



Nabor Felix Cortez



Erik Williams



Andrew House



Jorge Ramirez



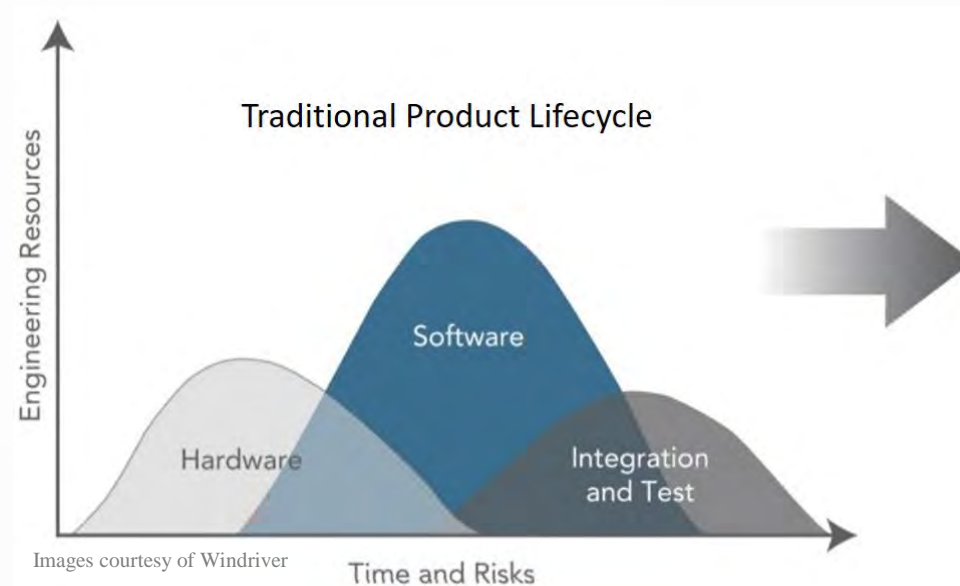
Virtualization: Unlocking Software Modularity of Embedded Systems

Abstract



Traditional avionics-embedded development requires writing software applications that are specific to hardware (HW) configurations and the operating system (OS). In the early stages of the lifecycle, this proves cumbersome because complete architectures usually need to be defined before any software development can begin. This constraint also limits reuse and refactoring during hardware end-of-life or HW refreshes, which leads to higher and longer lifetime costs. By requiring physical access to the hardware for testing, this process hinders the fundamentals of Agile, pushing the lifecycle further right.

As Digital Engineering becomes mainstream, it is increasingly critical to understand the implications of various certifying bodies and contracting support to fully utilize its associated benefits. DevSecOps designed for embedded development is a highly rigorous, complicated, and time-consuming problem to overcome. Supporting embedded systems beyond application development with matching acquisition support is crucial to unlocking their modularity.



"Virtualize embedded systems to remove road-blocks to accelerate DevSecOp practices"

Embedded System Components



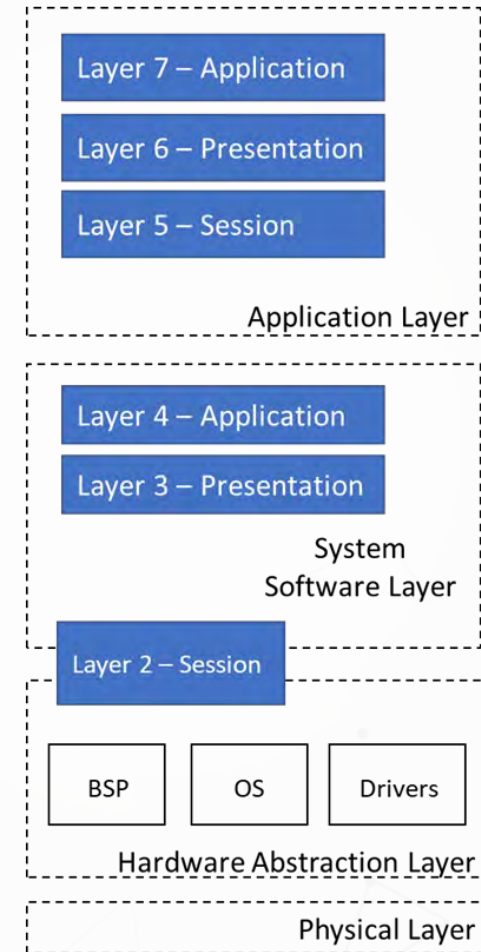
The adoption of standard interfaces alone does not guarantee modularity and reuse. Intellectual Property (IP) can be included in delivered solutions using unique hardware, firmware, and I/O devices.

Swapping software created by different contractors has the potential to fail due to hidden IP if it is not done early in the development process. The modularity of a system can be limited by incompatibilities caused by "hidden IP" or other "gotchas."

To avoid this dilemma, the embedded system must be generalized to the point where any incompatibilities would be obvious. The ultimate goal is to create "application stores" for each specific system.

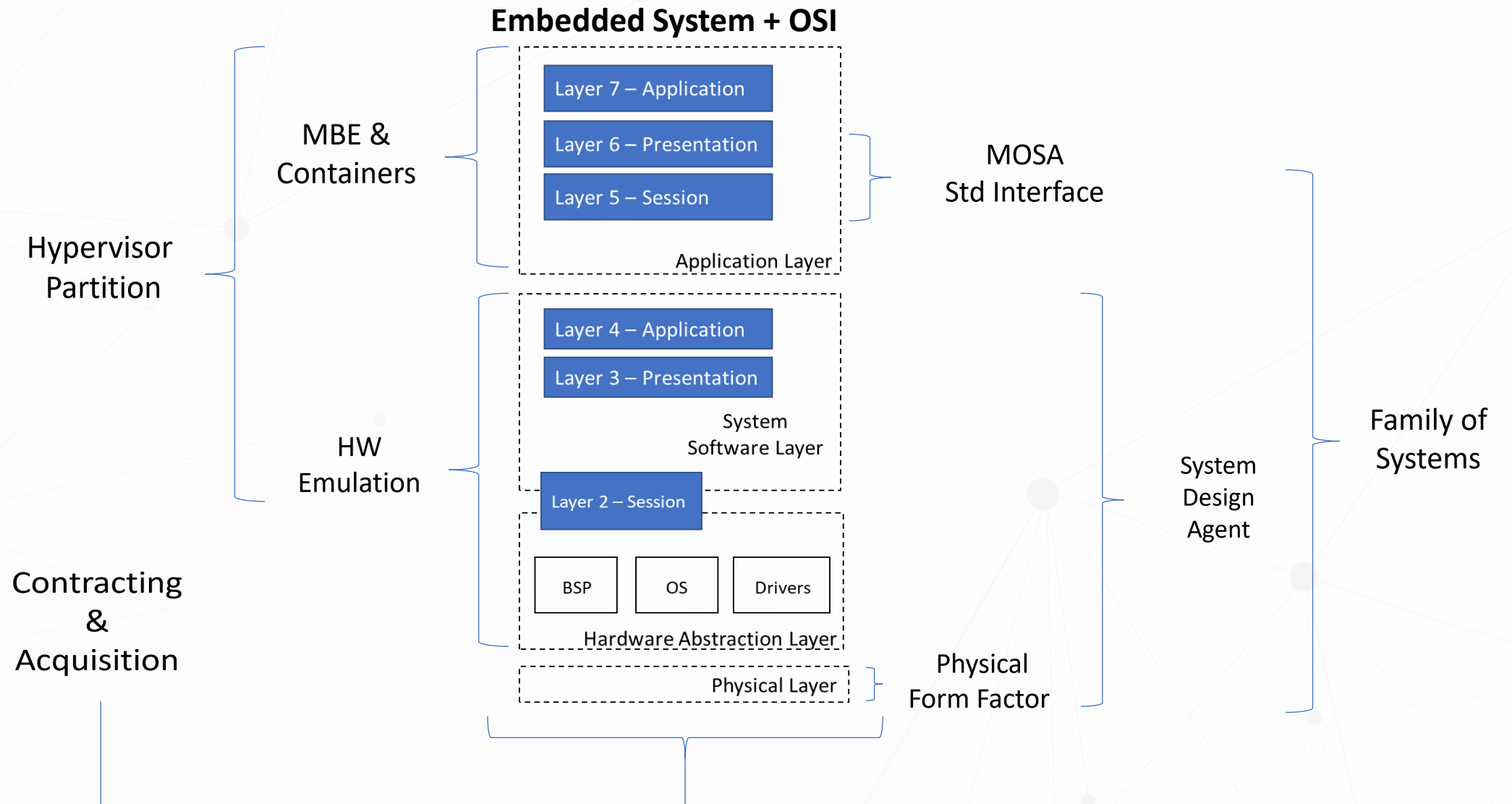
Equally important is, the development of a robust acquisition and contracting strategy must respect the trade-offs between the underlying proposed technologies and processes being adopted.

