

NORTHROP GRUMMAN DEFINING THE FUTURE

Mission Systems

March 5, 2003

Usability Constraints on Architecture Development and Use

Ground Systems Architecture Workshop
Manhattan Beach, CA

John Reeves

Department Manager, Domain Systems
Engineering



Architecture Usability Constraints

- **Stable Early in the System Development Lifecycle**
- **Distinguish Component Boundaries and interfaces**
- **Promote Communication between Component Developers**
- **Organize System Level Construction, Integration and Testing Plans**
- **Support Component Level Trade Studies**



System Architecture Characteristics

- **Descriptive Requirements - Define a decomposition of the system such that**
 - 1. System level behavior can be described**
 - 2. Component requirements/capabilities can be allocated**
- **Usability Heuristics**
 - **Minimize number of components**
 - **Minimize the amount of documentation**
 - **Minimize domain specific jargon and representations**



Architecture Methodology

- **Architecture Definition (Pre-PDR)**
 - **Operational Model (Use Case) - Define system level behavior**
 - **Structural Model – Define the components and how they interact**
 - **Construction Plan – How each component will be built**
- **Architecture Implementation (Post-PDR)**
 - **Structural Model is stable, used to organize requirements, development**
 - **Operational Model is used to define testing and integration program**



Architecture Usability (1 of 2)

Architecture Use	Recommendations
As a Blueprint for Evolution	Used for evolution from system to system. Organizational use in system development.
Multiple Views	Limit to structural and operational, other views generated as needed for design, construction or trades
As a Decision Making Tool	Used for front-end component level trades
Architecture Representation	Lowest common denominator tools
Component based Architecture	Components should be defined to allocate functionality, identify interfaces, and plan their construction



Architecture Usability (2 of 2)

Architecture Use	Recommendations
Requirements Verification	Components should address requirements allocation only
System Testing	Operational view should outline system level test plan (I.e. Use Case driven testing)
System Implementation	Components should address feasibility of construction
Managing COTS products change	COTS shouldn't be part of the architecture, but part of the construction that realizes an architecture
COTS product independent architecture	



Architecture Recommendations

- **Minimal C4ISR Framework**
 - **Operational Model – Small set of Use Cases supporting key system behaviors**
 - **Systems Model - Small set of Components**
 - **Technical Model – Development Constraints**
- **Focus on External Behavior of System, Components**
 - **Defer internal behavior to design**
- **Focus on Common Understanding of Development Team**
 - **Defer formal detail to design**