Objectives:

Discuss systems engineering practices of NSWC Dahlgren Division when carried out in the system of systems environment.

Discuss Dahlgren SoSE efforts and related system context, lessons learned, and challenges.

Opinions expressed are those of the principal author, and do not reflect official policy or positions of the Navy, Navy or DoD programs of record, or NSWCDD. With grateful acknowledgement to co-authors G. Goddin, J. Heil, J. McConnell, P. Pierce, G. Rivera, S. Such for valuable discussion and perspectives on best practice and lessons learned in SoSE.
NSWCDD's Systems Engineering Process
NSWCDD Perspective and inputs to the OSD SoSE Guide
Case studies, Best practice, Lessons learned

Chem-Bio Architecture Engineering
Naval Integrated Fire Control – Counter-Air
Combat Systems Engineering across Surface Ship Classes
Aegis Ballistic Missile Defense
Software Engineering
Affordable Weapons Systems
NSWCDD Systems Engineering*

Requirements Analysis

- Translate needs into feasible concepts
- Determine logical architecture alternatives; understand functional relationships; determine where and how functions are performed in the system
- Define the system solution

Functional Analysis and Allocation

- Develop clear understanding of customer needs
- Determine Operational Environment
- Decompose Functions
- Allocate Requirements to Functions
- Capture Functional Analysis & Allocation Results

Synthesis

- Decompose functions & build schedule

Systems Analysis and Control

- Provide visibility to ensure product quality and conformance to project plan; ensures best practice

* Based on MIL-STD 499 B &EIA/IS-632, 1994
Mission Level Requirements
Flow Into Programs of Record

A Subset of System Requirements
Addresses Integration for Mission Capability

Requirements Flow Down as Architectural Elements of Platforms and Systems

Multiple Missions, Multiple Acquisitions, Requirements Flowing to Different Levels Concurrently
A Highly Complex Engineering Endeavor Requiring Discipline, Competence and Tools
DoN Engineering of Systems
(a spectrum of Systems Engineering)

Systems Integration takes place at each level of the hierarchy and requirements are passed between levels of the hierarchy.
System-of-Systems Integration

Requirements

Production

V, V&C

Decompose

Force Focus

SoS

Capability Focus

System

Continual Assessment

In-Progress

Re-Aggregate

Functional Focus

Component

End Item Focus

Verify, Validate, Certify

In-Progress

In-Progress

In-Progress
Chem-Bio Architecture Engineering: System of Systems Approach to Counter the Threat

CB Threats & Hazards

- Agent Delivery
- Doses on Target
- Downwind Dispersal
- Doses Absorbed
- Symptoms

Medical Pretreatment

Individual & Collective Protection

Installation Force Protection

Contamination Avoidance and NBC Battle Management (Detection, Identification, Reconnaissance & Warning)

Medical Treatment

Information Systems

Decontamination, Restoration

Sustained Combat Power
Chem-Bio Architecture Engineering
Best Practices

- Architectures are useful in managing complexity
- Architecture framework (DODAF) facilitates the sharing of information and requirements among systems engineers and architects
- SE and Architecture tools are necessary to manage the complexity
- Managing CBRD requirements and gaps facilitates the identification of S&T opportunities that affect cross-Service capability
- Managing CBRD requirements for the services facilitates the identification of common elements resulting in life-cycle savings
- Open architecture concepts promote the ability to leverage needed subcomponent elements (specific algorithms from components rather than the total component)
Objective: Achieve Naval and Joint Integrated Fire Control capability against over-the-horizon and below-the-horizon AAW threats by distributing the AAW fire control loop across multiple PoR platforms.

Approach: Form a collaborative government/industry SoS systems engineering team by collecting lead engineers and managers from across all participating PoR systems. Develop IFC-unique operational concepts, systems engineering products and trade studies and allocate results to PoR programs.
Leadership, teaming and collaboration are essential to success for SoS development.

DODAF architecture is essential for definition and organization of SoS capability and the eventual allocation of unique functionality to existing and future PoR programs.

- Define capability within the OVs
- Compare OVs to similar SoS
- Expand the intermediate SVs
- Allocate functions to PoR SV-4s
- Add new PoRs via their SV-4s
Conduct SoS-unique systems analysis and trade studies as needed for critical functions.

