

Effective Systems Engineering Training

Rick Hefner, Ph.D.
California Institute of Technology
Industrial Relations Center 1-90
Pasadena, CA 91125-9000
310-304-0533
rhefner@processimprovement.com

Abstract—The need for systems engineering training is steadily increasing, as both the defense and commercial markets take on more complex “systems of systems” work. A variety of universities and commercial training vendors have assembled courses of various lengths, format, and content to meet this need. This presentation looks at the requirements for systems engineering training, and discusses techniques for increasing its effectiveness. Several format and content options for meeting these requirements are compared and contrasted, and an experience-based curriculum is shown.

TABLE OF CONTENTS

1. INTRODUCTION	1
2. TARGET AUDIENCE.....	1
3. BODY OF KNOWLEDGE	2
4. DELIVERY METHOD.....	2
5. DELIVERY FORMAT	3
6. EXAMPLE	3
7. SUMMARY.....	5
8. ACKNOWLEDGEMENTS	5
9. REFERENCES	5
10. BIOGRAPHY	5

1. INTRODUCTION

The need for trained systems engineers is steadily increasing, as both the defense and commercial markets take on more complex “systems of systems” work. Although leading universities are starting to offer system engineering (SE) training to undergraduate and graduate students, the greatest demand is within practicing engineers.[1] Competition in the engineering marketplace calls for reduced cycle times, reduced risk of project failure, and a greater ability to meet complex systems requirements.*

To that end, many universities are developing systems engineering training programs. These programs are designed to enhance existing systems engineering skills by addressing the need to:

- Produce designs and cost estimates of complex products faster and cheaper;
- Work with customers to better translate user needs into outstanding delivered products;

1 _____
1-4244-0525-4/07/\$20.00 ©2007 IEEE

* IEEEAC paper #1419, Version 6, Updated July 1, 2006

- Define optimal verification and validation programs tailored to a company and customer’s risk threshold;
- Effectively use advanced information systems technology to effectively manage design data and interfaces and increase productivity;
- Optimize designs based on Cost As an Independent Variable (CAIV) and related methods;
- Manage risk in a cost constrained environment;
- Trade project parameters, including technical performance, quality, cost, and schedule.

The purpose of organizational training is to develop the skills and knowledge of people so they can perform their roles effectively and efficiently. An organizational training program involves:

- Identifying the training needed by the organization;
- Obtaining and providing training to address those needs;
- Establishing and maintaining training materials;
- Establishing and maintaining training records;
- Assessing training effectiveness;

The training strategy and tactics employed will greatly influence cost, quality, retention of knowledge, and student satisfaction. The design of an effective systems engineering training program must address three issues: target audience, body of knowledge, and delivery format. Each of these is addressed below.

2. TARGET AUDIENCE

Different practitioners benefit in different ways from comprehensive systems engineering training. Software and hardware engineers at the subsystems level can better understand their role in the systems lifecycle, and can become more aware of the context of their decisions and their impact on the overall system performance. Those engineers with interest and aptitude sometimes change over to a systems engineering career path. Similarly, project managers can develop a broader perspective of the technical work they must plan and track. Support personnel, such as quality assurance specialists may obtain the same benefits with exposure to systems engineering concepts.

This may be better addressed within the broader context of a competency model [2]. A competency is a set of behaviors that encompasses skills, knowledge, abilities, and personal

attributes that are critical to successful performance at a particular job. A key component of a competency is that it be observable and measurable through certain behaviors. These behaviors, taken together, provide a model for superior performance on the job. When integrated with an organization's strategic goals and individual performance plans, a competency model provides a powerful mechanism for identifying gaps in individual and workforce-wide skills sets, and aligning learning resources to minimize these gaps.

3. BODY OF KNOWLEDGE

The selection of systems engineering topics to be trained usually relies on an industry standard, such as the INCOSE Systems Engineering Body of Knowledge (SEBOK) [3], Department of Defense guide to systems engineering [4], EIA-632 [5], or IEEE 1220 [6]

One of the traditional debates is whether the course should address "systems engineering" as a discipline or "engineering a systems" as a process. The difference is whether the course material will cover the technical management of the system. In the same sense that managers benefit from understanding the technical aspects of SE, technical people can benefit from understanding the management aspects. To that extent, sources such as the Project Management Body of Knowledge (PMBOK) [7] and the Capability Maturity Model Integrated (CMMI) [8] should be considered.

