Introduction

- Target Audience for This Presentation:
  - SW Architects at all levels on a project
  - Stakeholders for SW Architecture and Design
    - Government Customers, Prime Contractor, Managers, System Engineers, SW designers and implementers, Testers, …

- Your Software Will Have an Architecture
  - Whether you document it or not
  - Whether you manage its definition or not

- SW Architect Defines the Vision for all the Architecture and Design that Occurs
  - This Includes Architectural Viewpoints and Views (IEEE 1471) in UML

- Views Described Here based on Wiley book on Large-Scale Software Architecture (reference at end)

- While Focus is Large-Scale, Approaches Described Here also Work Well for Smaller Systems and Embedded Systems
Run-Time vs. Build-Time Architecture and Design

- **Run-Time**
  - SW Components
    - Run-time grouping of objects with well-defined interfaces
  - SW Processes
    - Look at Task Manager process tab or Unix ps command
  - SW Deployment
    - How does SW map to hardware processors, nodes, etc.

- **Build Time**
  - Subsystems
    - Collection of classes, maps directly to directory in build
  - Classes
    - Encapsulation of attributes and methods

- Keep Run-Time / Build-Time Distinction Clear in all SW Design Documents
Run-Time / Build-Time View Examples

Run-Time
- COMPONENT
- PROCESS
- DEPLOYMENT

Build-Time
- CLASS
- SUBSYSTEM
Hierarchical Component (Run-Time) Architecture

Top-Level Component View

Domain/Segment Level Comp Views

CSCI-Level Comp Views

Colors indicate CSCI mapping of Components
Top-Level SW Architecture Document (TLSAD)

- TLSAD Document Contents
  - System Overview
  - Top-Level Component View
    - Component Descriptions
    - Interface Descriptions
  - Domain-Level Component Views
    - Component Descriptions
    - Interface Descriptions
  - Other Information
    - System-Wide Decisions / Constraints
    - Mapping of Interfaces to Documents

- Owned by SW Architect for the System Under Design
- Interface Definitions Best Managed in a Tool or DB
Mapping (Reuse) of TLSAD Information

TLSAD
- System Overview
- Top-Level Component View
- Domain1 Comp View / Interfaces
- Domain2 Comp View / Interfaces

SDD-1
- System Overview
- Top-Level Component View
- Domain1 Comp View / Interfaces

SDD-2
- System Overview
- Top-Level Component View
- Domain1 Comp View / Interfaces
Effectively Capturing SW Architecture and Design in the SDD

SDD-x (Content maps to MIL-STD-498 and DoD 2167A)

1. Overview
   System Overview
   Top-Level Component View
   Domain1 Comp View / Interfaces

2. CSCI Architecture
   Design Decisions
   CSCI Comp View / Interfaces

3. Hardware Utilization
   Deployment View / Process Descriptions

4. Detailed Design
   Subsystem1 Views / Descriptions
   Subsystem2 Views / Descriptions

© 2007 General Dynamics, All Rights Reserved
Conclusions

- Management of SW Architecture Complexity is Critical to Success of Large-Scale SW Systems
- Keys to Success
  - Run-Time vs. Build-Time
  - Hierarchical Definition of SW Architecture
  - TLSAD Provides Centralized Management of Component Architecture and Interfaces
  - Consistent Approach to SW Architecture and Design Enhances Communication
    - Within the Project and with Customers
References

- Large-Scale SW Architecture (Wiley) – Garland/Anthony
  - largescalesoftwarearchitecture.com
- Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives (Addison-Wesley) – Rozanski / Woods
- Pattern-Oriented Software Architecture, Volume 1 (Wiley) – Buschmann, et al.
- Patterns of Enterprise Application Architecture (Addison-Wesley) – Martin Fowler