Product Line Development Through Systematic Reuse Using Commercial Processes

Jeffrey A. Shaw
CCT Program Manager
Raytheon Corporation
(303) 344-6153
jshaw@redwood.dn.hac.com
• Product Lines
  – Support multiple users
  – Support multiple missions
  – Core underlying architecture

• Systematic Reuse
  – Forward engineer legacy systems
  – Leverage previous investments

• Best Practices
  – Organize for focus
  – Demand value-added processes and products
  – Collaborate with a community

*Lower Cost, Decrease Schedule, and Increase Quality*
Scoping a Product Line

- Domain Definition
- Generalized System Requirements Specification
- Variation Points
- Reference Architecture

DCCS Reqs
SSCS Reqs
Malta Reqs
Architecture Survey
A Definition of a Product Line

Office Building, Department Store, and Warehouse...

- Common building materials are used across different projects
- Standards ensure compatibility of building materials from competing vendors
- Each project has its own architecture
- Each project uses common design and construction processes
- Projects customize only where necessary
- Projects share benefits of common infrastructure
A Satellite Command and Control System Product Line

- Open standards-based common software components provide building blocks for flexible software architectures

- Well-defined interfaces between software components permit technology evolution and lessen reliance on specific vendors

- Each program defines its own system architecture using the common components

- Programs customize and extend where necessary to implement their concepts of operation

- Programs share common development processes

- Programs achieve their unique ground system solution

- Programs share in development and maintenance costs
Take Advantage of What’s Available

Legacy Code Bases

Product Line Architecture

Proven Algorithms

COTS Products

Process-Driven, Systematic Reuse
Frequent Working TIMs

Engineering Product Focus

Accurate Progress Metrics

Quality Through Process
Focus on Risk Reduction

- Overall system requirements and architecture are defined up front
- Detailed development follows a spiral approach
  - Create system, category, component skeletons early
  - Focus on high-risk capabilities
- Support a front-loaded integration strategy
- Allow incremental review of maturing products
  - Design artifacts
  - Developmental prototypes
  - System thread demonstrations
A Collaborative Development Environment

- Utilize the expertise of all community members
  - Technical Support
  - End-users
  - Technologists
  - Developing contractor

- Extranet
  - Constant, instantaneous connectivity
The End Goals

- **Lower Lifecycle Costs**
  - Product line allows sharing of development and maintenance expenses

- **Decrease Schedule Duration**
  - Common capabilities provided
  - Focus on new unique requirements

- **Increase Quality**
  - Large base of existing capabilities with depth and breadth of operational usage