THE Open GROUP

FACE™ Future Airborne Capability Environment

SOSA™ Sensor Open Systems Architecture

FACE Technical Overview

Christopher Crook
PEO AVN G10 Support
from DEVCOM AvMC

Huntsville Alabama
September 14, 2021

U.S.ARMY

PROGRAM EXECUTIVE OFFICE AVIATION

FACE™
Future Airborne Capability Environment

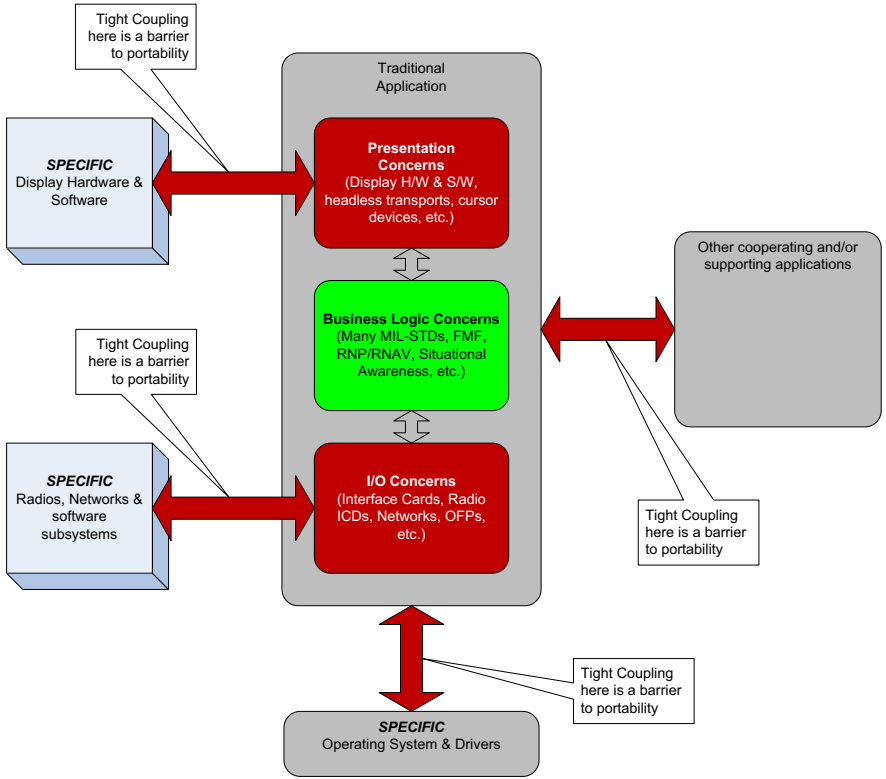
Agenda

- Why a standard?
- FACE Technical Standard
  - What does it define?
  - FACE Reference Architecture
  - FACE Data Architecture
  - What does it all mean?
- How do I use it?
- How do I start?

Why a Standard?

Main Technical Drivers:

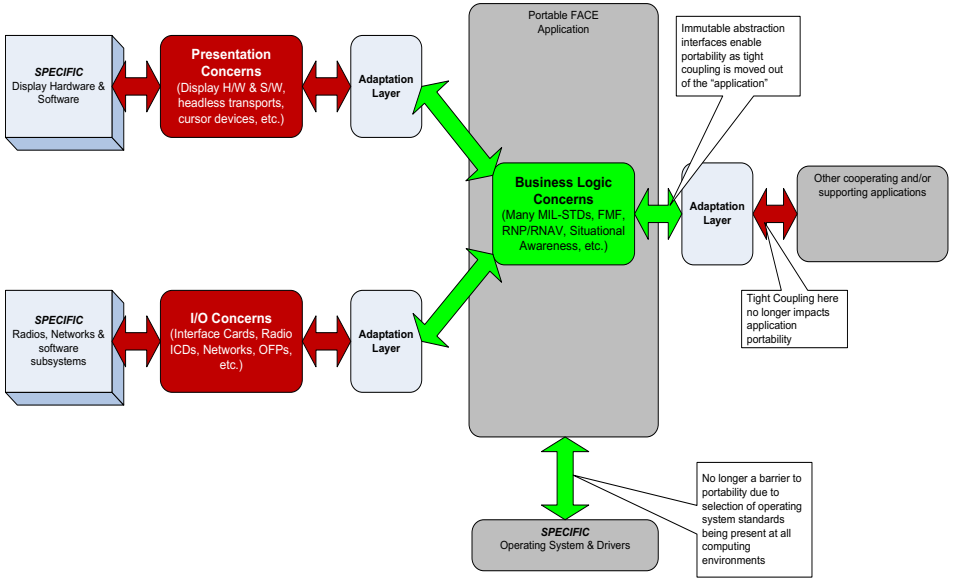
- Software Portability and Reuse
- Tightly coupled software
- Proprietary interfaces



Why a Standard?

