Project Performance International

SYSTEMS ENGINEERING NEWSLETTER

brought to you by

Project Performance International (PPI)

SyEN 47 – August 16, 2012

Dear Colleague,

SyEN is an independent free newsletter containing informative reading for the technical project professional, with scores of news and other items summarizing developments in the field, including related industry, month by month. This newsletter and a newsletter archive are also available at www.ppi-int.com.

Systems engineering can be thought of as the problem-independent, and solution/technology-independent, principles and methods related to the successful engineering of systems, to meet stakeholder requirements and maximize value delivered to stakeholders in accordance with their values.

If you are presently receiving this newsletter from an associate, you may receive the newsletter directly in future by signing up for this free service of PPI, using the form at www.ppi-int.com. If you do not wish to receive future SE eNewsletters, please reply to the notifying e-mail with "Remove" in the subject line, from the same email address. Your removal will be confirmed, by email.

We hope that you find this newsletter to be informative and useful. Please tell us what you think. Email to: contact@ppi-int.com.

WHAT'S INSIDE:

Quotations to Open On

Read More…

Feature Articles:

- Developing Systems Engineers: A Practice-based Approach - Further Developments

- Software Quality Metrics: Three Harmful Metrics and Two Helpful Metrics Part 2

Read More…

Systems Engineering News

- Condolences for the Family of a Beloved and Respected Leader

- INCOSE IS12 International Symposium Conducted in Rome

- PMI, INCOSE and Lean Advancement Initiative (LAI) at MIT Partner to Find Best Practices for Delivering Successful Programs
- Worcester Polytechnic Institute Partners with INCOSE
- Systems Thinking in Education
- Guide Available for the Application of SE in Construction
- Swiss Society of Systems Engineering Formed
- Survey/Reports on Systems Engineering and Embedded Software Best Practices

[Read More…](#)

Robert's Reflections - Confusion over the term "System of Systems Engineering" (SOSE)

[Read More…](#)

Featured Societies – The OR Society

[Read More…](#)

INCOSE Operations; INCOSE Student Division

[Read More…](#)

Systems Engineering Tools News
- Integrating Spec Data in Models
- Open Source SysML Editor
- SysML Designer (Indigo version) 1.0.2
- Free/Open Source Simulation Software Tools

[Read More…](#)

Systems Engineering Books, Reports, Articles, and Papers
- The Guide to Lean Enablers for Managing Engineering Programs
- Large Scale Complex System and Systems of Systems Engineering Case Studies
- Strategies to the Prediction, Mitigation and Management of Product Obsolescence
- Reliability Engineering, 2nd Edition
- Visual Models for Software Requirements
- Article - Requirements Traceability: The black art of design and development
- Call for Papers - Scientific Research and Impact
- July 2012 Volume 15 Issue 2 of INCOSE INSIGHT

[Read More…](#)

Conferences and Meetings

[Read More…](#)

Education and Academia

- U.S. News and World Report Rates the Tagliatela College of Engineering
- Transnet Centre of Systems Engineering (TCSE) Positions, South Africa

[Read More…](#)

Some Systems Engineering-Relevant Websites

[Read More…](#)

Standards and Guides

- DoDAF (USA Defense Architecture Framework) Future Development
- MIL-STD-881 Revision C, Work Breakdown Structures for Defense Materiel Items

[Read More…](#)

A Definition to Close on – Capability

[Read More…](#)

PPI News

[Read More…](#)

PPI Events

[Read More…](#)

Quotations to Open On

It is time for us all to stand and cheer for the doer, the achiever - the one who recognizes the challenge and does something about it.

-- Vince Lombardi

Men build too many walls and not enough bridges.

-- Sir Isaac Newton

Feature Article

Developing Systems Engineers: A Practice-based Approach

Further Developments

Duarte Gonçalves

+27 12 841 3963
dgoncalv (at) csir.co.za

Council for Scientific and Industrial Research (CSIR)

Abstract

South Africa has been experiencing a shortage of systems engineers. This seems also to be true internationally. Ironically, we in South Africa seem to have only introductory systems engineering courses at local universities. Systems engineers have developed by means of experience on the job. Experience can be a good teacher, but it is also slow and expensive.

This paper is focused on providing a strategic solution to this problem. We start by considering a number of reasons why systems engineering is difficult to learn. A framework for defining the required system engineering competencies is introduced. A practice-based approach is presented as part of the solution, including the roles of universities, students, and industry within this approach. Finally, we elaborate on a proposed curriculum for a practice-based SE educational program.

The shortage of systems engineers requires strategic action. In order to accelerate the development of highly competent systems engineers, we will need to adopt new approaches, some of which have been available for a long time but have fallen into disuse. That which is being proposed will require considerable effort, but is expected to yield substantially improved results that will contribute to developing the next generation of systems engineers.

1. Introduction

Systems engineering (SE) is a critical capability for developing large, complex projects in the South African industry, for example in the defense and aerospace industries, as well as in any organization that applies SE. However, systems engineers with the appropriate levels of competence are in short supply, not only in South Africa, but internationally. One of the reasons for this is that systems engineering is difficult to teach at universities using traditional models. Systems engineering requires both explicit knowledge (which can be taught at a university) and tacit knowledge (like skills and judgment, which are learned through doing). Another factor is that unless students have experienced some of the problems that can occur on large development projects, students will not learn in a meaningful way. So, for the large part, systems engineers have developed through experience and this can be a lengthy process.

The systems engineering world as seen through the International Council on Systems Engineering (INCOSE) is moving from document-centric approaches to model-based systems engineering (MBSE). Many international projects are using MBSE and tools are widely available today. MBSE is seen as a growth area in INCOSE's vision for 2020. So, in addition to the learning issues, emerging developments in architecture frameworks in combination with MBSE require a curriculum that prepares a new breed of systems engineers.

In order to address the learning issues, we propose developing a practice-based systems engineering (PBSE) program as a joint collaboration between universities and industry: Universities present the theoretical background and participating industry organizations provide opportunities to apply material on real projects and work under an experienced systems engineer (a coach).

What we are proposing has been done elsewhere in the world. An ex-South African, Alistair Campbell, has been involved in setting up a similar program at the University of South Australia in collaboration with Australian industry (Campbell and Cropley, 2009). In Europe, European Aeronautic Defense and Space (EADS) is a lead user of a practice-based training platform that not only uses a practice-based approach, but also seeks to develop collective skills (Fournier et al. 2010). There are also parallels with the field of medical education where a number of models (such as the SPICES model), have been developed which are also practice-based (Gonçalves, 2008).

Since the model being proposed represents a change from traditional teaching models, a pilot program has been implemented to understand the implications for universities, industry, and students. The intention is to expand it to the broader industry.

This paper presents the plan for a PBSE program. Accordingly, the next section describes PBSE education in more detail, beginning with why SE is difficult to learn. This is followed by the requirements and considerations for a PBSE course. A SE curriculum is proposed, aimed at SE for developing systems in the early phases of the systems life-cycle.

2. A Practice-based Approach to SE Education

We discuss SE education in the context of achieving engineering work objectives, i.e. delivering successful, working systems. Learning and development are seen as secondary goals that are to be achieved in the process. Bobbit, in his seminal work produced in 1918 (Bobbit, 1918), argues for "work-activities as the only possible normal method of preparing for the work of the world". It is unrealistic to expect employers to accept responsibility for 'preparing' every employee. In general, SE has some significant differences from other professions (discussed below) which require a 'new' approach. Bobbit elaborates further saying that the student "…examines every fact and principle in relation to his practical problem and not merely as a field of intellectual sight-seeing".

Requirements relating to a new program concerned with the development of SE competence, raised in (Gonçalves, 2008), are:

1. A new SE program must transfer both explicit and tacit knowledge components of knowledge relating to SE. Transferring tacit knowledge is more difficult than transferring explicit knowledge, and requires approaches that depart from traditional engineering education.

2. The program must provide a learning context in which SE will be applied. The industry, the types of products being developed, and level of these products in the systems hierarchy and the life-cycle phase of real projects define this context. The context is essential for situated learning.

3. There are three different levels of learning that need to be addressed: individual, group (or team), and intra/inter-organizational. The last two levels require participation (social interaction which includes 'seeing' and doing). Many current programs in South Africa are focused on the individual level.

4. Students wishing to participate in the program must have sufficient "absorptive capacity": prior knowledge that allows SE knowledge and skills to be properly assimilated. As a guideline, students should have at least three years of engineering experience.

5. Any program needs to be student-centered because more important than what is taught is that which the student learns.

A practice-based SE course, in conjunction with screening of students, is believed to address these requirements. To summarize, the proposed approach is characterized by at least three aspects: situated learning; learning that can take place at the three different levels: individual, group (or team) and organization; and tacit knowledge that can be transferred. Explicit knowledge would be transferred by means of lecture modules in conjunction with group exercises. Thus the foundation is built on SE principles (not just "experience"). A fourth aspect, one could argue, is a potentially higher level of emotional involvement, which is a major factor in the level of student learning.

Why do we need this approach? In order to answer this question, a five-level model of SE education is proposed in Table 1. At the highest level, "why", relates to purpose and typically the domain of leadership; "when" is judgment regarding tailoring (to some extent the purpose can also be important in this regard); "what" being the process or knowledge for achieving the purpose; and 'how' is linked to skills. One of the issues with SE standards is that they deliberately avoid the 'how' level because this is dependent on the industry and product under development and the life-cycle phase. The 'who' describes the requisite characteristics of the person or team, an aspect that has received attention locally and internationally (Gonçalves and Britz, 2008).

Why	Purpose
When	Judgment
What	Process (knowledge)
How	Skills
Who	Personal Characteristics

Table 1: The WWWHW model for SE learning

The WWWHW model is a framework that assists in identifying the various SE aspects to be learned. At least three levels of consideration are proposed (see Table 2): Systems engineering (global level); process level; and a method level. The number of levels is determined by the complexity of the project and the system requirements. If we consider the process level, then it is the 'what' of the SE level, with some of the 'how' being considered at the analysis level (a similar model is used by Vincente). Courses may not focus on all the knowledge levels and also not address sufficient levels of consideration. There are some that deal with the 'how', but few are able to cover all levels.

Level of consideration		

	Global	Process	Method
	Systems Engineering	Requirements Analysis	Behavior Analysis
Why	Deliver value to stakeholders	Increase quality of system requirements	Model system's behavior to improve quality of behavior related requirements
When	On all but very small projects	The requirements are not provided or the quality of the specification is a risk to the project	The system has moderate to complex behavior
What	Analyze requirements, develop the architecture and other processes	Analyze context and behavior, develop operational concept description, develop a value model write specifications and other methods	Analyze states and modes, analyze scenarios and other techniques
How	"How" is the next lower level of consideration.		
Who	Person with general SE knowledge, skills and other characteristics	Person with RA knowledge, skills and other characteristics	Person with behavior analysis knowledge, skills and other characteristics

Table 2: Levels of Consideration in the WWWHW model - example

There are a number of challenges in implementing a PBSE program. First, engineers are away from work while they are doing theory. We need to implement the PBSE Program in a way that minimizes impact on work. Secondly, we need to ensure systematic practice. In other words, we need to have either a variety of projects or a single large project where the student can practice the required variety of SE competencies. This may not always be available at one organization. The other concern is that security and confidentiality may hamper the program. Also, the delivery of theory needs to be synchronized to practice in order to close the theory-practice loop. This may make scheduling challenging.

One proposal to address some of these issues is to partition the course into small modules that cover theory and practice-based learning in a specific module. There could be a basic module that covers introduction to SE and a number of other modules, possibly including the process level. The advantage of this approach is that we can deliver, for example, a requirements analysis module, when it is needed on a project. The student is only away for the duration of one module. The practice-based part of the module can be linked to the project that required the competency. We are not attempting to cover an entire SE course all at once. The assumption here is that there are large projects or a sufficient number of smaller projects to provide the practical exposure. Once sufficient modules have been completed, the course is considered to be complete. If the courses are offered largely on a student-centered model (Gonçalves, 2008), then we need to be more flexible concerning when we present the modules. In the next section of this paper, we consider the stakeholders of a PBSE program.

3. Stakeholders of the PBSE Program

There are four main categories of primary stakeholders of the PBSE program (discussed below): the universities, organizations, students (some of whom would be at a university or in an organization), and the SE profession.

Defense, Peace, Safety and Security (DPSS), a unit of the CSIR, is taking the lead on developing the program. While the CSIR as a science council is not part of industry, we will include it under industry as an employer of systems engineers for the purposes of this discussion. It is intended that the PBSE program be extended to the broader industry to mitigate the shortage of systems engineers in South Africa as a national initiative. Based on interactions with a number of industry organizations, there is broader interest beyond just DPSS. This interest needs to be developed further.

At least three categories of students could be considered: undergraduate students, post-graduate study immediately after

the first degree, and study after an initial period of approximately three years of industry experience. Because of 'absorptive capacity' considerations (discussed above and in more detail in Gonçalves, 2008), the PBSE program will not consider undergraduate students. Characteristics of students that should be taken into account are described in Table 3**.**

Secondary stakeholders include clients who require the skills but may impose security requirements. Another secondary stakeholder would be the accreditation institutions.

INCOSE's South African Chapter represents the interests of systems engineers in South Africa and by association also those of their employers, with a number of the employers having a need for developing SE competencies.

View	Study immediately after obtaining a degree Age < 25 years	Study as a working student Work experience >3years
Financial	Low income.	Steady junior engineering salary.
Experience	Little to no work experience. Fresh experience as a student.	Organizational, technology, and project experiences.
Project access	Access may be limited or somewhat artificial.	Ongoing access to projects.
Teams	Not applicable.	The student is a member of a team that delivers the product(s), service(s), or system.
Coach	Access may be limited.	Ongoing access to a coach (although the coach may not always be available).
Theory	Good access to the extent that the SE skills are available at a university.	Study of theory may be limited by the pressures of work. The student may not get a good framework.
Time	Attending classes 2-3 hours each weekday.	Working typically 8 hours per weekday.
Motivation source	Self-motivated.	Self-motivated, plus work-related motivations (such as building a career, reputation, future assignments, bonuses, and prestige).

Table 3: Characteristics of Students to be Considered for PBSE

4. Requirements for a PBSE Program

In this section, the need for systems engineers is defined in terms of SE competencies and the ability to deal with problems and solutions. Some requirements that enable learning and organization-specific requirements are identified. Matters relating to certification, accreditation, projects, supervision, and assessment of students, intellectual property, and security are discussed. The section concludes with a concept for the PBSE program roles and responsibilities.