Embedded System + OSI



"We must look beyond standard interfaces to unlock software modularity"

Virtualization: Embedded System Strategies



Golden Horde Colosseum (GHC)



Acquisition processes and tool-chain utilizing virtualization to remove Software/Hardware dependencies

- Digital Sandbox
 - Built on AFSIM (Government owned M&S framework)
 - Scenarios created to support mission requirements
 - Scoring evaluation and playback against requirements
 - Supports rapid prototyping of embedded software such as NCA
- Software V&V via build-up approach through SITL, HITL, and LVC
- Challenge Based Acquisition
 - Other Transaction Authority (OTA) Agreements for agile contracting
 - Prototype OTA used for competitive selection
 - Working software delivered for each competition
 - Production OTA for "winner" follow-on sole source
 - Can be used in conjunction with System Design Agents



The Colosseum will expand the industrial base by lowering entry barriers, building a sandbox for traditional and non-traditional entrants, enabling rapid development, demonstration of new NCA technologies, and providing quicker acquisition and transition on-ramps.

GHC Virtualization – Interfaces / Containerized Software



GHC is effective because it utilizes unclassified challenges, non-traditional DoD contracting methods, open standards, software/hardware virtualization, and provides a level playing field for the evaluation of vendor software. Everything but the core functional requirements are provided (i.e. Virtualized). This allows a more equitable opportunity to win government contracts.

"Framework is provided to contractors, so the focus is on embedded software and not underlying software or hardware dependencies"

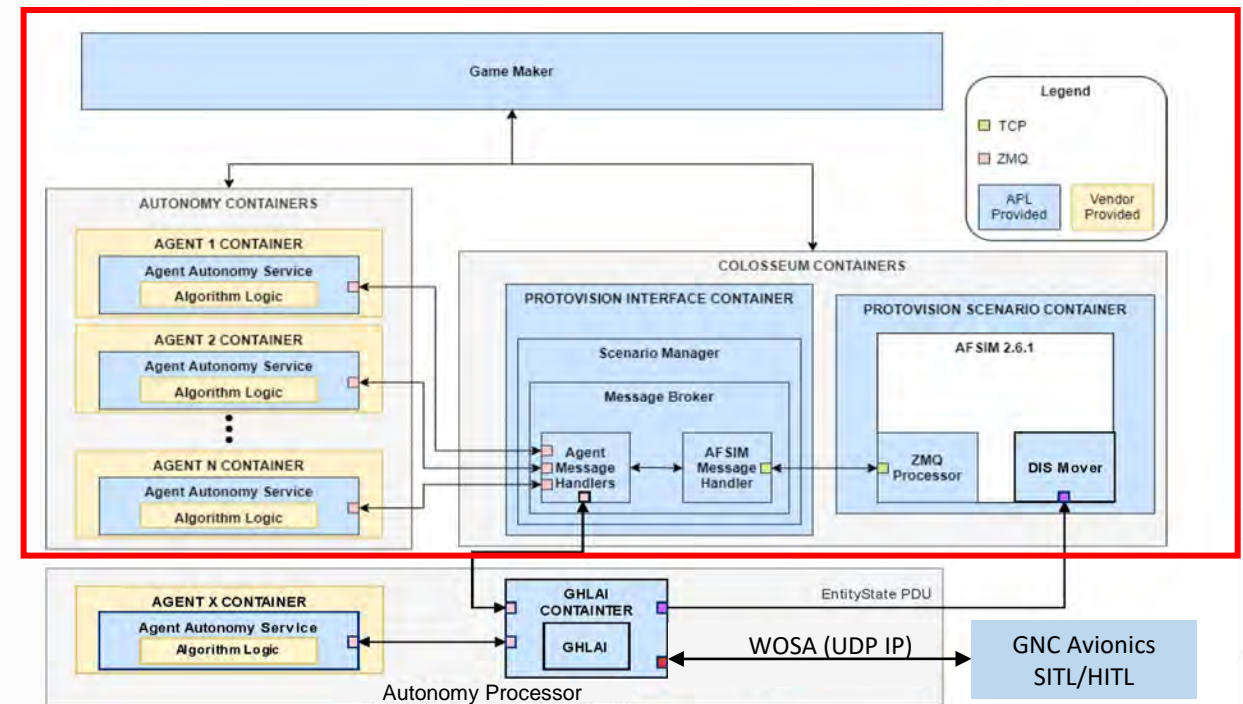
Virtualization through standardization

- Interface standardization and translation
- Software containers

Software V&V via build-up of "in-the-loop" components with mission simulation (AFSIM): model, software, processor, and avionics hardware

Approach

- Container Protobuf/ZMQ Interface to AFSIM
- MOSA/WOSA integration
- GHLAI translation and abstraction between software containers, GHC, and real hardware interfaces
- Vendor IP protected within container



GHC Virtualization – Live Virtual Constructive (LVC)