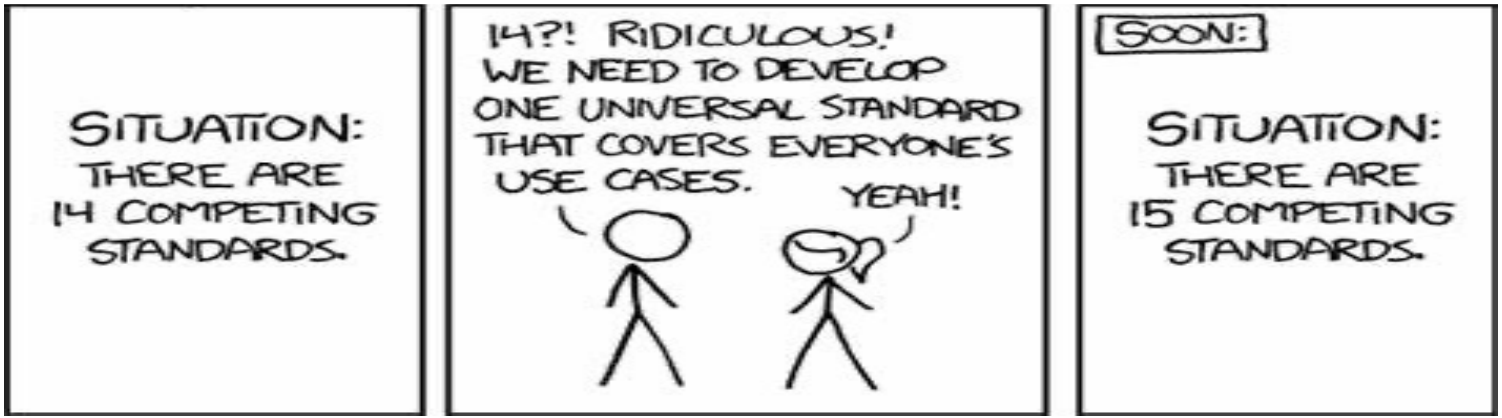
Technical Goals:

- Focus on modular design
- Define interfaces to abstract common tasks (e.g., data transport, graphics, OS-level functions)
- Define a reference architecture that uses open standards (e.g., POSIX, ARINC 653, OpenGL)
- Define requirements for conformance



Why a Standard?

What we wanted to avoid:



FACE Technical Standard

What does it define?

- A **Reference Architecture** that uses standardized interfaces and provides requirements for developing software components that will reside in architectural segments.
- A FACE Data Architecture for describing data and its semantics
- IDL definitions for **FACE Interfaces**
- Levels of criticality for conformance (Security, Safety, General Purpose)
- **Programming Language Mappings** from **IDL** to the following languages:
  - C
  - C++
  - Ada
  - Java

FACE Technical Standard

The **FACE Technical Standard** abstracts software capabilities into logical segments where variance occurs, referred to as '**Segments**'.

- Think of a 'Segment' as a group
- We want to group software based on:

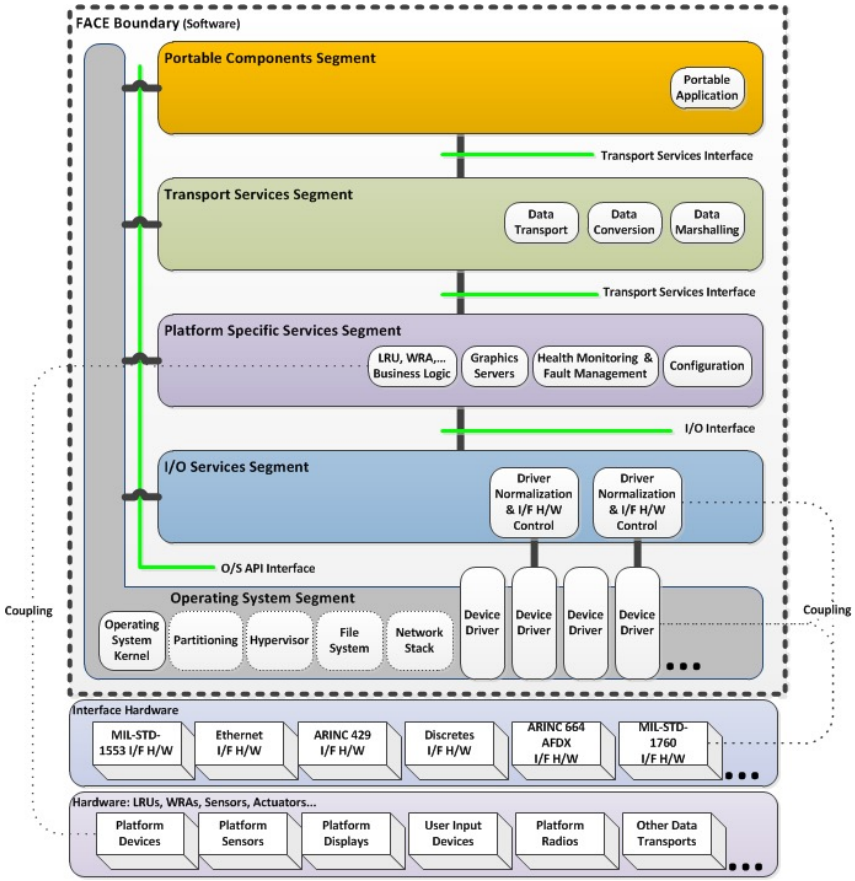
  "What it does" and "What it needs to communicate with"