4.1. Need for Systems Engineers

Frameworks used to define the need in terms of SE competencies and the ability to deal with problems and solutions include:

1. For competencies, INCOSE UK Systems Engineering Competencies Framework (INCOSE UK, 2006)

2. For the ability to deal with problems and solutions, Kasser et al. Five Types of SEs (2009).

The industry need is to develop highly competent systems engineers. Competence requires knowledge, skill, and psychological characteristics. We propose using the INCOSE UK Systems Engineering Competencies Framework (INCOSE UK, 2006) that defines 21 competencies (see Table 4) and four levels of competence: awareness (A), supervised practitioner (SP), practitioner (P), and expert (E). High-competence is defined as practitioner or expert level. The priority for developing each competency is high (H), medium (M) or low (L). The DPSS priorities are requirements analysis (which is distinguished from requirements management) and architecture. It is likely that these two competencies would be a priority across industries, but we expect priorities of other competencies to vary across industries. The DPSS priorities for other competencies are preliminary. Modules do not need to be structured along the lines of competencies. In fact, we may want to structure modules along broad process lines or life cycle while avoiding industry specific terminology. The basic concepts

should be applicable across industries, but this would need to be validated (a research project relating to this topic is planned). Students would need to pick a set of four to six competencies that they would focus on based on the industry organization.

Category	Competency	Competence Level	Priority
Systems Thinking	System Concepts	P	H
	Super System Capability Issues	P	H
	Enterprise & Technology Environment	P	H
Holistic Lifecycle View	Determining and Managing Stakeholder Requirements	P	H
	Systems Design – Architectural Design	P	H
	Systems Design – Concept Generation	P	M
	Systems Design – Design for "ilities"	SP	M
	Systems Design – Functional Analysis	P	H
	Systems Design – Interface Management	P	H
	Systems Design – Maintain Design Integrity	P	M
	Systems Design – Modeling & Simulation	P	M
	Systems Design – Select Preferred Solution	P	H
	System Design – System Robustness	SP	L
	System Integration & Verification	P	M
	Validation	P	M
	Transition To Operation	SP	L
Systems Engineering Management	Concurrent Engineering	SP	L
	Enterprise Integration	SP	M
	Integration of Specialties	SP	M
	Lifecycle Process Definition	SP	M
	Planning, Monitoring & Controlling	P	H

Table 4: Typical SE Competency Requirements for DPSS

However, this competency framework has some issues. It is focused on developing systems where the requirements are already largely developed. At DPSS, we need some of these skills, but most critically the ability to define the problem. The INCOSE UK Framework is limited in the area of requirements analysis. It refers to functional analysis, but this is actually functional design. Functional analysis (which uses the same notation, but with different rules), as part of requirements analysis, is not considered. Problem definition will not be fully covered initially but is a critical skill for systems engineers working in the early system lifecycle phases. Five types of SEs can be defined based on their ability to deal with problems and solutions, described in Table 5 (Kasser et al., 2009).

Type I	This type is an "apprentice" who can be told "how" to implement the solution and then be expected to implement it.
Type II	This type is the most common type of systems engineer. Type II's have the ability to use the systems engineering process to figure out how to implement a physical solution, once they are told what conceptual solution to implement. Most systems engineers fall into this category.
Type III	Once given a statement of the problem, this type has the necessary know-how to conceptualize the solution and to plan the implementation of the solution.
Type IV	This type has the ability to examine the situation and define the problem.
	This type combines the abilities of the Types III and IV- that is, has the ability to examine the situation, define the

| Type V | problem, conceptualize the solution, and plan the implementation of the physical solution. |

Table 5: Five Types of Systems Engineers

Type I is a transitory educational level. The main DPSS requirement is for SEs at levels III to V, with the main focus on Type IV. This is consistent with the fact that DPSS develops products in low volumes and its main focus is on the feasibility part of the system life-cycle.

Both the CSIR and the universities are required to produce research outputs. There are three categories where research outputs could be produced:

1. Development and evaluation of a PBSE program,

2. Systems engineering, and

3. Project related research.

While it is not a requirement of this program to produce such outputs, should they be produced, they can be published. It may be possible to publish project-related research, but this would need to be discussed between the university and the organization on a case-by-case basis, subject to intellectual property and security considerations.

4.2. Organization Specific Requirements

It is likely that each organization will have specific requirements. For example, DPSS may screen candidates in terms of psychological characteristics. Demographic transformation goals are also important for DPSS as a government funded organization. It is anticipated that universities also might want to utilize the SE modules as part of other programs, where DPSS is for example looking more towards certificate courses.

4.3. Roles and Responsibilities

The roles and responsibilities presented here are based on competence in the various areas and align with the organization's mission. Universities would be best suited to presenting the SE principles (if lecturers with the knowledge and skills are available), evaluating the students and accrediting the course. The universities have long-standing experience in certification and accreditation of students and courses respectively. We are not currently looking for SE researchers (although this is a longer term consideration); rather systems engineers with theoretical grounding and skills. While a Masters degree could be used as a mechanism, a certificate course might be preferred. The best course of action appears to be university specific and on this issue we will follow the university's preference. Two certificates may be required (it would be necessary to clearly distinguish between them):

- A certificate for organizations in which a culture of SE does not yet exist. In these cases, only the theory would be offered as a certificate.

- A certificate for the full practice-based approach, including the theory.

Whatever approach is chosen, the program would need to be accredited (with, for example, the South African Qualification Authority) following execution of the pilot program in order to increase industry acceptance and ensure quality.

The industry organization would provide real projects, a SE coach to guide the student, and a suitable and stable environment for learning. SE work will be done in the context of a live project with defined deliverables. Students should not be leading such projects if there is significant project risk and should work with a coach from the organization. The student will develop SE competencies by applying SE principles on real projects within a team and organizational context. Working as part of a team leads to development of social skills required for SE. The material provided by Young (2006, Chapter 5) provides discussion of both the need for teamwork and how to foster effective teamwork. The university supervisor, the coach, and the student's immediate supervisor or project manager would perform the assessment of the student, with defined roles led by the university supervisor. To ensure a consistent standard across industry, we would need to define module level objectives to be achieved in practice. Students would be assessed on these as proposed in the following section.

Development of the PBSE pilot program might be done jointly by universities and DPSS. An alternative being explored is to re-use suitable material from other courses. Each module (discussed in the following section) would be presented by a competent lecturer, whether from industry, a university, or some other organization.

5. Proposed PBSE Pilot Curriculum

This section provides an outline of the PBSE modules and how these map to the required SE competencies.

5.1. Outline of SE Modules for the PBSE Program

An overview of the PBSE modules that are envisioned is presented in Figure 1. The pilot will focus on the SE introduction (mandatory first module), requirements analysis (RA), architecture, and SE management modules. Modeling and simulation, integration, verification, validation, and specialty engineering will follow, once the practical issues of a practice-based approach have been resolved. Building the foundation for MBSE will be a central theme all the modules. Many international projects are using MBSE and tools are widely available today. We will delve into those modules that are directly relevant to the pilot, bearing in mind the why, when, what, how, and who parts of the framework which are discussed in the next sub-section.

The focus of the modules is on material that has broad applicability. For example, SE standards represent best practice. Companies will select a certain standard over another. The modules should endeavor to give an overview of such areas, but will not address such material in detail. It would be better for the student to learn this within his/her company context, including company-specific tailoring.

A number of sources of information were consulted in compiling this curriculum:

1. Literature (Kasser 2007, Squires and Cloutier 2009)
2. Other programs (PPI Course notes, MIT course material 2009, University of South Australia, 2009).
3. Personal experience of the author and discussions with colleagues.

Many aspects of problem definition are addressed as part of the RA module. While some of the approaches that are used in RA can be applied, additional tools may be required. Other areas, such as designing the developing organization and enterprise integration will also need to be developed.

5.2. The SE introductory Module

The SE introductory module (Figure 2) presents the objectives, principles, and overview of the PBSE course. The practice-based format will be new to students, so the roles and expectations of the university and those of the employer will need to be defined.

The SE core module seeks to introduce the basic concepts and motivation for applying SE. The module starts by looking at why projects fail. This is the reason for the existence of SE and leads to its purpose. Basic concepts must be introduced, for example, "What is a system?", "What is a system life-cycle?" and "What is systems thinking?" These themes will be reiterated through the other modules. SE principles should be covered in this module along the lines of, for example: Understand the problem before committing to the solution. Modeling notations for SE are introduced in the core module. These representations support understanding, reasoning, and communication aspects of the system, which are fundamental issues in SE. This is true not only for technical aspects of engineering but also for management aspects such as planning. Architecture frameworks are not explicitly presented in the module because these are application specific. However, there is an implicit architecture framework underlying the modeling notation discussed in this module. The two fundamental viewpoints are behavior and structure – essential in understanding architecture frameworks. The criteria for selecting a modeling notation need to be presented – syntax and semantics (expressiveness), rigor, and understandability (Buede, 2000), before a number of modeling notations are introduced.
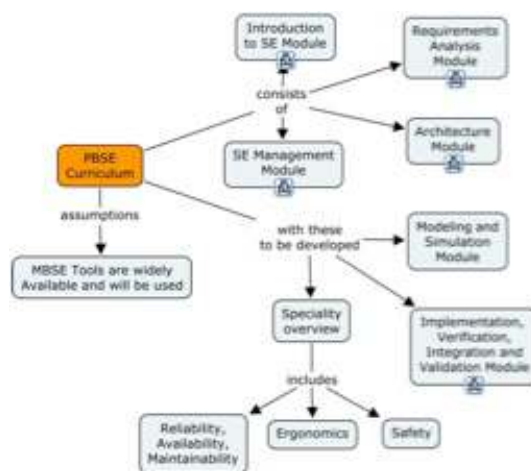

Figure 1: PBSE - Curriculum Overview

5.3. Requirements Analysis Module

One of the larger modules will be requirements analysis (RA) – this is appropriate given the importance of requirements. The view taken here is that requirements elicitation is tightly interleaved with analysis.

Figure 3 provides an overview of the RA Module. Starting with the purpose of RA, the requirements process and requirements types need to presented. An area which needs some attention is elicitation techniques, using scenarios for example, and sources of requirements. Considerable effort is spent on techniques for RA, including the purpose and applicability of each technique. These are essential to defining the problem before any specifications are written. Students will need to develop the discipline of separating the problem from the solution. The characteristics of good requirements (requirements quality) should be addressed in conjunction with writing specifications. Managing RA ranges from planning a RA effort to creating traceability to stakeholders and operational concepts. A healthy dose of emphasis on iteration is required.

Product scoping, as proposed by Hooks and Farry, (2000), may be very useful in the context of RA to create a common vision and draw a boundary concerning what is or is not a requirement and a tool for gauging the size of the effort.
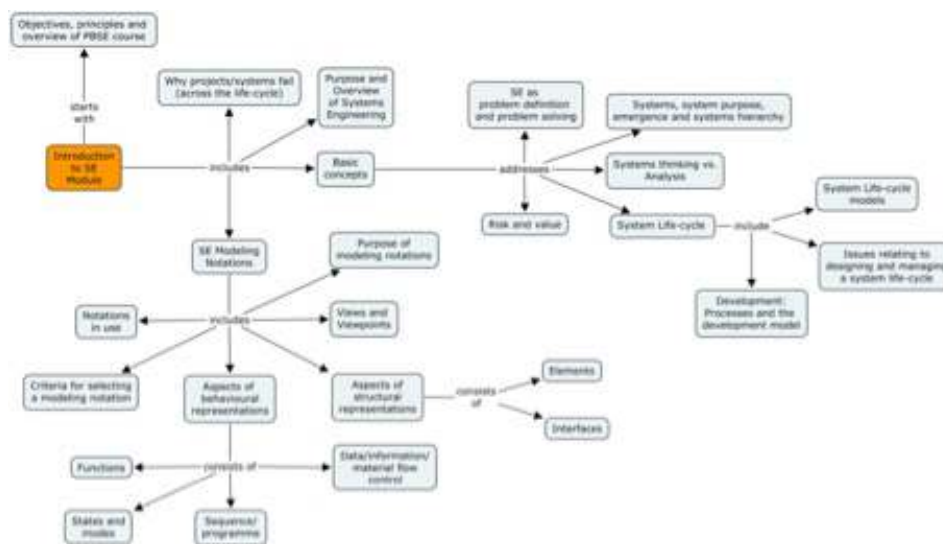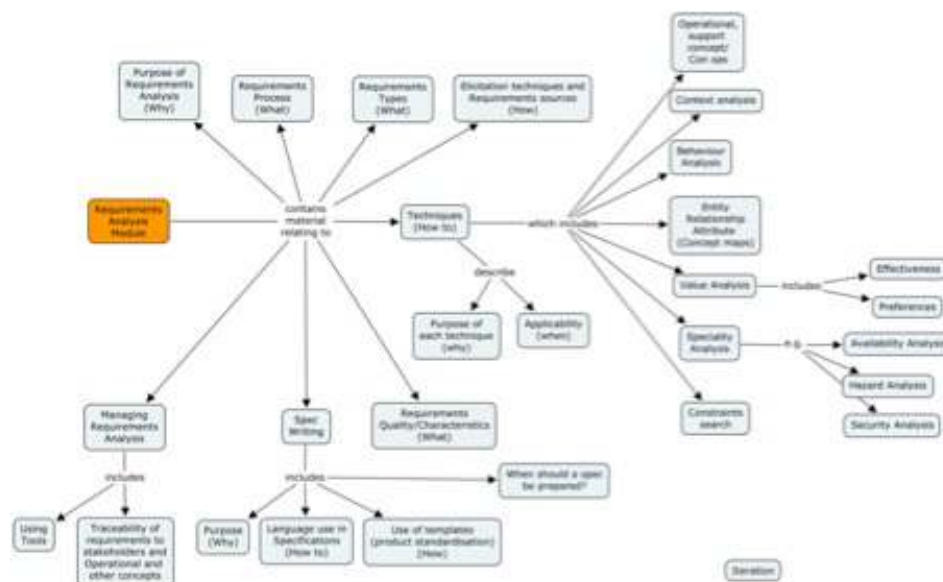

Figure 2: SE Introductory Module

Figure 3: Requirements Analysis Module

5.4. Architecture Module

An overview of the architecture module is presented in Figure 4. Architecture is not a mature field with a widely accepted underlying theory. For this reason, a number of approaches to architecture need to be presented. This depends on whether these are software, or hardware and the specific type of hardware systems e.g. largely signal processing, for example, radar. The importance of identifying the drivers of architecture early on needs to be communicated to students (why, what and how). Key to development of architecture is creativity, dealt with in concept generation. Alternatives need to be generated both at the system level and at function level. Concept generation is supported by behavior analysis (part of which is functional analysis). The architecture module would need to cover both the development of structural (physical) and behavioral aspects of architecting. Interfaces would be dealt with as part of the structural architecture. Concepts relating to

the development of alternatives, the evaluation of these and the selection of candidate architectures needs to be presented. The issue of traceability from requirements, functions, and allocation to system elements needs to be covered. The concept of technical budgeting supported by modeling and simulation needs to be introduced. Technology as the basis of any solution and the concept of technology maturity need to be presented. Again a healthy dose of emphasis on iteration is required and when to stop. As a final point solution is developed for the full life-cycle and consideration must be given for how it will be evolved once implemented as new requirements emerge and others change.
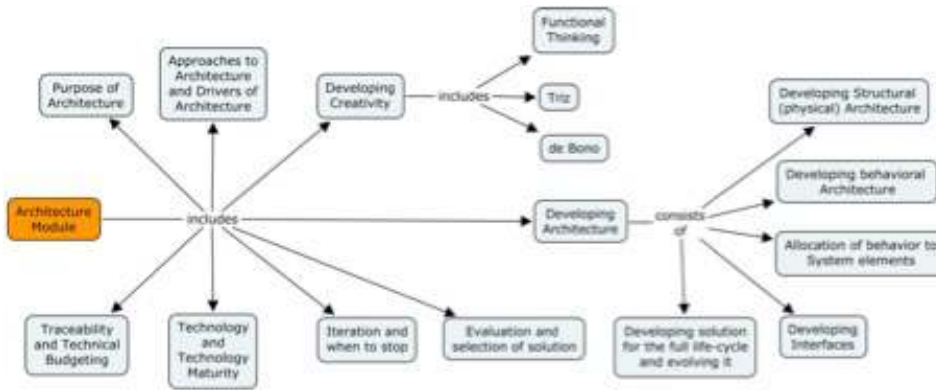

Figure 4: Architecture Module

5.5. SE Management Module

An SE management module is planned as presented in Figure 5. This module introduces risk management, configuration management, technical performance management, concurrent engineering management, specialty engineering, interface management and quality assurance. For each of these, the purpose, what needs to be done, how it should be approached will be presented.

