

Technical Paper Abstracts

The following section lists technical paper abstracts in the order in which they appear in the program. The number shown before each abstract title indicates the presentation order in the session and track. Papers can be referenced in the CD ROM Proceedings

SESSION 1

Session 1 Track 1: Systems Science

1.1.1 **START Analysis for ESAS Capability Needs Prioritization**

W. P. Lincoln, J. Mrozinski, H. Hua, S. Merida, K. Shelton,
V. Adumitroaie, C.R. Weisbin, J. Derleth
Jet Propulsion Laboratory

START is a tool to optimize research and development primarily for NASA missions. It was developed within the Strategic Systems Technology Program Office, a division of the Office of the Chief Technologist at NASA's Jet Propulsion Laboratory.

START is capable of quantifying and comparing the risks, costs, and potential returns of technologies that are candidates for funding. START can be enormously helpful both in selecting technologies for development – within the constraints of budget, schedule, and other resources – and in monitoring their progress.

START's methods are applicable to everything from individual tasks to multiple projects comprising entire programs of investigation. They can address virtually any technology assessment and capability prioritization issue. In this report, START is used to analyze the capability needs using data from NASA's Exploration Systems Architecture Study (ESAS).

1.1.2 **Defining the "Trade Space" for CAIV Optimization Using a Cost Model Derived from Linear Regression of NASA Project Data**

J. F. Krupa, *Westinghouse Savannah River Company*
P. J. Componation, *University of Alabama at Huntsville*

In 1995, Dr. Paul Kaminski, then Defense Undersecretary for Defense for Acquisition and Technology, established the principles of CAIV—"Cost as an Independent Variable"—to take advantage of aerospace's experience using Target Cost to battle the escalating cost of defense systems in a time of declining procurement and development dollars.

One of the key tenets of CAIV is to limit Key Performance Parameters (KPP) to the critical few, to establish thresholds and desired levels of performance, and to trade off performance, cost and schedule. This trade arena is called the "Trade Space". Although a number of authors have proposed methodologies for performance, cost and schedule trades, a clear process has not emerged. One difficulty that emerged from study was how to use existing cost estimating systems to perform "Trade Space" optimizations. This paper presents the use of a cost model derived from log linear regression analysis of data from eighty-six NASA projects, the Analytic Hierarchy process, and non-linear optimization to show how CAIV "Trade Space" can be developed from cost data.

1.1.3 Part Count and Design of Robust Systems

D. Frey, *MIT*; J. Palladino, *General Electric Aircraft Engines*;

J. Sullivan, *Pratt & Whitney*;

M. Atherton, *Rolls-Royce International Limited*

Systems engineering efforts are frequently undertaken to reduce part count with the goal of cutting costs, enhancing performance, or improving reliability. This paper examines the engineering practices related to part count, applying different theories – the Theory of Inventive Problem Solving (TRIZ), Axiomatic Design, and Highly Optimized Tolerance. Case studies from the jet engine industry are used to illustrate the complicated trades-offs involved in real-world part count reduction efforts. It is shown that no current theory fully accounts for the realities of system design, but that part consolidation at the component level generally proceeds consistent with the TRIZ ideality principle and that system part count and coupling are generally increasing driven by escalating demands for system performance and robustness.

1.1.4 Joint Cognitive Systems: Considering the User and Technology as One System

B. P. McKenna, J. W. Gualtieri, W. Elm, *ManTech – CSEC*

With the addition of advanced technologies and automation within systems, the tasks that operators are asked to complete have grown more complex. This has led to greater opportunities for malfunctions, critical incidents, and catastrophes. A major contributor to these problems is the human operator and the technology not being explicitly designed as a single Joint Cognitive System (JCS). The operator is typically only blithely considered during system design and is typically expected to adapt to inputs from the other subsystems. Thinking of the operator and the automation/technology from a holistic perspective, as a single JCS, instead of separate subsystems, allows for more effective, agile, and robust decision-making. The operator must have a complete understanding of the automated processes and the work domain, leading to more informed decision-making. Thinking of the entities as co-agents, rather than two isolated components that must interface, allows for more effective problem solving and decision-making.

Session 1 Track 2: SE Management Process

1.2.1 A Requirements Guide for All (REGAL)

J. Dick, *INTEGRATE Systems Engineering*;

G. Fanmuy, *PSA Peugeot Citroën*;

L-H Thevenet, *Université de Paris 1*

This paper describes an INCOSE initiative to collect from the systems engineering community information about good practice in requirements engineering, management and development. This initiative is the brainchild of the INCOSE Requirements Working Group, and is intended to provide a living requirements “Book of Knowledge” accessible in electronic form on the web, through which practitioners can contribute, evaluate and debate good requirements practice.

1.2.2 Enhancing SE Deployment in Large Organisations by Proactively Managing Service Quality of SE Training and Support Services

M. F. Kossmann, *AIRBUS UK*

When implementing Systems Engineering (SE) in large organisations, one key success factor that is absolutely essential for the acceptance of the related changes by concerned employees is the availability of high quality training and support services.

For many such employees that are meant to start applying SE in their daily work, the first and arguably most important impression is the one given in the corresponding training session. Also, once they have completed the training needed for their level of involvement, it is crucial that they find the necessary support to get started and keep on track.

The present paper will argue why it is worthwhile to proactively manage service quality when introducing SE in large organisations, give an overview of a proven Service Quality Management process for SE training and support services, illuminate the inherent advantages of such an approach and specifically look at a case study from Airbus, where the process in question has been used over a period of two years.

1.2.3 Role of Flow-Down Approach and Orthogonal Arrays in System Design and Testing

R. Jugulum, J. Singh, *MIT*

Design of systems to perform their intended functions is a creative and challenging activity because many requirements need to be satisfied under all customer usage conditions. In order to perform their intended functions over wide range of usage conditions, the systems must be designed in robust manner with a systematic flow-down of requirements at various levels of the design process. After completing the design they must be tested in various usage conditions and studying all possible main effects and interactions. This type of testing enables customers to have a robust product, which will avoid incidences of functional failures. In this paper we used axiomatic design's zig-zag approach for requirements flow-down and orthogonal arrays for testing the system's performance. Since software design is a good example of system design, we describe use of these techniques in system design by considering software design as an example.

1.2.4 How Do We Win This Game When the Rules Keep Changing? A Case for the Increased Application of Design for Six Sigma in Systems Engineering

N. A. Mackertich, D. G. Cleotelis, *Raytheon Network Centric Systems*

We all know the goal of the game: "Develop increasingly complex systems with smaller performance margins that meet the user's requirements in the shortest time, with high reliability, open and adaptable, and at the lowest cost". The environment we are working in continues to push Systems Engineering challenges to the next level. As Systems Engineers, we all want to win the game i.e. beat our competitors, satisfy our customers, develop unprecedented systems to address political and economic challenges. We have a proven process and a common set of tools, which we use and refine. Yet as we begin to participate in the game, we find that not only do we have huge challenges, but the rules keep changing. The customer/user requirements change, the conceptual design needs modification, the analysis reveals design challenges, budgets are reallocated. So how do we win this game? This paper presents a case for the increased application of a suite of enablers, processes, and practices referred to as Design for Six Sigma at appropriate integration points during the Systems Engineering process lifecycle.

