Tri-Service Convergence:

An Open Architecture for Embedded System Development

Embedded Tech Trends

January 23, 2018

Mike Hackert (NAVAIR)

Ben Peddicord (CERDEC)

Dr. Ilya Lipkin (AFLCMC)

CERDEC US ARMY – RDECOM

Introduction

- **DoD supports a Modular Open Systems Approach (MOSA)**
- **Industry has done a great job developing a number of standards**
  - OpenVPX has become the defacto standard for next generation embedded system designs
  - Other standards define all the key building blocks
    - For example: chassis management, communication protocols, etc.
  - Confusion remains on how to combine the “shopping list” of options into a holistic system specification
- **SOSA is developing a specification combining a number of standards**
  - All three services participating (Army CERDEC, Air Force AFLCMC, and Navy NAVAIR)
  - Participation includes both the end user community and the supplier base
  - Creating a layered approach applicable to a broad range of applications
- **All are invited to participate**

Outline

- **Overview**
  - What is HOST
  - What is CMOSS
  - What is SOSA
- **Open architecture goals**
- **What is different now?**
  - Convergence of HOST, CMOSS, SOSA

Who's Who
(or acronym definition)

- **HOST**
  - Hardware Open Systems Technologies standard
  - Initiated by US Navy's Naval Air Systems Command (NAVAIR) Patuxent River MD ~2014
- **CMOSS**
  - Command, Control, Communications, Computers, Intelligence, Surveillance and Reconnaissance (C4ISR) / Electronic Warfare (EW) Modular Open Suite of Standards
  - Initiated by US Army's Communications-Electronics Research, Development and Engineering Center (CERDEC) at Aberdeen Proving Grounds MD ~2013
- **SOSA**
  - Sensor Open System Architecture Standard
  - Initiated by US Air Force's Life Cycle Management Center (AFLCMC) at Wright-Patterson AFB, Ohio as an Open Group committee
    - Incubated in the Future Airborne Computing Environment (FACE) Consortium in ~2015
    - Stood up as consortium November 2017