SE planning receives considerable attention in this module. The diagram shows the planning for the development phase of the life-cycle. Planning for other life-cycle phases (production, transition to operation, operation and support, disposal) will need to be introduced. The emphasis should be on how to identify the life-cycle phases, based on technology maturity and other requirements.

Planning (in the development part of the lifecycle) deals with defining the processes to be followed, defining the SE products that will be produced (documents, models, etc.) and allocating responsibilities. How the processes will be sequenced is defined by the development model based on considerations such as risk and is dependent on application maturity (low maturity leading to an evolutionary approach) or technology maturity.

Critically, the relationship between SE and project management needs to be discussed. It is especially difficult to cost the development of any project before the first cut problem definition, RA, and architecture have been developed. The development of a work/product breakdown structure is an important tool for managing the cost, schedule and risk on a project.
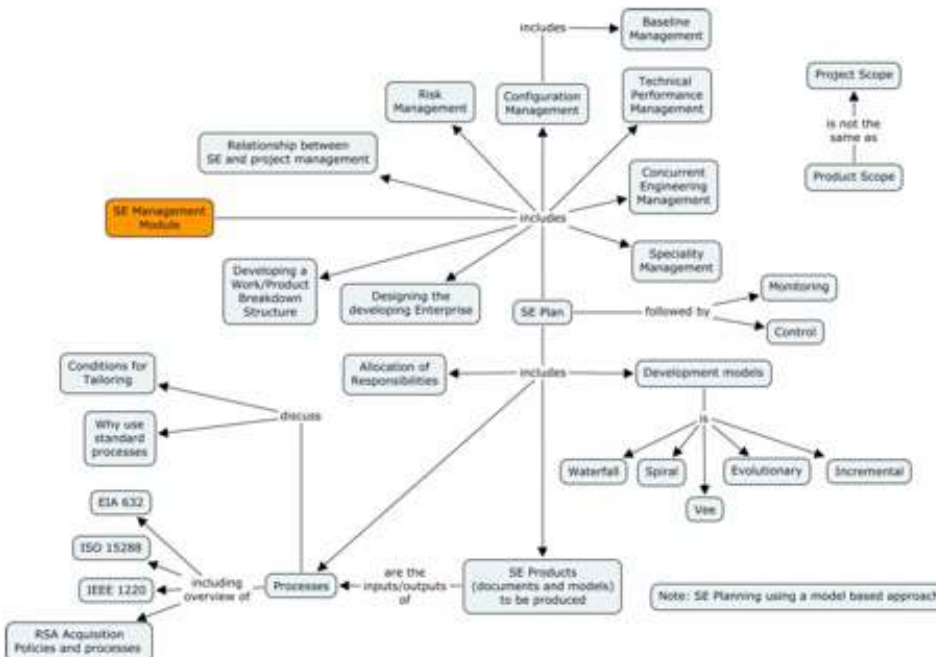
5.6. Implementation, Integration, Verification, and Validation

The Implementation, Integration, Verification, and Validation module deals with realizing the solution, integrating and verifying various elements of the solution, and checking that the solution meets the stakeholder needs. An overview of the module is presented in Figure 6. While the 'V' model is the traditional approach for explaining integration, verification and validation, it is a rather simplified model. More emphasis needs to be placed on a plan for implementation, verification, and integration. Depending on the nature of the system, there may be a transition to operation before validation can be performed.

Early validation is an important area to emphasize in this module. In the early phases of the system lifecycle, use of simulations can significantly reduce risk. This form of validation occurs before any physical implementation or integration. However, implementation and integration have been grouped with verification and validation because these are normally intimately related, a fact that is overlooked in some courses.
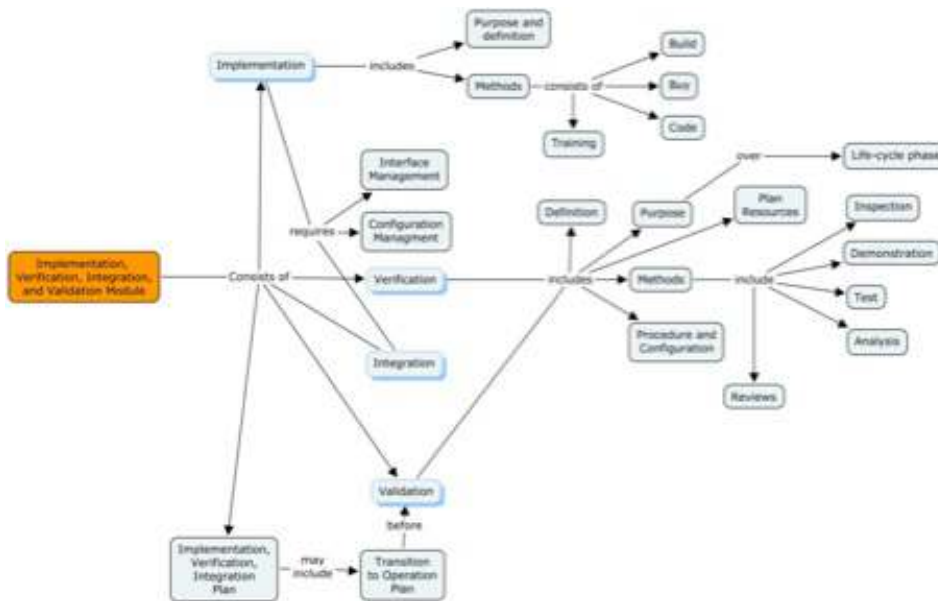

Figure 6: Implementation, Integration, Verification, and Validation

5.7. Mapping of SE competencies to SE modules

Six modules have been proposed as part of a PBSE program. The competencies addressed by the proposed modules are shown in a number of competency areas remain to be addressed, for instance, enterprise issues are not currently addressed. Other aspects that might need additional support are the integration of specialties and decision analysis.

Category	Competency	Module
Systems Thinking	System Concepts	SE core
	Super System Capability Issues	SE core, Requirements analysis
	Enterprise & Technology Environment	Partly addressed in Architecture
Holistic Lifecycle View	Determining and Managing Stakeholder Requirements	Requirements analysis
	Systems Design – Architectural Design	Architecture
	Systems Design – Concept Generation	Architecture
	Systems Design – Design for…	Partly addressed in Architecture
	Systems Design – Functional Analysis	SE modeling concepts and notations
	Systems Design – Interface Management	Requirements analysis, Architecture
	Systems Design – Maintain Design Integrity	SE management
	Systems Design – Modeling & Simulation	Modeling & simulation

	Systems Design – Select Preferred Solution	Foundations laid in Requirements analysis module, Architecture
	System Design – System Robustness	Reliability, availability and maintainability module
	System Integration & Verification	Implementation Integration, verification and validation
	Validation	
	Transition To Operation	
Systems Engineering Management	Concurrent Engineering	SE management
	Enterprise Integration	
	Integration of Specialties	Specialty overview
	Lifecycle Process Definition	SE management
	Planning, Monitoring & Controlling	

Table 6: Mapping of SE Competencies to Modules

6. Conclusion

The shortage of systems engineers in South Africa and internationally requires strategic action. This paper focuses on providing a strategic solution to this problem. A practice-based systems engineering (PBSE) program characterized by situated learning; learning that can take place at individual, group (or team) and organizational levels; and tacit knowledge such as skills and judgment that can be learned through doing. Six modules are proposed for the PBSE Program: PBSE Curriculum Overview, Requirements Analysis, Systems Engineering Introduction, Architecture, Systems Engineering Management, and Implementation, Integration, Verification, and Validation. A mapping of systems engineering competencies to the modules was provided. It is believed that this program will contribute to developing the next generation of systems engineers. In order to accelerate the development of highly competent systems engineers, we need to adopt new and re-energized approaches to education with current systems engineering content. What is proposed will require considerable effort, but is expected to yield substantially improved results. This paper does not address the broader national SE education in South Africa.

7. References

Bobbitt, J. The Curriculum. Boston: Houghton Mifflin, 1918.

Buede, D. M., The Engineering Design of Systems: Models and Methods, Wiley Interscience, 2000.

Campbell, A and Cropley, D, Creating a Master's Degree in systems integration: Capturing and embedding industry knowledge. Int. Journal of intelligent Defense Support Systems, Vol. 2, No. 3, 2009.

Fournier, F., Pohl-Winkelmann, C. & Wilke, D. M. Using training platforms to develop collective skills and performance in Systems Engineering Proceedings of the 7th biannual European Systems Engineering Conference (EuSEC), 2010

Gonçalves, D.P. Developing Systems Engineers, Portland International Centre for Management of Engineering and Technology (PICMET), July 2008.

Gonçalves, D.P. and J. Britz, Systems Engineering Research: Screening Candidate Systems Engineers. DPSS-SM-CPG-044 Rev 1, 2008.

Grady, J. O., System Requirements Analysis, McGraw-Hill, Inc., 1993.

Hooks, I. & Farry, K., Customer-Centered Products: Creating Successful Products Through Smart Requirements Management, American Management Association, 2001.

Honour, E., "Understanding the Value of Systems Engineering", Proceedings of the Annual International Symposium of INCOSE, 2009.

INCOSE UK, 'Systems Engineering Competencies Framework', Technical report, Issue 2. 2006. Available from World Wide Web http://www.incose.org.uk/Downloads.

Kasser, J.; Hitchins, D. & Huynh, T. "Reengineering Systems Engineering",

Proceedings of the 3rd Asia-Pacific Conference on Systems Engineering (APCOSE), 2009.

Kasser, J. E., Integrated Multidisciplinary Engineering for the 21st Century: Study Guide, 2007.

Squires, A. & Cloutier, R., "Evolving the INCOSE reference curriculum for a graduate program in systems engineering", Systems Engineering, Vol. 12. 2009.

Halligan, R., Systems Engineering: From awareness of need to retirement from use, PPI-XB10-003 162-2, Project Performance International, 2010.

Massachusetts Institute of Technology: MITOpenCourseware, http://ocw.mit.edu/OcwWeb/Engineering-Systems-Division/index.htm.

University of South Australia, Master of Engineering (Military Systems Integration), http://www.unisanet.unisa.edu.au/programs/program.asp?Program=LMSI&Year=2009.

Young, R.R., Project Requirements: A Guide to Best Practices. Vienna, Virginia (USA): Management Concepts, 2006.

Additional Feature Article

Software Quality Metrics:

Three Harmful Metrics and Two Helpful Metrics

Part 2 of 2

(Continued from Part 1, which was published in the previous edition of SyEN)

Capers Jones, VP and Chief Technology Officer
Namcook Analytics LLC

The Benefits of Function Point Metrics for Data Normalization

Function point metrics were developed Allan Albrecht and his colleagues at IBM and placed in the public domain in 1978. The International Function Point Users Group (IFPUG) took over the counting rules and function point training, and has grown into the largest measurement association in the world with affiliates in 24 countries.

In recent years a number of function point "clones" have been developed which differ slightly in counting rules. Among the many variants are COSMIC function points, FISMA function points, NESMA function points, function points light, engineering function points, feature points, and backfired function points.

There are also a number of specialized metrics that use some of the logic of function point analysis but mix in other counting rules. Two of the more common variants are use-case points and story points. Table 5 shows the comparative sizes of 15 functional metrics circa 2012:

	Functional Metrics	Size	% of IFPUG
1	IFPUG function points	1,000	100.00%
2	Backfired function points	1,000	100.00%
3	Cosmic function points	1,143	114.29%
4	Fast function points	970	97.00%
5	Feature points	1,000	100.00%
6	FISMA function points	1,020	102.00%

7	Full function points	1,170	117.00%
8	Function points light	965	96.50%
9	Mark II function points	1,060	106.00%
10	NESMA function points	1,040	104.00%
11	RICE objects	4,714	471.43%
12	SNAP non functional metrics	235	23.53%
13	Story points	556	55.56%
14	Unadjusted function points	890	89.00%
15	Use case points	333	33.33%

Table 5: Comparative Sizes of Functional Metrics Circa 2012

In 2011, IFPUG issued new counting rules for non-functional size elements such as quality, performance, and the like. These are called "SNAP metrics" and are too new to have much empirical data.

The reason that IBM spent several million dollars inventing function points and the reason that function points are the most widely used metric in the world is that they actually demonstrate standard economic concepts. Function points can be used to normalize data in a fashion that matches standard economic assumptions.

Recall that Table 1 showed a significant increase in cost per defect throughout the testing cycle. This is the basis for the urban legend that it costs 100 times more to fix a bug after release than before.

Let us revisit the underlying data from Table 1 and see what happens when we normalize defect removal effort using cost per function point instead of cost per defect. Table 6 uses exactly the same effort used in Table 1:

- Writing test cases takes 16.5 hours for every test stage;
- Running test cases takes 9.9 hours for every test stage; and
- Defect repair takes 5.0 hours for every defect found.

Table 6 shows the results normalized using function points instead of cost per defect (but the underlying effort is identical in the two tables):

	Writing Test Cases	Running Test Cases	Repairing Defects	TOTAL $ PER F.P.	Number of Defects
Unit test	$12.50	$7.50	$189.38	$209.38	50
Function test	$12.50	$7.50	$75.75	$95.75	20
Regression test	$12.50	$7.50	$37.88	$57.88	10
Performance test	$12.50	$7.50	$18.94	$38.94	5
System test	$12.50	$7.50	$11.36	$31.36	3
Acceptance test	$12.50	$7.50	$3.79	$23.79	1

Table 6 Cost per Function Point for Six Forms of Testing
(Assumes $75.75 per staff hour for costs)

Notice the complete reversal of costs when function point metrics are used. Instead of becoming more expensive when few bugs found, defect removal costs per function point steadily decline as fewer and fewer bugs are found!

In other words defect repairs do not increase over time, they become cheaper over time. Table 6 is nothing more than the data from Table 1 with normalization based on cost per function point.