- Identify SoS MOEs (measures of effectiveness).
  - Unique goals and objectives to be achieved by the SoS in order to accomplish the SoS mission.
- Identify PoR MOPs (measures of performance).
  - Parameters and functions unique to each PoR that contribute to overall SoS MOEs.
- Analyze and trade functionality and performance across the SoS.
  - Quantify results against the MOPs and roll up to the overall MOEs.
- Simulate and analyze SoS performance via low-fidelity (spreadsheets, MatLab tools) to higher-fidelity (federated PoR models) methods as feasible.
Combat Systems Engineering accomplished across platform combat systems via Product Line Approach

- Surface Combatant – Next Navy
- Surface Combatant – Navy after Next
- Product Line Systems Engineering
- New C SEA Functions
- Existing Platform Tech Teams
- CSEA Focus
- New Surface Combatants
- Current Surface Combatants
- Large-deck Combatant
- Each Program Executes to Its Schedule and Requirements (SIPM Focus)
Desired Future State
Product Line Acquisition

Scope Funding Allocated to Program

NCD Updates
Perf Studies
ICDs

Unique to ship / weapons system
Common across ships / weapons system

Platform
Specific

Unique Class 1
Common Class 1
Unique Class 2
Common Class 2
Unique Class 3
Common Class 3

Platform Combat System Upgrade Plan Based on Product Line Plan

PARM Roadmaps
Fleet Needs
Product Line SE
Product Line Development Plan
Objective Architecture Migration Plan

Fleet Needs
Product Line SE

Platform
Specific

Unique Class 1
Common Class 1
Unique Class 2
Common Class 2
Unique Class 3
Common Class 3

Platform Combat System Upgrade Plan Based on Product Line Plan

NCD Updates
Perf Studies
ICDs

Unique to ship / weapons system
Common across ships / weapons system

Platform
Specific

Unique Class 1
Common Class 1
Unique Class 2
Common Class 2
Unique Class 3
Common Class 3

Platform Combat System Upgrade Plan Based on Product Line Plan
Product Line SE / Upgrade Development Example – Notional Ship Class 1

<table>
<thead>
<tr>
<th>FY08</th>
<th>FY09</th>
<th>FY10</th>
<th>FY11</th>
<th>FY12</th>
<th>FY13</th>
<th>FY14</th>
<th>FY15</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product Line SE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ship Class 1 Obj Arch Definition</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Product Line Rqts. &amp; Architecture</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Capability POR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scoping</td>
<td>Capability Dev.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ship class 1 Systems Engineering (One for each Platform CS)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Requirements Architecture</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Production Build &amp; System Test</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SW DR SW DR SW DR SW DR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Component 1 Component 2 Component N</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RCIP-1A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RCIP-1B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Capability POR

Ship class 1 Systems Engineering (One for each Platform CS)

Ship class 1 Component Dev. & Integration (One for each Platform CS)

RCIP-1A

RCIP-1B

PEO

PARM

PSEA

Component Developer

RCIP Developer

22 Oct 09

NSWCDD Brief for NDIA 12 SE Conf 26-29 Oct 09
Common Weapons Control System Background

❖ CWCS is:
   ➢ Common system for preparing/launching various weapons across multiple warfare areas
   ➢ Applicable to various platforms (surface and sub-surface)
   ➢ Establishes open architecture environment for adapting/scaling new weapons/systems
   ➢ Moves Navy (& potentially Joint) weapons control away from creating NEW & modifying closed stove-pipe systems
   ➢ Leverages existing Naval systems (Tomahawk Weapons Control System, Navigation, C4I Systems, etc.)

❖ CWCS concept being evaluated by multiple NSWCDD department’s systems engineers

❖ Systems engineering artifacts and system prototype under development

Establish Common Weapons Control System for Navy Platforms and Weapons
CWCS SoSE Approach

- Leverage current surface and submarine systems
  - Weapons Control
  - Navigation
  - C4I
  - Networks (ship and sub-based)

- Follow established systems engineering processes

- Leverage established systems engineering products
  - Architecture, weapon system requirements specs, interface requirements, employment concepts, scenarios, etc.

