



Applied Systems Engineering

A 4-Day Practical Workshop

Planned and Controlled Methods are Essential to Successful Systems.

Today’s complex systems present difficult challenges to develop. From military systems to aircraft to environmental and electronic control systems, development teams must face the challenges with an arsenal of proven methods. Individual systems are more complex, and systems operate in much closer relationship, requiring a system-of-systems approach to the overall design.



This workshop is a review of the latest principles for systems engineering in context of standard development cycles, with realistic practice on how to apply them.

You should attend this workshop if you are:

- A leader or a key member of a complex system development team
- Concerned about the team’s technical success
- Interested in how to fit your system into its system environment
- Looking for practical methods to use in your team

Participants in this course practice the skills by designing and building interoperating robots that solve a larger problem.

The challenges today are changing, because systems are changing. Rather than separate, well-defined functions, systems today interact in complex ways to perform operational missions. Many interoperable systems are essential to normal operations, and seemingly minor design errors can cost lives.

The challenges today are changing, because acquisition methods are changing. Interoperability of systems is driving new iterative approaches that displace the older “waterfall” acquisition models. New developments require more integrated information quicker.

Error Cost Found in Phases

PHASE	COST
Definition	1
System Design	3
Prelim Design	10
Detail Design	50
Production	200
Use	500

The discipline and concepts of systems engineering provide ways to manage this complexity. By following systems engineering practices, teams organize their thought processes in such a way as to bring order out of chaos. Studies of complex programs have shown that the proper application of up-front thinking can reduce the cost impact of errors by as much as five hundredfold.

Systems engineering is a simple flow of concepts, frequently neglected in the press of day-to-day work, that reduces risk step by step. In this workshop, you will learn the latest systems principles, processes, products, and methods. This is a practical course, in which students apply the methods to build real, interacting systems during the workshop. You can use the results now in your work.

The workshop is aimed at **program managers, project managers, systems engineers, technical team leaders, and others who participate in defining and developing complex systems.**

Drawing from decades of accumulated knowledge in system development, this workshop provides you with highly effective tools to use in your work. The material is augmented by examples from real-life experience, including participative examples from the attendees. Lessons covered include:

How do We Work With Complexity? – Basic definitions and concepts. Problem-solving approaches; system thinking; systems engineering overview; what systems engineering is NOT.

- Objectives, requirements, processes
- Reductionist and system-level approaches
- Complexity theory, emergent properties, complex behavior
- Conceptual development and chunking
- System thinking principles
- Application in the system life cycle

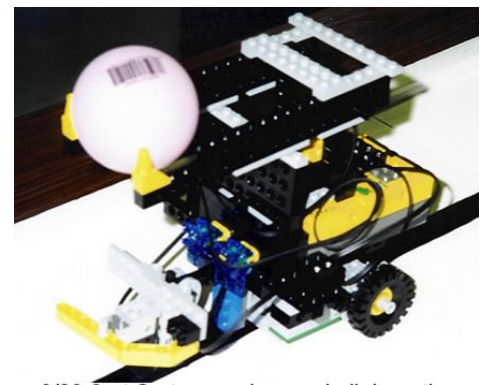
Systems Engineering Model – An underlying process model that ties together all the concepts and methods. Overview of the systems engineering model; technical aspects of systems engineering; management aspects of systems engineering.

- Model overview
- Incremental, concurrent processes
- Process loops for iteration
- Technical and management aspects of systems engineering

A System Challenge Application – Practical application of the systems engineering model against an interesting and entertaining system development. *Small groups build actual interoperating robots to solve a larger problem.* Small group development of system requirements and design, with presentations for mutual learning.

Where Do Requirements Come From? – Requirements as the primary method of measurement and control for systems development. How to translate an undefined need into requirements; how to measure a system; how to create, analyze, manage requirements; writing a specification.

- Determining the mission from an operational view
- Quantifying the mission
- Analyzing missions and environments
- Creating requirements
- Definition of requirements types
- Requirements analysis methods



3/02 Cart System carries one ball down the track, then turns around to go back for another

Where Does a Solution Come From? – Designing a system using the best methods known today. System architecting processes; alternate sources for solutions; how to allocate requirements to the system components; how to develop, analyze, and test alternatives; how to trade off results and make decisions. Getting from the system design to the system.

- Defining alternative concepts
- Architecting with patterns
- Analyzing and evaluating alternatives

-
- Establishing an allocated baseline
 - Implementing the baseline system
 - Production challenges for unique systems
 - System integration processes
 - Transitioning the system into use
 - System operations
 - Post-mission use and disposal

Ensuring System Quality – Building in quality during the development, and then checking it frequently. The relationship between systems engineering and systems testing.

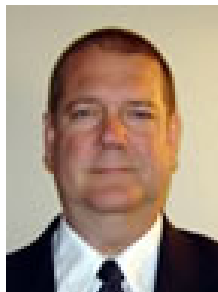
- Technical analysis as a system tool
- Verification at multiple levels: architecture, design, product
- Validation at multiple levels; requirements, operations design, product

Systems Engineering Management – How to successfully manage the technical aspects of the system development; virtual, collaborative teams; design reviews; technical performance measurement; technical baselines and configuration management.

- Technical teamwork and leadership
- Planning technical processes
- Assessing and controlling the technical processes, with corrective actions
- Technical performance measures
- Risk management methods
- Small case studies on process management

Summary - Review of the important points of the workshop. Interactive discussion of participant experiences that add to the material.

Mr. Eric Honour has been in international leadership of the engineering of systems for over a decade, part of a 37-year career of complex systems development and operation. His energetic and informative presentation style actively involves class participants. He is a former President of the International Council on Systems Engineering (INCOSE). He was selected in 2000 for Who's Who in Science and Technology and in 2004 as an INCOSE Founder. He has been a systems engineer, engineering manager, and program manager at Harris, E-Systems, and Link, and was a Navy pilot. He has contributed to the development of 17 major systems, including Air Combat Maneuvering Instrumentation, Battle Group Passive Horizon Extension System, and National Crime Information Center. BSSE (Systems Engineering) from US Naval Academy and MSEE from Naval Postgraduate School.



Dr. Scott Workinger has led innovative technology development efforts in complex, risk-laden environments for 30 years in the fields of manufacturing (automotive, glass, optical fiber), engineering and construction (nuclear, pulp & paper), and information technology (expert systems, operations analysis, CAD, collaboration technology). He currently teaches courses on program management and engineering and consults on strategic management and technology issues. Scott has a B.S in Engineering Physics from Lehigh University, an M.S. in Systems Engineering from the University of Arizona, and a Ph.D. in Civil and Environment Engineering from Stanford University.