SYSTEMS ENGINEERING Research Center

Architecture-Based Analysis of System ility Synergies and Conflicts

Barry Boehm, Jo Ann Lane, USC
Kevin Sullivan, U. Virginia

NDIA Systems Engineering Conference
October 30, 2013

SYSTEMS ENGINEERING Research Center

Outline

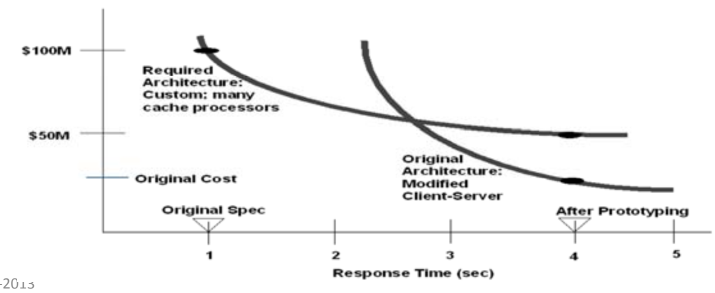
- **Critical nature of the ilities**
  - **Major source of project overruns, failures**
  - **Significant source of stakeholder value conflicts**
  - **Poorly defined, understood**
  - **Underemphasized in project management**
- **Challenges for cyber-physical-human systems**
- **SERC Foundations efforts**
  - **AFIT, GaTech, MIT, NPS, PennState, USC, Uva, WSU**
  - **Stakeholder value-based, means-ends hierarchy**
  - **Formal analysis of ility definitions and relations**
  - **Architecture strategy synergies and conflicts**

SYSTEMS ENGINEERING Research Center

Importance of ility Tradeoffs

Major source of DoD system overruns

- System ilities have systemwide impact
  - System elements generally just have local impact
- ilities often exhibit asymptotic behavior
  - Watch out for the knee of the curve
- Best architecture is a discontinuous function of ility level
  - "Build it quickly, tune or fix it later" highly risky
  - Large system example below

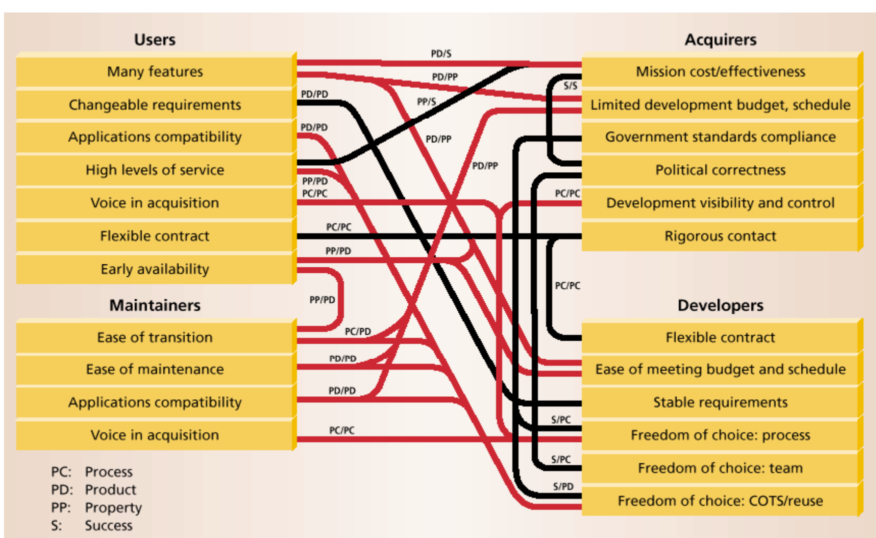


10-30-2013 3

SYSTEMS ENGINEERING Research Center

Role-Based Ilities Value Diversity

Bank of America Master Net; DoD?