- Integrate existing functionality to provide benefit to warfighter and taxpayer
  - CWCS integrates two existing systems
    - Naval Fires Control System (NFCS)
    - Tactical Tomahawk Weapons Control System (TTWCS)
  - Integrates Marine & Army fires networks and capabilities to all surface combatants
  - Coordination of fires
  - Reduces overall program cost and lifecycle support
CWCS SoSE Approach

- Leverage training curricula and documentation
- Leverage established training pipelines
- Joint interoperable with various systems (end-to-end)
  - Tasking from multiple sources
  - Battle Management & Coordination Systems
  - Situational Awareness systems (e.g. GCCS-M, J,…)
  - Manages and deconflicts multiple weapon variants for simultaneous weapon prep/launch
  - Threat data, obstruction data, etc.
- Leverage existing tactical data analysis and extraction applications/tools
- Leverage combat systems training and simulation functionality
Combat System Certification
Situation Before 2004

- Combat System Certification Processes Varied Widely Across Systems and Programs
  - Certification did not Occur for all Combat System Elements
  - Combat System Certification for SSDS & ACDS Ships was not Conducted
    - Fielded Through Existing SEA62 Fleet Delivery Readiness Review (FDRR)
  - Platform Certification for Aegis Ships was not Conducted
    - Assumed as Part of Aegis Combat System Certification
  - Certification Criteria not well Defined or Understood

- In-Service Programs Viewed Certification Largely as a T&E Event Vice a Continuous Process Throughout System Definition And Development
  - Quality Issues Drove Test / Fix / Test Loop
    - Drove Perception That Certification is Long and Expensive
  - Various Test Efforts Were not well Coordinated
    - Developer, Cert, CSSQT, DT / OT
**Combat Systems Certification**

**One Process – Four Phases**

**Examples**

<table>
<thead>
<tr>
<th>Aegis</th>
<th>Carrier</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reagan SG</td>
<td></td>
</tr>
<tr>
<td>DDG / CG</td>
<td>CVN / LH</td>
</tr>
<tr>
<td>ACS</td>
<td>SSDS CS</td>
</tr>
<tr>
<td>VLS</td>
<td>RAM</td>
</tr>
</tbody>
</table>

**Pre-decisional Draft**
Coordinating the Phases

CS Cert is Focal Point for all Certification Activities

- Coordinates and Aligns Element Certifications
- Administers and / or Oversees Critical System Integration
- Supports Platform Certification by Providing the Activities and Data to Fulfill the Warfare System-Related Platform Cert Criteria
  - Including Many of the WSIIT Requirements
- Also Provides Process and Means to Address Corrective Actions Required as a Result of Platform / Interoperability Cert Assessments

Well Coordinated Phases – No Duplication
Certification is Both a Process and an Act of Attestment

- **Continuous Assessment (Vice End-Game Only)**
  - Objectively Assess the Progress of the Development Effort to Reduce the Risk that the System will be Ready to Certify on Schedule
    - Assessment of Progress Versus Plan
    - Verification of System Efficacy and Quality
    - Identification and Resolution of Potential Certification Issues
  - **Authorizations**
    - Assess the Ability of a Specific Version of the System to Perform Specific, Well Defined Scenarios or to Perform a Limited Mission (Usually an At-Sea Test Or Trial)
    - Assess the Ability of the System to Operate Safely Within Documented Restrictions
  - **Certification Panels**
    - Assess the Readiness of a Specific Version of the System to Perform the Broad Mission Requirements of the System (Readiness for Full, Unrestricted Fleet Use)
    - Assess the Ability of the System to Operate Safely
Process Value-Added

- Provides a Structured, Systematic Assessment Methodology
- Applies Full Rigor: Defines Certification Activities, Detailed Tasks, Work Products, and Applicable Metrics IAW Industry Standards
  - Establishes Expectations for Developer-Executed V&V / Certification Activities and Artifacts
  - Assesses Developer’s V&V Work / Results
  - Defines Appropriate Degree of Independent Assessment Activity
- Fully Adapted to Evolutionary Acquisitions
- Coordinates With Other Critical Processes (e.g. Safety, CM, QA, etc.)
- Addresses: COTS / NDI, Reuse, HSI, Security, Safety, etc.
- Generates and Accumulates Technical Insight for Continuously Updated Assessments and Cert Status
- Builds in Accountability of the Cert Process Itself

Detects and Eliminates Defects and Risks Earlier
NSWCDD has 50+ year history of providing full spectrum SW Engineering and actual SW Development for multiple Combat and Fire Control Systems

Includes real-time, safety critical, complex algorithms, multi-process, multi-interface tactical and simulation sw design, code, and test

Participation in cross organizational and cross discipline (SE/SW/Test) IPT and Leadership of Industry and Government Engineering SW Development IPTs

- Pro-active SW expert participation in from Concept Development through System Requirements, System Development, Deployment, and Operational Support

Demonstrated success in developing Open Architecture based multi-platform capable, re-usable, scalable, and maintainable software components

Applied Software Engineering and Development Expertise
SoS Mapping Software Levels

SYSTEM

Component Level

Segment Level

SoS either targets an entire system or...

