

# *Working Group Session Summary*

## ACE4: Architecture-Centric Evolution of Software-Intensive Systems

*Session 10e*

Chairs

Dr. Sergio Alvarado

Dr. Phillip Schmidt

The Aerospace Corporation

# Session Goals

- Promote the central role of software architectures during the acquisition and development of software-intensive systems
- Elucidate high-level recommendations for improving software architecture representations, development, and design
- Focus on evolution, evaluation & elaboration of descriptive/prescriptive architectures within the system acquisition & development lifecycle

# Presenters/Panelists

- **Acquisition Perspective**
  - Col. Michael Coolidge, Air Force Space and Missile Center
  - Lt. Col. William Page, Air Force Space and Missile Center
  - 1Lt. DeWitt Latimer IV, Air Force Space and Missile Center
- **Oversight Perspective**
  - Frank Sisti, The Aerospace Corporation
  - Dr. Peter Hantos, The Aerospace Corporation
  - Dr. Phillip Schmidt, The Aerospace Corporation
- **Development Perspective**
  - Richard Anthony, General Dynamics
  - Leeha Herrera, John Hopkins University Applied Physics Laboratory
- **Research Perspective**
  - Eric M. Dashofy, University of California, Irvine
- **Moderator**
  - Dr. Sergio Alvarado, The Aerospace Corporation

# Key Points

- Acquisition
  - Challenge: Integration of systems that were never meant to be integrated
    - Follows from focus on net-centric systems
  - There is a “sweet spot” for standards/architecture development between overly-generic and stovepipe systems
  - Beginning to define architectures early in the acquisition process improves outcomes—acquirers should do their homework
  - Adopt a step-based approach to development with later blocks taking on less mature technology
  - Having dedicated software people/advocates inside the acquisition house promotes early guidance to the contractor

# Key Points

## ■ Oversight

- Leveraging software & architecture people from across the oversight organization increases horizontal engineering
- Lifecycle models are key in ensuring the synergy across architecture evolution, elaboration, and evaluation
- Architecture evolution should be evident in the evolution of architectural views
- Evaluating the product is just as important as evaluating the process to create the product
- Use UML profiles to manage evolutionary change of complex architectural models
- Reference architectures are key platform-independent models to characterize goals, requirements, feasibility, and system variability

# Key Points

- Development
  - Lessons moving from stovepipe to reuse to reference architecture
    - Increased reusability, shared cost
    - More validation by more programs
    - Platform for more future development
    - Higher initial costs
    - More communication issues
    - Configuration management issues
  - Develop high-level architectures early to use as drivers for the remainder of development
- Research
  - All stakeholders should agree early on the goals, benefits and limitations of a reference architecture
  - Architecture-centric tools, used properly, can provide improved architecture communication, representation, and analysis

# Conclusions

- Acquisition and Oversight Perspective
  - Increased recognition of the need for early definition of software requirements being reflected in ongoing organizational changes
- Development and Research Perspective
  - Case studies indicate that early inception and definition of architecture results in successful outcomes
- Stakeholder-centric views of architectures should serve as the basis for stakeholder negotiation