Sustaining Software-Intensive Systems - A Conundrum

Mary Ann Lapham
Agenda

What is Sustainment

Software Entrance Criteria for Sustainment

Selected Examples of Challenges

Conclusions
What is Sustainment?

Joint Publication 4-0 ( Doctrine for Logistic Support of Joint Operations): “The provision of personnel, logistic, and other support required to maintain and prolong operations or combat until successful accomplishment or revision of the mission or the national objective”

DoDI 5000.2: “The first effort of the Operations and Support (O&S) phase established and defined by DoDI 5000.2. The purpose of the sustainment effort is to execute the support program to meet operational support performance requirements and sustain the system in the most cost effective manner of its life cycle. Sustainment includes supply, maintenance, transportation, sustaining engineering, data management, Configuration Management (CM), manpower, personnel, training, habitability, survivability, environment, safety (including explosives safety), occupational health, protection of critical program information, anti-tamper provisions, and information technology (IT), including National Security Systems (NSS), supportability and interoperability functions. Sustainment overlaps the Full Rate Production and Deployment (FRP&D) effort of the Production and Deployment (P&D) phase.”
Software Sustainment - Definition

Software maintenance and software sustainment are often used interchangeably. We will make a distinction.

**Software Maintenance**: “The process of modifying a software system or component after delivery to correct faults, improve performance or other attributes, or adapt to a changed environment.”

No definitive definition of software sustainment has been found, so our working definition is:

**Software Sustainment**: “The processes, procedures, people, materiel, and information required to support, maintain, and operate the software aspects of a system.”
Software Sustainment - Distinction

Software maintenance consists of:
- correcting the faults
- improving performance or other attributes
- adapting to a changed environment

Software sustainment addresses other issues not always included in maintenance, such as:
- operations
- documentation
- deployment
- security
- CM
- training
- help desk
- COTS management
- technology refresh
Software Entrance Criteria for Sustainment

Example

- Signed Source of Repair Assignment Process (SORAP)
- Completed Operational Test & Evaluation
- Stable software production baseline
- Complete current software documentation
- Authority to Operate (ATO)
- Sustainment Plan
- Current & negotiated sustainment transition plan
- Sustainment staff training plan

Key:  
- Green: Good  
- Yellow: Caution  
- Orange: Warning  
- Red: Critical

Note: Based on SEI experience with actual programs
Selected Examples of Challenges

Programmatic

COTS

System Transition

Information Assurance (IA)

• Indicates Major Concern
Programmatic Issues

• Transition tasks in development contract
• Funding for development contractor
• Funding for sustainer
• Transition plans created and implemented
• Sustainer training created

• Signed SORAP required
COTS Obsolescence & Upgrade Planning

New complexity to handle

• Demand for features

System Context

• Unsupported releases
• No market demand

Simultaneous Definition and Tradeoffs

COTS Obsolescence & Upgrade Planning

Fielded System Instances

Incremental system releases

Marketplace

• System development
• Tech refresh, system sustainment

COTS Vendors

© 2006 by Carnegie Mellon University

Version 2

GSAW 2006 - page 9
COTS "Business" Issues

- New “business” issues (licenses, data rights, warranties) to resolve
System Transition

- Staffing issues – loss of staff
- Training issues – get staff up to speed
- Complete documentation – hand-off
- Expert knowledge transfer/loss
Information Assurance

• Was security involved in system engineering decisions?
• How secure is the overall system?
Conclusions

Sustainment is a balancing act
• Definitions (what’s included)
• Planning
• Coordination on issues
• Risk Mitigation
Questions

Contact Information:

Mary Ann Lapham  
Software Engineering Institute  
Carnegie Mellon University  
4500 Fifth Avenue  
Pittsburgh, PA  15213-3890

mlapham@sei.cmu.edu  
412-268-5498