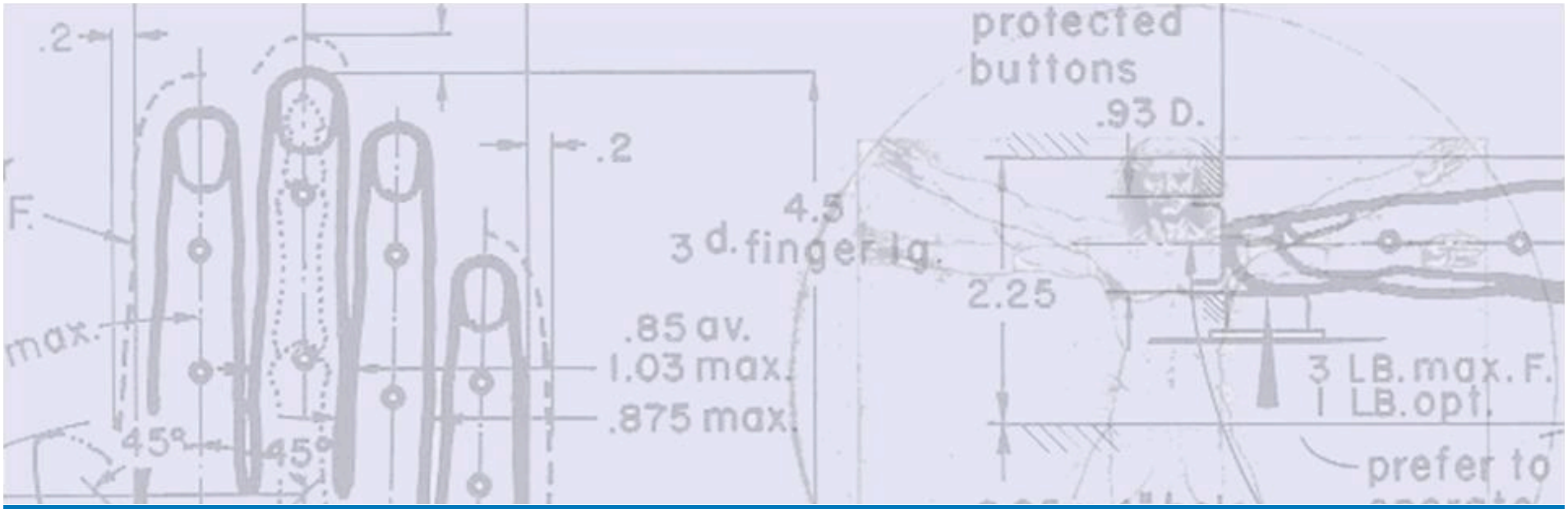


AN EVOLUTION OF USABILITY ANALYSIS

THE SYSTEM OF SYSTEMS USABILITY (SOS-U) FRAMEWORK

Frank C. Lacson, M.S.
Matthew R. Risser, Ph.D.

Pacific Science & Engineering Group

franklacson@pacific-science.com

National Defense Industrial Association. 15th Annual Systems Engineering Conference. October 22-26, 2012

Objectives

- Describe the limitations of conducting traditional usability analyses in a System of Systems (SoS) environment
- Communicate the implications of limited SoS usability on human and system performance
- Reveal a framework to scope development of SoS-specific usability requirements, methods, and outputs
- Provide examples and seed discussion topics to explore additional SoS-related usability issues, use cases, and methods

Usability for System of Systems (SoS)

A definition of Usability (ISO 9241)

"…the extent to which a product can be used by specified **users** to achieve specified **goals** with **effectiveness**, **efficiency,** and **satisfaction** in a specified context of use"

Are these SoS usable?

