MODELING, SIMULATION,
PROTOTYPING & VALIDATION

How the Cloud is a Mission Enabler for Embedded System Development

John Brabbs
B. Colby Jones, PhD

Overview

Introduction
Use Case
Testing Env
CASTLE Migration
Cloud Value Proposition
Cloud Challenges
Hybrid Cloud Approach
Conclusion

Disclaimer: Reference herein to any specific commercial company, product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the Department of the Army (DoA). The opinions of the authors expressed herein do not necessarily state or reflect those of the United States Government or the DoA, and shall not be used for advertising or product endorsement purposes.

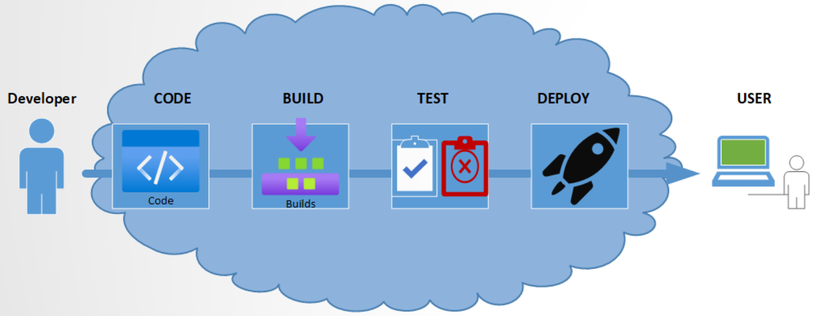
Introduction
Cloud – So What?

1. GVSC's SEC has put together a vision for software modernization to support the DoD Software Modernization strategy and the National Defense Strategy where the DoD will "prioritize speed of delivery, continuous adaptation, and frequent modular upgrades". [1]
2. GVSC's SEC supports multiple embedded system projects and needs the ability to "enable the delivery of resilient software capability at the speed of relevance." [2]
3. SEC sees the Cloud as a critical enabler to our software modernization strategy around embedded systems.

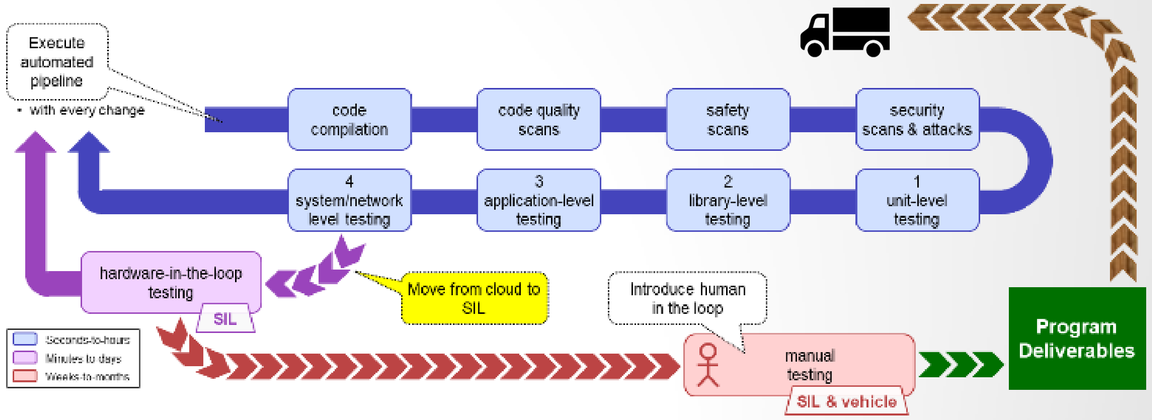
- The Cloud allows SEC to scale the automated test vision
- Supports the embedded system continuous integration (CI) pipeline.
- GVSC sees a Hybrid Cloud approach as the solution since it will allow data/applications to be shared seamlessly and allow SEC to selectively deploy/scale/move resources in the ways that make the most sense

USE CASE:
Embedded System Development

- The Army's ground vehicle embedded systems path to production differs from a web or mobile app path since the target hardware is a vehicle platform.
- Both can support Continuous Integration
- Web or mobile app can support Continuous Deployment, automated deployment
- Embedded Systems support Continuous Delivery, manual approval to deploy



Example of a web app path to production



Example of an Embedded System path to production [3]

Testing Environment

- Current environment is geared more towards manual testing of software in the SIL and the actual vehicle/system.
- Challenge - SEC is working towards a strategy to automated testing as shown in figure 1, that will provide the ability to scale using 100s to 1000s of tests with scalable computing before moving to the SIL and vehicle.
- On-Premises hardware only, doesn't provide the ability to easily scale the testing before getting to the SIL.