A second issue is the extent to which SE training should address enterprise-specific topics, such as the use of specific tools and methods, organizational policies and procedures, customer-specific acquisition practices, or domain-specific technologies. By including such topics, the value of the course can be greatly enhanced, but this also prevents using a standard, commercially available course.

EIA-632

EIA-632 [5], *Processes for Engineering a System*, was developed as a joint project of the Electronic Industries Alliance (EIA) and the International Council of Systems Engineering (INCOSE). The standard defines a systematic approach to engineering or reengineering a system, incorporating industry best practices. The top-level hierarchy is shown in Figure 1.

The scope of the standard includes the technical and managerial practices a SE contractor would use, and acknowledges the acquisition processes used by the customer. The processes decompose into 33 separate process requirements, as shown in Figure 2.

4. DELIVERY METHOD

SE courses may be delivered through classroom lectures, team projects, computer-based training, or some combination. Length and depth of the training must also be determined. Although there is strong motivation for reducing cost, the effectiveness of the training may be severely limited by choosing the wrong format or length for a given audience.

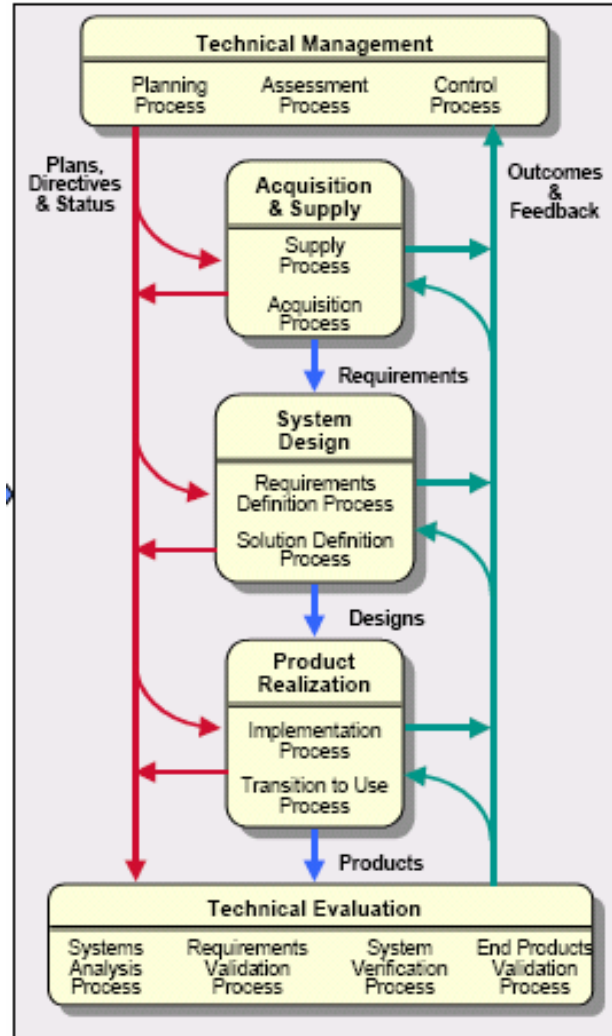


Figure 1. EIA-632 Processes

SUPPLY
1—Product Supply
ACQUISITION
2—Product Acquisition
3—Supplier Performance
PLANNING
4—Process Implementation Strategy
5—Technical Effort Definition
6—Schedule and Organization
7—Technical Plans
8—Work Directives
ASSESSMENT
9—Progress Against Plans and Schedules
10—Progress Against Requirements
11—Technical Reviews
CONTROL
12—Outcomes Management
13—Information Dissemination
REQUIREMENTS DEFINITION
14—Acquirer Requirements
15—Other Stakeholder Requirements
16—System Technical Requirements
SOLUTION DEFINITION
17—Logical Solution Representations
18—Physical Solution Representations
19—Specified Requirements
IMPLEMENTATION
20—Implementation
TRANSITION TO USE
21—Transition to Use
SYSTEMS ANALYSIS
22—Effectiveness Analysis
23—Tradeoff Analysis
24—Risk Analysis
REQUIREMENTS VALIDATION
25—Statements Validation
26—Acquirer Requirements Validation
27—Other Stakeholder Requirements Validation
28—System Technical Requirements Validation
29—Logical Solution Representations Validation
SYSTEM VERIFICATION
30—Design Solution Verification
31—End Product Verification
32—Enabling Product Readiness
END PRODUCTS VALIDATION
33—End Products Validation

Figure 2. EIA-632 Process Requirements

5. DELIVERY FORMAT

SE courses may be delivered through classroom lectures, team projects, computer-based training, or some combination. Length and depth of the training must also be determined. Although there is strong motivation for reducing cost, the effectiveness of the training may be

severely limited by choosing the wrong format or length for a given audience.