Flight Test Objectives

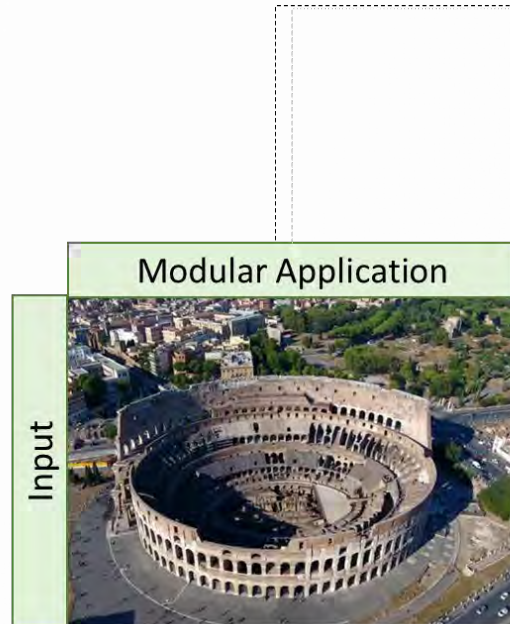
Demonstrate that the AI solutions are able to integrate onto live platforms and computing payloads

- *SWAP, Comms, real flight dynamics, latency etc.*

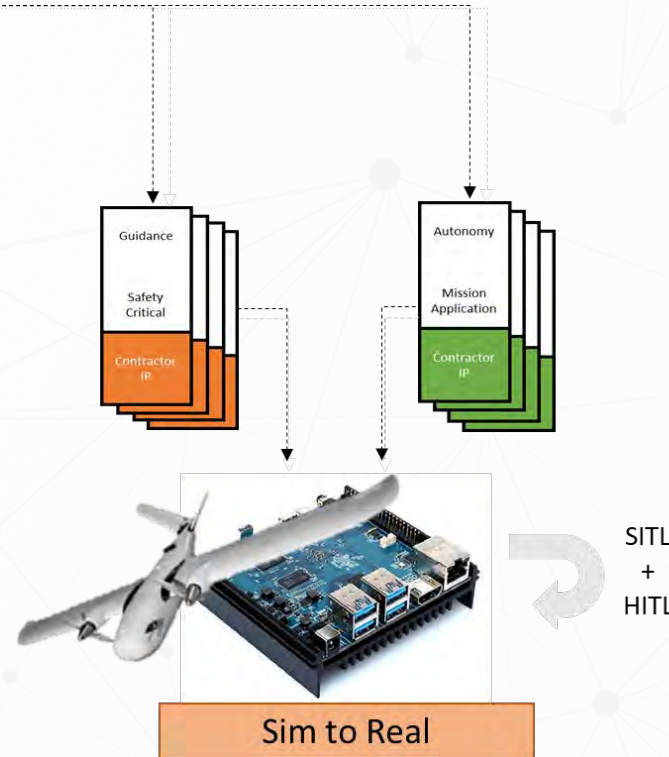
Demonstrate the ability to test new autonomy concepts on cheaper non-objective platforms

- *Critical to future development/fielding trajectories*

Modular Open Systems Approach (MOSA)
+
Mission Application Requirements
+
Digital Models And Supporting Ecosystem



- ✓ Create the Arena Challenge for a System – pick, shop, and choose Software and Hardware to meet the mission needs.
- ✓ Stay Generic.
- ✓ Digital First for early and often testing.



Modular Software is loaded on commercial developmental hardware with matching architectures. Digital Emulation of the hardware provides test beds on every developer's desk.

Flight test on a Surrogate platform with real hardware flying with virtual clones

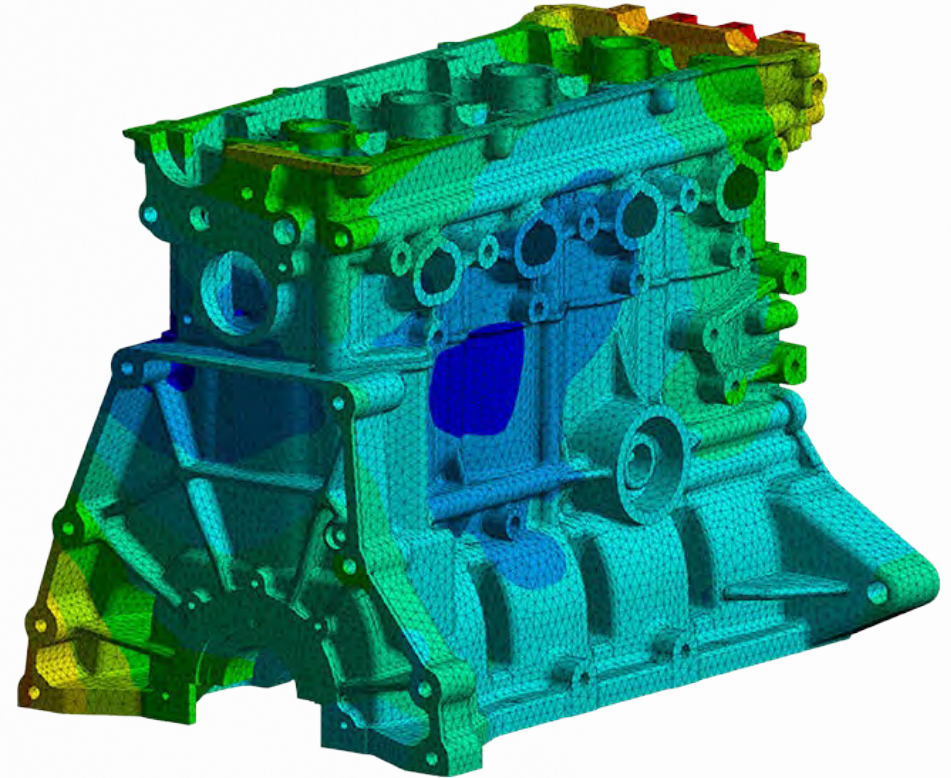
- *LVC Agents + Live Aircraft*
- *3x Elanus Duo Surrogate Aircraft*
- *Live Aircraft Interface: GHLAI*
- *Safety: Testing of Autonomous systems in a Complex Environment (TACE)*

Virtualization: Model-Based Engineering



What is Model-Based Engineering?

To put it simply, modeling is the use of something in place of something else for some cognitive purpose at a low cost. This allows us to use a simpler, safer, or cheaper alternative to "reality." A model represents reality for a particular purpose. However, it is an abstraction of reality because it cannot represent every aspect of it. As a result, we can deal with the world in a simplified manner, avoiding risk, complexity, danger, and irreversibility.