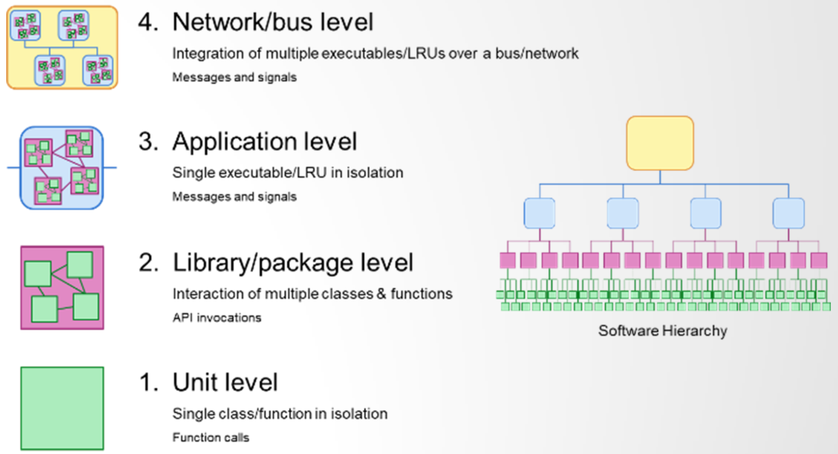
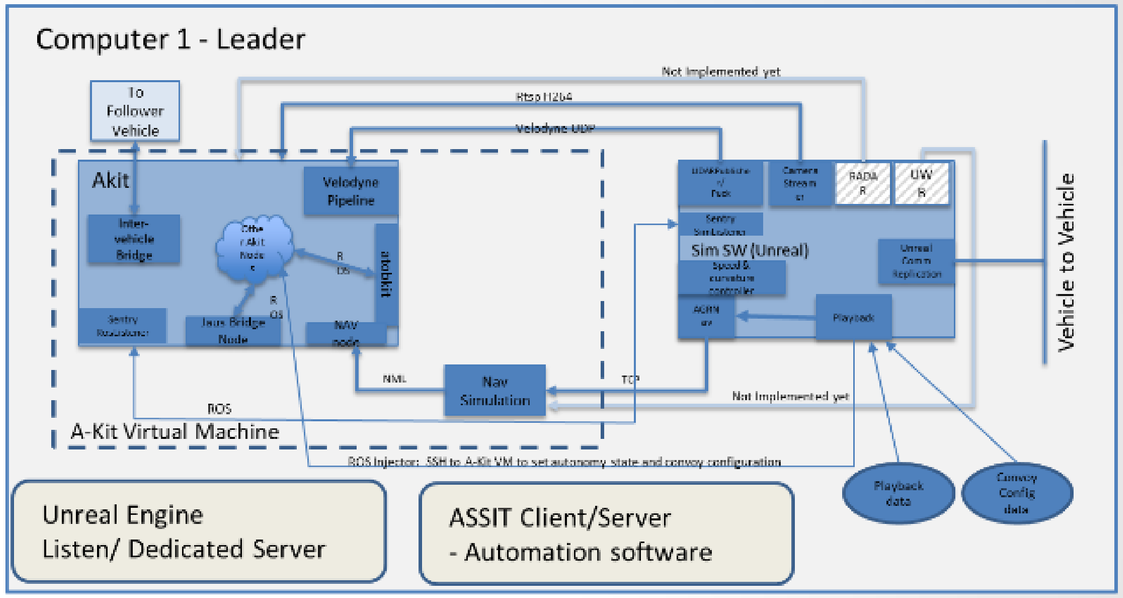