Component YY

Segment ZZ

SW CSCI 1

SW CSCI 2

SW CSCI ###

Critical Software Components

SAME OA REQUIREMENTS AT THE SW CSCI LEVEL AND BELOW

- Open Standards
- Reuse
- Modularity
- Extensibility

- Maintainability
- Interoperability
- Composability

SYSTEM

Functional Focus

COMMON COMPONENT

Object 1

Files 1

Files XXX,XXX

SLOC 1

SLOC 2

SLOC XXX,XXX

SoS targets a specific slice of software enabled functionality resident in the system

SoS Capability Focus

Common Hardware and Operating Systems

22 Oct 09

NSWCDD Brief for NDIA 12 SE Conf 26-29 Oct 09

25
Software Lessons Learned
Open Architecture is more than just “Reusability”

Composability
The System Provides Recombinant Components that can be Selected and Assembled in Various Combinations to Satisfy Specific Requirements

Interoperability
Ability of Two or More Subsystem to Exchange Information and Utilize that Information

Open Standards
Standards that are Widely Used, Consensus Based, Published and Maintained by Recognized Industry Standards Organizations

Maintainability
The Ease With Which Maintenance of a Functional Unit can be Performed in Accordance With Prescribed Requirements

Extensibility
Ability to add new Capabilities to System Components, or to add Components and Subsystems to a System

Modularity
Partitioning into Discrete, Scalable, and Self-Contained Units of Functionality, With Well Defined Interfaces

Reusability
Ability for an Artifact to Provide the Same Capability in Multiple Contexts

Diagram Key

These OA “ILITIES” Cannot be Easily Verified by System Testing Alone. Gov’t SW Expertise Insight Into Design and Code is Required to Ensure Reusable Software.

Designing and Coding for These “ILITYES” is the Key to Saving Significant $$$$$$$$. 

* Reference: OA Architectural Principles and Guidelines v 1.5.6, 2008, IBM, Eric M. Nelson, Acquisition Community Website (ACC) DAU Navy OA Website
**SW Lessons Learned:**
Levels of SW Complexity / Devil is in the Details

Gov’t SW SMEs must ensure OA req’s are met at the most detailed levels of SW design for:
- Open Standards
- Reuse
- Modularity
- Extensibility

Gov’t SW SMEs must understand the technical design and details for complex:
- Data & File Management
- Threading & Tasking Hierarchy
- Initialization /& Termination
- Time Critical & Deterministic Processing
- Intra & Inter Process Communications
- Fault Processing
- Process Prioritization

**System Component Relative Sizes**

<table>
<thead>
<tr>
<th>Tens</th>
<th>Hundreds</th>
<th>Thousand</th>
</tr>
</thead>
<tbody>
<tr>
<td>FD Req’s and I/Fs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Comp/Seg Req’s and I/Fs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SW CSCI Level Req’s and I/Fs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSCIs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CSCs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Objects</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Files</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SLOCs</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Common Hardware and Operating Systems**

*A single erroneous SLOC/Character can crash the entire system*
Open Architecture: Example Achieved at the CSCI and Class Level

Functional Domain Level

External Comm's | Sensor Mgmt | Ship Control | Display | Track Mgmt | Command Control | Weapon Mgmt | Vehicle Control | Infrastructure | Support

Component level

Combat System

Segment level

Tomahawk Weapon System Component

Missile Planning Segment | Weapon Control System Segment | Missile Segment

CSCI level

Command and Control CSCI | Engagement Manager CSCI | Display Layer CSCI | Missile Manager CSCI | Inter-LAN Comm’s Manager CSCI | Sim’s CSCI | Training CSCI | Operating Environment CSCI | Common Services CSCI | Support Services CSCI

Class/Object level

Object Oriented Design

Reusable, Scalable, Maintainable

Platform & Launcher Unique Objects

Open Architecture Scalability Achieved at the CSCI Object Level
Wrap-Up
SoSE Key Points

- Tailoring the Systems Engineering Process
- Technical Considerations in System- and Family-of-Systems Engineering
- Distributing Functionality across Systems
- Leveraging Commonality
- Life-cycle Affordability
- Development for System Certification