4 10-30-2013

SYSTEMS ENGINEERING Research Center

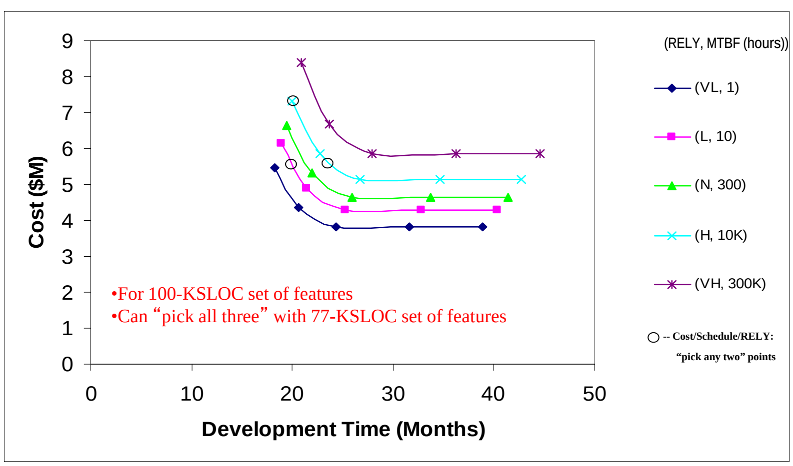
Example of Current Practice

- "The system shall have a Mean Time Between Failures of 10,000 hours"
- What is a "failure?"
  - 10,000 hours on liveness
  - But several dropped or garbled messages per hour?
- What is the operational context?
  - Base operations? Field operations? Conflict operations?
- Most management practices focused on functions
  - Requirements, design reviews; traceability matrices; work breakdown structures; data item descriptions; earned value management
- What are the effects on other –ilities?
  - Cost, schedule, performance, maintainability?

SYSTEMS ENGINEERING Research Center

USC: COCOMO II-Based Tradeoff Analysis

Better, Cheaper, Faster: Pick Any Two?



SYSTEMS ENGINEERING Research Center

Outline

- Critical nature of the ilities
  - Major source of project overruns, failures
  - Significant source of stakeholder value conflicts
  - Poorly defined, understood
  - Underemphasized in project management
- ➡ Challenges for cyber-physical-human systems
- SERC Foundations efforts
  - AFIT, GaTech, MIT, NPS, PennState, USC, Uva, WSU
  - Stakeholder value-based, means-ends hierarchy
  - Formal analysis of ility definitions and relations
  - Architecture strategy synergies and conflicts

10-30-2013 7

SYSTEMS ENGINEERING Research Center

Importance of Cyber-Physical Systems

Major gap in tradespace analysis capabilities

- Current ERS, DARPA tradespace research focused on physical system tradeoffs
  - Range, payload, size, weight, lethality, power and fuel consumption, communications bandwidth, etc.
  - Some focus on physical modularity, composability
- Current cyber tradespace research focused on software, computing, human factors tradeoffs
  - security, safety, interoperability, usability, flexibility, adaptability, dependability, response time, throughput, etc.
- Gaps in capabilities for co-design of hardware, software, and human factors; integration of tradespace analyses