Session 1 Track 3: SE Support Process

1.3.1 Quantifying the Benefit of Introducing Systems Engineering Processes - Myth or Reality?

E. Knippel, *BMW AG*; A. Schulz, *3D Systems Engineering GmbH*

In the past years the automotive industry had to face, especially in the field of electric/ electronic, an increase in complexity. To be able to keep highest quality standards BMW Group had to restructure its existing processes within electric/electronic development according to these new challenges. A Change Program was set up to shift those development processes towards a system orientated development.

For commercial driven companies an evaluation of cost vs. benefits is a prerequisite for approval and execution of such a change program. Therefore a business case was necessary. At the beginning of the change program a statement of anticipated likely impacts had to be made. At the end of the change program predicted impacts had to be validated. The approach taken was based on an aggregation of evaluations performed by the different subprojects of the change program.

This paper is illustrating – based on the experiences of a subproject (Configuration Management) – the approach taken to model and evaluate a business case at the beginning and the end of the change program. An example using neutralized data from BMW Group is demonstrating the approach. Lessons learned and success factors implementing the approach are discussed.

Evaluating or even quantifying the value of Systems Engineering Processes within product development is discussed within the SE community for quite a while, with very different positions. This paper aims at making a contribution to this discussion.

1.3.2 Context Based Measurement of Requirements Instability

M. A. Biddle, S. J. Moritz, *SAIC*

Requirements instability has always been and continues to be one of the more significant challenges for system development programs. Historical results across industry show that programs struggle and often fail to achieve stable and mature requirements baselines, and the impact of requirements instability on program cost, schedule and technical baselines is significant. This paper looks at some of the factors that affect requirements instability, and it outlines an approach for measuring requirements instability that takes contextual influences into account in order to provide insight as to why the requirements on a given program are unstable.

1.3.3 On the Alignment between System Architectures and Organizational Structures

T. Strandberg, D. Verma, *Stevens Institute of Technology*;
H. Burton, *Madera Consulting*

Enterprises of today need to be agile in how they manage systems over their life cycle. The design of system architectures has received a lot of focus as a means to facilitate agility. Less attention has been paid to the relationship of the system architecture to the organizational structures of all of the entities involved in the system life cycle.

A common, but inadequate, statement is that “the organizational structure shall mirror the product structure”. This research introduces processes as “middleware”, along with system architecture and organizational structure, to be included in the

trade-off for effective system life cycle management. Important additional factors that must be considered include informal organizational structures and the maturity of products, process and organizations.

This paper suggests a model of the problem description and recommendations for further research that can be adopted by practitioners and researchers in the fields of both engineering and organizational design. The paper builds on a previous paper by (Strandberg et al 2006) and elaborates on, in particular, the aspect of maturity.

Session 1 Track 4: Modeling & Tools

1.4.1 Using the Knowledge Pyramid to Characterize Systems

J. N. Martin, *The Aerospace Corporation*

Many systems help us increase our knowledge of the world in one way or another. Systems engineering needs a better way to understand how systems help or hinder the creation of knowledge. The Knowledge Pyramid was developed as a reference model to facilitate systems analysis with respect to data, information and knowledge. This paper describes the elements of the Pyramid and how they relate to system architecture.

1.4.2 An Approach to Simulation Effectiveness

D. P. Goncalves, *CSIR*

Simulation is an important aspect of engineering complex systems. In the real world, numerous problems can prevent the effective use of simulation. This paper looks at the tough question: When is a simulation effective? How would we know? The context and purpose of simulation are important in answering the question.

If the simulation is viewed as a system, it follows that it has stakeholders and requirements originating from the creating system. An important result is that measures of simulation effectiveness include fidelity, time-to-answer, and resource usage. The importance of a referent (codified knowledge) in defining fidelity and related pitfalls are discussed. Simulation effectiveness assessment enables simulation designers to trade simulation effectiveness against cost and risk subject to constraints. A brief overview of how abstraction and simulation method selection can be used for this trade-off is given. The impact of simulation effectiveness on risk is discussed. The benefits are balanced simulations with risk that is better matched to the problem at hand.

1.4.3 Cross Cutting Concerns and Ergonomic Profiling Using UML/SysML

M. C. Hause, *Artisan Software Tools, Ltd*

Modeling with UML/SysML involves many different organizing principles. These include inheritance and aggregation hierarchies, association networks, and dependency relationships. These can be addressed by the model architecture itself, and by the different UML/SysML diagrams. Also, there are requirements or areas of concern such as safety, risk, and traceability that can cut across the entire model at all levels. These need to be addressed early on in the project as they can have far reaching consequences for both product and process. Just as no architect would complete the design for a house, and only then address plumbing, gas, and electricity, no system architect would finish their system without having addressed safety, risk, and traceability. This paper will look at methods for integrating these cross-cutting concerns into a SysML model. It will demonstrate how ergonomic profiling

supported by an integrated database provides diagrammatic information in a format familiar to a domain specialist, maintains model and viewpoint consistency, and allows generation of customized reports to address specific concerns.

1.4.4 Introducing Structured Information Handling in Automotive EE Development

D. Malvius, O. Redell, S. Ritzen,
Royal Institute of Technology, KTH

One way for the automotive industry to cope with the demand of a more structured information handling is to adjust model-based development (MBD) to multidisciplinary needs. Many of the issues faced in this transition are as much organizational and managerial as they are technical. In a case study carried out at a global automotive manufacturer a project to improve the electrical and electronics (EE) development has been followed and analyzed. The project originated from different needs identified by management in their ongoing work effort towards MBD as well as by developers who experienced that tools did not support their work situation. This paper describes how the introduction of a new tool support was made in a project carried out within EE development, further it reports on benefits achieved by using the tool. Both the effect on the work of affected EE developers and the expansion of a new information model are discussed, leaving important implications for management. Success keys for putting new support tools into practice are identified here and include; a bottom-up approach, user involvement from the beginning, focus on the individual needs and adaptation to current work practice. Further, management support and adequate resources are essential for extracting long-term benefits.

SESSION 2

Session 2 Track 1: Systems Science

2.1.1 **Feelings and Physics: Emotional, Psychological, and Other Soft Human Requirements, by Model-Based Systems Engineering**

W. D. Schindel, *ICTT, Inc. and System Sciences, LLC*

Traditionally, engineering encourages requirements statements that are objective, testable, quantitative, atomic descriptions of system technical behavior. But what about “soft” requirements? When products deliver psychologically or emotionally-based human experiences, subjective descriptions may frustrate engineers. This challenge is important for products appealing to senses of style, enjoyment, fulfillment, stimulation, power, safety, awareness, comfort, or similar emotional or psychological factors. Automobiles, buildings, consumer products, packaging, graphic user interfaces, airline passenger compartments and flight decks, and hospital equipment provide typical examples. This paper shows how Model-Based Systems Engineering helps solve three related problems: (1) integrating models of “soft” human experience with hard technical product requirements; (2) describing how to score traditional “hard” technology products in terms of “fuzzier” business and competitive marketplace issues; and (3) coordinating marketing communication and promotion with the design process. The resulting framework integrates the diverse perspectives of engineers, stylists, industrial designers, human factors experts, and marketing professionals.