For the first time using function points, it is possible to actually study the economics of software quality in a fashion that matches standard economic assumptions, instead of distorting standard economic assumptions. This is why functional metrics are so powerful: they reveal real economic facts that are hidden by cost per defect, lines of code, and technical debt.

This article is based on IFPUG function points, but the same logic applies to COSMIC, FISMA, NESMA, and all the other function point variants. The only caveat is that the others will produce slightly different results.

The slow speed and high costs of manual function point counting have lowered the acceptance of these powerful metrics. As of 2012, several high-speed and low-cost function point methods are available that can reduce the costs of counting function points from more than $5.00 per function point counted down below $0.01 per function point counted. Within a few years these high-speed methods should make function points an industry standard.

The Benefits of Defect Removal Efficiency (DRE)

The most powerful and useful quality metric ever developed is that of "defect removal efficiency" (DRE). The reason for this claim is that improvements in DRE bring with them improvements in software schedules, software development costs, software maintenance costs, customer satisfaction, team morale, and stakeholder satisfaction. In other words, DRE is the central metric around which process improvements pivot.

DRE metrics were first developed inside IBM in the early 1970's as a method of evaluating the effectiveness of software inspections compared to software testing. As DRE usage expanded, it was found that DRE is perhaps the single most important software metric, because it is the best indicator of project health and also of development speed, costs, customer satisfaction, and quality.

Projects with DRE below 85% will always run late, will always be over budget, and will never have happy customers. On the other hand, projects with DRE above 95% will usually be on time, usually be under budget and usually have happy customers. No metric is perfect, but DRE is the best indicator of project health ever devised.

Defect removal efficiency is not too difficult to measure, nor is it an expensive metric. The essential math of DRE is to accumulate counts of all bugs during development. Then after 90 days of customer use, aggregate user defect reports with internal defect reports and calculate the percentage of bugs removed prior to release.

For example if your development team found 95 bugs before release and users reported 5 bugs in the first three months of usage, then DRE is obviously 95%.

(One caveat is that the International Software Benchmark Standards Group (ISBSG) uses only 30 days of customer usage in calculating DRE. Therefore, they always have higher DRE numbers than the author because 30 days of usage only reports about 20% of the bug volumes found in 90 days. Why ISBSG chose 30 days in unknown, since IBM and other companies have been using 90-day DRE measures since the early 1970's and the bulk of all published studies of DRE are based on 90 day windows.)

Table 7 illustrates two scenarios. Case A shows low quality without the use of pre-test inspections and static analysis. Case B shows high quality that includes the use of pre-test inspections and static analysis. Both Case A and Case B start with a defect potential of 1,000 defects. Case A uses only a standard sequence of testing. Case B uses pre-test static analysis and pre-test inspections before testing starts:

	Case A Low Quality		Case B High Quality	
Defect Potential		1,000		1,000
	Efficiency		Efficiency	
Pre-Test Removal				
Static analysis	0.00%	1,000	60.00%	400

Pre-Test inspection	0.00%	1,000	85.00%	60
Test Removal				
Unit test	25.00%	750	30.00%	42
Function test	27.00%	548	33.00%	28
Regression test	25.00%	411	30.00%	20
Performance test	12.00%	361	17.00%	16
Component test	33.00%	242	37.00%	10
System test	35.00%	157	40.00%	6
Acceptance test	15.00%	134	15.00%	5
Delivered defects		134		5
DRE	86.60%		99.50%	

Table 7: Examples of High and Low Defect Removal Efficiency

Note that pre-test inspections have a secondary benefit of raising testing efficiency levels. This is why testing efficiency is higher in Case B than in Case A.

Readers might think that while it is good to achieve high levels of defect removal efficiency, the costs might be prohibitive. This is a major economic misunderstanding by the software industry. High quality is not expensive. High quality is cheaper than low quality because testing costs are greatly reduced by pre-test inspections and static analysis. Table 8 show an approximate cost comparison of the differences between Case A and Case B:

	Case A Low Quality	Case B High Quality
Pre-Test Removal		
Static analysis	$0	$5,000
Pre-test inspection	$0	$50,000
Test Removal		
Unit test	$25,000	$10,000
Function test	$25,000	$15,000
Regression test	$10,000	$5,000
Performance test	$10,000	$5,000
Component test	$20,000	$15,000
System test	$25,000	$10,000
Acceptance test	$10,000	$5,000
Total Cost	**$125,000**	**$120,000**

Table 8: Cost Comparison of Low Quality and High Quality

In spite of the fact that pre-test inspections cost $50,000 the total cost of quality for the high-quality Case B is $5,000 less than the total cost of quality for the poor quality Case A.

The economic cost savings that accrue from high quality and high DRE cannot be measured using the three bad metrics of cost per defect, lines of code, and technical debt. But they can be measured using the combination of the two good metrics, function points and defect removal efficiency (DRE).

DRE is the most critical metric in all of software because it is the lynch pin of process improvements. Effective process improvement will raise DRE well above 95%. Any methodology that does not measure and seek to improve DRE is essentially ineffective.

Software Metrics Research Laboratories and Research Tools

Ideally every proposed metric would be formally evaluated at a metrics research facility at a university or non-profit think tank. Unfortunately, software metrics just pop up like mushrooms after a rain without any formal evaluation or examination under controlled conditions.

To improve the rigor of metrics study, the author and Namcook Analytics LLC have built a metrics research tool. This tool, Software Risk Master™ (SRM) allows metrics to be evaluated under controlled conditions. For example SRM supports side-by-side analysis of cost per defect, function points, lines of code, technical debt, and DRE for the same application.

Users have the ability to specify exact quantities of defects in requirements, design, code, user documents, and bad fixes. Then they can follow both defect prevention and defect removal through any possible sequence of inspections, static analysis, and testing. Several of the tables in this report are taken from Software Risk Master™.

As defects are removed, costs are normalized using function points, cost per defect, lines of code, and technical debt and the results are shown in a side-by-side format. This makes it easy to examine each metric in turn. Other metrics such as story points and use-case points could also be used.

The tool can also show the results of 32 different kinds of methodologies such as Agile, XP, pair-programming, RUP, TSP, EVO, PRINCE2®, Merise, waterfall, etc.

Controlled results using the Software Risk Master™ tool show the hazards of the three bad metrics. The results indicate that cost per defect rises steadily and of course rises to infinity for zero-defect results. These results clearly violate standard economics.

The costs per line of code for defect removal are cheapest for assembly language and rise steadily when newer and more powerful languages are used such as Java, Ruby, Perl, Objective C, Smalltalk, and the like. Here too, the results violate standard economics because defects are less numerous and repairs are cheaper with modern languages.

For technical debt, the large costs of pre-release defect removal and the overhead costs of customer support and change teams are not included, so technical debt covers only a small percentage of cost of quality.

With functional metrics, standard economics finally arrive in the software world and the true reduction in costs with better quality becomes visible. Functional metrics also work for pre-release defects and for overhead costs.

Simultaneous results are shown for all of these major metrics. Additional metrics such as use-case points and story points can also be tested in side-by-side form. But for the purposes of this article, SRM can be used to demonstrate the economic distortions of the three bad metrics using identical defect volumes and controlled sequences of defect removal activities.

Summary and Conclusions about Software Quality Metrics

For more than 60 years, the software industry has lacked solid economic understanding of basic topics such as cost of quality and defect removal costs. The three bad metrics cited in this article distort economic reality and give the false impression that software quality is expensive, when in fact high quality is cheaper than poor quality.

In order to create valid economic models of software development, maintenance, and quality control it is urgent to have accurate measurements that use accurate metrics. The industry cannot afford the gaps and errors of bad metrics such as cost per defect, lines of code, and technical debt.

The combination of function point metrics combined with defect removal efficiency metrics (DRE) can show the true cost of quality and illustrate the fact that achieving high quality is the most cost-effective way to build software.

References and Readings on Software Quality and Software Metrics

Boehm, Barry Dr.; Software Engineering Economics; Prentice Hall, Englewood Cliffs, NJ; 1981; 900 pages.

Booch Grady, Object Solutions: Managing the Object-Oriented Project; Addison Wesley, Reading, MA; 1995.

Bundschuh, Manfred and Dekkers, Carol; The IT Measurement Compendium; Springer; Berlin; 2008.

Capability Maturity Model Integration; Version 1.1; Software Engineering Institute; Carnegie-Mellon Univ.; Pittsburgh, PA; March 2003; http://www.sei.cmu.edu/cmmi/

Brooks, Fred: The Mythical Man-Month, Addison-Wesley, Reading, Mass., 1974, rev. 1995.

Charette, Bob; Software Engineering Risk Analysis and Management; McGraw Hill, New York, NY; 1989.

Charette, Bob; Application Strategies for Risk Management; McGraw Hill, New York, NY; 1990.

Cohn, Mike; Agile Estimating and Planning; Prentice Hall PTR, Englewood Cliffs, NJ; 2005; ISBN 0131479415.

DeMarco, Tom; Controlling Software Projects; Yourdon Press, New York; 1982; ISBN 0-917072-32-4; 284 pages.

Ebert, Christof and Dumke, Reiner; Software Measurement: Establish, Extract, Evaluate, Execute; Springer, Berlin; 2007.

Ewusi-Mensah, Kweku; Software Development Failures; MIT Press, Cambridge, MA; 2003; ISBN 0-26205072-2276 pages.

Gack, Gary; Managing the Black Hole – The Executives Guide to Project Risk; The Business Expert Publisher; Thomson, GA; 2010; ISBSG10: 1-935602-01-2.

Galorath, Dan; Software Sizing, Estimating, and Risk Management: When Performance is Measured Performance Improves; Auerbach Publishing, Philadelphia; 2006; ISBN 10: 0849335930; 576 pages.

Garmus, David and Herron, David; Function Point Analysis – Measurement Practices for Successful Software Projects; Addison Wesley Longman, Boston, MA; 2001; ISBN 0-201-69944-3;363 pages.

Gilb, Tom and Graham, Dorothy; Software Inspections; Addison Wesley, Reading, MA; 1993; ISBN 10: 0201631814.

Glass, R.L.; Software Runaways: Lessons Learned from Massive Software Project Failures; Prentice Hall, Englewood Cliffs; 1998.

Harris, Michael; Herron, David, and Iwaniciki, Stacia; The Business value of IT; Auerbach; 2008.

Hill, Peter R. Practical Software Project Estimation; McGraw Hill, 2010

Harris, Michael; Herron, David; and Iwanicki, Stacia; The Business Value of IT: Managing Risks, Optimizing Performance, and Measuring Results; CRC Press (Auerbach), Boca Raton, FL: ISBN 13: 978-1-4200-6474-2; 2008; 266 pages.

Humphrey, Watts; Managing the Software Process; Addison Wesley, Reading, MA; 1989.

Jones, Capers and Bonsignour, Olivier; The Economics of Software Quality; Addison Wesley, 2011; ISBN 978-0-13-258220-9; 57 pages.

Jones, Capers; Software Engineering Best Practices; McGraw Hill, 2009; ISBN 97800-07-162161-8; 660 pages.

Jones, Capers; Applied Software Measurement; McGraw Hill, 3rd edition 2008; ISBN 978-0-07-150244-3; 668 pages; 3rd edition (March 2008).

Jones, Capers; Estimating Software Costs; McGraw Hill, New York; 2nd edition, 2007; 644 pages; ISBN13: 978- 0-07-148300-1.

Jones, Capers; "Estimating and Measuring Object-Oriented Software"; American Programmer; 1994.

Jones, Capers; "Why Flawed Software Projects are not Cancelled in Time"; Cutter IT Journal; Vol. 10, No. 12; December 2003; pp. 12-17.

Jones, Capers; "Software Project Management Practices: Failure Versus Success";
Crosstalk, Vol. 19, No. 6; June 2006; pp4-8.

Jones, Capers; Software Assessments, Benchmarks, and Best Practices; Addison Wesley Longman, Boston, MA, 2000;

659 pages.

Jones, Capers; Software Quality – Analysis and Guidelines for Success; International Thomson Computer Press, Boston, MA; ISBN 1-85032-876-6; 1997; 492 pages.

Jones, Capers; Patterns of Software System Failure and Success; International Thomson Computer Press, Boston, MA; December 1995; 250 pages; ISBN 1-850-32804-8; 292 pages.

Jones, Capers; Assessment and Control of Software Risks; Prentice Hall, 1994; ISBN 0-13-741406-4; 711 pages.

Jones, Capers; Conflict and Litigation Between Software Clients and Developers; Version 10; Software Productivity Research, Burlington, MA; June 2009; 54 pages.

Jones, Capers; A New Business Model for Function Points; Version 1.0; Capers Jones & Associates LLC; Narragansett, RI; June 2009; 40 pages.

Jones, Capers; A Short History of Lines-of-Code Metrics; Capers Jones & Associates LLC; Narragansett, RI; September 2009; 20 pages.

Jones, Capers; A Short History of Cost per Defect Metrics; Capers Jones & Associates LLC, Narragansett, RI; October 2009; 22 pages.

Jones, Capers: "Sizing Up Software;" Scientific American Magazine, Volume 279, No. 6, December 1998; pages 104-111. Kan, Stephen H.; Metrics and Models in Software Quality Engineering, 2nd edition; Addison Wesley Longman, Boston, MA; ISBN 0-201-72915-6; 2003; 528 pages.

McConnell; Software Estimating: Demystifying the Black Art; Microsoft Press, Redmond, WA; 2006.

Laird, Linda M and Brennan, Carol M; Software Measurement and Estimation: A Practical Approach; John Wiley & Sons, Hoboken, NJ; 2006; ISBN 0-471-67622-5; 255 pages.

Park, Robert E. et al; Software Cost and Schedule Estimating - A Process Improvement Initiative; Technical Report CMU/SEI 94-SR-03; Software Engineering Institute, Pittsburgh, PA; May 1994.

Park, Robert E. et al; Checklists and Criteria for Evaluating the Costs and Schedule Estimating Capabilities of Software Organizations; Technical Report CMU/SEI 95-SR-005; Software Engineering Institute, Pittsburgh, PA; January 1995.

Pressman, Roger; Software Engineering – A Practitioner's Approach; McGraw Hill, NY; 6th edition, 2005; ISBN 0-07-285318-2.

Radice, Ronald A.; High Quality Low Cost Software Inspections; Paradoxicon Publishing Andover, MA; ISBN 0-9645913-1-6; 2002; 479 pages.

Royce, Walker; Software Project Management: A Unified Framework; Addison Wesley, Reading, MA; 1998.

Roetzheim, William H. and Beasley, Reyna A.; Best Practices in Software Cost and Schedule Estimation; Prentice Hall PTR, Saddle River, NJ; 1998.

Strassmann, Paul; Information Productivity; Information Economics Press, Stamford, Ct; 1999.

Strassmann, Paul; Information Payoff; Information Economics Press, Stamford, Ct; 1985.

Strassmann, Paul; Governance of Information Management: The Concept of an Information Constitution; 2nd edition; (eBook); Information Economics Press, Stamford, Ct; 2004.

Strassmann, Paul; The Squandered Computer; Information Economics Press, Stamford, CT; 1997.

