

Control Channel Toolkit

Methods and Technologies for a Reusable SGS Architecture



Mike Grier
CCT Lead Engineer
Raytheon Systems Company
16800 East CentreTech Parkway
Aurora Colorado 80011
mgrier@redwood.dn.hac.com



Position Summary

- COTS and legacy integration not the end goal
- Systematic reuse requires
 - Organizational focus
 - Architecture focus
 - Open standards focus
- These complement but do not require COTS use and legacy reuse



Common Goals

- Reduce Space Ground System (SGS) lifecycle costs
 - Development reuse
 - Common maintenance
- Increase functionality
- Improve integration and interoperability



Approaches

- Code cloning
 - Developmental savings
 - Multiple maintenance baselines
- COTS integration
 - Development and maintenance savings
 - Beneficial for 2σ solutions
 - Difficult to evolve
 - Requires strong middleware solution
- Product line development (CCT approach)



Product Line Development

- Stakeholder-focused organization
- Common system requirements
- Targetable software architecture
- Common but tailorable assets



CCT Product Line Approach

- Organize around customer and reusers
- Analyze variability
- Define component-based architecture
- Use open standards
- Establish common solution



Stakeholder-focused Organization

- Primary difference between CCT and single-vendor COTS solutions
- Cooperative methodology and business model
- Shared costs and decision making
- Direct reuser involvement



Variability Analysis

- Analyze for reuse
 - Analyze stakeholder specifications
 - Survey diverse systems for driving requirements
- Define product line requirements
 - Common capabilities
 - Variation points for integrating mission-unique capabilities



Component-based Architecture Definition

- Package for reuse
 - Group logical entities to enhance cohesion, limit complexity
 - Use common services throughout
- Flexible to operations concepts
 - Software architecture supports many target architectures
 - Reusers target towards operations or driving requirements



Open Standards

- Promote interoperability
- Permit COTS flexibility
- Applicable standards
 - C++/C/FORTRAN
 - Motif
 - POSIX
 - CORBA



CORBA

- Open component-based architecture
 - Access to growing array of services and facilities
 - Interoperability and flexibility
 - Growth in realtime systems
- COTS and Legacy Integration
 - CORBA multi-tier solution
 - Numerous platforms, languages supported
 - Defers coupling to CORBA binding



Establish Common Solution

- Identify costs of non-standard features
 - Discourage “novel” vehicle features
 - Permit migration of mission-unique features into product line
- Support multiple implementations
 - CCT-based
 - COTS-based
 - Legacy-based



Summary

- Product line approach focused on systematic reuse
- Open standards promote COTS and legacy integration
- CCT uses both