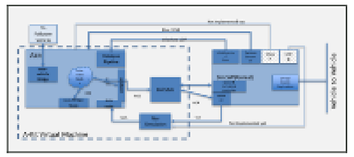
Figure 1: Automated Testing [3]

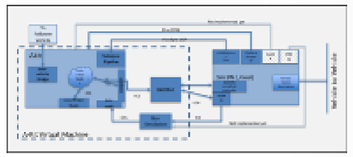
CASTLE Cloud Migration

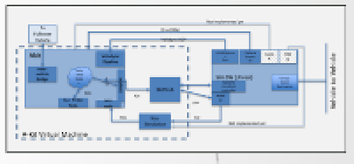
- Cloud migration of the CASTLE architecture to achieve compute scalability while allowing users to develop and run simulations remotely.
- Exploring two cloud architectures using containers: VMs and Kubernetes

Current Physical Architecture





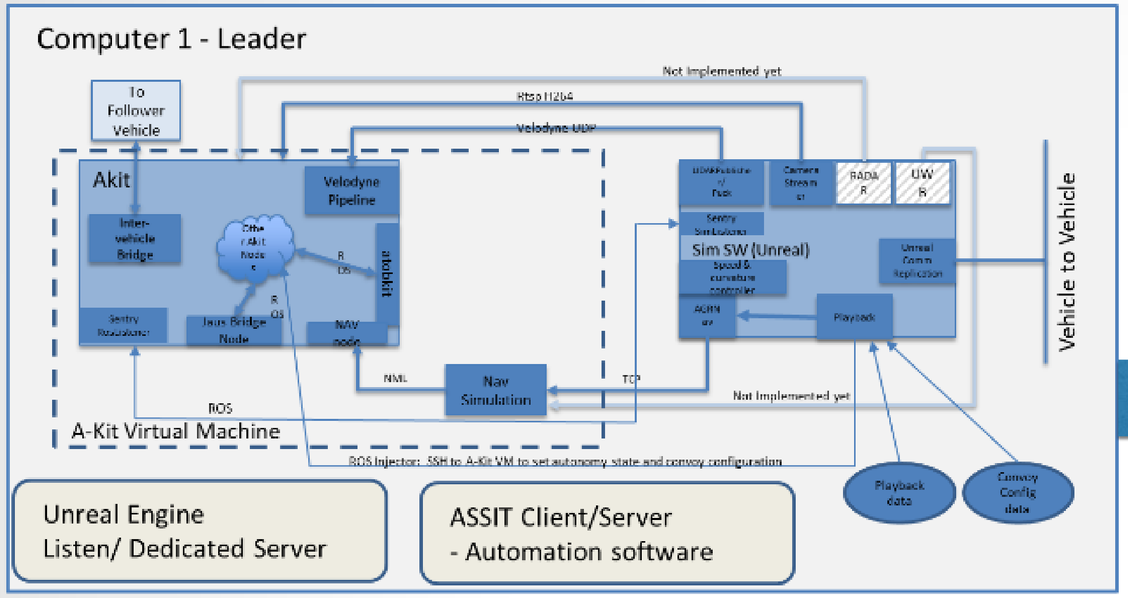




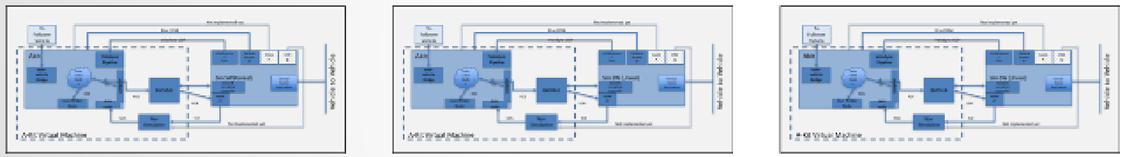
CASTLE Cloud Migration

- Re-architected on-premises solution running on multiple computers to run using containers.
- Leverage Infrastructure as Code (IaC) to spin-up and spin-down in the Cloud
- Solution#1: Run multiple containers in a VM for testing autonomy SW in Convoy scenario