2.1.2 **Designing Systems for Adaptability by Means of Architecture Options**

A. Engel, *Israel Aircraft Industries;*

T. R. Browning, *Texas Christian University*

Systems provide value through their ability to fulfill stakeholders’ needs and wants. These needs evolve over time and may diverge from a fielded system’s capabilities. Thus, a system’s value to its stakeholders diminishes over time. As a result, systems are replaced or upgraded at substantial cost and disruption. If a system is designed to be changed and upgraded easily, however, this adaptability adds to its lifetime value. How can adaptability be designed into systems so that they will provide maximum value to stakeholders throughout their lifetime? This paper describes the problem and an approach to its mitigation.

We adopt the concept of real options from the field of economics and extend it to the field of systems architecture. We coin the term architecture options for this next-generation method and the associated tools for the design of flexible systems. Architecture options provide a quantitative means of implementing the optimal degree of design flexibility in a system to maximize its lifetime value for varied stakeholders. Based on initial research to date, we believe that implementing this aspect of design for adaptability can increase a system’s overall stakeholder value by 15% at a very conservative minimum. We also present an extension of a method for measuring the dynamic value of a system.

Session 2 Track 2: SE Technical Process

2.2.1 Test Time Reduction by Optimal Test Sequencing

R. Boumen, ISM de Jong, J.E. Rooda,
J.M. van de Mortel-Fronczak
Technische Universiteit Eindhoven

Testing complex manufacturing systems, like ASML lithographic machines (ASML 2005), can take up to 45% of the total development time of a system. This test time can be reduced by choosing wisely which test cases must be performed in which sequence, without making investments in test cases or the system. With the test sequencing method, developed by (Boumen 2005, 2006), it is possible to make these decisions such that a time, cost and/or quality optimal test sequence can be constructed. This paper shows with two case studies that test time can be reduced up to 20% without loss of quality with this method. The first case study is related to the test phase during the manufacturing of a lithographic machine. The second case study is related to the reliability testing of a first of a kind lithographic machine.

2.2.2 Integration and Test Strategies for Semiconductor Manufacturing Equipment

ISM de Jong, ASML and Technische Universiteit Eindhoven
R. Boumen, J.E. Rooda, J.M. van de Mortel-Fronczak,
Technische Universiteit Eindhoven

The complexity of semiconductor manufacturing equipment is growing. This growth results in a complexity increase of the integration and test phase of these systems. Simply adding more test resources is not possible anymore, because of the cost involved. A better design of an integration and test strategy can help to optimize this hectic phase. However, methods to design and evaluate integration and test strategies for multi-disciplinary systems are hardly available. In this paper, we present a method to design and compare integration and test strategies. Following this method, an optimal integration and test strategy can be chosen from a set of possible strategies. A case has been performed where a system is integrated and tested using three different integration and test strategies: a time-to-market-driven strategy, a quality-driven strategy and a combined quality and time-to-market strategy.

2.2.3 Directed Energy Weapon System Architecture to Meet Network Centric Operations Requirements

P. R. Marbach, *The Boeing Company*

Network Centric Operational Warfare (NCOW), the Global Information Grid (GIG), a typical Directed Energy (DE) Weapon System, and elements within a Battlespace that the DE Weapon would communicate with, are concepts introduced in this paper. NCOW requirements are described as applied to a weapon system, a DE weapon in this application, but the requirements and the solution proposed could be applied to other systems as well. This paper introduces a general DE Weapon and an architecture that meets the attributes described in the DoD Net-Centric Data Strategy.

Session 2 Track 3: SE Management Process

2.3.1 A Practical Program of Research to Measure Systems Engineering Return on Investment (SE-ROI)

E. C. Honour, *University of South Australia*

Past analysis has shown that there is a quantifiable correlation between the amount, types and quality of systems engineering efforts used during a program and the success of the program. For any given program, an amount, type and quality of systems engineering effort can be selected from the quantified correlations. The optimal nature of these selections, however, has not yet been explored. An ongoing project, Systems Engineering Return on Investment (SE-ROI), aims to quantify the correlations by gathering data on current and completed programs. This paper describes the practical program of research being used in the SE-ROI project and the current state of that development. The research program involves defining categorization sufficient to explore the correlations, implementing that categorization onto data sheets, gathering data from real programs through a personal interview process with the program leaders, and then performing statistical work to reveal the correlations. The project expects to achieve practical results in the form of (a) statistical correlation of SE methods with project success, to understand how much of each SE method is appropriate under what conditions, (b) leading indicators that can be used during a project to assess the project's expected future success and risks, and (c) identification of good SE practices that are appropriate to generate success under different conditions.

2.3.2 Towards a Work Breakdown Structure for Net Centric System of Systems Engineering and Management

G. Wang, *BAE Systems*; J. Lane, B. Boehm, *USC*;
R. Valerdi, *MIT*

As the system engineering industry sees an increasing focus on the lifecycle development, acquisition, and sustainment of net-centric Systems of Systems (SoS) and Family of Systems (FoS), organizations find the need to evolve current processes and tools to better handle the increased scope, scale, and complexity of these efforts. One such tool, the Work Breakdown Structure (WBS) is important in planning and execution of program activities as requirements and goals of the program evolve. This paper provides an overview of the limitations of current WBSs with respect to SoS efforts and presents a proposed WBS structure that more adequately reflects the evolving processes and cross-organizational complexities.

2.3.3 Uniting Three Families of Risk Management—Complexity of Implementation x 3

T. H. Holzer, *National Geospatial-Intelligence Agency*

This case study presents how a government agency overcame difficult challenges to establish a solid foundation for the management of its three families of risk- program, enterprise, and joint enterprise. Establishing an efficient and effective risk management process in a single development program is very challenging- culture resistance to change and unwillingness to share information viewed as negative prevail. There is additional complexity convincing people to adopt a process that is part of the bigger organization and sharing information regarding their ability to achieve program objectives. Now convince two different organizations to share information with each other. The outcomes show how the process can be scaled, adapted, integrated and

adopted across the complex systems and diverse organizations and enable program teams to identify, assess, and mitigate complex risks. For an organization aspiring to establish an internal or joint enterprise risk management program, these findings are a head start.