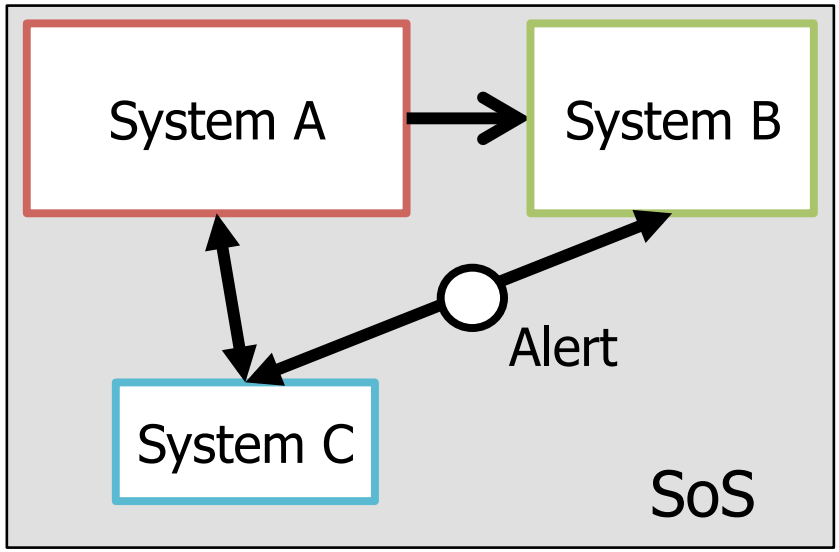
International Organization for Standards (1998). ISO 9241 – 11: Ergonomic requirements for office work with visual display terminals (VDTs) – Part II: Guidance on usability.

Problem Statement

- As Systems Engineering approaches move towards
  - Commercial Off The Shelf /Government Off The Shelf (COTS/ GOTS) technologies
  - Family of Systems (FoS), Systems of Systems (SoS)
  - Lag in effectiveness of usability techniques, mindset
- Why? As with system integration…
  - Sub-system usability does not guarantee system usability
  - Even more so for SoS usability, description of issues follow
- So what?
  - Unrealized HSI risks, reductions in human performance
  - Costly and unexpected re-work at testing
  - At the SoS level, add'l constraints in implementing solutions

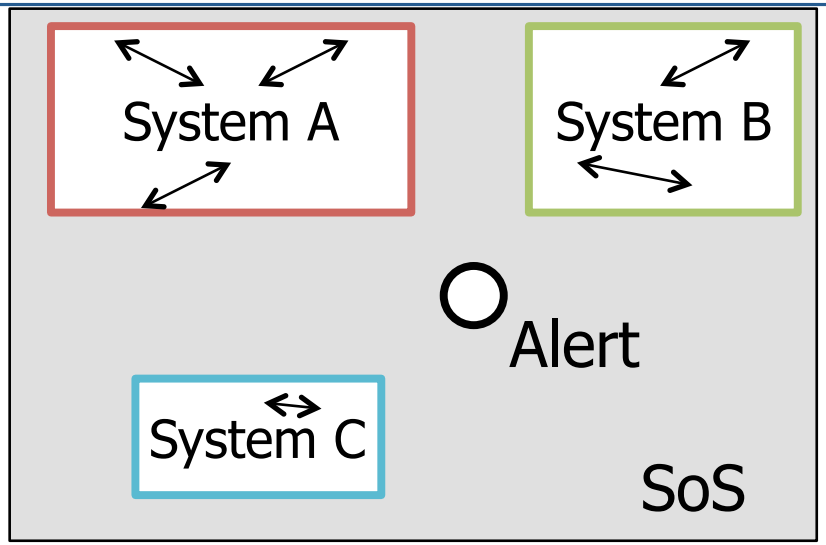
Example System of Systems

- Quick mental exercise: Think of a sample SoS
- Scope of sample SoS
  - Three systems (A, B, C) plus one SoS "Glueware" (e.g., Alert)
  - Different owners (by color) and budgets (size)
  - Different types of information flows: Push, pull, both
- SoS examples
  - ISR, METOC, and C2
  - IT system w/ Navy platforms



SoS Usability Issue #1: Unit of analysis and metrics limited in scope

- Analysis limited to individual systems and interfaces
- Two examples
  - "I think I would like to use this system frequently"[1]
  - "Minimize memory load"[2]
- Metrics limited to menu and control interactions, simple and repeated tasks
- Consumers consist of individual users
- In SoS, the number of users and systems are expanded, but not whole-cloth