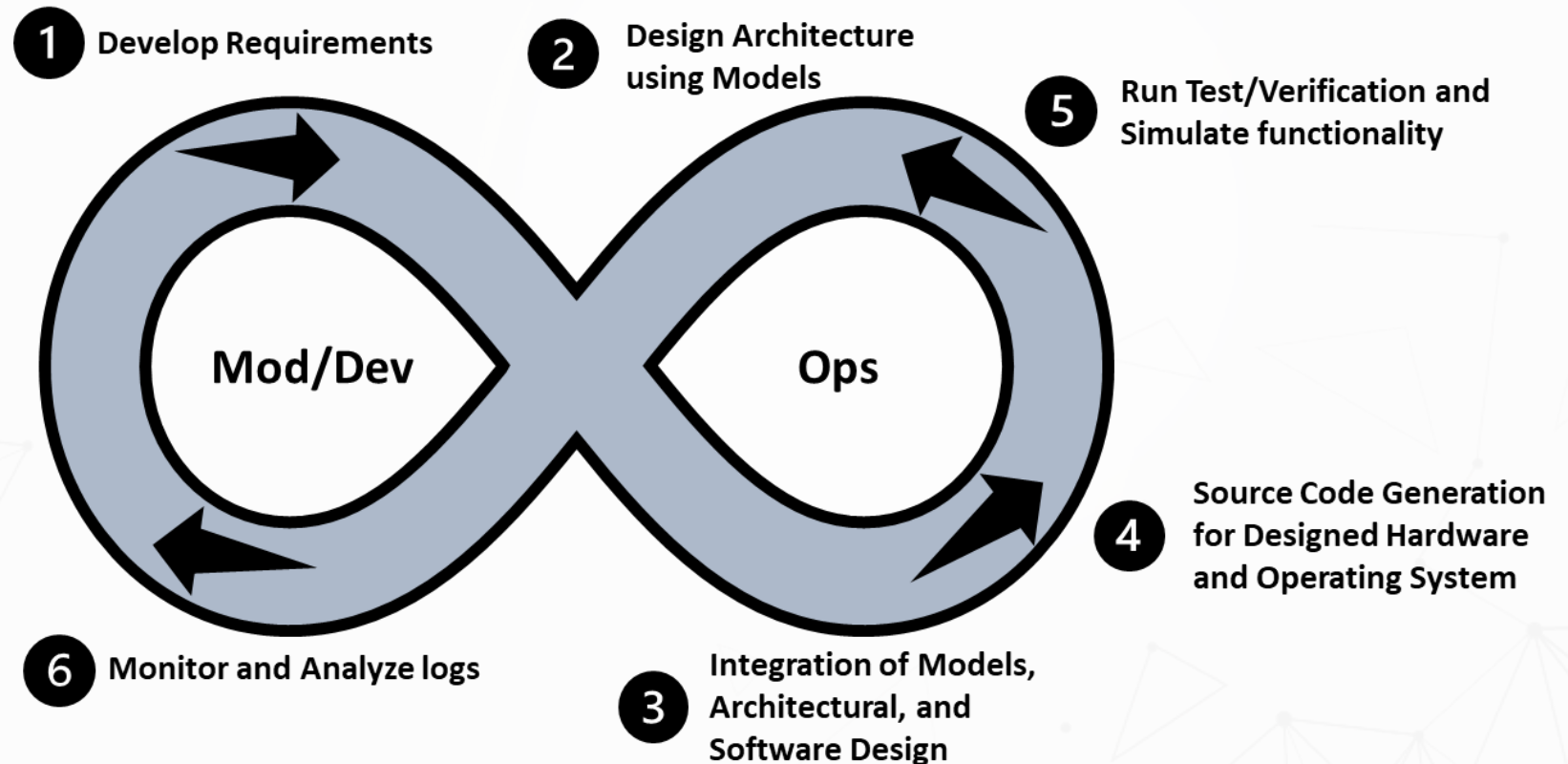


Virtualization Model Lifecycle: ModDevOps



Integration of various models, specifically System and Software models, can be used to optimize each key phase in the OODA Loop. The evolution of software abstraction from assembly code to object-oriented code has been a natural evolution over time. It is inevitable that software abstraction will continue in the form of models.

- Hardware Agnostic
- OS Agnostic
- *Logic separate from the Interface
- *Integrated Simulation & Test
- Inherit Traceability DO-331
- Auto Code & Document Generation
 - DO-330 Tool Qualification
 - Easy integration of APIs
- Source of Truth is a single Model
- Easy to share



Virtualization: Hardware Abstraction via Emulation



aka Software Digital Twin



A Digital Twin is: “An integrated multi-physics, multiscale, probabilistic simulation of an as-built system, enabled by Digital Thread, that uses the best available models, sensor information, and input data to mirror and predict activities/ performance over the life of its corresponding physical twin.” *(source: DAU Glossary of Defense Acquisition Acronyms and Terms)*

For the purpose of comparison with “Hardware Emulation” for software V&V, we are considering a Digital Twin to be defined as follows:

- A Digital Twin is the digital representation of an SBC(s) and/or components representing a functionally correct, predictable and reproducible representation of that board or system at the appropriate level of fidelity to perform software verification, performance analysis and software validation tasks.

Virtualization: Hardware Emulation for DevOps

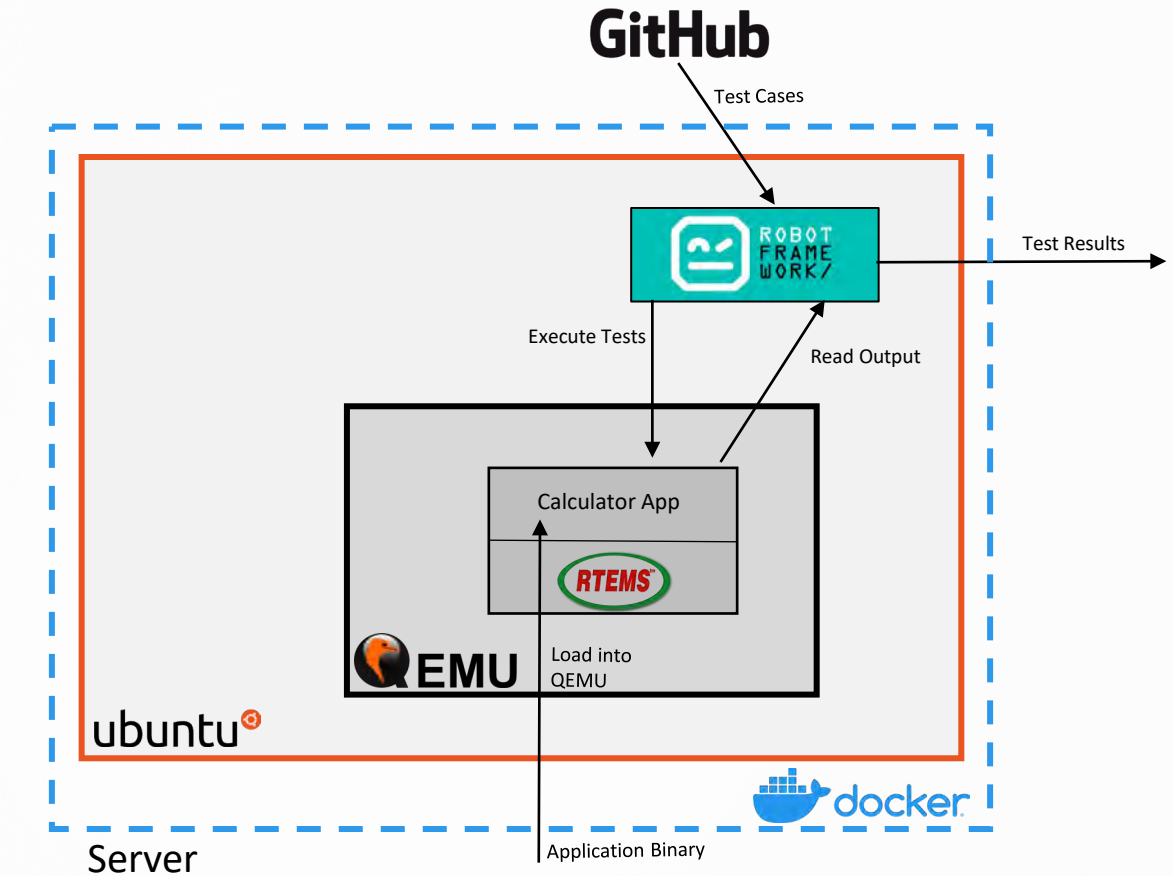
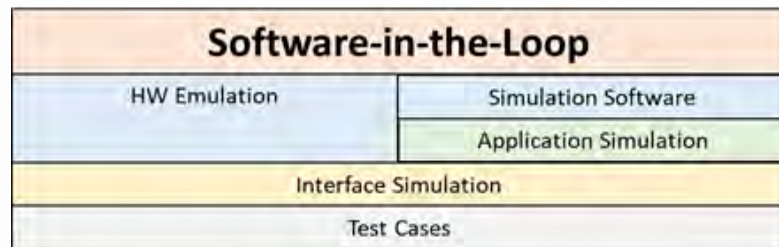