Session 2 Track 4: SE Support Process

2.4.1 Evolution of a Standard - EIA-632 From 1994 to 2006

R. M. Harwell, *SYSTEM Perspectives*

Process standards represent the best practices used by an industry or discipline. Most process standards are usually created by volunteer teams operating under the auspices of an authorized standards development body such as the Government Electronic and Information Technology Association (GEIA). This paper examines the evolution of one standard from its inception in 1994 to its current revision activities, noting the factors influencing scope, content, and structure. It also provides an insight into the latest decisions affecting the current revision efforts.

2.4.2 A Proposed Paper Template for Improving the Quality of Practitioner Written Papers at Conferences and Symposia

J. E. Kasser, *Systems Engineering and Evaluation Centre*

Conference papers are a major source of information in postgraduate education and research. However, the quality of many practitioner-written conference papers describing their experiences is less than optimal. This paper suggests a template to try to improve the quality of practitioner presentations and papers in the Case Study genre (prototyped at SETE 2004) to format practitioner papers as a way to link their experiences into the literature to provide data to assist researchers improving the practice of systems engineering. Examples of the use of the template are included.

SESSION 3

Session 3 Track 1: Modeling & Tools

3.1.1 COCOMO-SCORM Interactive Courseware Project Cost Modeling

R. Smith, L. Edwards, *SPARTA Inc.*

The U.S. Department of Defense has made significant investments in interactive courseware designed to train personnel distributed around the world. As these projects have multiplied, so too have the different approaches to estimating the schedule and staffing to create them. This paper presents the current status of a project that is creating a cost estimation algorithm for interactive courseware based on the COCOMO II family of software cost models. Our project focuses on specializing the COCOMO algorithm to account for the important variables to courseware development projects. We also will calibrate the algorithm to match historical data available from completed SCORM courseware projects.

3.1.2 Human Performance Modeling for Enterprise Transformation

G. Lintern, *General Dynamics-AIS*

Satisfaction of current demands for increased efficiency in the use of human resources requires nothing less than a revolutionary transformation in how we structure complex socio-technical systems. Human systems issues related to transformation of complex socio-technical systems pose substantive design challenges that must be addressed with human-centric analysis followed by human-centric design. In this paper, I integrate a cognitive work analysis strategy with a human performance modeling strategy which, when combined, constitutes a systematic, human-centric analysis-to-prototype design framework. I have described the cognitive work analysis strategy elsewhere and in this paper focus my discussion on cognitive workflow modeling. I illustrate the requirements of human performance modeling and methods of satisfying them with reference to a Brahms workflow model of time sensitive targeting based in part on a functional abstraction space developed within a cognitive work analysis and use that workflow model to motivate discussion of issues related to human-centric design for enterprise transformation.

3.1.3 Strategy For the Composition and Development of the Authoritative System Representation (ASR)

R. Clayton, *Booz Allen Hamilton*

Engineering a system in this era of hyper-complexity and massive scale, necessitates modeling and simulation (M&S) application at nearly every step of the system's life-cycle. The sequential nature of the Systems Engineering (SE) process provides opportunities to apply M&S that is particularly suited to each phase of the SE process. The credibility of M&S is strongly tied to the degree of similarity or fidelity of the M&S tool and its appropriateness in addressing the immediate engineering issues within the current SE phase. Correspondingly, engineering and developing the M&S tool requires a degree of specification analogous to the specification of requirements associated with the system. This specification serves to define what the tool is to do. It also serves to benchmark how well it is to do it so that it will be clear if the M&S tool was built correctly (through Verification and Validation). But unlike the larger system to which the tool will be employed, a majority of the M&S tool specification is not of

the tool, itself, but rather the portrayal of the object of its employment—the system/element under engineering or scrutiny. This is referred to as the system representation or simply, the representation.

This paper focuses on the development of a formal representation product for M&S mandated in the Defense (more specifically, Army) Acquisition system entitled the Authoritative System Representation (ASR). The paper's perspective is strongly oriented on the M&S developer's challenge of providing, to the System Engineer, tools that fulfil engineering needs. Every attempt is made to correlate System Engineering practices with Software Development practices of sound M&S construction. The ASR is a critical, but previously unheralded, component of the Defense Acquisition concept of Simulation Based Acquisition. This paper strives to share insight into this M&S product which is deserving of understanding by System Engineers.

Session 3 Track 2: Specialty Engineering

3.2.1 Corporate Social Responsibility(CSR)-the System Perspective & the Systems Engineering Role

A. Zonnenshain, *RAFAEL*

During the last decade, we have witnessed companies exercising Social Responsibility (SR) in their business. CSR is the responsibility that is incumbent upon a company, its managers and workers in their activities and decisions insofar as it affects the society. CSR includes issues and solutions in different areas such as: Environment, Hygiene, Safety, Sustainability, Working Environment, Labor Relationship, Human Resources, Product Life Cycle, Ethics, Investing in the Community, Corporate Governance and Transparency & Reporting. Leading companies are incorporating CSR requirements and issues into their Corporate Management System. This system includes strategy, policy, practices, processes, procedures, action plans, and success measures. This paper describes how to build and conduct CSR as a System. We do this by presenting CSR as a Management System by describing the principles of this System and through the experiences of my company – RAFAEL. Also, we describe the role of Systems Engineering in introducing and implementing CSR. We propose a role for INCOSE and its Chapters in promoting CSR.

3.2.2 Cost as an Independent Variable Balancing Performance with Affordability

E. J. Casey, D. M. Allen, *Raytheon Missile Systems*

The US Department Of Defense (DoD) Cost as An Independent Variable (CAIV) initiative replaced their Design To Cost (DTC) requirements. These changes were caused by the continuing reduction in the DoD budgets and the requirement to maintain the country's war fighting capabilities as the best in the world. In this peacetime environment, cost is the overarching constraint, not performance. As system providers, we are required to identify the customer's cost driving requirements. The customer/user can then makes the program decision to reduce system performance to maintain program costs.

CAIV is the trade-off of total ownership cost (TOC), also known as life cycle cost (LCC), performance, schedule and risk for a product that is developed, produced, and supported within the customer's expectations.

In order to easily evaluate a program's performance in the area of CAIV, a small series of metrics can be established. Since CAIV seeks to find the optimum balance between Cost, Performance,

Schedule and Risk, any set of CAIV Metrics should encompass these areas. These metrics would allow both senior management and the program manager a clearer picture of a program's status and progress, and, when applicable, be able to identify areas in which a program needs assistance or a concentration of forces.

This paper presents the CAIV process, program responsibilities, and a recommendation for a series of CAIV Metrics.

Session 3 Track 3: Systems Science

3.3.1 Network Centric Operation Implementations in Several Domains

C. Adler, *The Boeing Company*;

C. Dagli, *University of Missouri-Rolla*

A current focus in building military systems is the use of network based approaches. This is exemplified in the concepts of Network Centric Operations (NCO), Network Enabled Capabilities (NEC) or Network Based Defense (NBD). These concepts are, in part, inspired and motivated by the network based approaches that have revolutionized conducting business and operations in commercial domains across a wide spectrum of activities.