HOST – A Key Pillar of NAVAIR's Open Architecture Approach

CERDEC
US ARMY – RDECOM

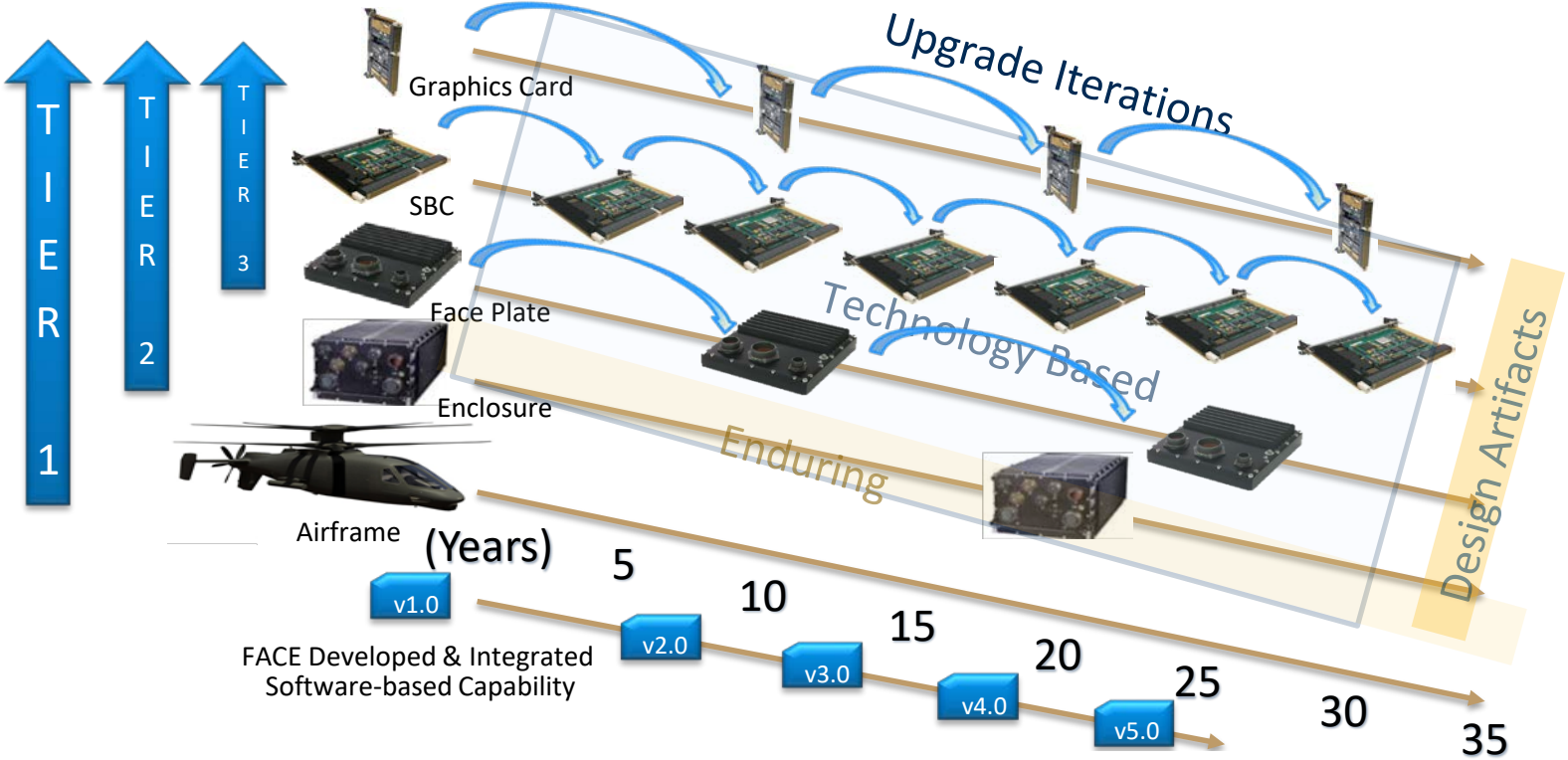
HOST Objectives

- **Create a hardware technical reference framework**
  - Used for developing embedded computing systems
- **Improve affordability**
  - Enables reuse
  - Increasing economies of scale opportunities
- **Enable effective and timely technology refresh cycles**
  - Abstract hardware from software
    - Acquisition community can pre-plan tech refreshes even during initial system acquisition program
    - Vendors know module interfaces, mitigating risk in new product investment
- **Initial focus on airborne mission processing**

HOST Creates Module Level Specification

- **Establish interchangeability at the module level**
  - Specify hardware interfaces (e.g. pin and protocol) allowing creation of standardize modules
  - Create generic module requirements (e.g. single board computers (SBCs) or switch cards)
  - Is silent on functional requirements for particular payload capabilities
- **Establish interoperability through a layered approach which adds requirements for programmable logic**
  - Allow reconfigurability so that one component (e.g. SBC) can operate in two different systems
- **Enable components to be interchanged across systems utilizing common reconfigurable logic (e.g. Field Programmable Gate Array)**

HOST Vision



HOST's Three Tiered Structure

- **Tier 1**
  - Establishes universal requirements that apply to all HOST components regardless of core technology
  - Extensible to multiple core technologies (e.g. other formats)
  - Only one Tier 1 ever anticipated
- **Tier 2**
  - Defines platform agnostic technical requirements for each core technologies
  - Currently only the OpenVPX Tier 2 standard exists
    - Considering other application space for the next Tier 2
    - Thoughts include
      - Unmanned vehicles or smaller format than 3U
      - Weapon systems with unusual formats (e.g. round format)

HOST's Three Tiered Structure

- **Tier 3**
  - Specifies module level requirements (e.g. one for every unique module)
  - Define hardware requirements that must be combined with system unique requirements
    - Must add requirements to be a complete specification
      - Programmable logic
      - Payload capabilities
      - Integration logic (e.g. system startup, hardware management, etc.)
    - Component level documents that will guide H/W development to facilitate modular components, Tier 3 reuse, and upgradeability
  - Allows end user to create a component registry of Tier 3 specifications
    - Useable for management of a family of products (e.g. modularized systems)
    - Useable for sharing modules across programs and services

