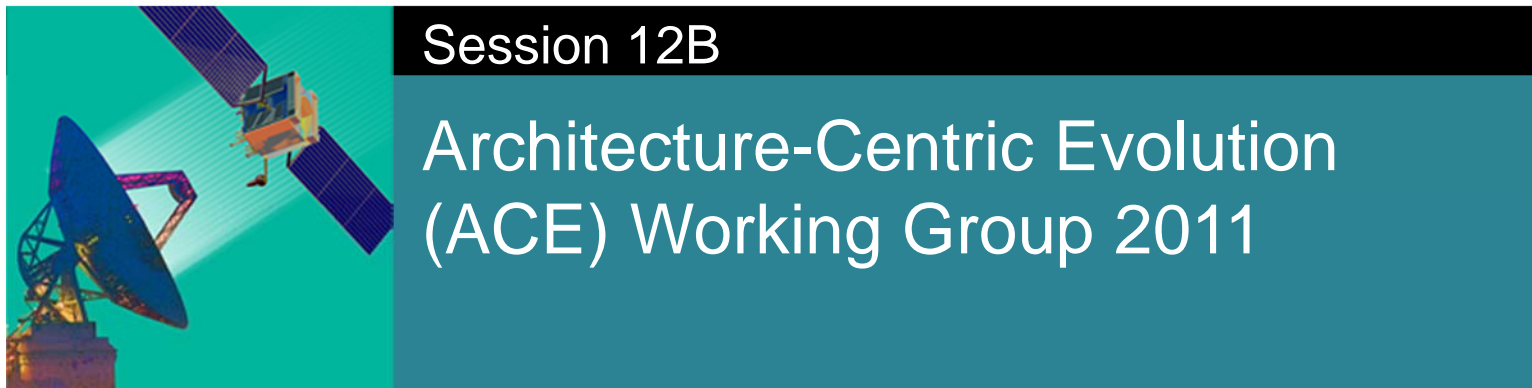


Working Group Outbrief

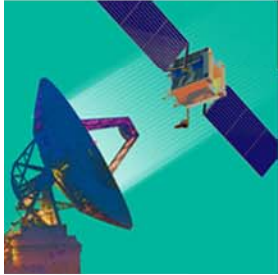
Ground System Architectures Workshop



Sheri Benator, Sergio Alvarado, The Aerospace Corporation

Team: Jeff Estefan, Jet Propulsion Laboratory; John Arcos, Phil Schmidt; Scott Hendrickson, The Aerospace Corporation

Ground System Architectures Workshop

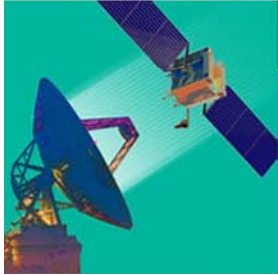


Session 12B

Session Goals

- Forum for acquirers, users, developers & researchers to collaborate to improve software architecture representation, development, & analysis
- Topic
 - Architecture Techniques & Challenges: Harmonizing new and reuse application and infrastructure software
- Presentations & panel discussions
 - Focus on integrating and harmonizing new, reuse, commercial, and open source application and infrastructure software
 - Current applications (experiences and lessons learned)
 - Research for future application

Ground System Architectures Workshop

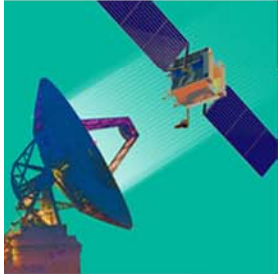


Session 12B

Presenters/Panelists

- Experiences and Lessons Learned
 - Reuse
 - Reuse: Dealing with the Hand You are Dealt
 - Jonathan Haulund, Los Angeles Air Force Base, SMC
 - Beyond the Code: Lessons Learned in Software Reuse
 - Amanda Ragan, Northrop Grumman
 - Enterprise Architecture
 - Architecture: Controlling Chaos by Defining Complex Interdependencies
 - Brian Giovannoni , JPL
 - Integrating Legacy Software: Lessons and Hurdles
 - John Chobany, Aerospace Corporation
 - Ground System as an Enterprise - Software Architecture Considerations
 - Gerald J. Dittberner, Harris Corporation