Network centric systems in the commercial domain display a number of interesting capabilities. These included the means to integrate extremely large groups of users into effective systems-of-systems with time constants measured on the order of hours/days. The basic network centric enablers used in these systems fall into several major categories.

This paper examines several high level architectures used in network centric operations for a number of non-military network based systems; a 'bricks and mortar' retailer, on-line 'eCommerce' businesses, and in software development. This examination is done by means of a literature survey and focuses on the interaction patterns that the systems must support, important capabilities the systems provide and important architectural enablers.

3.3.2 How Should We Use the Term "System of Systems" and Why Should We Care?

D. T. Cocks, *Lockheed Martin - MS2*

As an engineering discipline, systems engineering is relatively young. As this branch of applied science comes of age, it struggles with nomenclature and boundaries and roles of interaction with other disciplines. Yet before the foundations are quite set and cured, additional terms are being introduced to imply specialized subdivisions or independent offshoots of the yet-ill-defined context of systems engineering. This paper offers a proposed definition for "System of Systems" to distinguish it from "System" and "Family of Systems" that not only delineates the differences among these terms, but also postulates that there is value in recognizing these differences.

3.3.3 A Novel Modeling Approach Enhancing Classic Scenario Generation Techniques

S. J. Eelman, S. Föller, *Technical University of Munich*

Scenario methods are deployed in strategic planning to generate alternative environments as a mean to deal with uncertainties imposed by factors, which are hard to foresee or control. Current mainstream methodological approaches produce descriptive, steady-state scenarios with qualitative outcomes. For description of complex and unsteady environments they show deficiencies to reproduce dynamic behavior of systems and fall short to highlight scenarios of higher-than-average quality and value, restraining the efficiency to address uncertainty.

The paper presents an approach to comply with this need using established scenario techniques in an integrated way and incorporating a new technique to generate scenarios stepwise into the future. It utilizes procedures to cluster relevant scenarios and quantifies scenario quality with relevant indicators.

Results of a sample processes demonstrate that the approach behaves as complex system. Scenarios produced obtain a desired level of consistency and homogeneity within clusters while they establish high dissimilarity amongst one another. The achieved process quality contributes to an improved selection and assessment of relevant scenarios which form sound foundations for alternative strategy mappings and, thus, the understanding and treatment of system uncertainties.

Session 3 Track 4: SE Technical Process

3.4.1 Technology and Obsolescence Sustainment for Integrated Systems

T. E. Herald, D. J. Genaw, *Lockheed Martin*

Large, network-centric systems utilizing legacy elements, integrating newest commercial technologies and involving highly diverse joint-forces and international stakeholders have increased the magnitude and complexity of systems development and, subsequently, the system sustainment. This forces the need for additional bottoms-up tangible change management decision and optimization support for the systems and supportability engineers.

This paper addresses two critical aspects in the context of system-level obsolescence management. The first is System Obsolescence Life Cycle Forecasting and the second is Product and Technology Obsolescence Surveillance and Health Assessment. Proactive forecasting and assessment condense into a sustainment plan that encompasses the complete product operational life cycle. Product surveillance is the real-time data monitoring mechanism that calibrates the accuracy of the forecast and the implementation plan. This paper provides recommendations specifically for the supportability engineer and the systems engineer in order to optimize the design for system affordability and to monitor change through market surveillance of the system baseline elements toward continued support of program decisions.

This paper is broken into five parts in order to highlight the needs and solutions which apply to each life cycle phase. This provides a complete technology management strategy for an integrated system that is comprised of varying commercial technologies, readily available Commercial-off-the-Shelf (COTS) elements, and required legacy systems.

Part 1: Overview of Technology Management (TM) Requirements and Approach. Includes a discussion of the “life cycle mismatch” facing any operational system, wherein the desired system operation period is much longer than the life cycle of the constituent parts. This obsolescence mismatch must be managed at the system level to ensure program affordability.

Parts 2-4: Development, Production and Sustainment phase TM needs and solutions.

Part 5: Conclusions and Recommendations. Two distinct TM variables are covered; the life cycle phases and the program complexity. Understanding these variables guides the systems and logistics engineers to an optimal operational solution.

3.4.2 Capability Engineering for Strategic Decision Making

M. Lizotte, C. Nécaille, C. Lalancette,
Defence R&D Canada - Valcartier

The current paper summarizes the Capability Engineering Process (CEP) being developed to help decision making on strategic investments and divestments for the Canadian Forces and Department of National Defence. This effort is part of a technology demonstration effort called the Collaborative, Capability, Definition, Engineering and Management (CapDEM). The CEP introduces ways to increase strategic agility capability management in a world in constant evolution. A CEP application provides a set of options addressing a given capability gap. Among benefits, this process: (1) provides decision makers with timely strategic information through an iterative and incremental approach; (2) reduces time spent on unrealistic options by continuously pruning the solution space as early as possible; (3) provides operationally acceptable strategic options with direct involvement of the operational community into the solution development; and (4) provides feasible options by ensuring commitment and participation in developing solutions involving all of the organization's functional components: Personnel, R&D, Infrastructure, Concept development, Information management, and Equipment (known as PRICIE components in Canada).

3.4.3 Systems Engineering Net-Centric Solutions: An Analysis of Different Perspectives

D. H. Kemp, G. Crosby, D. Snell, *UK Ministry of Defence*

This paper reviews several different perspectives of net-centric operations (NCO) identified by the INCOSE NCO working group. It is one of three initial products of the group alongside a review of the Systems Engineering challenges posed by NCO and a literature survey of the state of the art in NCO.

The paper uses elements of Checkland's Soft-Systems Methodology to understand a range of different perspectives on NCO. It describes three different perspectives of the benefits of net-centric approaches, four perspectives of what constitutes a net-centric solution and four different approaches to realising net-centric solutions.

Finally the paper examines the relationship between the perspectives and draws some high level conclusions about the utility of the different approaches to different situations.

SESSION 4

Session 4 Track 1: SE Management Process

4.1.1 **Ten Design Principles: Some Implications for Multidimensional Quantification of Design Impacts on Requirements**

T. S. Gilb, *RPI*

Designs have multiple impacts on requirements, and can only be fully understood in terms of their impact on requirements. In other words, the contributions of designs towards the set of performance and resource requirements must be considered, when evaluating designs, in addition to their contributions to the function requirements.

This paper sets out ten principles, and outlines their various implications for design. These are basic ideas about designs, which we should explicitly acknowledge, teach and use in practice. I would be surprised to find any serious disagreement about these principles, but I would be surprised to find serious conscious practice and teaching of them today!

4.1.2 **Project-driven Adaptation of Software Life Cycle Model**

E. M. Barnard, *IFS South Africa*

Due to ever-changing technology and business requirements, the inefficiency and ineffectiveness of traditional plan-driven software development life cycle models are being questioned. Software engineers have to find ways of improving software development processes by applying more agile principles to the traditional plan-driven approaches. This paper addresses this issue and suggests how project and technical management can apply principles from the agile group of life cycle models to alleviate some of the problems encountered. Project characteristics that can be used to evaluate individual projects to determine the degree to which the software life cycle can be adapted, are indicated. Practices are also proposed for adapting traditional plan-driven life cycle processes that can benefit from more agile approaches.