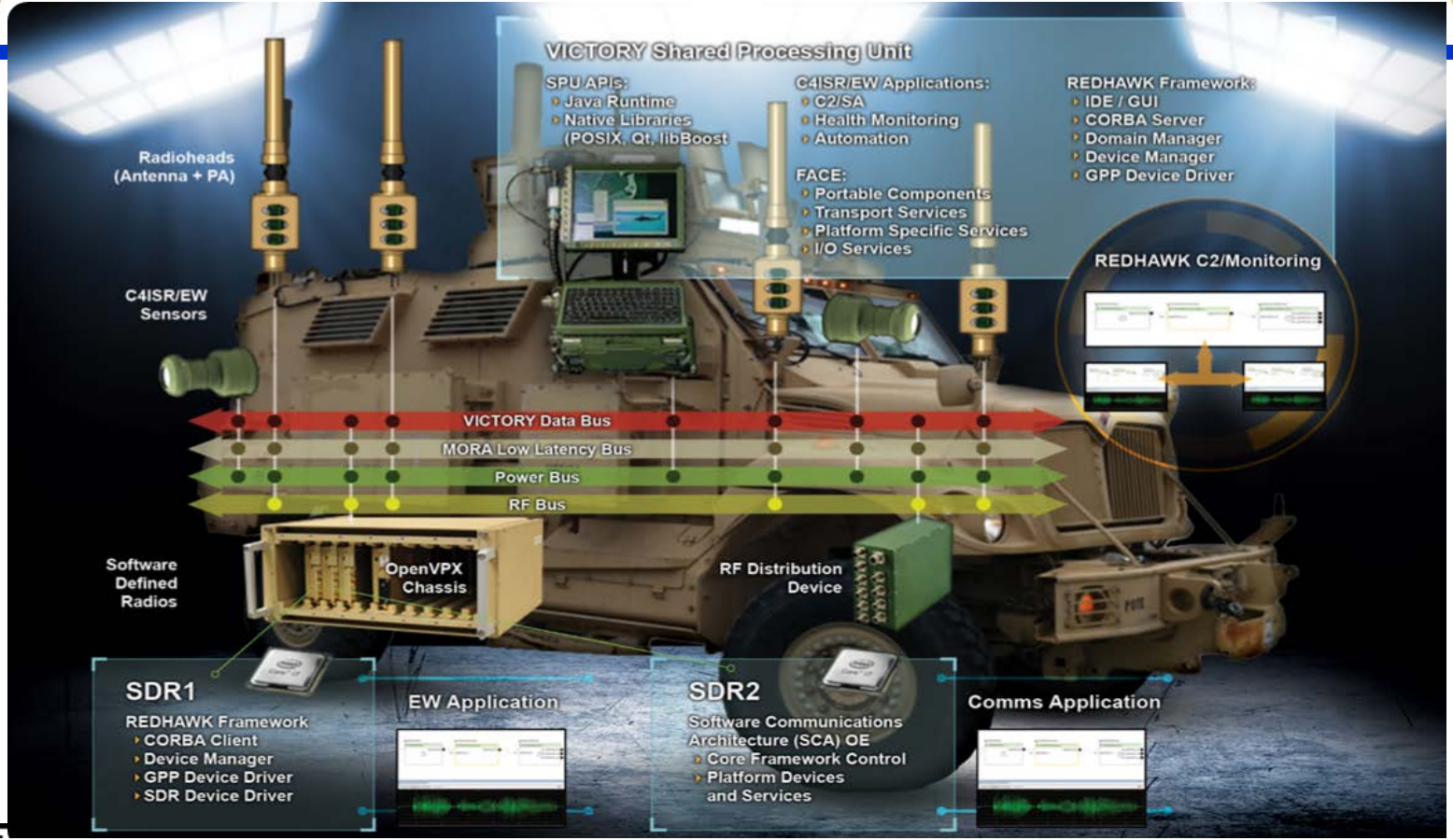
CMOSS

- **Defines an open architecture that reduces size, weight and power (SWaP) while enabling rapid insertion of new capabilities**
  - Focused on C4ISR systems
- **Utilizes a suite of standards (similar to HOST but different)**
  - Uses OpenVPX Hardware form factor to create capabilities as common cards
  - Establishes network interoperability using Vehicular Integration for C4ISR/EW Interoperability (VICTORY) to share services such as Time and Position
  - Decomposes functionality using the Modular Open RF Architecture (MORA) to share resources such as antennas and amplifiers
  - Software frameworks such as REDHAWK, Software Communications Architecture (SCA), and FACE to enable software portability

CMOSS Vision

- **Defines a Universal A-kit that eliminates the need for platform specific integration**
  - An A-kit is typically thought of as an integral part of a vehicle
    - A simple example is the wiring harness
- **Fields capabilities as payload specific cards in a common chassis and using existing cabling**
  - Modules become units of configurable capabilities
- **Takes the next step beyond HOST defining an additional layer of payload requirements**
  - Payload capabilities

