



## Acquisition Research Program: Creating Synergy for Informed Change



# Total Ownership Cost – A Decade into the 21<sup>st</sup> Century

Michael W. Boudreau COL (ret), Senior Lecturer, NPS

Brad R. Naegle LTC (ret), Senior Lecturer, NPS

# Total Ownership Cost—a Decade into the 21<sup>st</sup> Century

## 2003 – Where We Were

- Positive
  - Encouraged experimentation
  - Come up with new approaches
- Negative
  - Leadership wasn't heavily involved in affordability – not speaking with one voice
  - Lack of Discipline (e.g., technological maturity)
  - Lacked risk assessment tools (TRLs, MRLs, SMLs)



# Total Ownership Cost—a Decade into the 21<sup>st</sup> Century

## Today – Where We Are

- Cost estimating
  - Certifications
  - Nunn-McCurdy
  - Cost Databases
  - Affordability Slices of Mission Areas
- Collaborative IT
- Mandated reviews – MS B (KP-1), CDR-A (KP-2), MS C (KP-3)
- Navy Gate Reviews (affordability)
- Configuration Steering Boards (counter to requirements creep)
- Product Support Manager – Performance Based Logistics (affordable logistics)



# Total Ownership Cost—a Decade into the 21<sup>st</sup> Century

## Opportunities

- Cost estimating impediments
  - Statistical **Confidence Levels**
  - Useful **Cost Databases** – support **early** cost estimates?
  - **Nunn-McCurdy** Breaches using the **wrong metrics**
  - Cost vs. **Affordability**
- Collaborative IT
  - Are the **right stakeholders involved** in the conversation?



# Total Ownership Cost—a Decade into the 21<sup>st</sup> Century

## Summary

- Mandated discipline
- Bureaucracy
- Selective lack of tools
- Need to move to self-motivated discipline



# Software Intensive Systems & TOC

- Poor SW size & complexity estimates lead to understated SW O&S cost estimates
  - Requirements progression from user 'Capability Need' through PM 'Performance Spec' to contractor 'System Design' invites requirements interpretation
  - Interpretation leads to vague or missed requirements
  - Vague/missing requirements lead to poor SW size & complexity estimates
- 
- Repeat as necessary!



# Bridging the SW Requirements Gap

- The immature SW engineering environment is incapable of satisfying unstated requirements – especially supportability performance gaps
- Requirements gap analysis essential for attaining SW supportability performance – **MUIRS Analysis:** Maintainability, Upgradability, Interoperability, Reliability, Safety & Security
- **Goal:** Develop complete, well defined inventory of requirements, including stated, derived, and implied
- **Tools:** MUIRS Analysis & SEI's Quality Attribute Workshop (QAW)





# SW Design – The Key to O&S Performance

- Must drive the design for supportability performance
- Starts with a complete inventory of requirements, including supportability requirements resulting from a Logistics Supportability Analysis – **MUIRS**
- SW developer needs to know requirements in context – How will system be used & maintained? In what environments? What is the priority of essential functions & enhancing functions? How should it operate when stressed? What is the expected exception handling, fault tolerance, and recovery techniques? How will performance be verified?





# SW Design continued

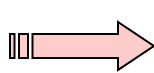
- User involvement in the SW design process is critical – they must develop scenarios for:
  - Use Cases: Including MUIRS focus for supportability
  - Growth: Anticipated changes over the life cycle
  - Exploratory: Expected performance when stressed, including FMECA prioritization of functionality/recovery
- Goal: Ensure SW developer understands warfighter expectations **before** system is designed
- Tools: SEI's Architectural Tradeoff Analysis Methodology (ATAM<sup>sm</sup>)