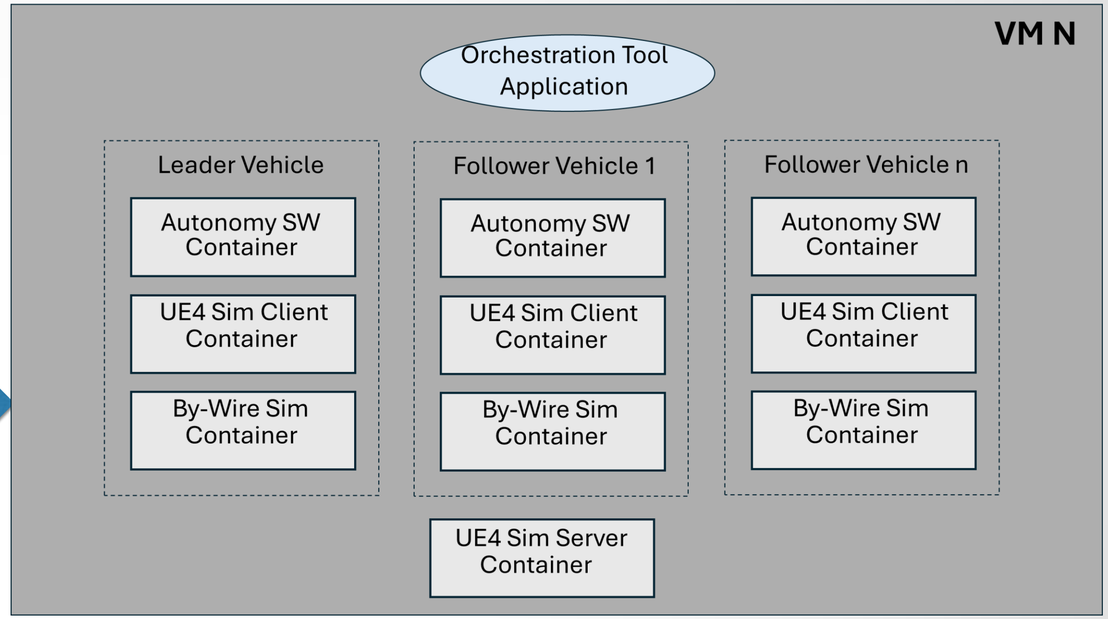
Current Physical Architecture



Computer 2 – Follower 1 Computer 3 – Follower 2 Computer 4 – Follower 3



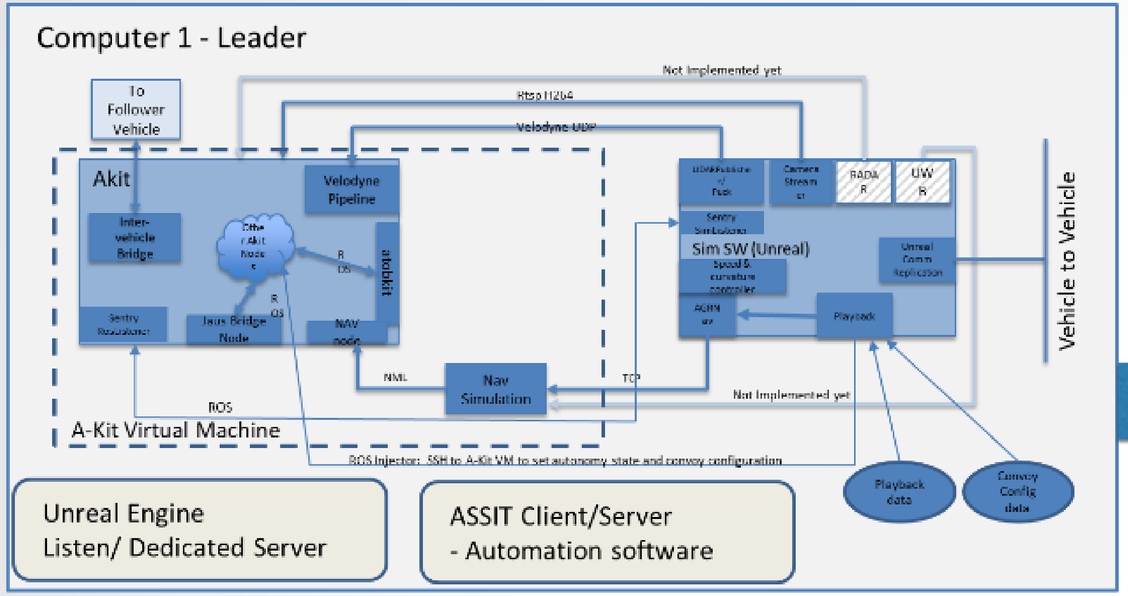
Cloud Virtual Architecture #1



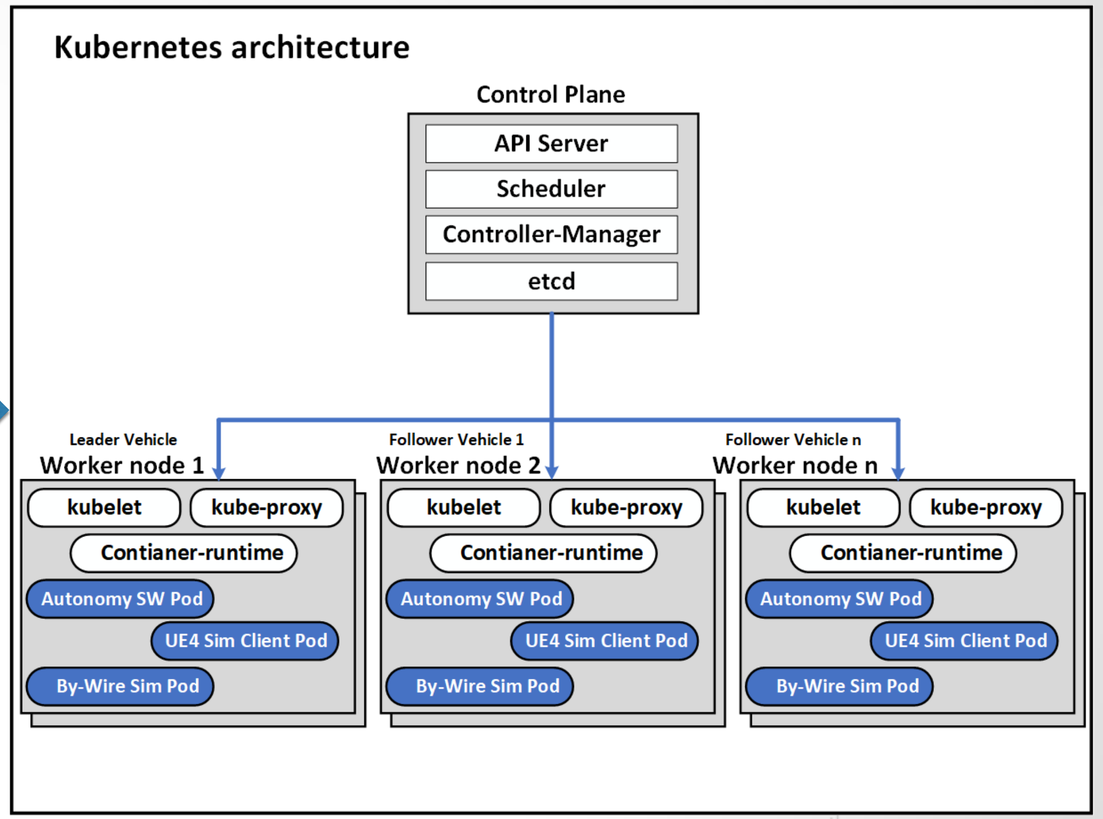
CASTLE Cloud Migration

- Re-architected on-premises solution running on multiple computers to run using containers.
- Leverage Infrastructure as Code (IaC) to spin-up and spin-down in the Cloud
- Solution#2: Run containers using Kubernetes for testing autonomy SW in Convoy scenario

Current Physical Architecture



Cloud Virtual Architecture #2



Cloud Value Proposition

Organizational Agility: Compute Resources with Flexibility

- The advent of Infrastructure as Code (IaC) and Configuration as Code (CaC) shifted the infrastructure build paradigm to treat environmental components as software constructs for placement under configuration management and reuse across the Cloud enterprise.