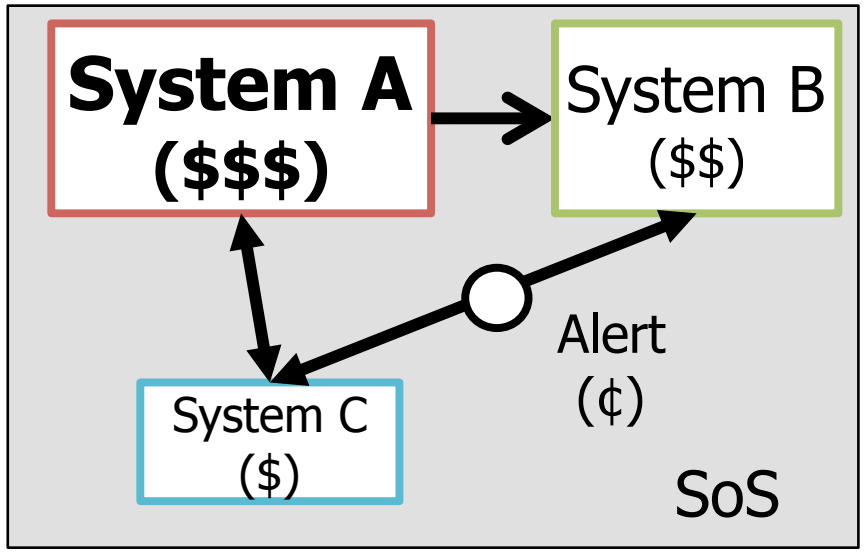
SoS Usability Implications

1. Support remains at component-level (not feature, capability level)
2. Higher-order tasks (e.g., collaboration) unsupported

System Usability Scale (Brooke, 1996); Usability Heuristic Evaluation (Nielsen, 1994)

SoS Usability Issue #2: Ineffective implementation of recommendations

- Single-system user interface recommendations are
  - Managed individually
  - Budgeted accurately
  - Traced clearly to requirements
- At the SoS-level, recommendations are
  - Managed across programs with different priorities
  - Varied in feasibility (legacy vs. dev. systems)

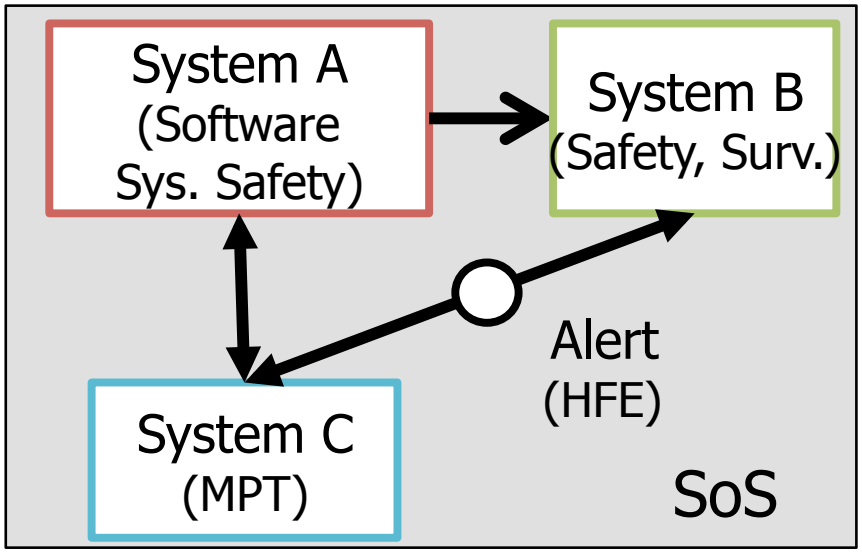


SoS Usability Implications

1. Unknown usability space: What is feasible to change?
2. ROI Uncertainty: Who pays? Who benefits?
3. Untraceable requirements: Where the benefits seen?

SoS Usability Issue #3: New implications for HSI domains

- Each system has an HSI profile (top HSI concern)
  - HFE, MPT, ESOH, Survivability, Habitability
  - E.g., "System C is a training system"
  - Stove-piped systems = predictable stove-piped solutions
- For SoS: Mix between IT Systems and Platforms
- Weather system example: How a shared map display (HFE) can affect manning



SoS Usability Implications

1. HSI domains are increasingly and unknowingly interrelated
2. Increased human performance risk (and performance benefits)

The SoS-U Framework

- Summary of SoS Usability Issues
  - #1: Unit of analysis and metrics limited in scope
  - #2: Ineffective implementation of recommendations
  - #3: New implications for HSI domains
- The System of Systems Usability (SoS-U) Framework
  - Provides a scoped functional path for addressing SoS usability issues
  - Organizes and categorizes usability techniques and tools
- SoS-U is not
  - A solution to a specific SoS or Program Of Record: Each SoS contains a unique set of constraints
  - Limited to user interfaces: Expansion of usability to meet specific SoS needs

A path – The SoS-U Framework

Architecture & Requirements	• Models of interfaces, data, and information flows • Derived requirements with SoS usability implications
Methods	• Collection of usability best practices suitable for the SoS environment
Metrics & Products	• Development of usability metrics to measure impacts on higher-order tasks (e.g., collaboration, information sharing)

