

Ground System Architectures Workshop

*Amplifying User Effectiveness:
Automation, Augmentation,
and AI/ML*

*February 24–27, 2025
Renaissance Los Angeles Airport Hotel
Los Angeles, CA*

Tailoring Ground Software Standards Towards Automation and Augmentation – Outbrief

***Alan Kwan
Jesus Rodriguez***

February 26, 2025



Overall Summary

Topics Discussed

- Comparison of the technical processes the two widely used software development standards
 - ISO-IEC-IEEE 12207-2017 Systems and software engineering – Software life cycle processes
 - SMC-S-012 – Space and Missile Systems Software Development Standard
 - *Business or Mission Analysis process*
 - *Stakeholder Needs and Requirements Definition process*
 - *System/Software requirements definition process*
 - *Architecture Definition process*
 - *Design Definition process*
 - *System Analysis process*
 - *Implementation process*
 - *Integration process*
 - *Verification process*
 - *Transition process*
 - *Validation process*
- Application of Agile (for Augmentation), DevSecOps (automation), and AI\ML (productivity)



Solutions / Recommendations

- Business or Mission Analysis and Requirements Definition Processes
 - *Stakeholder feedback loop*
 - Schedule at program increment or other major milestones
 - *Minimum Viable Product (MVP)*
 - Contract to define set of requirements that truly represent MVP to enable early transition to operations
 - Additional capabilities delivered in a true DevSecOps process
- Architecture, Design, Implementation, and Unit Test Processes
 - *Design for and prioritize CI/CD pipelines (e.g., modularity and interface for use of CI/CD tool sets)*
 - *Test as early as possible*
 - Test driven development (TDD)
 - Re-use unit tests to build CI/CD pipeline with automation
- Integration, Validation, Verification, and Transition Processes
 - *Integrate, validate, and verify incrementally and early*
 - *Transition MVP, then augment it*
- Measurement
 - *Early; software metrics; software factory metrics*

Test and deliver early (MVP), not just implement incrementally



Thank you