Automation, Automation, Automation

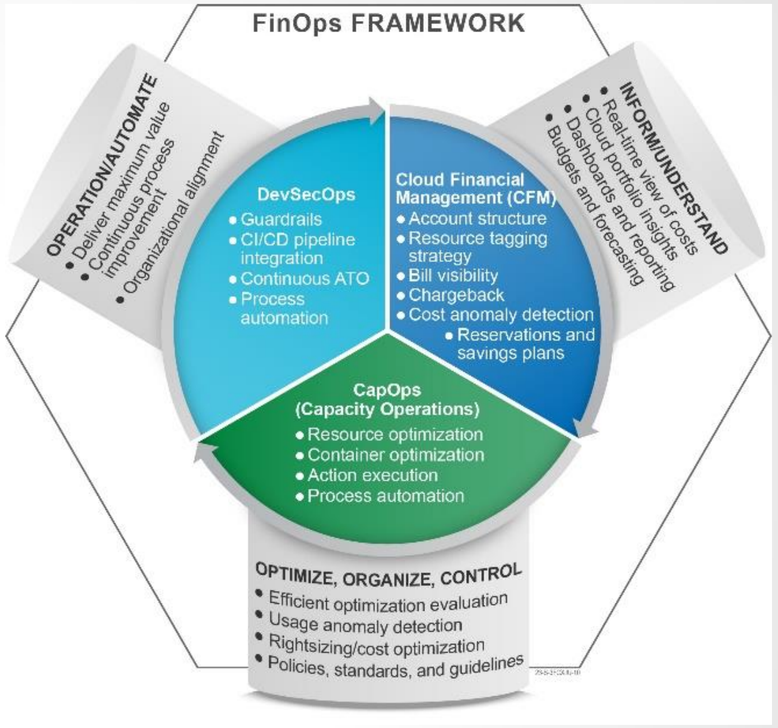
- Configuration version control
- Predictability
- Consistency
- Repeatability
- Composability

Fast feedback loops / Scalability

- Accelerate developer feedback

Cost Optimization

- Cloud resources are available on demand and torn down when not needed.



Cloud Challenges

Cost Management

- Orgs must manage the benefit to avoid unexpected costs. FINOPS.

Performance

- Understand usage models well, ingress and egress, GPU, network

Cybersecurity

- Cloud environments need continuous monitoring capability for intrusion detection, malware threats, user activity monitoring, and change management. ATO