4.1.3 **i-pub: Status, Insights and Visions**

E. Herzog, A. Pandikow, *Syntell AB*

The i-pub project aims at making INCOSE Symposia and other conference publications available on-line on the INCOSE intranet. In this paper we present the status of the project, insights gained from analysing the material collected and our visions for future extensions of the i-pub system

Session 4 Track 2: SE Management Process

4.2.1 **A Diagnostic Approach to Risk Driver Definition**

E. Stump, *Galorath Incorporated*

Brainstorming is a commonly used approach to generating project risk drivers. A consequence of brainstorming is that many proposed risk drivers are superficial symptoms, or at best next-level-down proximate causes.

Unfortunately, statements about symptoms and proximate causes are often assigned risk values such as probabilities or probability density functions. This paper will show that this practice can seriously overstate project risks. It can also lead to significant risk understatement due to issues of statistical independence.

4.2.2 How Planning for Success Can Lead to Catastrophic Failure

W. W. Schoening, *The Boeing Company*

Despite significant advances and continuing efforts to improve our processes, major programs continue to fail – not slowly and gracefully, but suddenly and catastrophically. Examination of failed programs over several decades suggests that our traditional systems engineering and program management methods are inadequate for discovering and mitigating the underlying causes because they have significant difficulty coming to grips with such huge failures. This paper identifies several symptoms related to potentially catastrophic failures during product development. Each symptom is accompanied by suggestions for remedies that can illuminate the underlying potential problems without undue impact on the program.

4.2.3 Systems Engineering Professional Development and Certification

G. H. Fisher, *The Aerospace Corporation*

In the mid-1990s, the Federal Government pursued “Acquisition Reform,” which resulted in significantly reduced government technical oversight of contractors. This caused less technical personnel to be hired in the government program offices for the last ten years. Recent investigations of space problems have recognized the need to revitalize the systems engineering workforce within the government program offices. Two years ago, the National Reconnaissance Office (NRO) embarked on the development of a professional development and certification program for space systems engineering. The NRO workforce is heterogeneous; it is comprised of military and civilian members of all DoD services as well as several intelligence community agencies. Our objective was to develop a program that maximized the synergy with parent-agency programs and avoided any redundant training requirements. A three-level certification program was established that required technical education, systems engineering experience, and systems engineering training. The training selected is a combination of existing NRO courses, off-the-shelf academic courses, commercial training classes, newly developed classes and INCOSE CSEP certification. After the first year, over 375 employees have attended at least one training class and we are certifying systems engineers at the rate of 10-12 per month. The success of this program has led to potential expansion into other areas of the government.

Session 4 Track 3: Modeling & Tools

4.3.1 Meeting the Challenge of Knowledge-Creating Systems

J. N. Martin, *The Aerospace Corporation*

Many systems help us increase our knowledge of the world in one way or another. Business application systems help managers understand the marketplace and how best to address market needs. Military systems provide command and control capabilities so we can have better battlefield awareness so we can fight to win. Chemical processing facility systems help us gain awareness of what is happening all along the processing chain. If all these systems help us gain knowledge of the world, does SE have the tools to analyze the knowledge-creation capacity of our systems? This paper will explore that question.

4.3.2 Platform Identification Using Design Structure Matrices

K. Kalligeros, O. de Weck, R. de Neufville, *MIT*;
A. Luckins, *BP Exploration and Production*

This paper introduces a methodology and algorithm for the qualitative identification of platform components at multiple levels of system aggregation, among variants within a family of systems. We assume that the architectural concept and the functional requirements for the variants are pre-determined, and use Sensitivity Design Structure Matrices (SDSM) to represent the sensitivities between the design variables of the variants. We then introduce a novel algorithm for the identification of platform variables given the SDSM for each variant. Finally, the methodology is extended to the qualitative identification of platforms at various levels of system aggregation, i.e., between systems, subsystems and components. The process is demonstrated in an example of platform identification among topsides facilities of Floating Production, Storage and Offloading (FPSO) units.

4.3.3 Applying Systems Modeling Language to A Simple Hardware System

J. C. Hsu, *The Boeing Company*

This paper used the Systems Modeling Language (SysML) to explore its applicability and effectiveness to a simple hardware system. The system and design requirements derived from the customer requirements were modeled in the requirements hierarchy diagram. The spring load equation with all the variables and constant were defined in the parametric constraint definition diagram. Parametric constraint on a block diagram was then used to design a conical helical spring. A link between design and requirements was demonstrated in the model. State machine diagrams depicted the finite state-transition on how a person emptied the mousetrap and how a mouse entered the trap. These diagrams were linked to verification requirements for system verification. SysML is shown to be an effective modeling language for hardware systems. The applicability of SysML to a combined hardware and software system has still to be demonstrated and proven.

Session 4 Track 4: Specialty Engineering

4.4.1 Fine-grained Method and Tool Integration for Better Automotive Software

F. Altheide, *University of Paderborn*;
K. Buhr, *Technische Universitaet Berlin*;
H. Doerr, *DaimlerChrysler AG*

In the automotive field, software development methods and tools are used to cope with the high complexity of automotive software development. However, problems occur with the tracing of information, the assessment and monitoring of the development status, late and thus expensive changes, etc, which lead to increasing development costs and time as well as quality deficits. Our experience with recent projects have shown that the key to solving these problems lies in providing proper answers to the following questions: (1) How can development methods be assembled? (2) How can the use of a variety of development methods be supported by an appropriate tool?

To answer these questions, we introduce a tool integration approach that is strongly based on the integration of development methods.

The contribution of this paper is (a) a method integration approach that considers fine-grained method objects and their specific structure in addition to the usually considered artifacts, (b) an approach to tool integration that overcomes the deficits of heterogeneous tool environments and utilizes the results of the methodical understanding.

4.4.2 Lessons Learned from Synchronizing Complex Systems Development within Automotive Industry

H. Negele, R. Schmidt, *BMW Group*;

S. Finkel, S. Wenzel, *3D Systems Engineering GmbH*

The automotive industry has encountered fundamental changes in the last decade, e.g. increased international competition, more complex and diversified products, and strongly changing product development processes. Nowadays, electrical/electronic (E/E) systems play a crucial role in vehicle development in order to offer innovative and attractive products to customers. Therefore, it is vital for an automotive OEM to master E/E systems over the entire life-cycle. Besides sound engineering, an effective project and risk management becomes more and more important. Synchronizing all the development activities and all involved stakeholders is a major challenge here. This is of key importance, as the E/E system represents a triplicate highly distributed structure, in terms of process, organization, and the product itself.