“How do you create a build container to run non-x86 architectures?”

As opposed to a server-hosted VM, containerization can be utilized for maximum portability/minimum resources.

Embedded system emulation requires additional levels of virtualization beyond simple docker containerization.

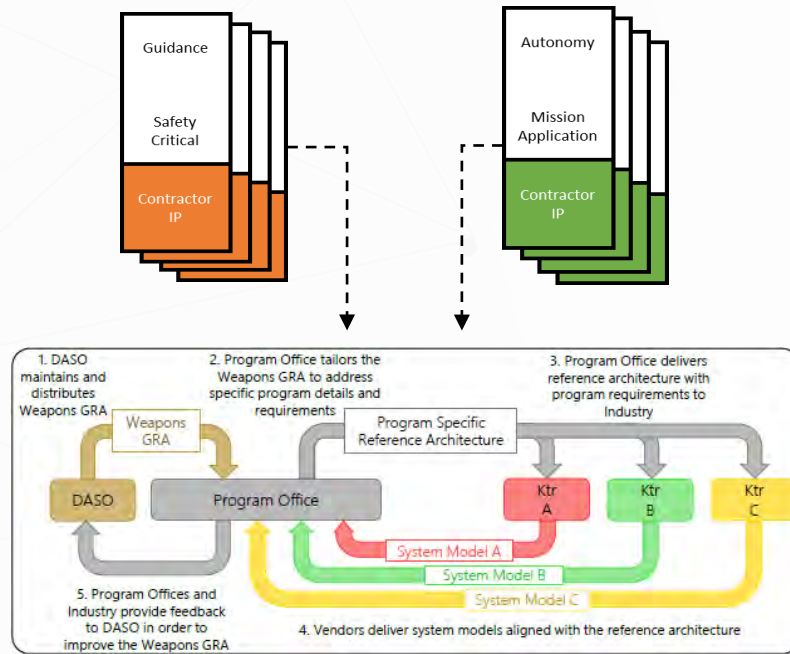
- Emulated HW specific images
- Increased resource requirements
- Potential for onboarding to Iron Bank/Registry One
- Easy transition to HWTL



Virtualization: Embedded Hypervisor



In an embedded system, an embedded hypervisor allows different operating systems, software, components, and functions to work independently while sharing information via a standardized bus (ARINC 653).

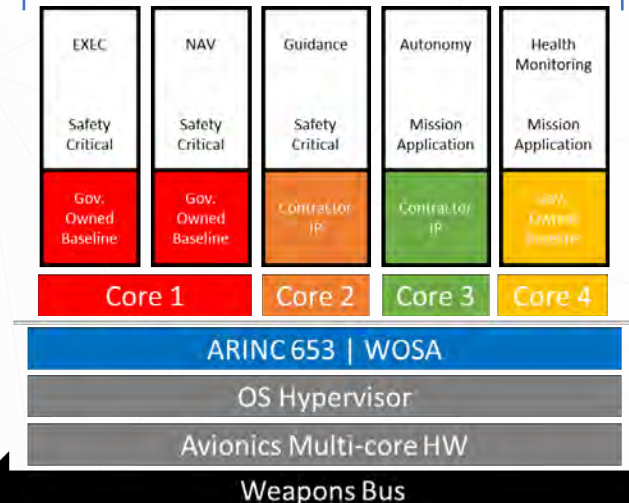


Government Reference Architecture



MOSA Compliant Modular Software

Government Owned Baseline
(Model Based Engineering with
Auto Code Generation)
+
Challenge Based Acquisition
Contractor Applications



MOSA Compliant Avionics Hardware

Virtualization: Software Design Agent



What is a Design Agent?

- Hired to Develop a Government-owned Technical Data Package (TDP)
- TDP Serves as Basis for Competitive Production and Life-Cycle Sustainment

Benefits

- Splits traditional Primary Contractor role between Design & Integration
- Non-Proprietary Data, Government Has Unlimited Data Rights
- Risk Reduction; SDA is contractually responsible for success
- Selects, awards, and manages subcontracts to domain experts
- System Engineering tasks to transition designs to production
- Increased Govt insight during design phase and selection