Stukes, Sherry, Deshoretz, Jason, Apgar, Henry and Macias, Ilona; Air Force Cost Analysis Agency Software Estimating Model Analysis ; TR-9545/008-2; Contract F04701-95-D-0003, Task 008; Management Consulting & Research, Inc.; Thousand Oaks, CA 91362; September 30 1996.

Symons, Charles R.: Software Sizing and Estimating—Mk II FPA (Function Point Analysis), John Wiley & Sons, Chichester, U.K., ISBN 0-471-92985-9, 1991.

Wellman, Frank, Software Costing: An Objective Approach to Estimating and Controlling the Cost of Computer Software, Prentice Hall, Englewood Cliffs, NJ, ISBN 0-138184364, 1992.

Whitehead, Richard; Leading a Development Team; Addison Wesley, Boston, MA; 2001; ISBN 10: 0201675267; 368 pages.

Wiegers, Karl E.; Peer Reviews in Software – A Practical Guide; Addison Wesley Longman, Boston, MA; ISBN 0-201-73485-0; 2002; 232 pages.

Various authors; The IFPUG Guide to IT and Software Measurement, Auerbach Publishers, April 2012; 828 pages.

Systems Engineering News

Condolences for the Family of a Beloved and Respected Leader

The INCOSE Community has lost a beloved and respected leader. On Friday, 20 July 2012, David Wright, President-elect, passed away while vacationing with his family in Spain. David was a long time member and contributed to INCOSE in many ways throughout the years. Persons wishing to share a memory or extend condolences to the family can leave a message on the INCOSE Forum. These messages will be sent to his family. A special memorial is planned for the next issue of INSIGHT.

More information

INCOSE IS12 International Symposium Conducted in Rome

Leaders of the worldwide systems engineering community converged on wonderful Rome, Italy over 9 to 14 July for the 22nd International Symposium of INCOSE, the International Council on Systems Engineering. The event was a huge success, with a smorgasbord of keynote speakers, paper presentations, panels, tutorials, exhibits, and competitions.

Several awards were presented, including:

2012 Best Paper Awards:

– "Next Generation Requirements Engineering", by John Favaro, Silvia Mazzini, Hans-Peter De Koning, Rudolf Schreiner, Xavier Olive
– "Understanding Airlines' Value Perceptions for Value-Based Requirements Engineering Of Commercial Aircraft" by Xinwei Zhang, Guillaume Auriol, Claude Baron, Hakki Eres, Mario Kossmann
– "Entering a Brave New World – Applying Systems Engineering to American Infrastructure Projects - Case Study: The California High-Speed Train Project" by Oliver Hoehne
– "Engineering Clean Energy Systems" by Alex Pavlak

2012 Best Student Paper Award:

– "Platforms for Engineering Experimental Biomedical Systems" by Matthew Mosteller, Mark Austin, Shah-An Yang, Reza Ghodssi

SE Journal Outstanding Paper Award:

– "Managing the Interstitials, A System of Systems Framework Suited for the Ballistic Missile Defense System" by Robert K. Garrett Jr., Steve Anderson, Neil T. Baron, James D. Moreland Jr.

The next INCOSE International Symposium will be in Philadelphia, USA over June 24 - 27, 2013.

More information

PMI, INCOSE and Lean Advancement Initiative (LAI) at MIT Partner to Find Best Practices for Delivering Successful Programs

The Guide to Lean Enablers for Managing Engineering Programs Identifies Best Practices that Effective Teams and Organizations Can Use to Overcome Challenges.

Wasting time and financial resources are often dismissed as the cost of doing business when it comes to engineering programs. To help organizations overcome these challenges, reduce risk, and improve ROI, the Project Management Institute (PMI) and the International Council on Systems Engineering (INCOSE) – which first announced their partnership in September 2011 – teamed with researchers and industry members from the Lean Advancement Initiative (LAI) at the Massachusetts Institute of Technology (MIT) to develop The Guide to Lean Enablers for Managing Engineering Programs.

The guide is an in-depth study that identifies 300 lean enablers, or best practices, that effective teams and organizations can implement to reduce waste and increase project and program success. Access the full report on Dspace@MIT or view the catalog record with citable URI.

"The use of Lean Management principles is particularly potent for organizations, as they heavily emphasize the need for overall integration of the value of delivery across all process and organizational boundaries – including boundaries between program management and systems engineering," said Mark A. Langley, president and CEO of PMI. "While the study focused primarily on engineering programs, the findings can be applied to other programs as well, including IT, business transformation and community- and society-focused initiatives."

"LAI at MIT has focused, for almost two decades, on conducting enterprise-level research and developing unique tools and products to help organizations effectively and efficiently produce stakeholder value," explained LAI Director Prof. Deborah Nightingale. "The Guide to Lean Enablers is a very useful addition to the tools currently available and is the powerful result of a working collaboration between INCOSE, PMI, and LAI."

The study draws from three domains of management wisdom: lean management, systems engineering and program management. The research team defined 160 program management challenges, which were collected into 10 themes. The most common challenges are:

- Reactive execution: Programs are driven by outside influences rather than by strategic goals.

- Lack of accountability: Roles and responsibilities of individuals, teams, project, staff, and organizations are not clearly defined.

- Insufficient competency: The knowledge of individuals, teams, and the organization is inadequate, not transferred sufficiently, or not applied appropriately during the program.

The study also identifies 300 lean enablers that are grouped around key lean principles including:

- Respect for people

- Capturing value as defined by the customer

- Mapping the value stream

- Maintaining flow through value-adding processes

- Letting customers' needs determine value

- Pursuing perfection in all processes.

To ensure the applicability of the lean enablers to actual programs, two PMI Project of the Year Award finalists – the Prairie Waters Public Works Project in Aurora, Colorado, and the Dallas Cowboys Stadium in Dallas, Texas, served as case studies. In both cases, the researchers found the programs applied more than 75 percent of the recommended enablers.

"This latest research and careful examination of highly successful programs illustrate how collaboration between program managers and systems engineers, paired with the adoption of lean enablers, contribute enormously to the success of programs," said John A. Thomas, president of INCOSE. "By strategically solving specific challenges, lean thinking removes waste and creates a valuable core competency around delivering value to customers."

For more information, look for #leanenablers on Twitter and visit http://hdl.handle.net/1721.1/70495.

About Project Mangement Instutute (PMI)

PMI is the world's largest project management member association, representing more than 600,000 practitioners in more than 185 countries. As a global thought leader and knowledge resource, PMI advances the profession through its global standards and credentials, collaborative chapters and virtual communities and academic research. When organizations invest in project management supported by PMI, executives have confidence that their important initiatives will deliver expected results, greater business value and a competitive advantage. Visit PMI at www.PMI.org, www.facebook.com/PMInstitute and on Twitter @PMInstitute.

About International Council on Systems Engineering (INCOSE)

INCOSE is a not-for-profit membership organization founded to develop and disseminate the interdisciplinary principles and practices that enable the realization of successful systems. INCOSE's mission is to share, promote and advance the best of systems engineering from across the globe for the benefit of humanity and the planet. INCOSE has grown significantly since its formation in 1990. Today, there are over seven thousand members representing a broad spectrum – from student to senior practitioner, from technical engineer to program and corporate management, from science and engineering to business development. Over 50 chapters have been established worldwide and 70 organizations from industry, academia and government are active members of the Corporate Advisory Board. Additional information is available at http://www.incose.org/.

About the MIT Lean Advancement Initiative (LAI)

The Lean Advancement Initiative (LAI) at MIT, together with its international Educational Network (EdNet), offers organizational members from industry, government, and academia the newest thinking, products, and tools related to enterprise transformation and architecting. LAI enables the focused and accelerated transformation of complex enterprises through collaborative stakeholder engagement in developing and institutionalizing principles, processes, behaviors, and tools for enterprise excellence. Please visit lean.mit.edu for more information.

Worcester Polytechnic Institute Partners with INCOSE

Worcester Polytechnic Institute (WPI) (USA) became a key strategic partner with the International Council on Systems Engineering (INCOSE) at the 22nd Annual INCOSE International Symposium in Rome, July 9 – 12, 2012. Through an agreement, WPI will be assisting its students and alumni of the MS and graduate certificate in systems engineering with obtaining either their Associate Systems Engineering Professional ASEP certification or their Certified Systems Engineering Professional CSEP certification. WPI's assistance will consist of helping with the application process, reviewing documents, and providing a review course on systems engineering that students can take before they sit for the INCOSE exam which is required as part of the certification process.

More information

Systems Thinking in Education

The Waters Foundation recently held the second annual Camp Snowball in Tucson, Arizona, USA. The camp is designed to build capacity in teachers and students for systems thinking and sustainability education locally, nationally and throughout the world.

Marv Adams, Chief Operating Officer at TDAmeritrade, and Darcy Winslow, former Global General Manager and VP, Women's Footwear, Apparel and Equipment, Nike joined Peter Senge in speaking and a public forum discussing connections between education and business as teachers help foster growth of 21st Century future ready graduates.

"The purpose of Camp Snowball is to provide opportunities for communities and schools to learn how to enable their students to think deeply and critically and to achieve academically in order to become responsible, thoughtful citizens of the interdependent world that they will inherit." Next year, the camp will be held at Wake Forest University and cosponsored by Winston Salem Forsyth County Schools.

More information

Guide Available for the Application of SE in Construction

A Guide available from INCOSE covers the application of Systems Engineering (SE) practices to Large Infrastructure Projects (LIPs). Such projects include the construction of infrastructure (e.g., highways, railways, electricity generation and distribution, water collection, storage, and distribution, and waste water collection and transfer), and the construction of major industrial plants, such as oil & gas platforms, refineries, mines, smelters, water and wastewater treatment and steel works. These projects may include a design stage, if this has not been completed prior to going to construction, but the emphasis of this Guide is on how to use SE practices to better perform the construction stage of a project. The focus is on the realization of the designed (or engineered) solution during construction and the transition into service of the resulting built product, and as a consequence, the application of SE practices is concentrated more on the construction process than on the design of the product or on the continuing operation and maintenance stage.

Members may download the Guide from the product area of INCOSE Connect. There is no charge to members for softcopy.

Swiss Society of Systems Engineering

The Swiss Society of Systems Engineering (SSSE) formed in 2011, as a Chapter of the International Council on Systems Engineering (INCOSE), and is active with many interesting events, including a Requirements Day, evening technical events, and working group leadership. Anyone interested in the Society's quarterly newsletter may subscribe at the Society's website. The Society's website supports French, and Italian, with German on the way, we understand.

More information: http://incose.ch/?language=en

Survey/Reports on Systems Engineering and Embedded Software Best Practices

The Aberdeen Group (http://www.aberdeen.com/) invites engineers to share their experiences and opinions by participating in a survey to help define what Aberdeen Group describes as "Best-in-Class systems engineering practices".

The survey explores topics such as where to start when improving systems engineering processes, requirements management, and change management. Two reports will be coming from this survey, one on systems engineering and one of embedded software. Those who complete the survey will automatically be sent copies of both reports as they become available.

Take the survey

Robert's Reflections

The time has come to address the rampant confusion over the term "System of Systems Engineering (SOSE)". If ever a term needed to be replaced, this is it. Literally, a cell-phone is a system of systems (SoS). And if that is the case, System of Systems Engineering is synonymous with Systems Engineering.

However, there is a class of system that, in its engineering, poses challenges that are much more difficult than those of systems engineering in general, specifically systems with some combination of:

- Operational independence of the subsystems - each of the individual systems within a SoS has a "life of its own" and can function acceptably and provide useful service to human users.

- Managerial independence of the subsystems - the individual systems within a SoS are under different authorities.

- Independent development - the different subsystems within the SoS are developed and upgraded on uncoordinated schedules.

Henceforth, I will be using the terms "System of Autonomously Managed Systems (SOAMS)" and "System of Autonomously Managed Systems Engineering (SOAMSE)" in relation to the above, unless someone comes up with better terms. Not exactly warm cuddly acronyms, I know. But the status quo of use of a term which inevitably leads to misunderstanding is much worse. We learned that lesson with Work Breakdown Structure (WBS).

Robert Halligan FIE Aust

Featured Society

The OR Society

The OR Society is the trading name of the Operational Research Society, which is registered in England and operates in the United Kingdom. Member-based, the Society's main functions are to enable members to acquire, share and exchange knowledge about operational research. It does these by providing information and technical ideas through its website, through its many publications (all available online), through a range of conferences and courses, and via branches

throughout the U.K.

Operational research (O.R.) is the discipline of applying advanced analytical methods to help make better decisions. By using techniques such as problem structuring methods (sometimes known as 'Soft O.R.') and mathematical modeling to analyze complex situations, operational research gives engineers and managers the power to make more effective decisions and build more productive systems based on:

- More complete data

- Consideration of all available options

- Careful predictions of outcomes and estimates of risk

- The latest decision tools and techniques.

Many of the problems OR tackles are messy and complex, often entailing considerable uncertainty.

The OR Society has five grades on membership: Student Member, plus four accreditation-based member grades: Candidate Associate of The OR Society (CandORS), Associate of The OR Society (AORS), Associate Fellow of The OR Society (AFORS), and Fellow of The OR Society (FORS).

Special Interest Groups as follows provide much of the professional focus:

- Community OR Network

- Complex Systems Discussion Group

- Criminal Justice

- Decision Analysis

- Defense

- Health & Social Services

- Independent Consultants' Network

- Information Systems

- Local Search

- Mathematical Programming

- OR and Strategy

- OR for Developing Countries

- OR in the Third Sector

- Problem Structuring Methods

- Simulation

- SD+.

These SIGs run a number of open emailing lists focusing on aspects of OR.

Major research activity supported by The OR Society: The LANCS Initiative

The LANCS Initiative is built on collaboration between four U.K. universities: Lancaster, Nottingham, Cardiff and Southampton. The U.K.'s Engineering and Physical Sciences Research Council (EPSRC) granted £5.4 million to support the development of research. See www.lancs-initiative.ac.uk.

Major research activity supported by The OR Society: NATCOR

The National Taught Course Center in Operational Research (NATCOR) is a collaboration between ten U.K. universities, to develop and deliver taught courses in Operational Research (OR) to PhD students. NATCOR is funded by EPSRC for the first five years of its life (October 2006 to September 2011) and is supported by the OR Society, which is a collaborating partner. See www.natcor.ac.uk/node/1

More information

INCOSE Operations

INCOSE Student Division

The International Council on Systems Engineering (INCOSE) revitalized the Student Division (SD) program in 2008 in an effort to increase the number of INCOSE student members. These students provide not only new concepts and research concerning systems engineering, but also represent the growth and future of INCOSE. The SD program is predicated on the value proposition of each of the Four-Way Benefit Model stakeholders as identified in Figure 1. The stakeholders include INCOSE, universities, enterprises, and students themselves.

Figure 1: Four-Way Benefit Model

Students are individual student members who are enrolled in academic programs and choose to participate in INCOSE, and may also be student members participating in student divisions at universities. Although there are several opportunities provided to students within the INCOSE framework; the focus of this article is the Student Division program, its alignments, and its initiatives.

