Army PEO Aviation, Mr. Joe Carter, U.S. Army PEO Aviation, Mr. Jason York, U.S. Army PEO Aviation, Mr. Matthew Sipe U.S. Army Futures Command, and Mr. Jesse Givens, U.S. Army PEO Aviation.

We also acknowledge the contributions of others, including other government, industry, and academia organizations as we reference only a small portion of their works in this paper, while knowing full well that those not specifically referenced were also consulted within either technical meetings, phone calls, emails, and working groups and other technical reference papers over many years – *thank you!*

## REFERENCES


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

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

[3] FACE - The Open Group (2017), FACE™ Technical Standard, Edition 3.0; retrieved from www.opengroup.org/library/c17c

[4] AHS 72nd Annual Forum – "Next-generation Model-Based Systems Engineering processes and Tools Supporting the Airworthiness efforts of Cyber Physical Systems (CPS)", Simi, Mulholland, Merrit, 2016.

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

[6] UCS-SPEC-MODEL Version 3.4.eap – "UAS Control Segment (UCS) Architecture Model", 27 Mar 2019.

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

[8] DoD Instruction – Information Technology Standards in the DoD, Number 8310.01, 2 Feb. 2015, Incorporating Change 1, 31 July 2017.

[9] DoD Directive 5000.1, "The Defense Acquisition System, May 2003.

[10] TES-SAVi AWESUM® Product line - Visit https://tes-savi.com/awesum-products/ for additional information on these MBSE products

[11] Army FACE™ TIM Paper – "Model-based Code Generation for the FACE™ Technical Standard – FACE Transport Services Segment (TSS) Type Specific Code and Configuration File", September 2018

[12] The Open Group FACE GSG - Open Group Guide FACE™ Software Supplier Getting Started Guide, Version 1.0, 2017.

[13] NAVAIR FACE™ TIM Paper – "FACE™ Cross-Integration Successes – Honeywell, RTI, Wind River, and Mercury Systems, 2017 June BITS Event – Model-based tools used for rapid FACE development and integration", October 2017

[14] Method and Apparatus for Interfacing with Multiple Objects using an Object Independent Interface Protocol," U.S. Patent No. 8,239,586, Tucson Embedded Systems, Inc., Capability Driven Architecture (TES CDA), 2009.


## BIOGRAPHY

***Stephen M. Simi****, TES - serves as TES Vice-President and Program Manager for Military Aviation programs, and TES-SAVI's Vice President. Stephen has 33 years of experience design and developing engineering and scientific applications, and managing multiple programs. Since 2010, Stephen has been active in the FACE Consortium's Integration WorkShop (IWS) – Vice co-Chair, and Outreach, Conformance, and Airworthiness sub-committees, and has exhibited at every FACE Technical Interchange Meeting (TIM). 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 authored technical publications and presented to the AHS, AOC, AIAA/IEEE societies, FACE Consortium, and MITRE on areas of software development, reusable systems, and advanced modeling and simulations of those systems. Stephen currently manages U.S. military programs for the Army Aviation, Air Force, and Vertical Lift Consortium. 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 TES, Stephen served as the Director of Software Development, and Director of Software Business Development at world-renown optics company Breault Research, served as a Technical Fellow at the MITRE Corporation; The Boeing Co. working on the International Space Station; was a Computer Science college professor, and served various other organizations designing, developing, and testing engineering and scientific applications over his 33-year technical career. He has authored over 20 technical publications.*

***Sean P. Mulholland****, TES - Co-founder of Tucson Embedded Systems, Inc. currently serves as TES CEO and President. Sean has 30 years of experience in software intensive system development, design, integration and testing, especially as it relates to mission critical and safety critical systems. Sean has designed and built several product lines that produced significant advancements in the areas of Geographic Information Systems, Military Ground Systems, Unmanned Ground Vehicles, Unmanned Aerial Vehicles, and Manned aircraft systems. Sean is a contributing author to the FACE™ technical reference* *architecture and has been active serving as a key resource in FACE Data Architecture development. Sean's current work is focusing on the development of a process and supporting tool suite for optimizing the system development of safe and secure systems for military and commercial systems. Sean has a B.S in Computer Science and Systems Design from the University of Texas at San Antonio.*

***Joe Uidenich, Raytheon** – serves as a software technical lead at Raytheon Mission Systems for Advanced Missile Systems programs. Joe has 18 years of experience in software architecture and development. He has designed and developed software system architectures of various mission planning systems and components for the US Air Force and US Navy. Additionally, he has designed and developed collaborative and autonomous open architecture frameworks and systems for the DARPA. Joe has a B.S. in Computer Science from Arizona State University.*


***Dr. James Head, Raytheon** - serves as engineering lead for Modular Open Systems Approach at Raytheon Missile Systems. Jim has 20 years of experience in technology development, innovation engineering, and systems engineering and architecture working programs in air, sea, land, and space domains. He was a AAAS Science and Technology Policy Fellow in the U.S. Department of State Office of Space and Advanced Technology and is a Lecturer in the Whiting School of Engineering, Johns Hopkins University. He has authored or co-authored over 40 technical publications. Jim holds a B.Sc. in Astronomy & Physics / Mathematics from Texas Christian University and a Ph.D. in Planetary Sciences from the University of Arizona.*