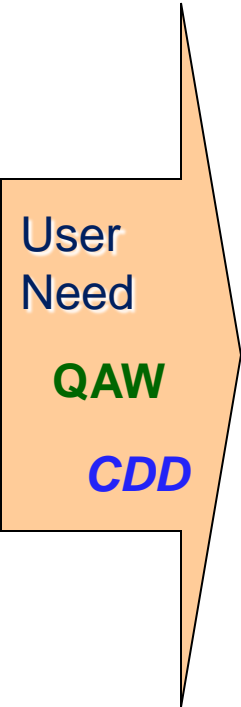
**ATAM  
Input**



**Scenario Development**



**Test Case Development**



**Use Cases**

**-Performance**

**- MUIRS**

**Growth Scenarios**

**-Performance**

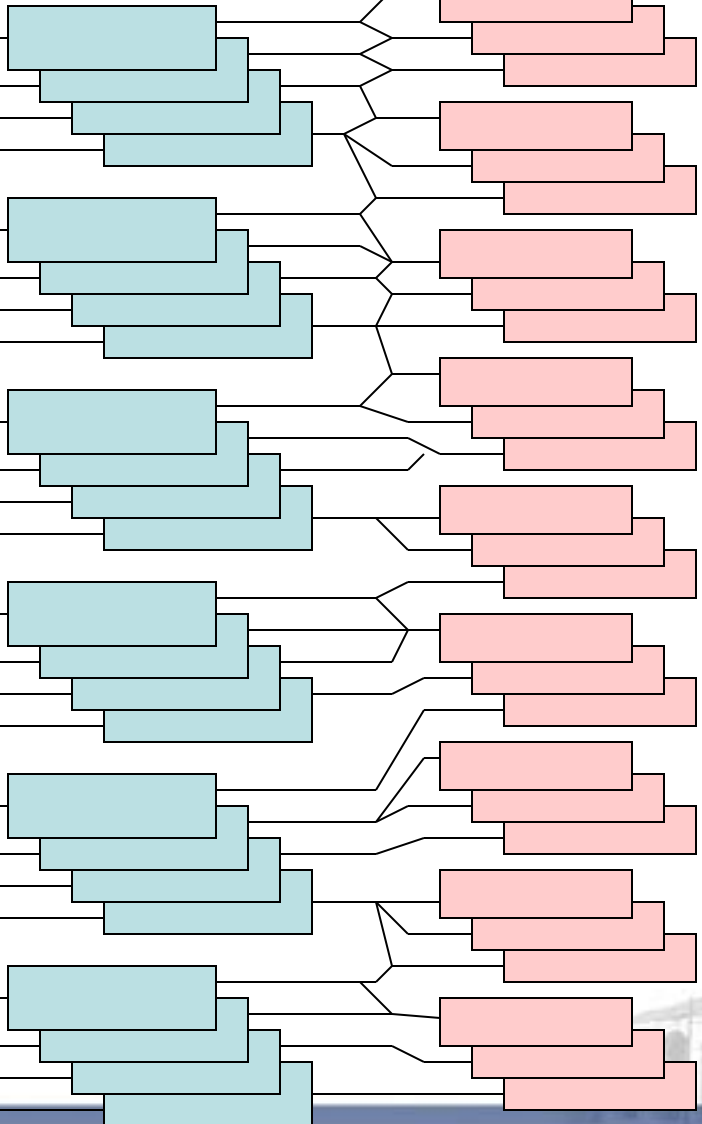
**-MUIRS**

**Exploratory Scenarios**

**-Performance**

**-FMECA**

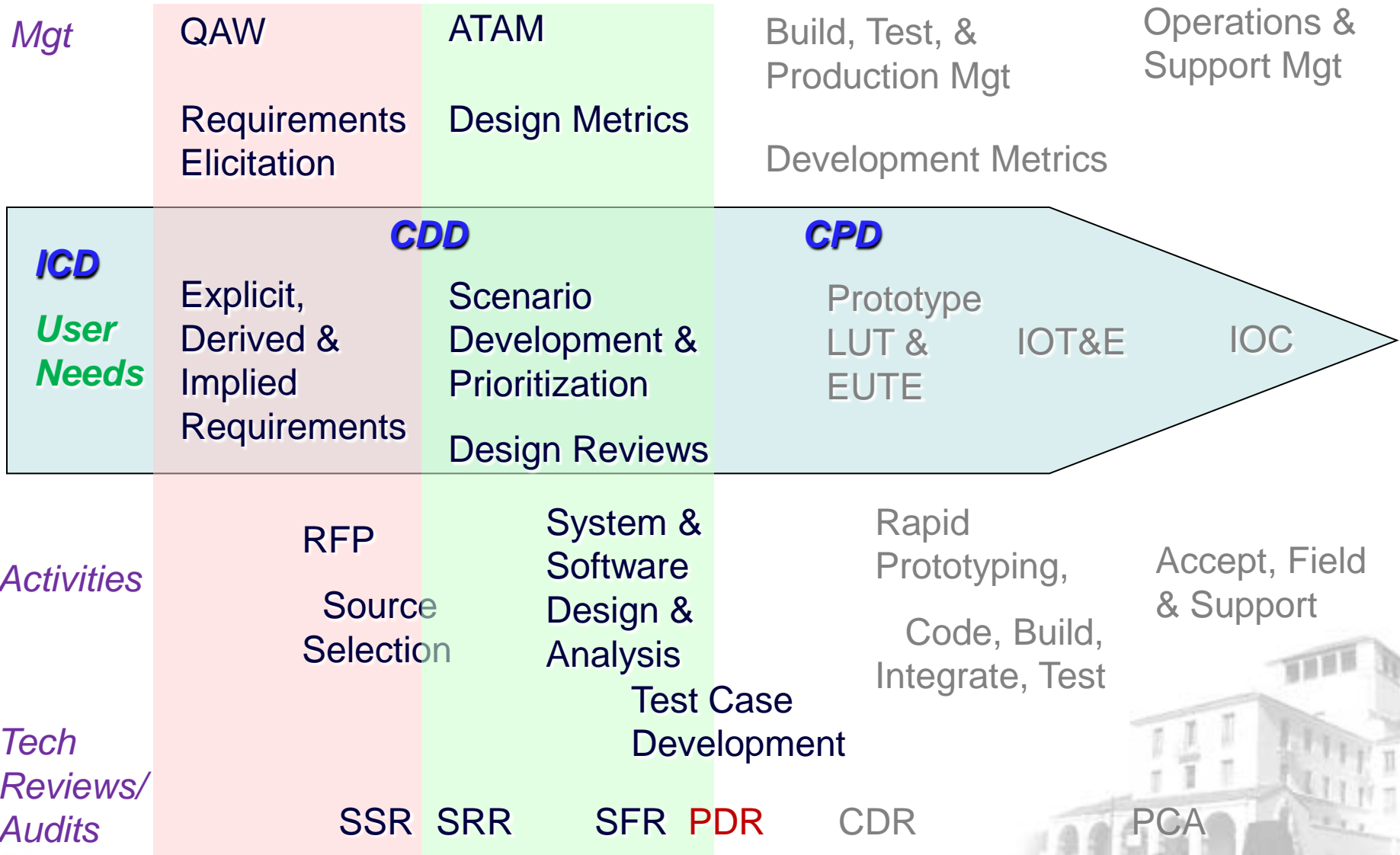
**-MUIRS**



**Integrated  
into test  
program**



# QAW & ATAM Integration into SW Lifecycle Management



# Summary: Improving SW TOC Performance

- Break the cycle: Poor requirements/designs = difficult and costly SW sustainment
- Complete the inventory of derived and implied SW supportability requirements with MUIRS analysis
  - Tools: MUIRS Analysis technique and SEI's QAW
- Drive the system design for improved supportability performance, critical for Software
  - Tool: SEI's ATAM <sup>sm</sup>
- Ensure test program includes supportability performance testing, stress testing, fault handling, and recovery techniques
  - Tool: SEI's ATAM <sup>sm</sup>