Module 1: SoS Usability Architecture and Requirements

- Definition
  - Models of interfaces, data, and information flows
  - Derived requirements with SoS usability implications
- Functions
  - Visualize and scope SoS usability space
  - Identify UI dev. constraints, data and information flows
  - Develop and analyze usability-related requirements
- Artifacts
  - Modified DoDAF artifacts, FoS User Interface specifications
- Anticipated Impact
  - Improved **effectiveness** of usability solutions by focusing on key SoS interfaces and human performance risks

Module 2: SoS Usability Methods

- Definition
  - Collection of usability inspection and testing methods suitable for the SoS environment
  - Best practices that range from general SoS properties to specific support of FoS functions
- Functions
  - Identify reusable SoS usability guidelines
  - Ensure a feasible range of usability analysis options
  - Leverage both existing and emerging usability techniques
- Anticipated Impact
  - Improved **efficiency** of usability recommendations via a fundamental understanding of core SoS usability issues

Module 2: Example artifact - SoS Usability Heuristics

Traditional Usability Heuristic	Anticipated SoS Trait	SoS Usability Heuristic (Converted)
Visibility of System Status "The system should always keep users informed about what is going on…"	Information overload due to expanded scope of system	Visibility of SoS Context "The SoS should always keep users informed **on what systems are being accessed**…"
Error prevention "… a careful design … prevents a problem from occurring in the first place"	Excess workload on error management for systems beyond user control	Error robustness "… a careful design … prevents **critical errors or propagation errors** from occurring in the first place"

Nielsen, J. and Molich, R. (1990). Heuristic evaluation of user interfaces. Proc. ACM CHI'90 Conf.

Module 2: Example artifact - SoS Usability Heuristics (cont.)

- Existing collections of usability heuristics can be converted based on key SoS characteristics: e.g.,
  - Nielsen heuristics: Match between system and real world, consistency and standards…
  - Gerhardt-Powals: Automate unwanted workload, Reduce uncertainty, Fuse data…
- Sample SoS heuristic development process
  1. Determine suitable systems/FoS for the heuristic
  2. Highlight key usability issues for the SoS
  3. Leverage existing heuristics, best practices
  4. Specify heuristics, adhere to templates
  5. Link to case studies, CONOPS, user tests

Rusu et al. (2011). A Methodology to Establish Usability Heuristics.
ACHI 2011: The Fourth International Conference on Advances in Computer-Human Interfaces.

Module 3: SoS Usability Products

- Definition
  - Development of usability metrics to measure impacts on higher-order tasks (e.g., collaboration, information sharing)
  - Documentation of human performance implications
- Functions
  - Quantify and validate human and system performance impacts
  - Adapt to evolutions in software development, SE
  - Explore interdependencies between HSI Domains
- Artifacts: M&S, inputs to SE docs, trade studies
- Anticipated Impact
  - Improved **satisfaction** of **users** and customer **goals** via quantification of usability benefits

Way Forward

- Architecture & Requirements
  - Use case development: Common categories of SoS?
  - Quick SoS Assessment: Where to start?
- Methods
  - Integrate industry standards (e.g., ISO, EIA, IEEE)
  - Expand SoS usability test methods
  - Apply lessons learned: Usability of web applications, tablets/ handhelds, widgets
- Metrics and Products
  - Update usability metrics to reflect higher order SoS activities (e.g., collaboration, information sharing, automation use)
  - Adapt existing M&S tools (e.g., IMPRINT) for SoS modeling

SoS-U Framework

A definition of Usability (ISO 9241)

"…the extent to which a product can be used by specified **users** to achieve specified **goals** with **effectiveness**, **efficiency,** and **satisfaction** in a specified context of use"

Architecture & Requirements	• Improved **effectiveness** of usability solutions by focusing on key SoS interfaces and human performance risks
Methods	• Improved **efficiency** of usability recommendations via a fundamental understanding of core SoS usability issues
Metrics & Products	• Improved **satisfaction** of **users** and customer **goals** via quantification of usability benefits

Take-Home Message

- Usability assumptions and techniques must adapt to emerging Systems Engineering approaches
- The SoS-U Framework provides a organizing representation for architectures, requirements, methods, metrics, and usability products
- Traditional usability tools can and should be leveraged - yet tailored - to meet SoS-specific system and human performance requirements

Questions?

For more information, please contact:

Pacific Science & Engineering Group
(858) 535-1661

Frank C. Lacson
franklacson@pacific-science.com