Ground System Architectures Workshop

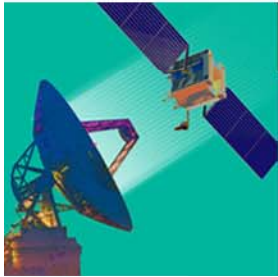


Session 12B

Presenters/Panelists

- **New Research Techniques**
 - Modeling and Evolving Product Line Architectures Using Change Sets and Relationships
 - Scott Hendrickson, UCI
 - Cataloging and Detecting Architectural Smells
 - Joshua Garcia, USC

Ground System Architectures Workshop

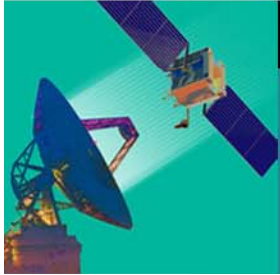


Session 12B

Key Points - Reuse

- Reuse: Large, complex and esoteric implementation with a proven track record... why not?
- Reuse Reality
 - Good, Bad, and Ugly software reuse
 - Prototypes and engineering software become operational products
 - Reuse of legacy software to support new missions that are not compatible with the legacy systems
 - Architecture of SW being reused and reuse expertise are major success drivers
- Reuse Recommendations
 - Perform Feasibility Studies
 - Spend time and money before you waste time and money
 - Analyze reuse architecture/design in context of new product requirements
 - Prototype for new environment/new requirements
 - Need to consider several areas
 - Code Design; Process; Stakeholders; Standards

Ground System Architectures Workshop

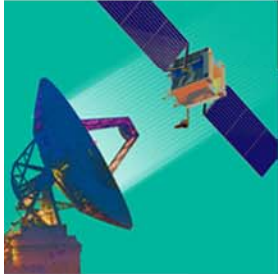


Session 12B

Key Points – Enterprise Architecture

- Impacts/Potential Obstacles to Cohesive Architecture
 - Organizational Structure
 - Process & Cultural Differences
 - Conflicting goals, schedules
 - Management Support
 - Control of the Money
- Social engineering as important as technical engineering

Ground System Architectures Workshop

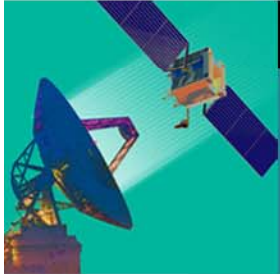


Session 12B

Key Points – Enterprise Architecture

- Consolidating legacy “stove-pipes” for commonality
 - Not always the best alternative for reducing program costs
 - Transition costs to go from legacy to new not always assessed
 - Interface complexity plays an important role
 - At some point wrapping legacy code reaches diminishing returns on performance
 - Concept studies/upfront modeling enable informed programmatic decision making
 - Development and maintenance costs of common services (or shared capabilities) need to be supported by the missions using those services
 - Challenge to implement common service and mission-unique approaches within the same ground system architecture
- Harmonizing Opportunities
 - Off the Shelf vs custom development
 - Security control with separate zones
 - Separate environments for operations, development and integration and test
 - Integrating the components

Ground System Architectures Workshop

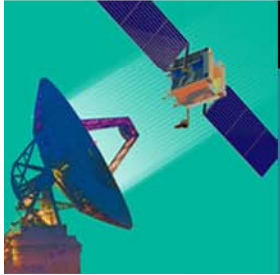


Session 12B

Key Points – Research

- When and where to refactor a software system's architecture?
 - Architecture Erodes over the Years – Understandability, Testability, Extensibility, Reusability
 - Categorization of Architectural (Bad) Smells
 - Architectural decisions that negatively impact lifecycle properties
 - Architecture Recovery/Representation Techniques & Smell Detection
- Change Sets and Relationships are a New Approach to Modeling and Evolving Software Product Line Architectures
 - Change sets encapsulate logically-related variability
 - Relationships govern valid change set combinations
 - Reduces overall redundancies, scattering, tangling

Ground System Architectures Workshop



Session 12B

Conclusions

- Social aspects often dominate technical aspects
 - Importance of a unifying vision
 - Consensus building at the grass roots level
- Reuse needs to be analyzed for appropriateness
 - Design decisions of the original system may not match the new context
 - Reevaluate prior system behaviors/trade space in new context
- Following best practices in SW architecture enables opportunity for reuse
 - Architect for deployment neutrality