**2024 NDIA MICHIGAN CHAPTER
GROUND VEHICLE SYSTEMS ENGINEERING
AND TECHNOLOGY SYMPOSIUM
MOSA TECHNICAL SESSION
AUGUST 13-15, 2024 - NOVI, MICHIGAN**

LESSONS LEARNED FROM APPLYING THE FUTURE AIRBORNE CAPABILITY ENVIRONMENT (FACE$^{\text{TM}}$) TECHNICAL STANDARD IN MILITARY GROUND VEHICLE ARCHITECTURES

Leonard Elliott[1], Christopher Crook[2], Joel Sherrill, PhD [2]

[1]U.S. Army DEVCOM GVSC, Warren, MI
[2]On-Line Application Research (OAR), Huntsville, AL

ABSTRACT

Over the past decade, in an effort to reduce system lifecycle costs and accelerate fielding of capabilities, military ground vehicle projects and programs have engaged in various initiatives to incorporate Modular Open System Approach (MOSA) principles into their electronics and software architectures. There are multiple strategies for increasing the modularity and openness of emerging vehicle electronics. Each strategy/approach targets a particular set of key interfaces and prioritizes one or more business objectives. Under the PEO-Ground Combat Systems Common Infrastructure Architecture (GCIA), adoption of the Future Airborne Capability Environment (FACE^{TM}) Technical Standard has been the primary approach for standardizing software interfaces. This adoption was motivated by the same software portability/modularity and vendor-independence objectives that led to the creation of the FACE Technical Standard for aerospace applications. This paper presents the status of efforts towards leveraging the FACE Technical Standard to build software under the Enhanced Vetronics (E-Vetronics) project, as well as the benefits and drawbacks that have been encountered during the process.

Citation: L. Elliott, C. Crook, J. Sherrill, “Lessons Learned from Applying the FACE Technical Standard in Military Ground Vehicle Architectures,” In *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 13-15, 2024.

1. INTRODUCTION

With the enactment of the United Stated Code (USC) Title 10 MOSA requirement for major defense acquisition programs, interest in creating modular and open architectures and designs, as well as proving conformance has surged. U.S. Army ground vehicle programs are using GCIA to achieve MOSA requirements. Foundational principles of MOSA, as described by Open Systems Joint Task Force (OSJTF) [1], include:

1. Modular Design
2. Identify Key Interface
3. Use Open-Standards
4. Establish Supporting Ecosystem
5. Verify Conformance

For the most part, the body of MOSA-related literature consists of an abstract narrative of how adhering to these principles will result in cheaper, faster development of systems. Detailed guidance and published lessons learned about how to achieve specific business objectives in a particular technical or application domain is very limited. Additionally, there is a lack of clarity about how various MOSA technical approaches relate to each other, explored by Elliott, et al [2], and whether an approach developed for a given domain (e.g. aerospace) can be leveraged in adjacent application domain, explored by Snyder, et al [3].

This paper attempts to add clarity and evidence to the MOSA body of knowledge by describing recent efforts to apply the FACE Technical Standard to increase software modularity, portability, and vendor-independence in U.S. Army ground vehicle research projects.

2. RELATION TO GCIA

GCIA is the U.S. Army's primary means of achieving MOSA in new ground vehicle programs. GCIA is a collection of standards, mechanisms, and specifications that seek to achieve network, hardware, and software modularity and interoperability. Figure 1 shows the major building blocks of GCIA,