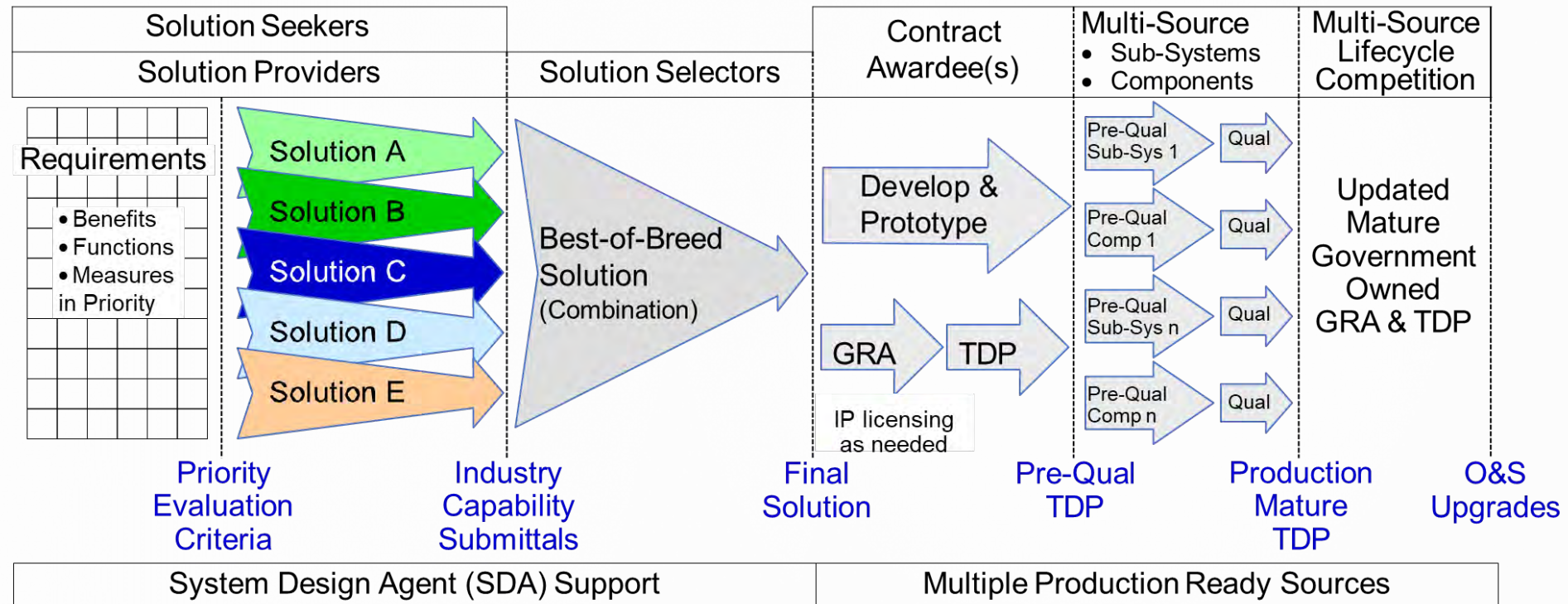
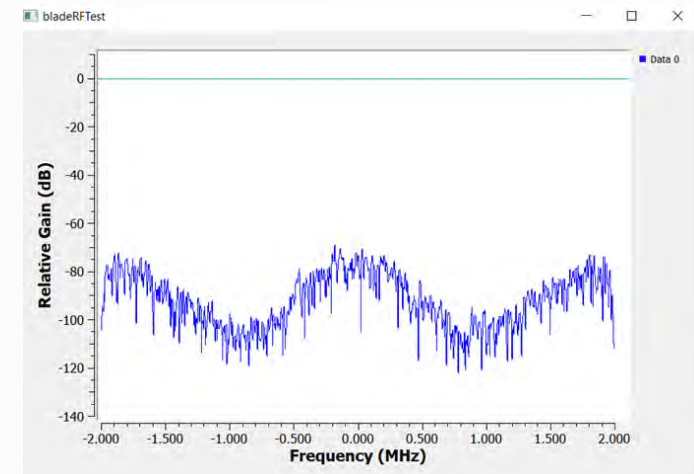
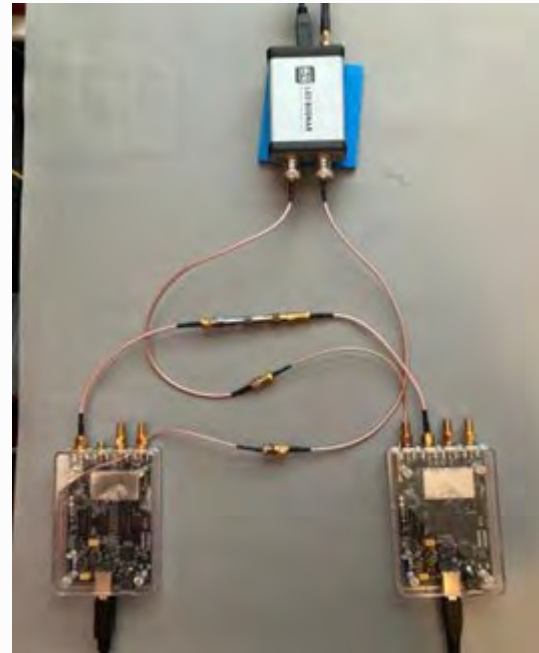
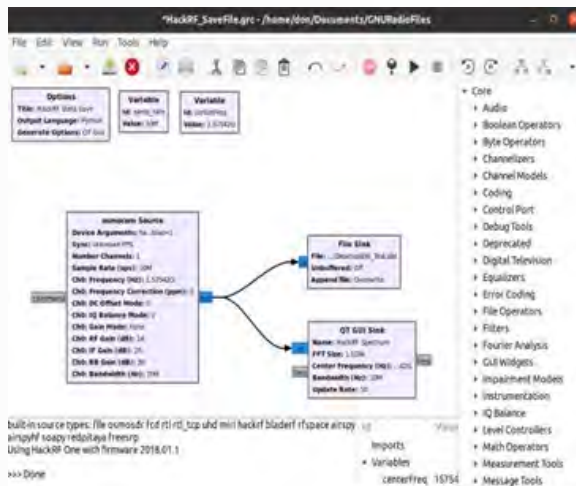


Image courtesy of IS4S

Virtualization: Family of Systems



Software Defined Hardware (SDH) is a combination of software and hardware technologies that enables programmable architectural and characteristic changes to the system. For example, a Software Design Radio (SDR) that is used across multiple platforms, and has different attributes such as frequencies, is a Family of Systems (FoS).



Virtualization: Family of Systems

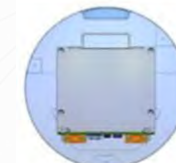


The VNX+ SWaP-C attributes make the standard a natural fit for SFF ATR-style avionics boxes and weapon systems requiring high-performance sensor interfaces in close proximity to FPGA and MPSoC signal processors, computers, radios, and platform I/O available as COTS MOSA modules with standardized electromechanical backplane interfaces.

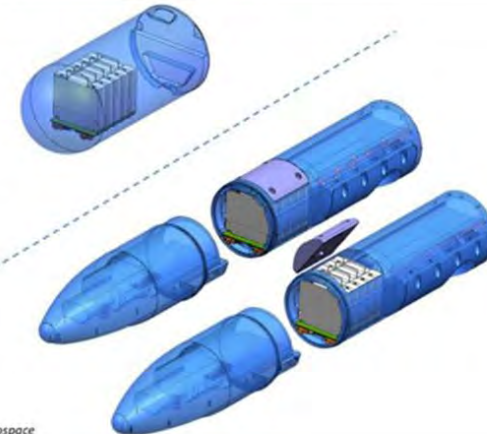
VNX+ is the only backplane-centric COTS MOSA standard which can be deployed as vertically oriented conduction-cooled modules on a traditional horizontal backplane, mounted longitudinally which is ideal for WDL munitions.