- Segments are separated by defined interfaces

FACE Reference Architecture

Segments:

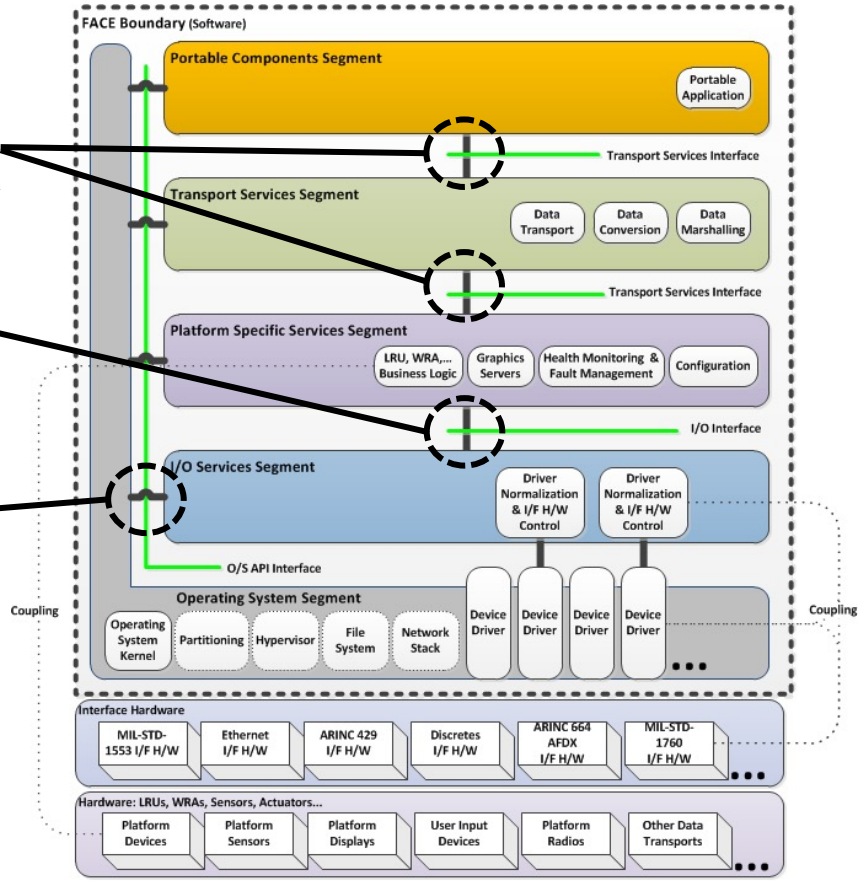
- **Portable Components** Segment (PCS):
  - Portable software (typically things that perform a function for the user)
- **Transport Services** Segment (TSS)
  - Software applications that move data
  - Examples: Pub/Sub implementations, POSIX socket implementations
- **Platform-Specific Services** Segment (PSSS)
  - ICD- or device-specific software, common services, and graphics
- **I/O Services** Segment (IOSS)
  - Services for communicating with devices, buses, or hardware
- **Operating System** Segment (OSS)
  - OS-level functionality, Run-Times, Partitioning



FACE Reference Architecture

The interfaces:

- **Transport Services Interface**
  - Provides datatype-specific interface to move data messages between applications in the PCS and PSSS
- **I/O Service Interface**
  - Provides an interface to provide data movement and external access to/from devices or external hardware
- **OSS Interface**
  - Provides a standardized means for software to use the services within the operating system and other capabilities related to the OSS.
  - POSIX
  - ARINC 653
  - Configuration Services



FACE Reference Architecture

Terms defined by the FACE Reference Architecture:

- A software component residing in a FACE Segment, designed to the requirements specified for that particular segment, is referred to as a **Unit of Conformance (UoC)**
- Once a UoC has been through the FACE Conformance Program, it is known as a **FACE Conformant UoC**
- Units of Conformance that communicates with the TSS must provide a **Unit of Portability (UoP) Supplied Model (USM),** according to the FACE Data Architecture.
- The ability to host and integrate FACE software components is dependent on a **FACE Computing Environment**, which is an implementation of the following:
  - FACE TSS
  - FACE IOSS
  - FACE OSS
  - Common Services required for operation

FACE Data Architecture

What is it?