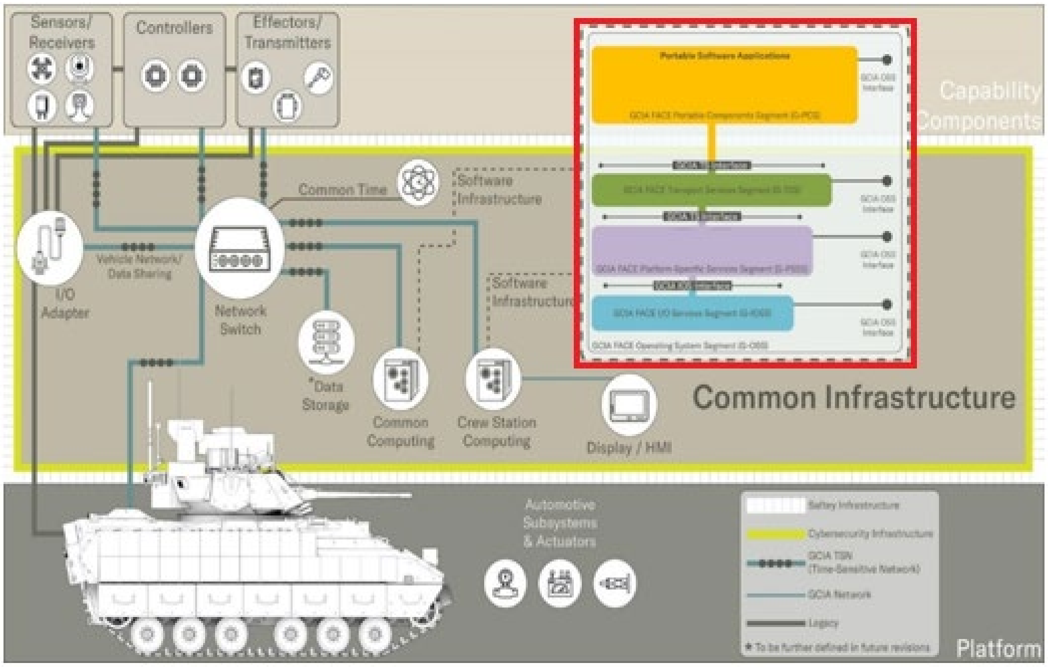


Figure 1: GCIA Conceptual Diagram [3] (Software Infrastructure Highlighted)

and much interest is paid to the hardware elements which include Network, I/O Adapters, Crew station, and Common Compute. These are tangible items that the ground vehicle community has a long history of trying to make more modular and interoperable. Vendors have a relatively mature business model and understanding of how to compete and provide value in the hardware space.

A less often discussed aspect is the GCIA software infrastructure which is highlighted in red in Figure 1. The ground vehicle community has only recently begun to pursue modular, reusable, mission/safety critical, and vendor neutral software. Senior leaders, however, understand the central importance of software in today's vehicles and how GCIA's use of the FACE Technical Standard supports key business objectives[5]. The following sections will discuss practical issues identified related to applying the FACE Technical Standard as part of the DEVCOM-GVSC-VEA Enhanced Vetronics (E-Vetronics) Science and Technology (S&T) effort.

3. OTHER SOFTWARE STANDARDS CONSIDERED

FACE is a software open-standard that is leveraged by architectures such as GCIA. There are other software standards that are relevant and were considered as candidates for meeting the business requirements of GCIA. These standards include AUTOSAR and ROS, which were examined and deemed unsuitable for licensing reasons, applicability, and/or safety-pedigree. AUTOSAR has structural similarities in its architectural approach to those of the FACE Reference Architecture. However, it requires a paid membership, whereas the FACE Technical Standard is open and free. ROS is specific to the robotics domain and its use of STL and Boost libraries would present challenges in reusing its infrastructure in a safety-critical environment. The Technical Standard, on the other hand, is consensus based with the buy-in of three DoD entities, plus that of most major defense contractors. It is written with safety-pedigree in mind while also fostering portability and reusability.

4. USE OF THE FACE TECHNICAL STANDARD IN E-VETRONICS

E-Vetronics is the U.S. Army's premier S&T effort for validating and incubating ground vehicle MOSA aligned network, computing, software standards, and components. Under E-Vetronics, various GCIA related software and hardware components are being prototyped and evaluated. As shown in Figure 2, the software infrastructure defined by the FACE Reference Architecture is a layered approach and provides modularity via Operating System Segment (OSS), Input/Output Services Segment (IOSS), Transport Services Segment (TSS), Platform Specific Services Segment (PSSS), and Portable Components Segment (PCS). The software services that are the subject of this paper provide access to common vehicle data such as position, orientation, Global Positioning Satellite (GPS), automotive, and other vetronics data. These services typically reside in the PSSS or PCS and are made available on the network via the Application Programming Interfaces (APIs) provided by the FACE Reference Architecture as shown in Figure 2. It should be noted that the FACE Reference Architecture shown in Figure 2 serves as the foundation for the GCIA Computing Environment shown in Figure 1.