The Student Division program is structured around the educational culture within the United States. Figure 2 illustrates the Americas model of the Student Division. This model also highlights the mentoring framework among the Four-Way Benefit Model stakeholders at the local level, the connection to the INCOSE Academic Forum, and the INCOSE structure at the sector and the CAB.
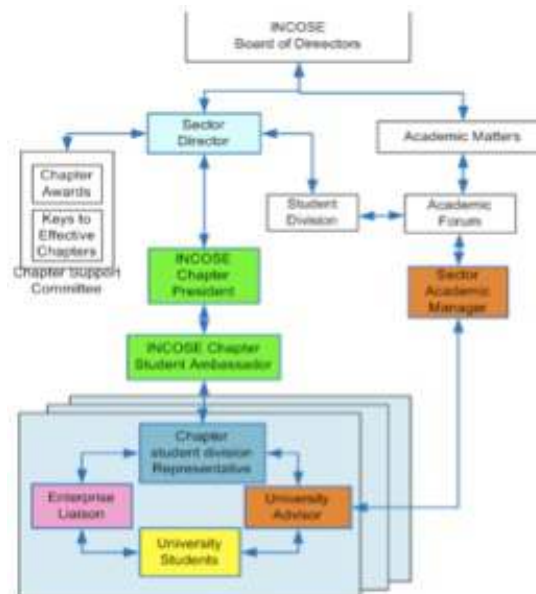

Figure 2: The Americas Model of the INCOSE Student Division

In 2011, INCOSE recognized the cultural differences among its non-American members and countries, resulting in the formation of three sectors; I) Americas: II) AERIT (Africa, Europe, Russia, Israel, and Turkey: and III) Asia-Oceania. This sectoring also provides the opportunity for development of the SD structure to align with the regional and educational

practices within each country. The student division models for each sector are in the process of development and alignment to the overall Student Division program.

Each 'student division' is an engineering 'club' at a university offering an accredited engineering program, and sponsored by a chartered INCOSE chapter. The INCOSE membership brings the structure and opportunities of the not-for-profit organization advocating systems engineering along with members who are practicing professional engineers and members of academia. This interwoven association forms the heart of the Four-Way Benefit Model with the students being the central focus of the Student Division program.

Each student division has the challenge and opportunity to identify its unique value propositions and success metrics during the formulation process, which includes the ratification of the student division by-laws. These value propositions are the significant elements of what is important to each stakeholder in supporting the development and sustainment of a student division. The stakeholders convert these value propositions into annual success metrics to measure their success, and advocate the value of systems engineering and the sustainment of the student division by the associated stakeholders.

The Student Division program has created:

- Presentation of the Student Division Program at several universities and conferences.
- An Americas' model to standardize the structure for each student division.
- A student division by-laws template.
- An INCOSE website for student divisions.
- Acknowledgment of students at the INCOSE International Symposium (IS) and International Workshop (IW).
- Registration fees for students who win research poster competition.

Figure 3 illustrates the combined number of student members in INCOSE through 2012 (red). The 2012 data indicate the number of new students for the first six months. The Student Division growth is approximately 15x between 2008 and 2012.
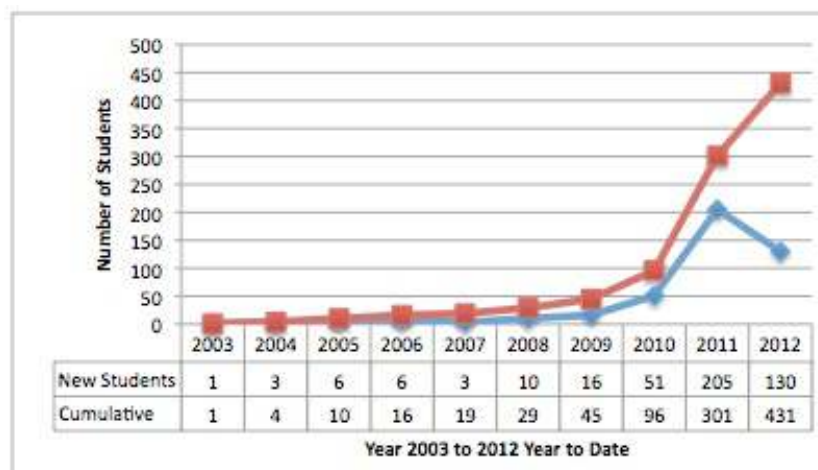

	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012
New Students	1	3	6	6	3	10	16	51	205	130
Cumulative	1	4	10	16	19	29	45	96	301	431

Year 2003 to 2012 Year to Date

Figure 3: INCOSE Student Division Members

The process of growing brings with it the knowledge for improvements. Opportunities for improvements are presented at the International Symposium to the Academic Council and at the Student Division Panel. The theme for the 2012 Student Division panel is 'How Can We Continue to Grow the INCOSE University Student Divisions (SD) Program?' Some of the topics planned for discussion include:

- Reduction of student load requirement from ¾ to ½ time.
- Development of an on-line student division.
- Initiation of Student Division Webinars.
- Updating the INCOSE website (tools/templates/register)
- Addition of the Omega Alpha Assoc Honor Society
- LEfSE competition
- Open to students of all disciplines, not only SE

- Improvements in the POC for Sectors/Chapters

- Alignment of SD Advisors with the Academic Forum.

- Tracking students who convert to full membership.

More information

Systems Engineering Tools News

Integrating Spec Data in Models

For infrastructure construction teams, considering the impact of contractual requirements earlier in the design and construction process allows projects to stay on budget and reduce change orders down the line—which is why integrating specification information with modeling tools is becoming more common as of late.

One such example comes from TEEC (The Engineering Essentials Co.), www.teecspecs.com, Concordville, PA. The company provides SpecWave, which allows for creation of specifications including warranties, material types, and other data using classification codes during design, procurement, verification, documentation, inspection, installation, and operations. The specification system also integrates with design models, allowing for the data to be available for reference, design review, transmittals, and handover, among others.

More Information

Open Source SysML Editor

An open source SySML editor may be downloaded at: http://percivalkirk.typepad.com/blog/2012/07/download-open-source-sysml-editor.html

---

SysML Designer (Indigo version) 1.0.2

This designer, from Obeo Network, provides a set of common diagrams to work with SysML models. The intent is to provide an easy way to make the transition from SysML to domain specific modeling. This designer is free (open-source with EPL license).

More information

Free/Open Source Software Simulation Tools

The EUROSIS website has a very extensive list of free/open source software simulation and gaming development packages (or ones thought by EUROSYS to be "reasonably priced"), complete with tool overviews and links. Included are systems engineering software tools for applications such as 3D Modeling, Agent Based Simulation, Discrete Event Simulation, Finite Element Modeling, Fluid Dynamic Modeling, General Simulation Platform, Monte Carlo Simulation, Object-Oriented Simulation, and Petri Nets.
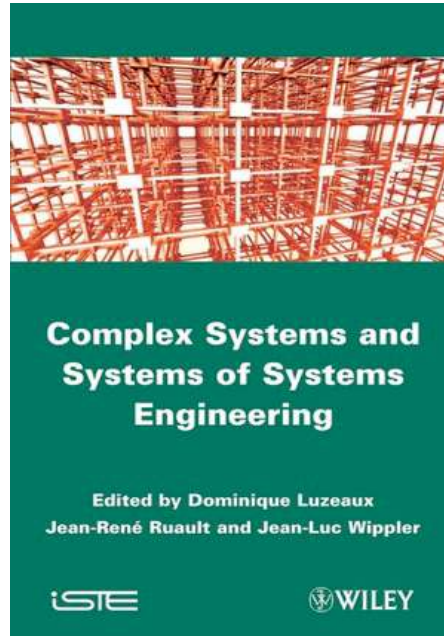
More information

Systems Engineering Books, Reports, Articles and Papers

The Guide to Lean Enablers for Managing Engineering Programs

Oehmen, Josef; Oppenheim, Bohdan W.; Secor, Deborah; Norman, Eric; Rebentisch, Eric; Sopko, Joseph A.; Steuber, Marc; Dove, Rick; Moghaddam, Kambiz; McNeal, Steve; Bowie, Mark; Ben-Daya, Mohamed; Altman, Wolf; Driessnack, John (Joint MIT-PMI-INCOSE Community of Practice on Lean in Program Management, 2012-05)

Large Scale Complex Systems and Systems of Systems Engineering Case Studies

Dominique Luzeaux

ISBN: 978-1-84821-253-4

Format: Hardcover

Publication Date: December 2011, Wiley-ISTE

Book Description (From John Wiley web site):

With the growing maturity of information and communication technologies, systems have been interconnected within growing networks, yielding new services through a combination of the system functionalities. This leads to an increasing complexity that has to be managed in order to take advantage of these system integrations. This book provides key answers as to how such systems of systems can be engineered and how their complexity can be mastered.

After reviewing some definitions on systems of systems engineering, the book focuses on concrete applications and offers a survey of the activities and techniques that allow engineering of complex systems and systems of systems. Case studies, ranging from emergency situations such as Hurricane Katrina and its crisis management or a generic scenario of a major traffic accident and its emergency response, to the establishment of a scientific basis in the Antarctic region illustrate key factors of success and traps to avoid in order to cope with such situations.

"The five parts of this book will provide the reader with a detailed description of all the elements that make up a RFID system today, including hot topics such as the privacy concerns, and the Internet of Things." (Radio-Electronics.com, 1 December 2011)

[More Information](#)

Strategies to the Prediction, Mitigation and Management of Product Obsolescence

Wiley Series in Systems Engineering and Management

Bjoern Bartels, Ulrich Ermel, Peter Sandborn, and Michael G. Pecht
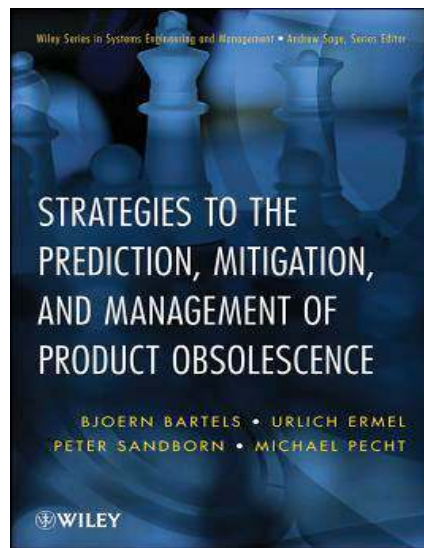
Publisher: John Wiley and Sons Ltd

Format: Hardcover

Publication Date: June 2012

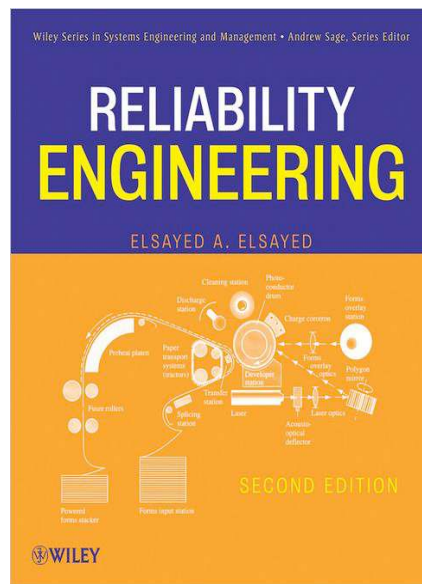**Book Description (From John Wiley web site):**

Supply chains for electronic products are primarily driven by consumer electronics. Every year new mobile phones, computers and gaming consoles are introduced, driving the continued applicability of Moore's law. The semiconductor manufacturing industry is highly dynamic and releases new, better and cheaper products day by day. But what happens to long–field life products like airplanes or ships, which need the same components for decades? How do electronic and also non–electronic systems that need to be manufactured and supported of decades manage to continue operation using parts that were available for a few years at most? This book attempts to answer these questions.

This is the only book on the market that covers obsolescence forecasting methodologies, including forecasting tactics for hardware and software that enable cost–effective proactive product life–cycle management. This book describes how to implement a comprehensive obsolescence management system within diverse companies. Strategies to the Prediction, Mitigation and Management of Product Obsolescence is a must–have work for all professionals in product/project management, sustainment engineering and purchasing.

More Information

Reliability Engineering, 2nd Edition

Elsayed A. Elsayed

ISBN: 978-1-1181-3719-2

Format: Hardcover

Publication Date: June 2012

Book Description (From John Wiley web site):

Reliability is one of the most important quality characteristics of components, products, and large and complex systems—but it takes a significant amount of time and resources to bring reliability to fruition. Thoroughly classroom- and industry-tested, this book helps ensure that engineers see reliability success with every product they design, test, and manufacture.

Divided into three parts, Reliability Engineering, Second Edition handily describes the theories and their practical uses while presenting readers with real-world examples and problems to solve. Part I focuses on system reliability estimation for time independent and failure dependent models, helping engineers create a reliable design. Part II aids the reader in assembling necessary components and configuring them to achieve desired reliability objectives, conducting reliability tests on components, and using field data from similar components. Part III follows what happens once a product is produced and sold, how the manufacturer must ensure its reliability objectives by providing preventive and scheduled maintenance and warranty policies.

This Second Edition includes in-depth and enhanced chapter coverage of:

- Reliability and Hazard Functions

- System Reliability Evaluation

- Time- and Failure-Dependent Reliability

- Estimation Methods of the Parameters of Failure-Time Distributions

- Parametric Reliability Models

- Models for Accelerated Life Testing

- Renewal Processes and Expected Number of Failures

- Preventive Maintenance and Inspection

- Warranty Models

- Case Studies

A comprehensive reference for practitioners and professionals in quality and reliability engineering, Reliability Engineering can also be used for senior undergraduate or graduate courses in industrial and systems, mechanical, and electrical engineering programs.

[More Information](#)

Visual Models for Software Requirements

Joy Beatty and Anthony Chen
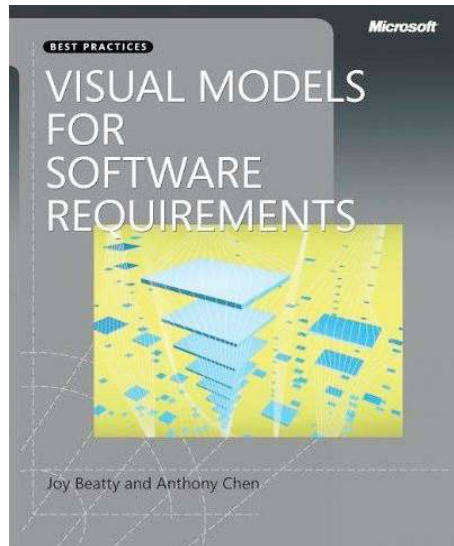
Published by: Microsoft Press (Best Practices Series)

Publication Date: July 23, 2012

Print ISBN: 978-0-7356-6772-3, ISBN 10:0-7356-6772-1

EBook ISBN: 978-0-7356-6775-4, ISBN 10:0-7356-6775-6

Formats: Paperback, E-book, Safari Books Online

Book Description (From Amazon):

Apply best practices for capturing, analyzing, and implementing software requirements through visual models—and deliver better results for your business. The authors—experts in eliciting and visualizing requirements—walk you through a simple but comprehensive language of visual models that has been used on hundreds of real-world, large-scale projects. Build your fluency with core concepts—and gain essential, scenario-based context and implementation advice—as you progress through each chapter.