The Kirkpatrick model [9] suggests four increasingly accurate ways to measure the effectiveness of training:

- Level 1 – Collect student and instructor reaction to the training;
- Level 2 – Measure student learning through testing;
- Level 3 – Measure transference of learning to the job;
- Level 4 – Measure impact on job performance.

Level 1 (“smile factor”) is the industry standard for most technical training. Students are typically given a one page form to complete at the end of the class, which asks how well they like the course, whether they felt the class was valuable, what parts should be lengthened/shortened, etc. Although this provides some qualitative feedback, it does not measure whether the student actually learned anything. In addition, students, being less experienced, may not be the best judge of whether they were taught appropriate topics or the appropriate duration.

6. EXAMPLE

Several universities offer systems engineering programs for the practicing engineer [1]. For example, the Caltech Industrial Relations Center (<http://www.irc.caltech.edu>) offers programs primarily for major aerospace contractors and government agencies. The Caltech IRC Systems Engineering Center packages the systems engineering training in a variety of formats, designed to fit specific budgets and needs:

- The One Day Systems Engineering Overview - A summary of the end-to-end systems engineering process;
- Systems Engineering Implementation Workshops - Focused workshops which allow project teams to apply system engineering processes to solve current project problems, under the guidance of a experienced systems engineer;
- Three Day Systems Engineering Course - Principles of systems engineering applied within practical, group problem solving applications
- 40-Hour Systems Engineering Certificate Program - 5 full days or 10 half-days of lectures and group exercises covering all aspects of systems engineering
- 56-Hour Systems Engineering Certificate Program - Expanded topics and lectures with an in-depth threaded exercise, typically a full, one-day course per month for seven consecutive months;
- Space Missions, Systems, and Subsystems Program - An intensive, highly technical 40-hour training experience derived from Caltech's graduate level course;

<p>Understanding Systems and Systems Engineering</p> <ul style="list-style-type: none"> ■ Systems and systems engineering in the project/program environment ■ The system development life cycle ■ The systems engineering process <p>Creating a High-Performing Team for Systems Engineering</p> <ul style="list-style-type: none"> ■ Conducting effective communications ■ Identifying group decision-making processes ■ Adapting to work style differences ■ Effectively resolving team conflicts ■ Giving and receiving feedback <p>Constructing Systems Engineering Requirements</p> <ul style="list-style-type: none"> ■ Understanding user requirements ■ Defining requirements ■ Specifications ■ Guidelines and lessons learned <p>Performing a Functional Analysis</p> <ul style="list-style-type: none"> ■ Functional analysis process ■ Functional requirements <p>Developing a Systems Architecture</p> <ul style="list-style-type: none"> ■ Selecting the system architecture ■ Software architecture ■ Defining, controlling, and managing interfaces ■ The Work Breakdown Structure (WBS) <p>Performing System Design and Development</p> <ul style="list-style-type: none"> ■ Hardware and software design ■ System modeling and effectiveness ■ Defining and managing system performance parameters ■ Trade-off analyses 	<p>Conducting Technical Reviews and Audits</p> <ul style="list-style-type: none"> ■ Types of technical reviews ■ Planning technical reviews ■ Conducting technical reviews <p>Using Configuration Management</p> <ul style="list-style-type: none"> ■ Configuration identification and change control ■ Configuration status accounting and audits <p>Verification and Validation Testing</p> <ul style="list-style-type: none"> ■ Test and evaluation requirements ■ The verification process and software testing <p>Managing Risk</p> <ul style="list-style-type: none"> ■ Risk management methodology ■ Cost/schedule risk analysis ■ The risk management plan <p>Managing System Cost and Schedule Estimation</p> <ul style="list-style-type: none"> ■ Estimating system cost and development time ■ Cost control approaches ■ Life cycle cost, schedule control, and earned value management <p>Producing the System</p> <ul style="list-style-type: none"> ■ Producibility ■ Cost of production ■ Examples of volume production <p>Systems Engineering Management and Planning</p> <ul style="list-style-type: none"> ■ Management responsibilities ■ Organizing systems engineering ■ Selecting and using metrics ■ Assessing systems engineering capability ■ Planning systems engineering
--	---

Figure 3. Systems Engineering Topics in the Caltech SE Program

All courses draw from a common body of course material, developed by a team of experienced systems engineers. The topics (Figure 3) represent a balance between traditional systems engineering, project management, and domain-specific issues germane to the aerospace community, such as Department of Defense contracting methodology. The courses are focused on engineering a system, and address all the topics of EIA-632. The instructors are experienced systems engineers and managers, so they understand current SE trends and the SE challenges faced by the students.