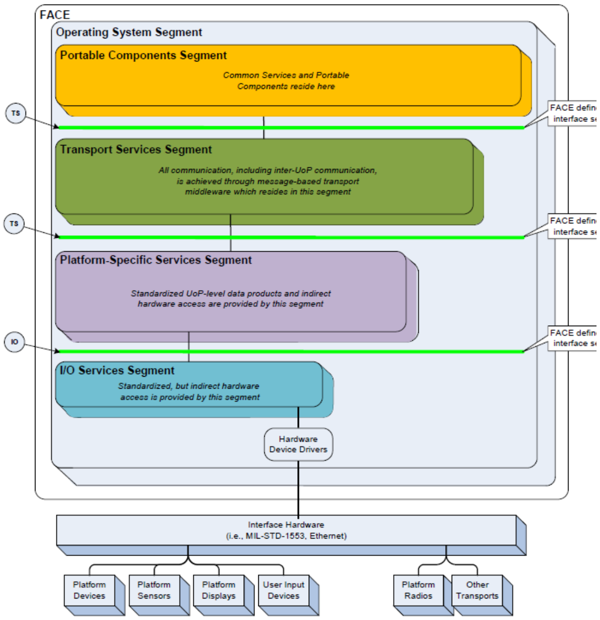


Figure 2: FACE Reference Architecture [6]

The remainder of the paper will discuss various aspects of using the FACE Technical Standard within the E-Vetronics S&T effort. When specific software module and service details are provided, an over-simplified example, Hatch Status Service, which provides information about whether a vehicle hatch is open or closed, will be discussed.

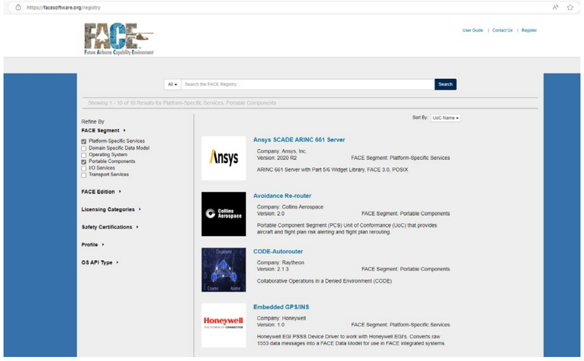
4.1. Adopting FACE Requirements

Here we discuss aspects of adopting FACE requirements for the E-Vetronics S&T effort. The FACE Technical Standard is intended, but not required for use in safety-critical environments with air vehicles being the original target environment. Not all the aspects of the airborne domain are applicable to a ground vehicle environment, such as ARINC-653. The FACE Technical Standard is large with optional requirements and multiple profiles. Having been written to address avionics systems of varying criticality, the selection of these options and use on safety-critical ground-based systems is not immediately obvious. However, the foundations of the FACE Technical Standard's profiles and optional requirements do reflect best practices that cross domains. The GCIA specification itself provides a selection of options and tailoring of the FACE Technical Standard for ground-based systems however there is still a learning curve for engineers to understand, implement, and/or develop software in accordance with the selected profiles/options. For instance, many software developers were unaware of the restrictions associated with developing safety-critical software, so realizing that limitations such as the lack of C++ Standard Template Library (STL) support in the Safety-Extended C++ Programming Language capability set were very surprising.

In summary, the FACE Technical standard does not always paint a clear picture on how certain requirements for FACE Units of Conformance (UoC) are meant to be interpreted and so education and guidance on the FACE Technical Standard from a third party was tremendously beneficial.

4.2. Procuring FACE Software

One of the business objectives of the FACE Approach is to lower costs by reducing duplicate development of software through portability and reuse [7]. U.S. Army ground vehicles programs obviously want to reduce costs and seek to reuse software whenever appropriate. In the E-Vetronics project, we wanted to try to assess how many existing FACE Units of Conformance (UoC) were available and appropriate for reuse in U.S. Army ground vehicle programs. The FACE Software Registry is the place where information about reusable software components is published. The registry, shown in Figure 3, is a site which allows you to search for readily available software UoCs.



Showing 1 - 10 of 10 Results for Platform-Specific Services, Portable Components

Ansys SCADE ARINC 661 Server
Company: Ansys, Inc.
Version: 2020 R2
FACE Segment: Platform-Specific Services
ARINC 661 Server with Part 5/6 Widget Library, FACE 3.0, POSIX

Avoidance Re-router
Company: Collins Aerospace
Version: 2.0
FACE Segment: Portable Components
Portable Component Segment (PCS) Unit of Conformance (UoC) that provides aircraft and flight plan risk alerting and flight plan rerouting.