This paper illustrates lessons learned from implementing synchronization management within the E/E system development as part of a major process improvement initiative. The paper starts by highlighting the paradigm shift within the development processes of an automotive OEM from “component-driven” development to “system-oriented” engineering. Based on this background the need for synchronization of all involved “players” within E/E system development will be pointed out. The development and implementation of a generic synchronization/master plan as well as the usage and tailoring of this plan for specific projects will be described. Also, embedding this E/E synchronization management within the overall vehicle development process will be discussed (e.g. how to handle changes within the development process). The paper finally highlights lessons learned during the phases of conception, implementation, and application of the generic master plan and the synchronization management process.

4.4.3 Extending Platforming to the Sequential Development of System Families

R. C. Boas, E. F. Crawley, *MIT*

Creating system designs that efficiently satisfy an evolving set of needs is critical for improving program lifecycle properties such as cost, development time, and risk; properties that are often in conflict with near-term system requirements. This paper proposes that system evolution can be viewed as a family of sequentially developed systems and that existing work in the platforming domain serves as a useful foundation. An overview of related work is provided with emphasis on the platforming literature in an attempt to strengthen the connection between the systems engineering and product platforming communities. A classification scheme is proposed for multi-system design strategies. Preliminary cost model work is presented, with the ultimate goal being a model that provides program managers with a clear understanding of the economic trades associated with sequential platform development and alternative strategies.

SESSION 5

Session 5 Track 1: Systems Science

5.1.1 Enterprise Opportunity and Risk

B. E. White, *The MITRE Corporation*

In traditional systems engineering (TSE) environments, and particularly in recent years, many are concerned about risk management. So much so that risks are identified early and often, and mitigation techniques are aggressively pursued. In contrast one does not hear as much about opportunity management. What about enterprise systems engineering (ESE) environments? Do traditional methods of handling risk and opportunity carry over, or should we be concerned about potential changes in the way we view the problem? Assuming there are new perspectives to bring to bear on this topic, what are they, and what principles might be discovered and applied to the enterprise to better deal with opportunity and risk? This paper offers some answers to these questions

- There is duality in treating risks and opportunities.
- In ESE, be aggressive with opportunity and accepting of risk.
- The greatest enterprise risk may be in not pursuing enterprise opportunities.

5.1.2 Cross-Cultural Issues Associated with the Application of ISO/IEC 15288 Standard

T. L. Ferris, *University of South Australia*

This paper reviews the history of systems engineering standards documents from the perspective of the interaction of the standards with culture. The early standards were largely prescriptive procedural documents arising from concerns of the defense and aerospace industries, largely in USA. The latest standard, ISO 15288, follows an issues related approach and is designed to be flexible enough to address the systems engineering needs of a broad range of industries in many countries. The paper argues that the new standard presents different cultural challenges than the earlier standards. The cultural analysis of Hofstede is used as a theoretical construct with which to assess the interaction of the change from earlier standards to ISO 15288 with national and organizational cultures.

5.1.3 Systems Engineering an INCOSE Chapter

P. R. Davies, *Thales UK Ltd.*

In this paper, the setup of INCOSE UK is used as a case study 'infrastructure project', an example of a 'peopled system'. To begin with, the ad-hoc developments in the typical formation stages of an INCOSE chapter are shown, and this is used to point out the relative immaturity of the processes from a systems engineering point of view. Then we look at how we made the Chapter more self-sustaining, by a strategic overhaul of the 'system' using SE principles:

- Business Case drivers
- Stakeholder Identification
- Stakeholder needs (requirements) analysis
- Architecting the Organisation
- Allocation of system requirements to organisational responsibilities
- Decision analysis of likely return on effort invested
- Validation and Verification via Event Q&A surveys

Comparisons are drawn with IEEE1220, EIA632 and CMMI. Finally, the Metrics that we have recently started to examine are presented, as evidence of increasing maturity of the process.

Session 5 Track 2: SE Management Process

5.2.1 A National Approach to Systems Integration Skills Base Development in Australia

S.C. Cook, *CEDISC, UniSA*;

L. Sciacca, B. Bates, D. Nandagopal, *DSTO*;

S. Allison, *Allison & Assoc.*;

D.Shackleton, *Shackleton Management Solutions Pty Ltd*;

A. Yates, *Government of S. Australia Defence Unit*

Large-scale systems, such as rail networks, air traffic control systems, and networked defence systems are vital to national well-being. However, modern systems are becoming increasingly complex in nature and their development is proving problematical both in Australia and around the World. The paper opens by describing the nature of the problem and the best-practice approach taken to overcome the difficulties. From here, the Australian needs for systems integration research and skills development are discussed. Finally, the paper outlines the approach taken to create a research concentration in systems integration that will form a coherent basis for a national response to the Nation's systems integration research and skill-base capability requirements.

5.2.2 Optimizing Quality Assurance for Better Results

N. Malotaux, *N R Malotaux - Consultancy*

We know all the stories about failed and partly failed projects. Apparently, too many defects are generated by developers, and too many remain undiscovered by checkers, causing too many problems to be experienced by users. Solutions are mostly sought in technical means like processes, metrics and tools. If this really would have helped, it should have shown by now.

Oddly enough, there is a lot of knowledge how to reduce the generation and proliferation of defects and deliver the right solution quicker. Still, this knowledge is ignored in many development organizations.

In a recent booklet: *How Quality is Assured by Evolutionary Methods* [2], I describe how to organize projects using this knowledge successfully. This paper extends the use of this knowledge to QA, in order to optimize the contribution of QA to project success. Important ingredients are: a change in attitude, taking the Goal seriously, focusing on prevention rather than repair, and constantly learning how to do things better.

5.2.3 Capitalizing On Systems Engineering

J. J. Sherey, *ICTT, Inc*

Project managers often find it difficult to justify allocating significant resources and schedule to systems engineering tasks when "real" engineering has to be done. With ever-decreasing time to market demands, systems engineering continually loses out to design, integration, and test. A new method of systems engineering called Pattern-Based Systems Engineering (PBSE) enables companies to transfer portions of systems engineering costs out of project specific budgets and into company capital asset accounts. Such a change in accounting provides a series of benefits that include improved documentation and management of core corporate intellectual property, best practices, and standards as well as not having to reserve as much precious project money on tasks that need constant re-justification to product development managers with typically constrained budgets. This paper reviews PBSE, relevant accounting standards, and how much of systems engineering can be performed as a company-wide capital asset development program instead of as project overhead.

Session 5 Track 3: SE Technical Process

5.3.1 Lessons Learnt From the Applications of QFD to the Definition of Complex Systems

A. Hari, *SEEC UNISA*; J. E. Kasser, *University of South Australia*; M.P. Weiss, *Technion*

Quality Function Deployment (QFD) has been used to elucidate and negotiate requirements for at least 20 years. QFD has a lot to offer but when used to specify complex systems, has been found to have a number of deficiencies, has been perceived as a time consuming process and has been found as being difficult to apply to complex systems. This paper summarizes the top five lessons learned from 15 years of experience in using QFD in the elicitation, elucidation and negotiation of requirements for complex systems.