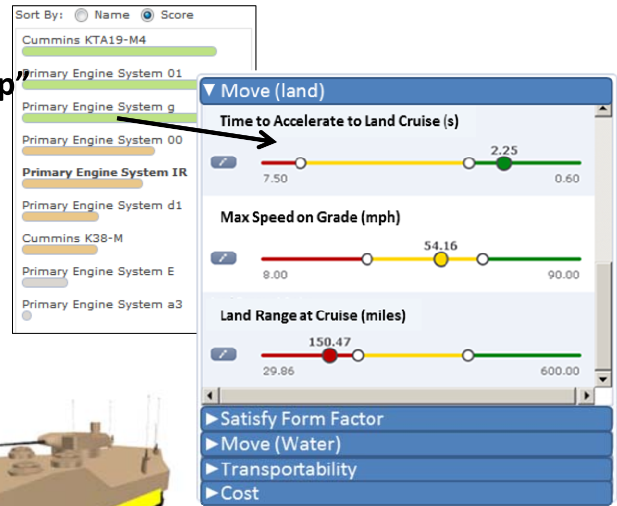
10-30-2013 8

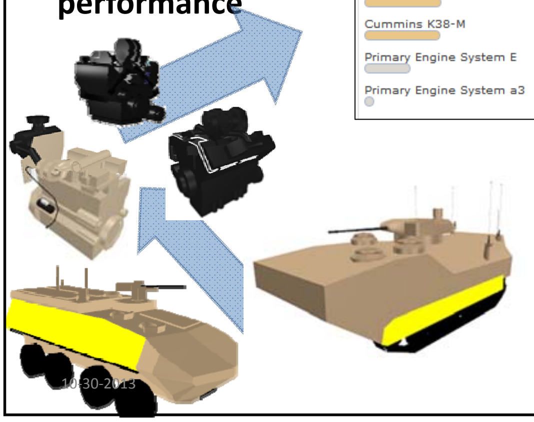
SYSTEMS ENGINEERING Research Center

GaTech – FACT Tradespace Tool

Being used by Marine Corps

- Configure vehicles from the "bottom up"
- Quickly assess impacts on performance





10-30-2013

SYSTEMS ENGINEERING Research Center

Prioritized JCIDS ilities

User View by Combatant Commands: Top priority first

- Intelligence, Surveillance, and Reconnaissance
  - Comprehensive Persistent Survivable Integrated Timely Credible Adaptable Innovative
- Command and Control (note emphasis on Usability aspects)
  - Interoperability Understanding Timeliness Accessibility Simplicity Completeness Agility Accuracy Relevance Robustness Operational Trust
- Logistics: Supply
  - Responsiveness Sustainability Flexibility Survivability Attainability Economy Simplicity
- Logistics: Maintenance
  - Sustainability Responsiveness Attainability Flexibility Economy Survivability Simplicity
- Net-Centric: Information Transport
  - Accessible Capacity Accurate Timely Throughput Expeditionary Latency

10-30-2013 10

SYSTEMS ENGINEERING Research Center

Outline

- **Critical nature of the ilities**
  - **Major source of project overruns, failures**
  - **Significant source of stakeholder value conflicts**
  - **Poorly defined, understood**
  - **Underemphasized in project management**
- **Challenges for cyber-physical-human systems**
- ➡ **SERC Foundations efforts**
  - **AFIT, GaTech, MIT, NPS, PennState, USC, Uva, WSU**
  - **Stakeholder value-based, means-ends hierarchy**
  - **Formal analysis of ility definitions and relations**
  - **Architecture strategy synergies and conflicts**

10-30-2013 11

SYSTEMS ENGINEERING Research Center

SERC Value-Based ilities Hierarchy

Based on ISO/IEC 9126, 25030; JCIDS; previous SERC research

- **Individual ilities**
  - **Mission Effectiveness**: Speed, Physical Capability, Cyber Capability, Usability, Accuracy, Impact, Endurability, Maneuverability, Scalability, Versatility
  - **Resource Utilization:** Cost, Duration, Personnel, Scarce Quantities (capacity, weight, energy, …); Manufacturability, Sustainability
  - **Protection:** Security, Safety
  - **Robustness:** Reliability, Availablilty, Maintainability, Survivability
  - **Flexibility:** Modifiability, Tailorability, Adaptability
  - **Composability:** Interoperability, Openness, Service-Orientation
- **Composite ilities**
  - **Comprehensiveness/Suitability:** all of the above
  - **Dependability:** Mission Effectiveness, Protection, Robustness
  - **Resilience:** Protection, Robustness, Flexibility
  - **Affordability:** Mission Effectiveness, Resource Utilization

10-30-2013 12

SYSTEMS ENGINEERING
Research Center

Means-Ends Framework: Affordability

Affordability Improvements and Tradeoffs

- **Get the Best from People**
  - Staffing, Incentivizing, Teambuilding
  - Facilities, Support Services
  - Kaizen (continuous improvement)
- **Make Tasks More Efficient**
  - Tools and Automation
  - Work and Oversight Streamlining
  - Collaboration Technology
- **Eliminate Tasks**
  - Lean and Agile Methods
  - Task Automation
  - Model-Based Product Generation
- **Eliminate Scrap, Rework**
  - Early Risk and Defect Elimination
  - Evidence-Based Decision Gates
  - Modularity Around Sources of Change
  - Incremental, Evolutionary Development
  - Value-Based, Agile Process Maturity
  - Risk-Based Prototyping
- **Simplify Products (KISS)**
  - Value-Based Capability Prioritization
  - Satisficing vs. Optimizing Performance
- **Reuse Components**
  - Domain Engineering and Architecture
  - Composable Components,Services, COTS
  - Legacy System Repurposing
- **Reduce Operations, Support Costs**
  - Automate Operations Elements
  - Design for Maintainability, Evolvability
  - Streamline Supply Chain
  - Anticipate, Prepare for Change
- **Value- and Architecture-Based Tradeoffs and Balancing**

