Integration of Software Intensive Systems

Tom Wissink
Lockheed Martin Fellow
IS&GS, Gaithersburg, MD
301-240-6244
tom.wissink@lmco.com

October 2008

Copyright 2008 Lockheed Martin Corporation. All rights reserved.
Agenda

- Introduce the Problem
- Integration Definitions
- Integration throughout a Development Lifecycle
- Integration: Techniques, Methods
- Integration Support Activities
- Wrap-Up
Introduce the Problem

- No consistent definition & process:
  - Results of a Web Search “System Integration is the successful integration of a new technology into the system by analyzing the technology's system effects and resolving any negative impacts that might result from its broader use.”
  - From the International Council on System Engineering (INCOSE) web site – “Integrate: . . . Systems, businesses and people must be integrated so that they interact with one another. Integration means bringing things together so they work as a whole. . .”
Introduce the Problem (cont’d)

- My favorite published definition:
  - Integration is defined as the act of mating hardware and/or software components, subsystems, systems or elements at their respective interfaces and verifying the compatibility and proper operation of the integrated units.
  - From a paper entitled “Integration Challenges of Complex Systems” written by Bill Haskins and Jack Striegel for the 16th Annual INCOSE International Symposium,
  - No complete guidance on how to do Integration
Integration throughout the Lifecycle

SRR
- System Requirements
- Operational Scenarios
- System Verification Plan
- Interface Control Docs
- Integration Threads
- System Integration Plan

SDR
- Subsystem Requirements
- Interface Design Documents
- Subsystem Verification Plans
- Subsystem Integration Plans

PDR/CDR
- Product Specifications
- Product Verification Plans
- Product Integration Plans

Goal I&V Documents:
- Draft at SRR, SDR & PDR
- Finals at CDR

Customer Requirements
- Customer Concept of Ops
- Customer Validation Plans
- Cost, Schedule, Risks

Operations Environment
- TEMP Driven

System & SoS Validation
- Procedures/Reports
- Operational Test & Evaluation

System Verification Testing
- Verification Test
- Procedure/Report
- System Integration

Subsystem Verification
- Verification Test
- Procedure/Report
- Subsystem Integration

Product Verification
- Verification Proc/Report
- Product Integration
- HW & SW Component/Unit Test

Detailed Design & Assemble/Build Hardware & Software Products
Integration: Techniques, Methods & Tools

Two Techniques:

- Non-Incremental* (Big Bang) vs Incremental*
  - Incremental is the way to go for most systems and large applications
  - Integrate/Build-Up – starting small and continually increasing capability/complexity

* References for Techniques and Methods
Integration: Techniques, Methods & Tools

Three Methodologies:
- Top-Down*, Bottoms-Up* & Thread-Based
Three Methodologies:

- **Top-Down**

Diagram showing a top-down integration methodology with nodes labeled A, B, C, D, and others connected in a hierarchical structure.
Integration: Techniques, Methods & Tools

Three Methodologies:
- Bottom-Up

Driver B
F
J

Driver A
H
K
L

Diagram showing the integration process with nodes A through I connected to H, K, and L.
Integration: Techniques, Methods & Tools

Three Methodologies:

- Thread-Based
  - Experience indicates this is the preferred method for most large complex applications and or systems
Integration Support Activities

- Interface Matrices (Interface Coverage)
  - Account for all internal & external interfaces

- Hardware/Software/System Build Plan
  - Thread based and negotiated with the developers

- Dedicated Integration Laboratories
  - Separate from Test Laboratories

- Early “ilities” Checkout during integration phases
  - Stability
  - Reliability
  - Performance
  - Capacity
Wrap-Up

- Integration requires a different skill set than Testing.
- Lessons learned have shown that Integration is a key weakness on most medium to large software intensive projects.
- Perform the Top Ten Integration steps and you will have a robust Integration process.
Top Ten Integration Steps

1. Document the Integration and Test process
2. Hire and train the right staff for the role of Integrator
3. Review and analyze requirements to ensure testability and included requirements to ensure visibility into system data while it is operating
4. Ensure all interfaces at all levels of the architecture have been identified and are implemented, tested, tracked, and statused
5. Identify & plan other testing activities to start during the integration test conduct phase (i.e. stability, performance, reliability, etc)
6. Develop and maintain a Project “Build Plan”
7. Define and ensure sufficient Integration and Test laboratories available
8. Design integration tests and test data for all levels of the architecture
9. Ensure functional testing is also being conducted at each level of the architecture
10. Ensure sufficient simulation/stimulation capabilities are available