CMOSS Vision



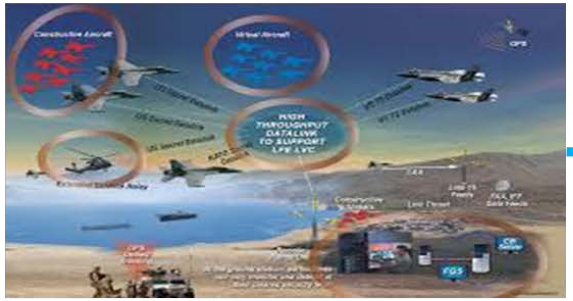
CMOSS's Game Changing Approach

- **Establish a Universal A-kit for Army vehicles**
  - Revolutionizes sustainment by defining standardized interfaces into which both common and specific modules can be easily replaced, swapped, or upgraded
  - Reduces logistics tails by enabling common sparing (both within and across systems)
    - Examples include single board computers and switch cards
- **Common modules increase competition and economies of scale**
  - CMOSS abstracts software from hardware
  - Reduces sustainment costs
  - Eliminates need for "End of Life" buys for a 30+ years sustainment
  - Enables hardware modernization every 5-10 years or less

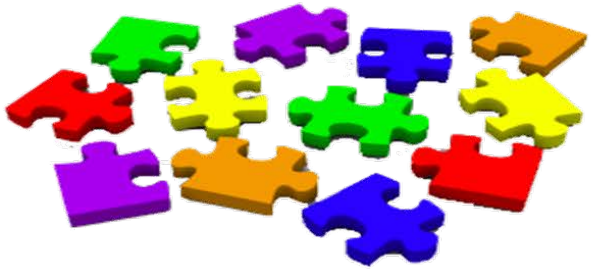
SOSA Overview

- **Collaborative effort across C4ISR development community**
- **Includes users and suppliers**
  - Air Force, Army, Navy, and Other Government Agencies (OGAs)
  - Industry partners
    - Prime contractors, integrators, systems developers, module and payload developers, etc.
  - See SOSA Open Group Forum for complete list of participants
- **Jointly developing common standards for sensor subsystems at the electrical, mechanical and software interfaces**
- **Extending its application space to new capabilities**
  - Including those created by general processing / reconfigurable resources

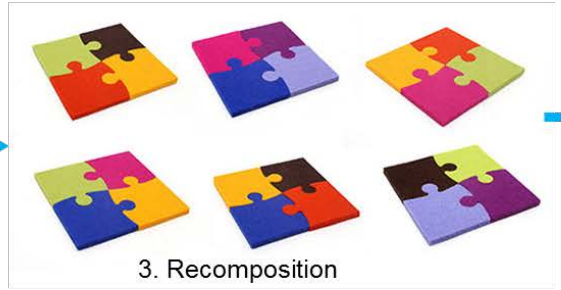
SOSA Vision



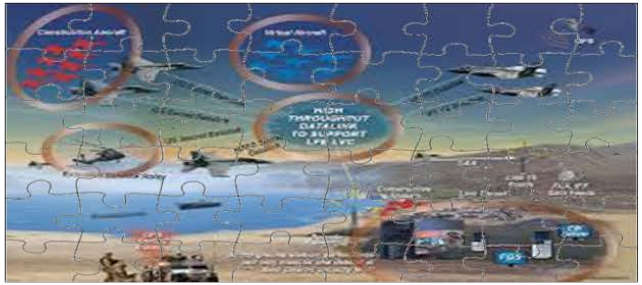
1. Current Mission Met Through Point Solutions and Workarounds



2. Decomposition into Functional Components



3. Recomposition Into Reusable Capabilities



4. Affordable Mission Effectiveness Through Systematic Reuse

Open Architecture Goals[1]
(Equivalent to HOST, CMOSS, SOSA Goals)

- **DoD seeks five primary benefits of MOSA:**
  - Enhance competition – open architecture with severable modules, allowing components to be openly competed
  - Facilitate technology refresh – delivery of new capabilities or replacement technology without changing all components in the entire system
  - Incorporate innovation – operational flexibility to configure and reconfigure available assets to meet rapidly changing operational requirements
  - Enable cost savings/cost avoidance – reuse of technology, modules, and/or components from any supplier across the acquisition life cycle
  - Improve interoperability – severable software and hardware modules to be changed independently

1. https://www.acq.osd.mil/se/initiatives/init_mosa.html

Summary

- **Tri-service Coordination for a Common Architecture Approach**
  - Establishing a common approach to embedded system standardization across all three services
  - Using industry standards wherever possible (e.g. VITA, VICTORY, MORA, etc.)
    - Adding specificity where necessary for interchangeability or interoperability
  - Creating / extending standards where necessary
    - Examples include chassis / hardware management
- **Converging in one standards body – SOSA**
  - HOST being mapped into CMOSS and added to SOSA
  - CMOSS being absorbed under SOSA
  - SOSA being extended and broadened to fulfill open architectures goals