Example of VITA 90 VNX+ backplane and module



Coyote (6" Diameter)



AIM-9X (5" Diameter)

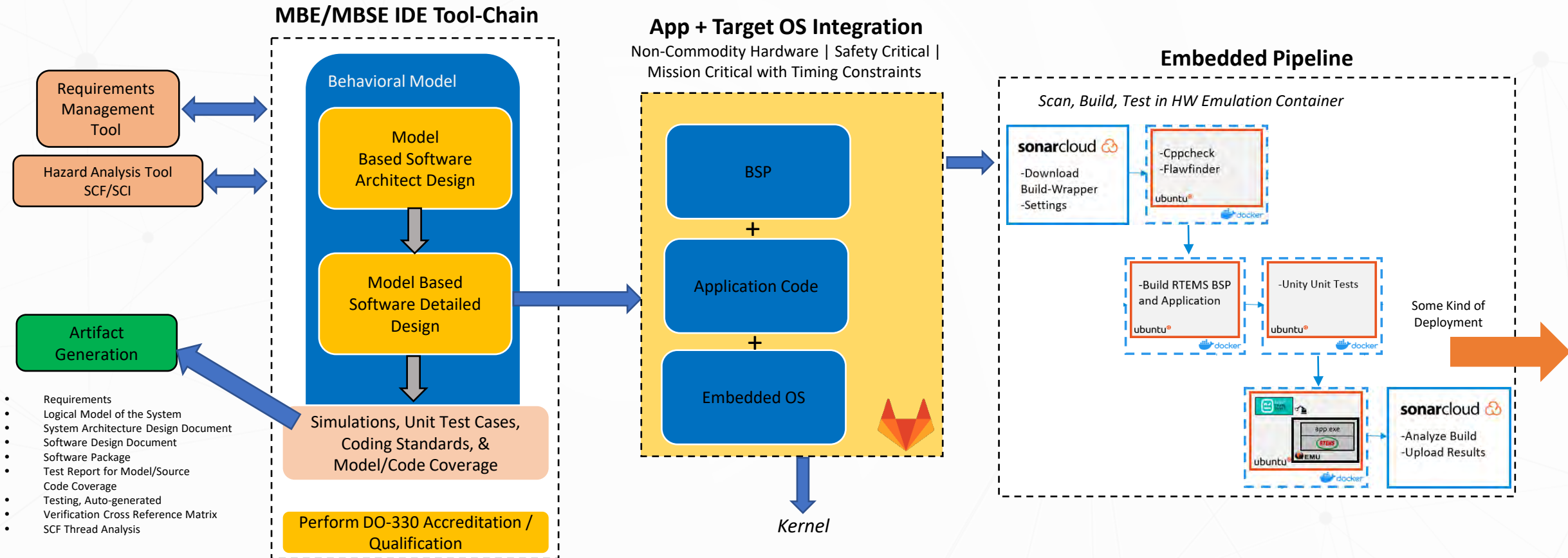
Images Courtesy of Collins Aerospace

VNX+ example system in Coyote and AIM-9X Sidewinder

Digital Transformation of the Embedded Development Lifecycle



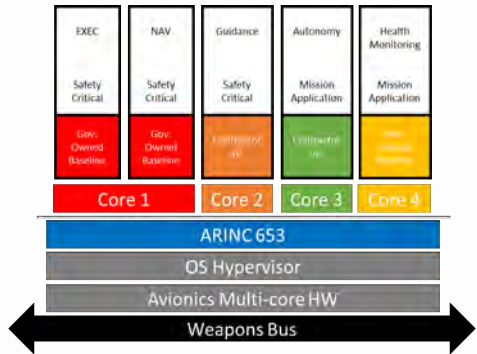
MBE & Hardware Abstraction Pipeline integration



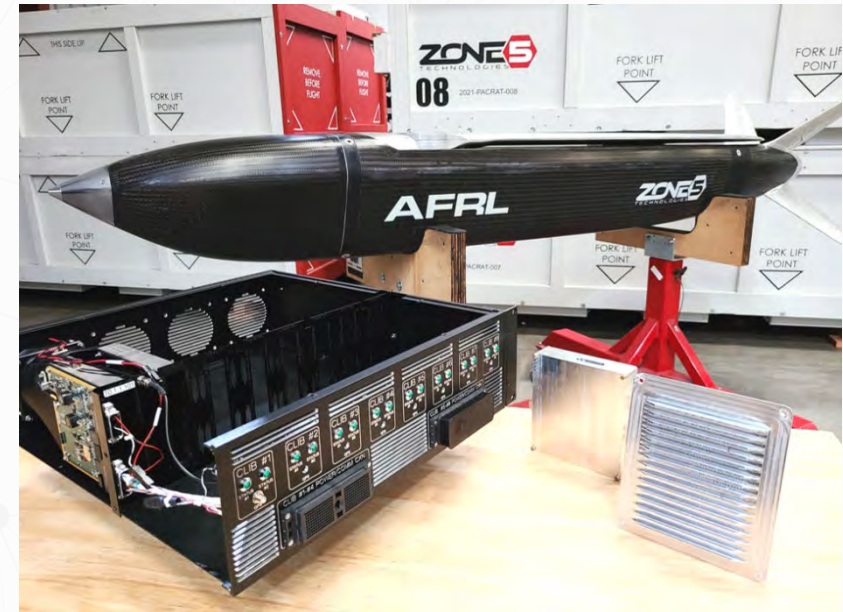
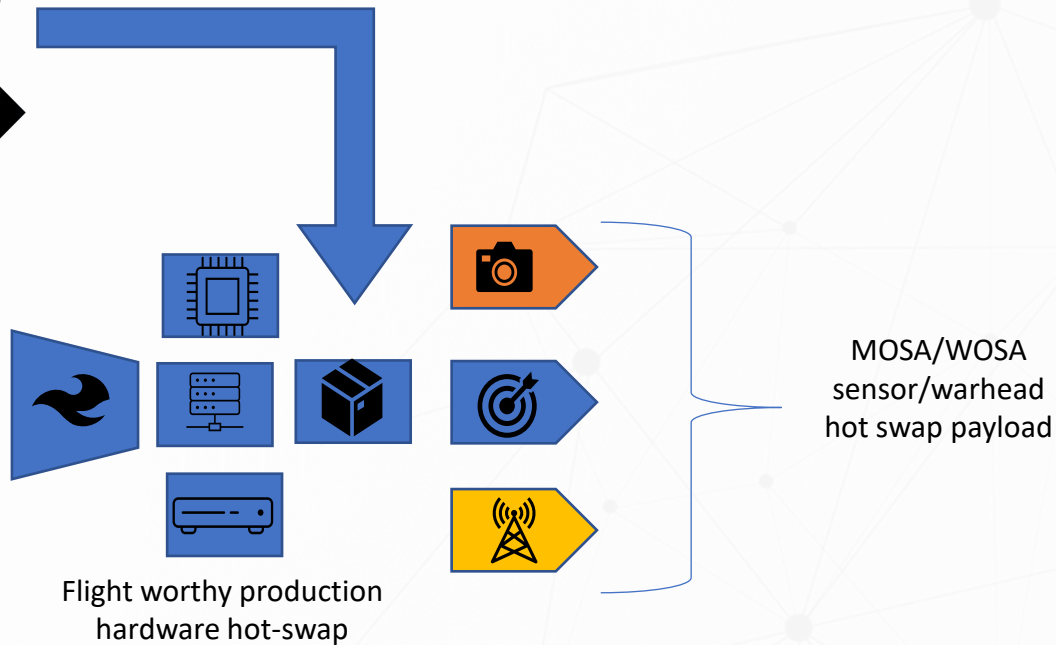
Virtualization: Open Weapon Platform (OWP)



The idea of an open weapon platform is complete control over the modularity between hardware requirements and software requirements to allow for testing without the immediate availability of real weapon system, all at a fraction of the cost. One such project, CLEVER, is setup to be an OWP.