The target audience for the current SE certificate programs is practicing subsystem and systems engineers with 5-10 years of experience. This group is seeking to understand how their current engineering activities fit within a broader context, and to expand the depth of their knowledge in selected areas. They are also seeking to understand the interconnections and balance between various engineering activities, and heuristics for guiding their work. Some of

these engineers are anticipating management careers where a broader perspective is useful; others are considering moving from subsystem engineering to systems engineering.

The multi-day courses are typically offered over several months. This format fits better with project schedule pressures, and allows for learning outside of the classroom. The students are divided into teams, and given a project to specify, analyze, and design, e.g., the autonomous ground vehicle discussed in the DARPA (Defense Advanced Research Projects Agency) Urban Challenge (<http://www.darpa.mil/grandchallenge>). Between the lectures, the students apply the lecture material to the vehicle design. Their results are presented in the next class, through a series of mock milestone reviews, with the instructors acting as the customer. The exercises fit with Level 3 of the Kirkpatrick model; the students apply the learned material in a job-like environment. Feedback has

been very positive, and indicates the exercise is one of the most valuable parts of the course in learning both the technical and interpersonal aspects of systems engineering.

7. SUMMARY

There is an increased need for systems engineering training, especially training aimed at the practicing engineer. To meet the need course audience, format, and content must be considered. There is significant synergy systems engineering and project management, as reflected in current SE standards, and the training should reflect this synergy.

8. ACKNOWLEDGEMENTS

CMM and Capability Maturity Model are registered in the U.S. Patent and Trademark Office. CMM Integration, CMMI, SCAMPI, and IDEAL are service marks of Carnegie Mellon University.

9. REFERENCES

- [1] Brown, Frederick S., and Haynes, Ray M. Haynes, "Industry-Driven Systems Engineering Education," Second Annual Conference on Systems Engineering Research, Los Angeles, California, 2004.
- [2] Shippmann, J. S., Ash, R. A., Battista, M., Carr, L., Eyde, L. D., Hesketh, B., Kehoe, J., Pearlman, K., and Sanchez, J. I., "The practice of competency modeling," *Personnel Psychology*, 53, 2000, pp 703-740.
- [2] Widmann, E. R., "The Taxonomy of Systems Engineering Competency for the New Millennium," 2000.
- [3] *Guide to the Systems Engineering Body of Knowledge*, INCOSE, 2004.
- [4] *Systems Engineering Fundamentals*, Defense Acquisition University Press, 2001.
- [5] EIA-632, *Processes for Engineering a System*, Electronic Industries Alliance, 1999.
- [6] IEEE 1220-2005, IEEE Standard for Application and Management of the Systems Engineering Process, 2005.
- [7] *A Guide to the Project Management Book of Knowledge*, IEEE Std 1490-2003, Project Management Institute, Inc., 2004.
- [8] "Capability Maturity Model Integration (CMMI), Version 1.1: CMMI for Systems Engineering and Software Engineering, Staged Representation, CMU/SEI-2002-TR-002/ESC-TR-2002-002, December 2001.
- [9] Kirkpatrick, Donald L., "Evaluating Training Programs: The Four Levels", Berret-Koehler Publishers, San Francisco, 1994, 231 pp.

10. BIOGRAPHY

Rick Hefner, Ph.D., is an instructor in the California Institute of Technology Industrial Relations Center, where he teaches systems engineering, project management, and risk management. Dr. Hefner received his BS and MS from Purdue University in Interdisciplinary Engineering, and his Ph.D. from UCLA in Applied Dynamic Systems Control. He has over 25 years of experience in software development, research, and management, and has served in industrial, academic, and government positions. Dr. Hefner is an SEI-authorized Lead Assessor, a charter member of the CMM Integration (CMMI) project, and a Six Sigma Black Belt and trainer