CODE-Autorouter
Company: Raytheon
Version: 2.1.3
FACE Segment: Portable Components
Collaborative Operations in a Denied Environment (CODE)

Embedded GPS/INS
Company: Honeywell
Version: 1.0
FACE Segment: Platform-Specific Services
Honeywell EGI PSSS Device Driver to work with Honeywell EGI's. Converts raw 1553 data messages into a FACE Data Model for use in FACE integrated systems.

Figure 3: FACE Software Registry [8]

The site allows you to search and filter on parameters such as the relevant FACE software segment (OSS, IOSS, TSS, PCS, PSSS), DO-178C Design Assurance Level (DAL) certification, license type, and operating-system profile, etc. Recently, the site has also listed Domain Specific Data Models (DSDM) that are available for reuse which is significant because data-modeling is an important and expensive part of many current projects and reuse of data models would drastically help reduce cost.

Table 1 shows the distribution of publicly listed FACE Conformant UoCs and the types of safety certifications that are advertised with DAL-A and DAL-D being respectively the most and least rigorous. Table 1 shows that OSS components are readily available and that the other segments do have components as well. Additionally, there are UoCs that have been developed that are not listed in this registry. For instance, on the E-Vetronics project the OAR-developed Reusable, Reconfigurable Avionics Data Exchange (RRADE) TSS software is being used. RRADE is Government Purpose Rights (GPR). The PCS and PSSS represent the segments where GCIA services would most likely be hosted, however many of the components in the FACE Registry relate to flight-control software and domains that do not apply to those of ground vehicles. However, there 'is' a GPS Receiver UoC that is listed in the FACE Registry, though inquiries with the vendor have not been returned. There is additionally a SysLog UoP that might be used for GCIA data-logging and discussions are being held to determine the suitability of that UoC for reuse. The lack of PCS and PSSS components in the FACE Registry, it was difficult to know what FACE Software existed and whether it could be reused to create GCIA services. There is the possibility that FACE Software has been developed by companies and not marketed as standalone UoCs but reused within a given company's internal projects.

Table 1. FACE Software Registry UoC Roll-Up

Segment	Number	DO-178 DAL Range
OSS	20	B, C, D
IOSS	9	C, D
TSS	5	C, D
PSSS	3	D
PCS	7	C, D
DSDM	2	N/A

4.3. Integrating FACE Software

The portability and modularity objectives of the FACE Technical Standard should make integrating software relatively easy. E-Vetronics has so far contracted the development of various UoCs, such as the GCIA State Management Service (SMS) and Entity Index (EI), which were developed by OAR Corporation. Those UoCs were distributed to teams outside of VEA and feedback was received indicating that the software was easy to use and integrate. The author also has personal experience on one other ground vehicle subsystem project where the FACE Technical Standard was used. On that effort it was observed and reported that integration challenges were reduced by using the FACE Approach. These data points provide evidence that the FACE Approach enables software reuse and reduces integration burden in ground-vehicle applications.

4.4. FACE Impact on Software Engineering and Development

Both the FACE Technical Standard and GCIA make heavy use of data modeling, which is a paradigm shift for the military ground vehicle community and represents a significant learning curve. There is an additional challenge in that GCIA adopts the FACE Technical Standard as the primary mechanism for achieving software modularity and portability, yet GCIA adopts a different overarching data modeling approach. The GCIA data modeling approach and FACE Data Model Language do not align one-to-one and there is confusion over how to bridge those differences. For the purposes of E-Vetronics, we began to model the relevant software interfaces using MagicDraw and the Modeling Tools for FACE (MTF) Plug-In that was developed by Vanderbilt University with U.S. Army funding [9]. This allowed us to generate the datatype-specific source code required to integrate FACE UoCs with a FACE TSS. A version of the Magic Draw MTF Plug-In is freely available and Distribution A. It contains the metamodels and constraints defined by the FACE Technical Standard and the Universal Domain Description Language (UDDL) Standard [10] from which our data interfaces could be modeled. Figure 4 shows how the FACE Shared Data Model (SDM) expresses vehicle-centered azimuth at the conceptual level. Many of these types are common to ground and air vehicles and so can be reused for E-Vetronics.