- A **data modeling approach** to describe the data going in or coming out of a PCS/PSSS component, in the context of the entities of concern to the software component, to enable an integrator to combine software components to provide a larger capability.
  - **In laymen's terms**: "Describe concepts we want to communicate about well enough for everyone to clearly understand what we mean.
  - Example: When a component defines a position message, is it WGS84 or ECEF?

When do I need it?

- Only when your UoC is communicating using the TSS

FACE Data Architecture

What does it consist of?

- Data Model Language
- A set of Data Model Language bindings that map Data Model Language elements to each of the supported programming languages (C, C++, Ada, & Java)
- The Shared Data Model (SDM)
- Rules for the construction of UoP Supplied Models (USM) and Domain Specific Data Models (DSDM)

What is the Shared Data Model?

- The starting point for all USMs and DSDMs
- It defines conceptual observables, logical measurement systems/axes, and platform IDL types
- You use it to build your USM or DSDM

Useful information

- Each version of the FACE Technical Standard has an equivalent SDM & Governance Plan

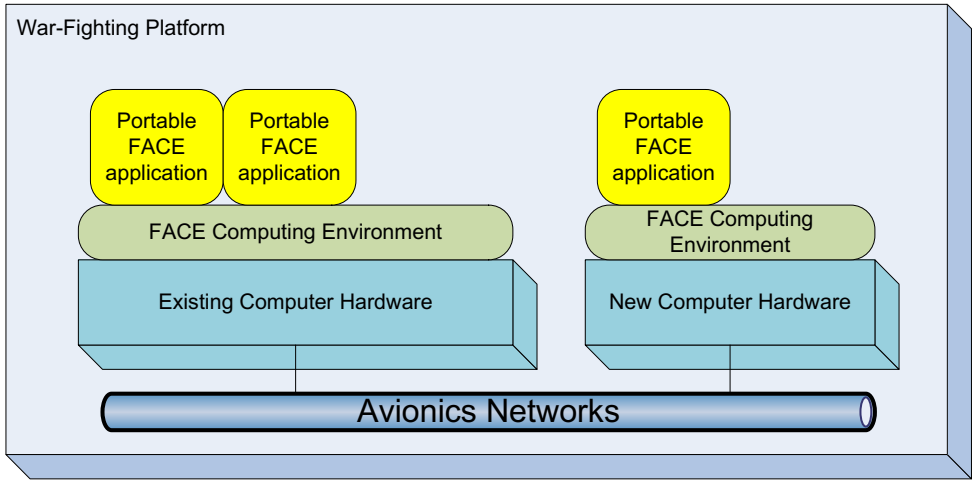
What Does it All Mean?

“Basically, the FACE Technical Standard provides a standardized means of designing software applications to promote portability and reusability in a ‘measurable’ way.”

It is meant for producing modular, safety-critical and general purpose software components, as well as providing an environment to integrate them.

It is not:

- A System Architecture
- A tool
- A coding standard
- A solution for ‘everything’
- An SDK



How Do I Use It?

Using the FACE Technical Standard is very much a "learn, adjust, and incorporate" process.

- Become familiar with the standard and check out:
  - Software Supplier's Getting Started Guide
  - Integration Guide for FACE Technical Standard, Edition 3.x
- Read the contracting guide for information on how to incorporate 'FACE' into software requirements
- Learn to navigate the standard. Start with UoC requirements, then identify requirements for the particular segment of interest.

  Bonus: Become familiar with the language requirements.

How Do I Use It?

When ready to develop, acquire/identify the following:

- An Edition of the FACE Technical Standard
- FACE Conformance Test Suite
- SDM
- FACE Data Model tool(s)
- A FACE Computing Environment and FACE support infrastructure (use the BALSA ones if you're new. They're free!)
- If FACE Conformance is one of the goals, contact a FACE Verification Authority early on and ask questions.

How Do I Start?

Go to opengroup.org/face and look around.

For documents, you will need to create an account. There's even a YouTube page link with videos for a variety of things. Check out the third-party tools page while you're at it.

ASK QUESTIONS!!!

FACE™
Future Airborne Capability Environment

Thanks!

Any questions?

Christopher Crook
chris.crook@intrepidinc.com