- Transcend the limitations of text-based requirements data using visual models that more rigorously identify, capture, and validate requirements

- Get real-world guidance on best ways to use visual models—how and when, and ways to combine them for best project outcomes

- Practice the book's concepts as you work through chapters

- Change your focus from writing a good requirement to ensuring a complete system.

More information

Article
Requirements Traceability: The black art of design and development

Shan Bhattacharya

The requirements-driven development mantra and all that it encompasses has been documented and discussed thoroughly for almost two decades in the pursuit of building better and more reliable applications. This mantra has been the undertone of software processes, certifying authorities, and industry standards focused on realizing requirements-driven development in its utopian form.

Despite these efforts, the majority of defects in embedded software space are still requirements related. One of the prime contributors to this problem is the continuous flux in requirements introduced by the changes in product scope. To shield

themes from the added risks contributed by this flux, organizations need to learn to manage change in the requirements and subsequent implementation phases of the product life cycle.

Good requirements management practices usually involve a well thought out breakdown or decomposition of requirements from system level on down. If this decomposition is managed well, its artifacts are configuration controlled, and the various engineering disciplines work well together, flux can then be handled throughout the development phases of the lifecycle.

More Information

Call for Papers - Scientific Research and Impact

Introducing ''Scientific Research and Impact'' (SRI), a multidisciplinary, peer-reviewed journal, published monthly by Science Park Journals. SRI is dedicated to increasing the depth of research across all areas of this subject. SRI welcomes the submission of manuscripts that meet the general criteria of significance and scientific excellence in this subject area, and will publish:

- Original articles in basic and applied research

- Case studies

- Critical reviews, surveys, opinions, commentaries and essays

SRI invites you to submit your manuscript(s). A decision on submitted manuscript(s) will be made within four weeks of submission. Following acceptance, a paper normally will be published in the next issue. SRI is an Open Access Journal. One key request of researchers across the world is unrestricted access to research publications. Open access gives a worldwide audience larger than that of any subscription-based journal and thus increases the visibility and impact of published works. It also enhances indexing, retrieval power, and eliminates the need for permissions to reproduce and distribute content. SRI is fully committed to the Open Access Initiative and will provide free access to all articles as soon as they are published.

This information was made available by Dr. Richard Griggs, Editor, Scientific Research and Impact (SRI) - Email: sri (at) scienceparkjournals.org;
Website: http://scienceparkjournals.org/sri/

More Information

July 2012 Volume 15 Issue 2 of INCOSE INSIGHT

The July 2012 Volume 15 Issue 2 of INSIGHT is ready to view or download on INCOSE Connect at the INSIGHT Library along with all past issues of INSIGHT.

Special Feature: Systems of the Third Kind

"Systems of the third kind include variations of currently popular labels such as chaotic, complex-adaptive, autonomous, resilient, sustainable, agile, and human activity. They move among us already: cars that drive themselves in urban environments, helicopters that land autonomously, lethal weapons that decide when and where to shoot, unmanned aircraft in the national airspace. Some work alone; others are being taught to work in packs and swarms. Emergent behavior is expected, with consequences, and with virtually no systems engineering guidance." The theme editors are Rick Dove, Jack Ring, and Thomas Tenorio.

More information

Conferences and Meetings

KSE 2012. The 4th International Conference on Knowledge 〜〜 **NEW**
August 17 - 19, 2012, Danang, Vietnam
More information

The 7th International Conference on Availability, Reliability and Security (ARES 2012)
August 20 - 24, 2012, Prague, Czech Republic
More information

EmpiRE 2012 : Workshop on Empirical Requirements Engineering
August 25, 2012, Chicago, USA
More information

Workshop on Model-Driven Engineering for Networked Ambient System (MDE4NAS)
August 27 - 29, 2012, Niagara Falls, Canada
More information

9th INCOSE SA Conference: Systems Engineering - The Jewel in the Crown
August 29 – 27, 2012, Pretoria, South Africa
More information

18th International Symposium on Formal Methods
August 27 - 31, 2012, CNAM, Paris, France
More information

Twelfth International Conference on Parallel Problem Solving from Nature (PPSN XII) NEW
September 1 – 5, 2012, Taormina, Italy
More information

Summer School 2012: Verification Technology, Systems & Applications NEW
September 3 – 7, 2012, Saarbrücken, Germany
More information

OR54 Annual Conference of the OR Society NEW
September 4 – 6, 2012, Scotland, United Kingdom
More information

International Annual Conference of the German OR Society 2012
September 4 – 7, 2012, Germany
More information

The Dutch Model Checking Day NEW
September 5, 2012, Amsterdam, Netherlands
More information

3rd IMA Conference on Numerical Linear Algebra and Optimisation
September 10 - 12, 2012, Birmingham, UK
More information

3rd International Summer School on Domain Specific Modeling - Theory and Practice NEW
September 10 - 14, 2012, Lisbon
More information

Sixth IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO 2012)
September 10 - 14, 2012, Lyon, France
More information

AIAA Complex Aerospace Systems Exchange NEW
11 - 13 September 2012, Pasadena, California
More information

International Workshop on Enterprise Integration, Interoperability and Networking (EI2N'2012)
September 12 - 13, 2012, Rome, Italy
More information

ORSSA 2012 41st Annual Conference of the Operations Research Society of South Africa
September 16 - 19, 2012, Johannesburg, South Africa
More information

Team Software Process (TSP) Symposium 2012

September 17 – 20, 2012, St. Petersburg, FL, USA
More information

Matheuristics'2012 – Fourth International Workshop on Model-Based Meta-Hueristics
September 17 - 20, 2012, Angra dos Reis, Brazil
More information

10th International Conference on Formal Modeling and Analysis of Timed Systems (FORMATS 2012)
September 18 - 20, 2012, London, United Kingdom
More information

12th International Workshop on Automated Verification of Critical Systems (AVoCS 2012)
September 18 - 20, 2012, Otto-Friedrich University in Bamberg, Germany
More information

2012 Interdisciplinary Symposium on Complex Systems   NEW
September 19 - 24, 2012, Kos island, Greece
More Information

Risk Engineering Society Conference (RISK 2012)
September 20 – 22, 2012, Sydney, Australia
More information

Verifikation a validierung Herausforderungen Bei Kurzen Entwicklungszeiten (V&V Forum)
September 21, 2012, Frankfurt, Germany
More information

RePa 2012 : Second International Workshop on Requirements Patterns
September 24, 2012, Chicago, USA
More information

20th IEEE International Requirements Engineering Conference
September 24 - 28, 2012, Chicago, IL, USA
More information

CLAIO XVI – 16th Conference of the Latin-American Association of Operations Research   NEW
September 24 - 28, 2012, Rio de Janeiro, Brazil
More information

20th International Requirements Engineering Conference (RE 2012)
September 24 - 28, 2012, Chicago, Illinois, USA
More information

SAFECOMP 2012 - 31st International Conference on Computer Safety, Reliability and Security
September 25 – 28, 2012, Magdeburg, Germany
More information

2nd Requirements Symposium
September 27, 2012, Berlin

MODELS 2012, ACM/IEEE 15th International Conference on Model-Driven Engineering Language & Systems - Call for Papers - Deadline 19 March 2012
September 30 - October 5, 2012 – Innsbruck, Austria
More Information

SAM (System Analysis and Modelling) Workshop 2012   NEW
October 1 – 2, 2012, Innsbruck, Austria
More information

6th INCOSE Annual Great Lakes Regional Conference 2012
October 12 – 13, 2012, Schaumburg, Illinois, U.S.A
More information

World Engineering Education Forum (WEEF12)   NEW
October 15 - 18, 2012, Buenos Aires, Argentina
More information

19th Working Conference on Reverse Engineering
October 15 - 18, 2012, Kingston, Ontario, Canada
More information

ASME 2012 Dynamic Systems and Control Conference (DSCC2012)
October 16 - 20, 2012, Ft. Lauderdale FL , USA
More information

2012 International Annual Conference of the American Society for Engineering Management, Agile Management
October 17 - 20, 2012, Virginia Beach, VA, USA
More information

ESM'2012 26th Annual European Simulation and Modelling Conference
October 22 - 24, 2012, Essen, Germany
More information

Human Factors and Ergonomics Society HFES 2012 Annual Meeting
October 22 - 26, 2012, Boston, MA, USA
More information

ICSSEA 2012 - International Conference on Software & Systems Engineering and their Applications
October 23 - 25, 2012, Paris, France
More information

2012 Canadian Society of Value Analysis (CSVA) Annual Conference
October 24 - 25, 2012, Calgary, Alberta
More information

The World Congress on Engineering and Computer Science 2012
October 24 - 26, 2012, San Francisco, USA

8º Congresso Brasileiro de Sistemas
October 25 - 26, 2012, campus da PUC Minas em Poços de Caldas, MG, Brasil
More information

The 19th International Conference on Industrial Engineering and Engineering Management
October 27 - 29, 2012, ChangSha, China
More information

Building Business Capabilities (BBC) 2012 ꙮꙮꙮ **NEW**
October 28 - November 2, 2012, Fort Lauderdale, FL, USA
More information

International Conference on Complex Systems (ICCS'12)
November 5 - 6, 2012, Agadir, Morocco
More information

12th Annual CMMI Technology Conference and User Group
November 5 – 8, 2012, Denver, USA
More information

INCOSE UK Annual Systems Engineering Conference 2012
November 7 - 8, 2012, London, UK
More information

Systems Engineering Day 2012 (TdSE 2012)
November 7 - 9, 2012, Paderborn, Heinz Nixdorf Museums Forum, Germany.
More information

14th International Conference on Formal Engineering Methods (ICFEM 2012)
November 12 - 16, 2012, Kyoto Research Park, Kyoto, Japan
More information

Complex Adaptive Systems Conference
November 14 – 16, 2012, Washington D.C., Dulles, USA

More information

PapersModel Based Systems Engineering 2012 Symposium
November 27 – 28, 2012, Edinburgh, South Australia

Operations Research Society of New Zealand (ORSNZ) Conference Includes a Systems stream: 'Systems Thinking, Systems Modelling and Systems Practice' NEW
December, 10 - 11, 2012, Wellington, New Zealand
More information

IEEE 2012 International Conference on Industrial Engineering and Engineering Management NEW
December 10 - 13, 2012, Hong Kong
More information

MESM'2012 The 13th annual International Middle Eastern Simulation and Modelling Conference NEW
December 10 - 12, 2012, Muscat, Oman
More information

3rd International Conference on Complex Systems Design & Management (CSD&M 2012) NEW
December 12 - 14, 2012, Cité Internationale Universitaire, Paris (France)
More information

INCOSE International Workshop IW2013
January 26 - 29, 2013, Jacksonville, Florida USA
More information

Conference Digital Enterprise Design & Management (DED&M 2013)
February 11 - 12, 2013, Paris, France
More information

International Symposium on Engineering Secure Software and Systems (ESSoS)
February 27 – March 3, 2013, Paris, France
More information

INCOSE IL 2013
March 5 – 6, 2013, Daniel Hotel Herzlia
More information

ASTEC 2013, Asian Simulation Technology Conference NEW
March 7 - 9, 2013, Shanghai, China
More information

The Requirements Engineering Track - 6th Edition at The 28th Annual ACM Symposium on Applied Computing (SAC 2013)
March 18 - 22, 2013, Coimbra, Portugal
More information

11th Annual Conference on Systems Engineering Research (CSER 2013)
March 19 – 22, 2013, Atlanta, Georgia, USA
More information

EMO 2013 - the 7th International Conference on Evolutionary Multi-Criterion Optimization
March 19 - 22, 2013, Sheffield, United Kingdom
More information

YoungOR 18
April 9 – 11, 2013, University of Exeter, Exeter, Unoted Kingdom
More information

ECEC'2013, 20th European Concurrent Engineering Conference NEW
April 15 - 17, 2013, Lincoln, United Kingdom
More information

SysCon 2013 - IEEE Systems Conference
April 15 - 18, 2013, Orlando, FL, USA

SETE 2013 (Systems Engineering - Test and Evaluation)
April 29 – May 1, 2013, Canberra, ACT, Australia

ISC'2013, 11th Annual Industrial Simulation Conference NEW
May 22 - 24, 2013, Ghent, Belgium

KIM2013 Knowledge and Information Management Conference
June, 4 - 5, 2013, Meriden, United Kingdom

12th International Symposium of the Analytic Hierarchy Process/Analytic Network Process (ISAHP 2013)
June 23 – 26, 2013, Kuala Lumpur, Malaysia

IS 2013 (International Council on Systems Engineering International Symposium)
June 24 – 27, 2013, Philadelphia, Pennyslvania USA

ISSS 2013: The 57th World Conference of the International Society for the Systems Sciences NEW
July 14 – 19, 2013, Hai Phong City, Viet Nam

The Sixteenth SDL FORUM - SDL2013 NEW
Date and location to be determined, 2013

ASME 2013 International Design Engineering Technical Conference and Computers and Information in Engineering Conference (IDETC/CIE2013)
August 4 - 7, 2013, location TBA, USA

OR55 Annual Conference of the OR Society NEW
September 3 - 5, 2013, Exeter University, Exeter, United Kingdom

International Conference on Operations Research NEW
September 3 - 6, 2013, Rotterdam, The Netherlands

APCOSE 2013 (Asia-Pacific Council on Systems Engineering)
September 9 - 11, 2013, Keio University in Japan

SIMEX'2013 NEW
September 10 - 13, 2013, Brussels, Belgium

27th European Simulation and Modelling Conference - ESM'2013
October 2013, Lancaster, UK

ASTEC 2014, Asian Simulation Technology Conference NEW
March 2014, Digipen Institute of Technology, Singapore

ISC'2014 - 12th Annual Industrial Simulation Conference NEW
June 11 - 13, 2014, Skövde, Sweden

19th World Congress of the International Federation of Automatic Control (IFAC 2014)
August 24 - 29, 2014, Cape Town, South Africa

SIMEX'2014 NEW
September 2014, Brussels, Belgium

INCOSE Europe, Middle East & Africa (EMEA) Sector: 1st EMEA Systems Engineering Conference 2014 (formerly EuSEC)
October 2014, Cape Town, South Africa

ASTEC'2015, Asian Simulation Technology Conference ᴧᴧ NEW
March 2015, Japan

ISC'2015 13th Annual Industrial Simulation Conference ᴧᴧ NEW
June 2015, St.Petersburg, Russia

SIMEX'2015 ᴧᴧ NEW
September 2015, Brussels, Belgium

ISC'2016 14th Annual Industrial Simulation Conference ᴧᴧ NEW
June 2016, Bucharest, Romania, (To be confirmed)

Education and Academia

U.S. News and World Report Rates the Tagliatela College of Engineering

U.S. News and World Report rated the Tagliatela College of Engineering, New Haven, Connecticut (USA) in the top tier of undergraduate engineering programs nationwide in their 2012 "Best Colleges" edition.