The primary point of variance in adopting the FACE Data Model Language to existing engineering processes pertained to the idea of modeling interfaces versus data connections that would be realized as interfaces. In the FACE Data Architecture, the notion of an interface is not a common element.

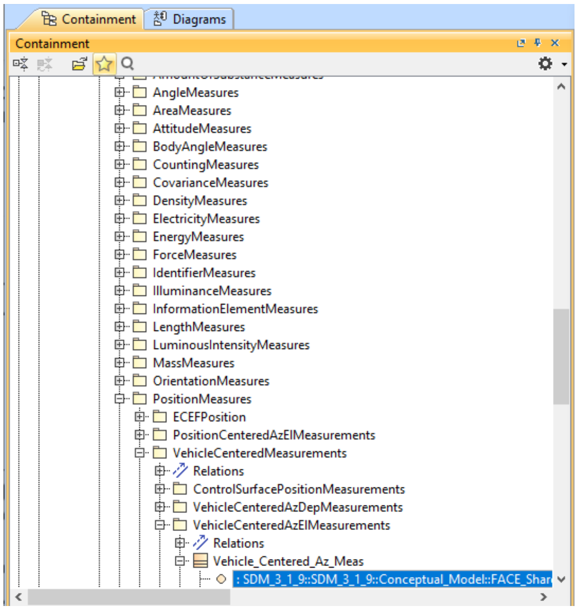


Figure 4: Magic Draw FACE Plug-In SDM

The focus of FACE Data Modeling is on data semantics and its relationships. The Transport Services (TS) Typed Interface, which provides message interaction between FACE UoCs in the PCS/PSSS, is instantiated based on the modeled message within a FACE UoP Supplied Model (USM). This was not readily understood, which involved a learning curve to understand that the point of a FACE Data Model is not model the software component, but the data contract when it came to a software components inputs and outputs. The overall point in this approach is to address data semantic variance and serve as a source of truth to define a FACE UoC and its data needs when integrating with a FACE TSS.

5. CONCLUSION

Adoption of the FACE Technical Standard does appear to ease integration challenges by ensuring modularity, portability, and conformance. There is a large initial investment in terms of learning about the FACE Technical Standard and with modeling of domain-specific software interfaces.

6. FUTURE WORK

Future work would include encouraging unregistered UoCs to be put into the FACE software registry. Further articulating and working out the relationship between GCIA data-modeling and FACE Data Modeling is on the horizon for the two to interoperate at the model-based systems engineering (MBSE) level.

7. REFERENCES

[1] Open Systems Joint Task Force (OSJTF), "A Modular Open Systems Approach (MOSA) to Acquisition, Version 2.0", September 2004.

[2]L. Elliott, S. Jenkins, M. S. Moore, and H. Yee "Potential for VICTORY and FACE Alignment – Initial Exploration of Data Interoperability and Standards Compliance", proceedings of the Vehicle Electronics and Architecture (VEA) and Ground Systems Cyber Engineering (GSCE) Technical Session, 2019 NDIA Ground Vehicle Systems Engineering and Technology Symposium, August 2019.

[3]M. Snyder, C. Allport "Using FACE^{TM} Technical Standard Features to Address Interoperability Between Ground Vehicle Domain Open Standards," In Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium (GVSETS), NDIA, Novi, MI, Aug. 16-18, 2023.

[4]Michael Doran, Mark Russell, Leonard Elliott, "Deterministic & Modular Architecture for Embedded Vehicle Systems" In Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium (GVSETS), NDIA, Novi, MI, Aug. 16-18, 2023.

[5]Freedberg, S, "Bradley Replacement , OMFV, Will Live or Die by Software" [Online] https://breakingdefense.com/2023/01/bradley-replacement-omfv-will-live-or-die-by-software/

[6]FACE Technical Standard, Version 3.1, https://www.opengroup.org/face, July 2020.

[7]FACE Business Guide, Version 3.0, https://www.opengroup.org/face, October 2020.

[8]FACE Software Registry [Online] https://facesoftware.org/registry

[9]Vanderbilt University FACE Tools [Online] https://www.isis.vanderbilt.edu/FACE

[10]The Open Group, "Open Universal Domain Description Language (Open UDDL), Edition 1.0", July 2019.