10-30-2013 13

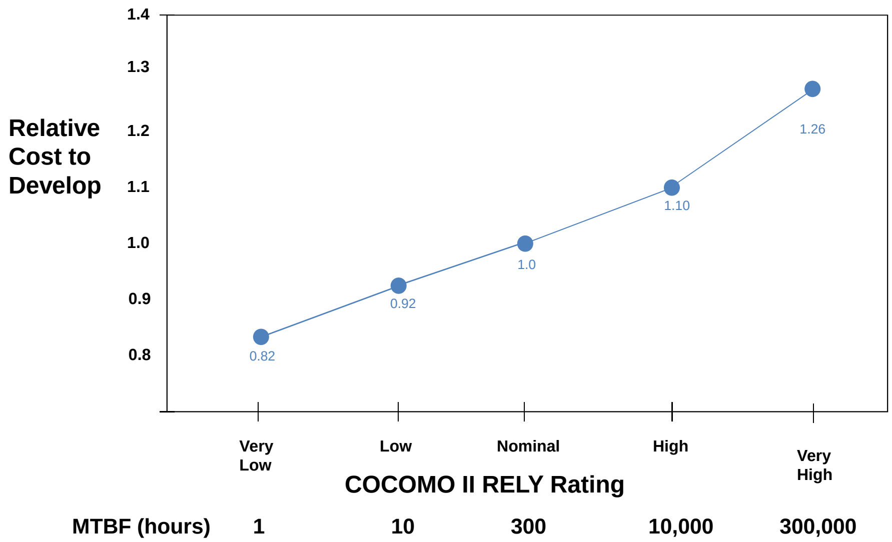
SYSTEMS ENGINEERING
Research Center

Architecture Strategy Synergy-Conflict Matrix

	Reliability	Modifiability	Interoperability	Cost
eliability		• Nanosensor-based smart monitoring improves reliability, makes mods more effective • Domain architecting (using domain knowledge in defining interfaces) improves reliability and modifiability • Modularity (high module cohesion, low module coupling) improves modifiability and reliability	• Domain architecting improves reliability, interoperability within the domain • High-cohesion, low-coupling modules improve interoperability and reliability • Common, multi-layered services and architecture improve interoperability and reliability	• Automated input, output validation reduces human costs • Increased reliability reduces life cycle ownership costs • Product line architectures reduce cost, increase reliability
odifiability	• Reliability-optimized designs may complicate fault diagnosis, system disassembly • Domain architecting assumptions complicate multi-domain system modifiability		• Modularization around sources of change improves modifiability and interoperability • High-cohesion, low-coupling modules improve modifiability and interoperability • Open standards, service-oriented architectures improve both modifiability and interoperability	• Modularization around sources of change reduces life cycle costs • High-cohesion, low-coupling modules reduce life cycle costs • Domain architecting enables domain product lines, reducing costs • Providing excess capacity improves modifiability and decreases lifecycle cost
nteroperability	• Data redundancy improves reliability, but updates may complicate distributed real-time systems interoperability • Optimizing on reliability as liveness may degrade message delivery, accuracy	• Domain architecting assumptions complicate multi-domain system interoperability		• Common, multi-layered services and architecture reduce life cycle costs • Product line architecture improves interoperability, reduces cost of later systems
ost	• Increased reliability increases acquisition costs • Hardware redundancy adds cost • Making easiest-first initial commitments reduces early costs but degrades later reliability, adds later costs • Formal verification adds cost	• Fixed-requirements, fixed-cost contracts generally produce brittle, hard-to-modify systems • Domain architecting increases multi-domain system costs • Providing excess capacity improves modifiability but increases acquisition cost	• Neglecting or deferring interfaces to co-dependent systems will reduce initial costs, but degrade interoperability • Product line architecture increases cost of initial system	

10-30-2013 14

Software Development Cost vs. Quality



Software Ownership Cost vs. Quality