More Information

Transnet Centre of Systems Engineering (TCSE) Positions, South Africa

Applications are invited for the positions of SENIOR MANAGER PROGRAMMES (SMP) and the CHIEF TECHNOLOGY / SYSTEMS OFFICER (CTO) in the Transnet Centre of Systems Engineering (TCSE) hosted by the Faculty of Engineering and the Built Environment (FEBE) within the University of the Witwatersrand, Johannesburg, South Africa. These two positions will have strong links with the School of Mechanical, Industrial and Aeronautical Engineering and other Schools within the Faculty as well as with Transnet's Operating Divisions.

More information

Some Systems Engineering-Relevant Websites

---

http://electronicdesign.com/article/embedded/use-excel-to-develop-a-traceability-matrix9584

This page describes a simple, Microsoft Excel-based implementation of requirements traceability by Aubrey Kagan

http://en.wikipedia.org/wiki/Capability_Immaturity_Model

This Wikipedia webpage overviews the Capability Immaturity Model (CIMM). The "Capability Im-Maturity Model" asserts that organizations can and do occupy levels below CMM level 1. An original article by Capt. Tom Schorsch USAF as part of a graduate project at the Air Force Institute of Technology provides the definitions for CIMM. He cites a paper by Professor Anthony Finkelstein as an inspiration. The article describes situations that arise in dysfunctional organizations. These situations are not uncommon, and occur in organizations of all kinds undertaking development, i.e. they are more properly in-practice characterizations of the management of specific projects, since they can occur even in organizations with positive CMM levels. The CIMM levels of:

 0 Negligent
-1 Obstructive

-2 Contemptuous
-3 Undermining,
make an amusing read, with a touch of reality.

http://www.omgwiki.org/MBSE/doku.php

This wiki supports the activities of the Model-Based Systems Engineering (MBSE) Initiative that is sponsored by the International Council on Systems Engineering (INCOSE) and the Object Management Group (OMG) Systems Engineering DSIG. Content includes:

- MBSE Initiative Sponsoring Organizations

- MBSE Initiative Overview

- Challenge Teams

- Activity Teams

- Affiliated INCOSE Working Groups

- Tool Vendor Collaboration with MBSE Teams

- MBSE Events and Related Meetings.

http://www.cesames.net/

This is the website, in French, of L'association Cesames (Centre d'Excellence sur l'Architecture, le Management et l'Economie des Systèmes). The site contains an extensive selection of downloadable papers relevant to system architecting, organized into twenty categories. Most of the papers are in English. There is also a books list, a list of standards (which is essentially a list of systems engineering standards), and a page of useful links.

http://www.nasa.gov/offices/oce/appel/ask/issues/47/index.html

The Academy of Program/Project and Engineering Leadership (APPEL) and ASK Magazine aim to help NASA managers and project teams by sponsoring knowledge-sharing events and publications, providing performance enhancement services and tools, supporting career development programs, and creating opportunities for project management and engineering collaboration with universities, professional associations, industry partners, and other government agencies. The site includes a Project Management and Systems Engineering (PM&SE) Development Gateway.

https://standards.nasa.gov/

This website of the NASA Technical Standards Program, sponsored by the NASA Chief Engineer, provides access to standards from over 100 standards-developing organizations, including US DOD and NASA.

http://ntrs.nasa.gov/search.jsp

This is the website of the NASA Technical Reports Server. Using the Advanced Search feature, a search on "systems engineering" yielded eighteen thousand hits, each a downloadable resource. About 10% of hits appear to be relevant to systems engineering in general.

http://www.nasa.gov/offices/oce/appel/knowledge/publications/case_studies.html

Case studies illustrate the kinds of decisions and dilemmas managers face every day, and as such provide an effective learning tool for project management. Due to the dynamic and complex environment of projects, a great deal of project management knowledge is tacit and hard to formalize. A case study captures the complex nature of a project and identifies key decision points, allowing the reader an inside look at the project from a practitioner's point of view. This site provides access to a range of very interesting case studies, with a project management theme but relevant also to systems engineering.

http://www.afit.edu/cse/cases.cfm

This AFIT page is populated with thirteen interesting systems engineering case studies.

http://marshmallowchallenge.com/Welcome.html

The Marshmallow Challenge is a fun and instructive design exercise at this website of Tom Wujec.

http://nomtbf.com/

This is a site by Fred Schenkelberg devoted to the eradication of the misuse of MTBF – Mean Time Between Failure. What started as a simple observation has developed into a personal mission by Fred Schenkelberg to stop the misuse, misunderstanding and misinformation circling around MTBF. The site explores the issues, problems, and faults around the improper use of MTBF. It also provides advice on how to spot correct and incorrect uses, plus how to 'help' individuals, teams, suppliers and organizations to 'get it right'.

http://www.cso.nato.int/abstracts.aspx

Publications of the NATO Science and Technology Organization may be searched and downloaded from this website. Publications include some systems engineering content.

http://asq.org/learn-about-quality/idea-creation-tools/overview/nominal-group.html

This page contains a useful overview of the Nominal Group Technique, a structured method for group brainstorming that encourages contributions from all participants.

http://www.thinksysml.org/index.html

This site by the Cornell University College of Engineering Systems Engineering Program contains a small but useful set of SysML (system modeling language) resources.

Standards and Guides

DoDAF (USA Defense Architecture Framework) Future Development

The Department of Defense Architecture Framework (DoDAF) is an architecture framework for the United States Department of Defense that provides structure for a specific stakeholder concern through viewpoints organized by various views. DoDAF defines a set of views that act as mechanisms for visualizing, understanding, and assimilating the broad scope and complexities of an architecture description through tabular, structural, behavioral, ontological, pictorial, temporal, or graphical means. It is especially intended for large systems with complex integration and interoperability challenges. DoDAF is presently at Version 2.02. The roadmap for its development is:

- DoDAF V 2.02 developing to V2.03
- DoDAF V 2.04 add DNDAF Security
- DoDAF V 2.05 MODAF integration
- DoDAF becomes a DoD/IC Standard
- DoDAF becomes an Object Management Group (OMG) Standard
- DoDAF becomes Unified Defense Architecture Framework (UDAF)
- UDAF becomes an OMG Standard
- UDAF becomes required in contracts.

More information

MIL-STD-881 Revision C, Work Breakdown Structures
for Defense Materiel Items

MIL-STD-881 was reinstated on 3 October 2011 with the release of Revision C so that it can be used for new and existing designs and acquisitions. Titled "Work Breakdown Structures for Defense Materiel Items," the revision replaces both the MIL-STD-881B from March 1993 and the MIL-HDBK-881A from July 2005 (which is cancelled by this new release).

[Download](#)

A Definition to Close on

Capability

Capability: The quality of being capable; ability.
Source: www.thefreedictionary.com/capability

Capability: A talent or ability that has potential for development or use.
Source: www.thefreedictionary.com/capability

Capability: The capacity to be used, treated, or developed for a specific purpose:
Source: www.thefreedictionary.com/capability

Capability: the quality or state of being capable; also: ability
Source: www.merriam-webster.com/dictionary/capability

Capability: a feature or faculty capable of development: potentiality
Source: www.merriam-webster.com/dictionary/capability

Capability: the facility or potential for an indicated use or deployment
Source: www.merriam-webster.com/dictionary/capability

Comment from Robert: I sometimes hear the word capability used as if it were a system or a product. The dictionary definitions make it clear that a system may have or provide a capability, but a system is not in itself a capability.

PPI News (see www.ppi-int.com)

States & Modes – Proper Training Is On The Way!

PPI has under development an exciting new course titled simply "States & Modes". The course aims to remove the shroud of confusion, uncertainty, conflict, misinformation and sheer insanity that seems to accompany states and modes in the engineering community (while the rest of society seems to cuddle up with states and modes quite nicely). The requirements application of states and modes is first addressed, then the design application, both in considerable detail. The course is a workshop from start to finish. BYOC (bring your own computer). Present intentions are to launch the course in the United States in March, 2013. Watch this space.

Copyright Restrictions on PPI DIDs Relaxed

Feedback from friends of PPI regarding the set of DIDs of key systems engineering documents which are downloadable from the the Resources menu selection at PPI's home page has been excellent. To further increase the usefulness of these resources, copyright restrictions have been reduced. The new copyright which applies is:

"Copyright Project Performance International. This document may be reproduced and distributed without restriction, except as below, provided that all reproductions contain the original copyright statement in the original form and location. Derivative works may be produced provided each derivative work contains a copyright statement referring to the content in which PPI holds copyright, in a form and in a location no less prominent than the copyright statement on the original. Copies and

derivative works may not be used for the delivery of training for profit."

The DIDs subject to this relaxed copyright restriction are:
• DID - Systems Requirements Specification (SyRS)
• DID - Software Requirement Specification (SRS)
• DID - Interface Requirements Specification (IRS)
• DID - Verification Requirements Specification (VRS)
• DID - Operational Concept Description (OCD)
• DID - System/Subsystem Design Description (SSDD)
• DID - Concept of Operations (CONOPS)
• DID - Interface Design Description (IDD)
• DID - Systems Engineering Plan (SEP).

Other downloadable resources are subject to copyright restrictions as marked.

[More information](#)

## Robert Halligan Visits the Republic of Karakalpakstan and Finds an Environmental Disaster

Where?

Karakalpakstan is an autonomous republic of Uzbekistan. Karakalpakstan is mostly desert and is located in western Uzbekistan including some of the Aral Sea, in the lowest part of the Amu Darya basin. It occupies the whole northwestern end of Uzbekistan.

I visited Karakalpakstan in July, to research the Aral Sea area as a system. The Aral Sea, once the world's fourth-largest lake, has been disappearing since the 1960s, when planners from the former Soviet Union began siphoning water to grow cotton in what is now Uzbekistan. Now 70 percent smaller, the Aral Sea is incredibly salty, containing six grams of salt per liter–three times the safe limit for human consumption. No plants or crops can grow in such salty water. As the evaporation continues, the remnants of agricultural and industrial pollution are left behind. When the wind blows in Karakalpakstan, a mixture of dust, salt, sand, and chemical pesticides threatens the health of plants, animals, and humans. Because the dust clouds are filled with contaminants like heavy metals and DDT, Karakalpaks report resulting health problems. Many people have respiratory diseases, and the surrounding areas have the world's highest instances of tuberculosis. In some places, infant mortality is higher, and people are developing liver and kidney diseases.

Robert (center) as a Guest of the Family
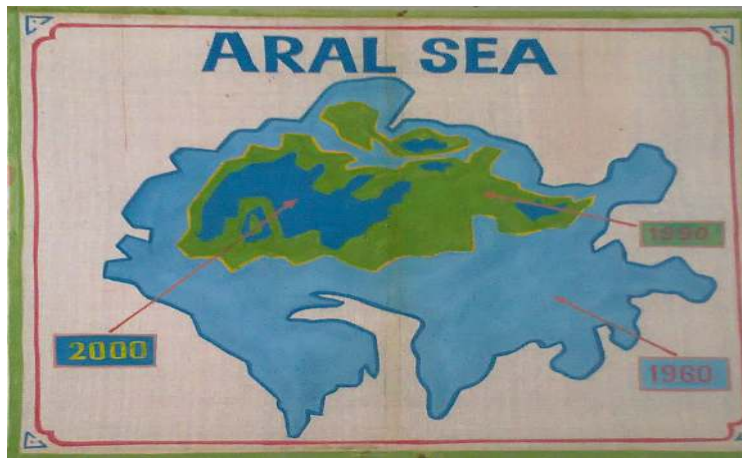
Aral Sea Shrinkage

Ship Graveyard

On a *very* positive note, the people of Karakalpakstan are warm and generous, family values prevail, and crime is low. We in the Western World could learn a lot from these people. I look forward to returning to their Republic, and to Uzbekistan in general.

Links:
Information Site: http://www.karakalpak.com/stangov.html
Official website of the Council of Ministers of the Republic of Karakalpakstan: http://sovminrk.gov.uz/lang/en/
Aral Sea Environmental Disaster: http://en.wikipedia.org/wiki/Aral_Sea

Robert Halligan FIE Aus

Introducing Brooke Mitchener

Brooke Mitchener joined PPI in July. Brooke is the newest member of the PPI family, working in administration. If you take a training course with PPI, there is every chance you will "meet" Brooke by email in relation to course administration.

PPI Events (see www.ppi-int.com)

Systems Engineering Public 5-Day Courses

Upcoming Locations Include:

- Amsterdam, The Netherlands

- Adelaide, Australia

- Brisbane, Australia

- Rio de Janeiro, Brazil

- Munich, Germany

Requirements Analysis and Specification Writing Public Courses

Upcoming Locations Include:

- Melbourne, Australia

- Stellenbosch, South Africa

- Las Vegas, USA

- Amsterdam, The Netherlands

Software Engineering Public 5-Day Courses

Upcoming Locations Include:

- Sydney, Australia

- Pretoria, South Africa

- Amsterdam, The Netherlands

OCD/CONOPS Public Courses

Upcoming Locations Include:

- Brasilia, Brazil

- Pretoria, South Africa

- Las Vegas, USA

Cognitive Systems Engineering Courses

Upcoming Locations Include:

- Adelaide, Australia

- Las Vegas, USA

CSEP Preparation Course (Presented by PPI subsidiary Certification Training International)

Upcoming Locations Include:

- Las Vegas, USA

- Austin, USA

- Munich, Germany

PPI Upcoming Participation in Professional Conferences

PPI will be participating in the following upcoming events. We look forward to chatting with you there.

- 9th Annual INCOSE SA Conference | Exhibiting | South Africa (27 - 29 August, 2012)

- 2nd European Defence Conference | Participating | Prague, Czech Republic (9 - 10 October, 2012)

- New Zealand Defence Industry Association Forum 2012 | Exhibiting | Wellington, New Zealand (16 - 17 October, 2012)

- Land Warfare Conference 2012 | Exhibiting | Melbourne, Australia (22 - 26 October, 2012)

- MilCIS 2012 Conference | Participating | Canberra, Australia (6 - 8 November, 2012)

- CSD&M 2012 Conference | Participating | Paris, France (12 - 14 December, 2012)

Kind regards from the SyEN team:
Robert Halligan, Managing Editor, email: rhalligan@ppi-int.com
Ralph Young, Editor, email: ryoung@ppi-int.com
Stephanie Halligan, Production, email: shalligan@ppi-int.com

Project Performance International
2 Parkgate Drive, Ringwood, Vic 3134 Australia
Tel: +61 3 9876 7345
Fax: +61 3 9876 2664
Tel Brasil: +55 11 3230 8256
Tel UK: +44 20 3286 1995
Tel USA: +1 888 772 5174
Web: www.ppi-int.com
Email: contact@ppi-int.com

Tell us what you think of SyEN: email to contact@ppi-int.com

If you do not wish to receive a copy monthly of SyEN in future, please reply to this e-mail with "Remove" in the subject line. All removals are acknowledged; you may wish to contact PPI if acknowledgement is not received within 7 days.