Interoperability and Flexibility

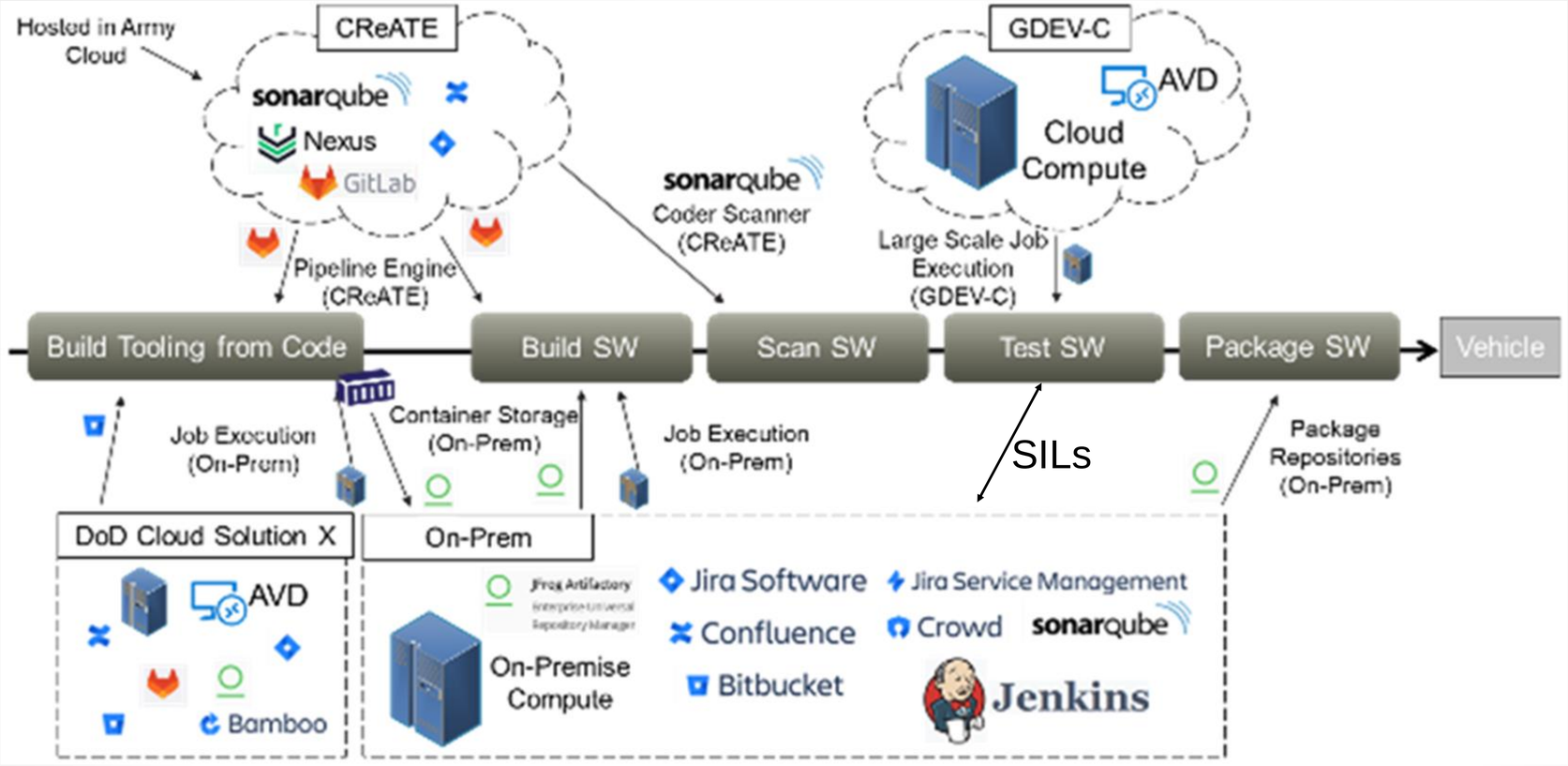
- Each CSP is different for Cloud-native services, this requires modifications to IaC/CaC when transitioning between Cloud platforms

Lack of Knowledge and Expertise

- SMEs to build, administer, and maintain the environment effectively.

Hybrid Cloud

The hybrid Cloud approach seeks to leverage the investment in on-premises compute and store resources while also tapping into the scalable, on-demand capabilities of the Cloud



Notional Hybrid Approach

Conclusion

- Cloud-based embedded software development requires a shift in thinking of the “Cloud” to a more **utilitarian** view rather than a “production” application hosting environment.
- A key piece of Cloud in supporting embedded system software development is creating simulations and emulations to **virtualize** the hardware for automated software testing, rather than only manual testing on real hardware.
- Cloud provides **scalability** in automated testing where 100s to 1000s of parallel tests are possible.
- Cloud allows multiple developers the **flexibility** to have consistent development environments and simulations accessible at any time, from any location.
- Extend current on-premise capabilities by adding additional **on-demand** software tools, simulations, emulations, and testing capabilities without investing in on-premises hardware that might sit idle for long periods until needed.

Questions

- Any questions?

REFERENCES

[1] J. Mattis, “Summary of the 2018 National Defense Strategy of the United States of America,” Department of Defense, 2018. Available: https://dod.defense.gov/Portals/1/Documents/pubs/2018-National-Defense-Strategy-Summary.pdf

[2] Kathleen Anne Holland Hicks, ‘‘Department of Defense Software Modernization Strategy,’’ Office Secretary Defense, Washington, DC, USA, 2022

[3] Wilkinson, Robert, “Software Modernization and DevSecOps”, Detroit Arsenal Acquisition Insight Days, AcqDays Topic 2-Cybersecuirty and DEVSECOPS, Nov. 2023, dau: https://www.dau.edu/events/acquisition-insight-days .