Lego Blocks of Hardware, Software, and Sensors on a Proven Surrogate

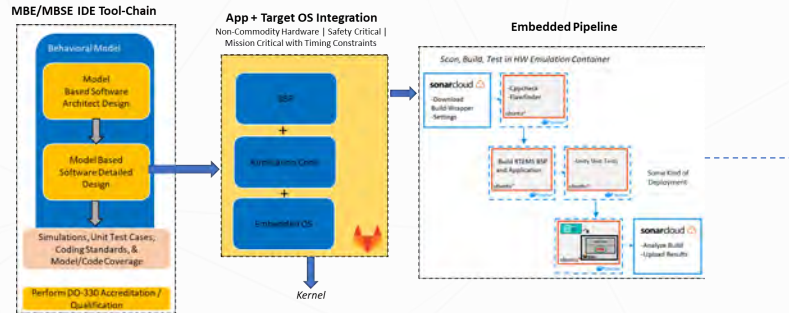


Images courtesy of Zone5

Proposed Virtualization Integration Strategy: Keyboard to Boom



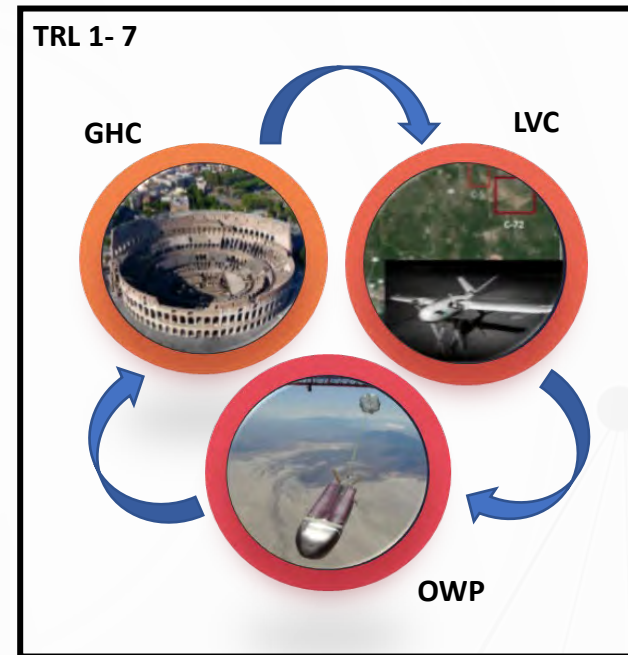
Keyboard to Boom aka "Ka-boom"



Selected Hardware Architecture
+
Digital Twin of Real Weapon
+
Baseline Specific Software for Configuration

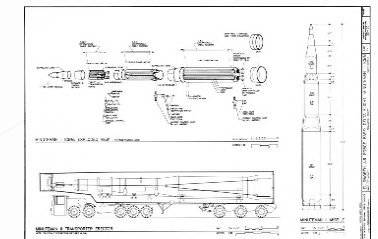
Selected Configuration for Proposed Program of Record

Avionics Hardware and Software Validation



Prototype/LRIP (TRL 8)

Once Configuration is validated and approved recommendations could move forward for full representation prototype for live fire and LRIP.



Fail Fast

- Configuration Changes
- Requirement Adjustments
- Design Flaws mitigated
- Reduce Tech Debt





“What Would You Attempt To Do If You Knew You Would Not Fail?”

– Robert Schuller



Two Questions You Should Ask About Your Embedded System

1. Is your development environment the same as your deployed environment?
2. Did you account for each Hardware/Software layer in your system?



XANATOS GAMBIT

Team Video
What Can We do for you?



Innovate
ARMAMENT DIRECTORATE
with



What is Commodity Hardware?
Embedded DevOps Video



Golden Horde

Intro Video
Are you ready to play a game?



DIGITAL EMBEDDED
DEVELOPMENT WITH A BANG
Presentation



IN THE
Nic of time
Lets Talk Embedded Software

Biographies



Nabor Felix Cortez is the Product Owner (PO) for the Xanatos Gambit (XG) team within the 76 SWEG's EDDGE division at Tinker Air Force Base. The XG team is devoted to advancing embedded technologies through DevOps and Digital Engineering applications. As a defense industry professional, Mr. Cortez has 16 years of combined public and private sector experience. With experience in more than eight weapon platforms, he also possesses a background in System of System Integration, sub-system design, OFP, RF/Radar, modeling, and simulation.



Felix Cortez

LinkedIn

He is the founder and Chair of Team 8, the embedded software consortium for the DOD DSAWG (DevSecOps DoD Authorization Working Group), which now resides within the AOWG (Action Officer Working Group) at the Pentagon. Team 8 is focused on applying industry best practices for embedded systems in the defense environment. In addition, it is focused on affecting policy within the DoD community to reform the way we approach embedded systems.

 nabor.cortez@us.af.mil




Erik M. Williams is a Senior Electronics Engineer at the Weapon Dynamics, Guidance, Navigation, and Control Branch of the Air Force Research Laboratory (AFRL) Munitions Directorate, Eglin Air Force Base, FL. He leads engineering and technical management activities for the Golden Horde Vanguard and has been in that role since Golden Horde concept inception in late 2018. His current focus is on development of the Colosseum, a fully integrated environment with software in the loop, hardware in the loop, and Live, Virtual and Constructive simulation capability for weapons that will enable rapid prototyping, evaluation, demonstration, iteration and transition of networked collaborative autonomous (NCA) technologies to the Air Force.



Erik Williams

LinkedIn

Prior to this post, he has served the Lead Engineer in the Air Force Lifecycle Management Center's Armament Directorate, Lethal SEAD/DEAD Branch (2013-2018), an Electronic Warfare Mission Data Engineer in the Air Combat Command (ACC) 53rd Electronic Warfare Group (53EWG) (2008-2013), and an enlisted Avionics Technician and Electronic Warfare Operator in the United States Navy (1999-2007).

 erik.williams.7@us.af.mil




Mr. Andrew House is currently a development engineer for the Xanatos Gambit team, part of the EDDGE division of the 76 SWEG at Tinker Air Force Base. With a background in process engineering, Mr. House specializes in DevSecOps for embedded software.



Andrew House

LinkedIn


He has been a lead for Team8, the embedded engineering team for the DSAWG (DevSecOps DoD Authorization Working Groups), now a part of the AOWG (Action Officer Working Group) at the DoD level. Team8 is focused on applying industry best practices for embedded systems in the defense environment, while also affecting policy within the DoD community to reform the way we approach embedded systems. Mr. House has been in embedded engineering for over a decade, working in the marine electronics industry before joining the government. For the past 6 years he has been a government civilian, involved in everything from OFP to hardware virtualization.

 andrew.house.4@us.af.mil



Major
Jorge Ramirez

Major Jorge Ramirez is an Acquisition Program Manager in the U.S. Air Force and currently assigned to the Air Force Life Cycle Management Center's Armament Directorate. His 14-year active duty career is underscored by the rapid delivery of multiple combat capabilities. In his current assignment, he leads the redefinition of Air Force agile software development for embedded systems, is pushing the bounds of networked, collaborative and autonomous munitions, and is optimizing challenge-based acquisition for modular systems.

 jorge.ramirez@us.af.mil