5.3.2 On the Systematic Use of Budget-Based Design

H. J. Freriks, *Océ-Technologies B.V*; WPMH Heemels, G. J. Muller, *Embedded Systems Institute*; J. H. Sandee, *Technische Universiteit Eindhoven*

In the majority of the engineering projects, the most important design decisions are taken at the early stages of the design. As these decisions have a considerable impact on the final implementation of the product, they ought to be made in a well-considered manner. The usage of higher-level methods to support making balanced decisions in initial stages of a design is considered very helpful. This paper deals with one of these methods: budget-based design. It is claimed that the systematic usage of budgets helps to speed-up the development process, to better assess project risks and results in better-founded design tradeoffs.

5.3.3 SysML-Based Systems Engineering Using a Model-Driven Development Approach

H-P. Hoffmann, *I-Logix*

More and more, systems engineers are turning to the System Modeling Language (SysML) to specify and structure their systems. This has many advantages, including verifiability and ease of passing off information to other engineering disciplines, particularly software. This article describes a SysML-based process that systems engineers can use to capture requirements and specify architecture. The process uses the SysML exclusively for the representation and specification of system characteristics. Essential SysML artifacts include use case diagrams, sequence diagrams, activity diagrams, statechart diagrams, and structure diagrams. The process is function-driven and is based heavily on the identification and elaboration of operational contracts, a message-based interface communication concept. The outlined process has been applied successfully at various customer sites.

SESSION 6

Session 6 Track 1: Specialty Engineering

6.1.1 The Application of Architecture Frameworks to Modelling Exploration Operations Costs

R. Shishko, *Caltech Jet Propulsion Laboratory*

Developments in architectural frameworks and system-of-systems thinking have provided useful constructs for systems engineering. DoDAF concepts, language, and formalisms, in particular, provide a natural way of conceptualizing an operations cost model applicable to NASA's space exploration vision. Not all DoDAF products have meaning or apply to a DoDAF-inspired operations cost model, but this paper describes how such DoDAF concepts as nodes, systems, and operational activities relate to the development of a model to estimate exploration operations costs. The paper discusses the specific implementation to the Mission Operations Directorate (MOD) operational functions/activities currently being developed and presents an overview of how this powerful representation can apply to robotic space missions as well.

6.1.2 Using Cognitive Engineering to Improve Systems Engineering

C. A. Bonaceto, K. J. Burns, *The MITRE Corporation*

Enterprise Systems Engineering (ESE) must go beyond the hardware and software of systems to address Human-Systems Integration (HSI). Towards that end, we propose that Cognitive Engineering techniques can and should play a key role in Systems Engineering efforts. In this paper we survey various methods in Cognitive Engineering, showing where these methods apply to specific problems in Systems Engineering from Concept Definition and Requirements Analysis, through Function Allocation and Performance Estimation, to Training Development and Performance Assurance. We also describe several uses of selected methods, including Cognitive Task Analysis, Computational Cognitive Modeling, and Critical Incident Analysis, to tackle specific problems in air traffic control. Taken together, these specific cases along with our general survey offer a roadmap for using Cognitive Engineering to improve Systems Engineering.

6.1.3 A Value-Based Theory of Systems Engineering

B. Boehm, A. Jain, *University of Southern California*

The INCOSE definition of "systems engineering" is "an interdisciplinary approach and means to enable the realization of successful systems." The Value-Based Theory of Systems Engineering presents necessary and sufficient conditions for realizing a successful system and elaborates them into an executable process. The theory and process are illustrated on a supply-chain system example, and evaluated with respect to criteria for a good theory.

Session 6 Track 2: Systems Science

6.2.1 Tradeoff Studies and Cognitive Biases

E. D. Smith, A. T. Bahill, *University of Arizona*

Tradeoff studies involving human subjective calibration and data updating are often distrusted by decision makers. A review of objectivity and subjectivity in decision making shows that Prospect Theory is a good model for actual human decision making. Similarities between the elements of tradeoff studies and the elements of experiments in cognitive science show that tradeoff studies are susceptible to human cognitive biases. Examples of relevant biases are given.

6.2.2 Systems Engineering Model for Integrability (SEMI): A Three Step Process for the Continuous Development of Highly Integrated Enterprise Applications

J. W. Lewis, *L3 Communications Titan*

Stovepipe analytic applications are the norm for the complex organizations involved in product design, law enforcement, and government. However, to be effective, these analytic applications must now rely on services provided by other organizations, exchange data with those organizations, and standardize on components defined by multiple integration architectures. Stovepipes are no longer adequate.

There is a bewildering array of technologies for developing integrated applications. Alternatives range from procedure libraries and relational databases to object-oriented databases and service-oriented architectures. Design decisions can have order-of-magnitude impact on performance, integrability, and maintainability. SEMI is a three-step process for structuring the decision making involved in the continuous development of integrated applications. The first step in SEMI is modeling the integration architectures; the second step is assessing the level of integration, and the third step is an integrated decision model.

6.2.3 Conflict in Systems Engineering Product Data Exchange Standardisation

R. Eckert, *EADS Deutschland GmbH*

Interoperability is the most critical issue facing business that need to access information stored in different CASE tools or information systems. Industry and the arch community have developed several standards and methodologies to facilitate interoperability among distributed and heterogeneous information systems.

The paper compares several available standards for the exchange of systems engineering data. As an example, the enormous overlapping between the available solutions for the exchange of requirements data and the resulting disadvantage for industry is shown. A better organised way for developing new standards is proposed.

Session 6 Track 3: SE Technical Process

6.3.1 Threads of Reasoning: A Case Study

H. Sandee, *Technische Universiteit Eindhoven*;
WPMH Heemels, G. J. Muller, *Embedded Systems Institute*;
P. van den Bosch, *Océ Technologies BV*;
MHG Verhoef, *Chess Information Technology BV*

In the design of technology intensive products like copiers, wafer steppers and televisions, one searches for a product that satisfies the product requirements as well as the business drivers. The main need in an early design phase is to bring structure in the typical chaos of uncertainty and the huge amount of realization options present. Potential realization choices all have advantages and disadvantages, which cause tensions and conflicts. The earlier the (essential) conflicts and tensions are identified, the better it is. Turning them from implicit to explicit helps the system architect in making the trade-off consciously or at least in selecting the most important tensions and conflicts that require further in-depth investigation. In this respect we demonstrate the effectiveness of a technique called “threads of reasoning”. The illustrative case study is the design of the paper flow control (sensors, actuators, control architecture, etc.) in a high-volume copier/printer.

6.3.2 No Cure No Pay: How to Contract for Software Services

T. Gilb, *RPL*

50% of all software projects are total failures and another 40% are partial failures according to widely quoted surveys in UK, USA and Norway. Large government projects in all 3 countries have been reported with spectacular failure and expense to taxpayers (Royal Academy of Engineering and British Computer Society 2004). What is the problem? Most discussions have centered on improving the software engineering process itself: better estimation, better requirements, better reuse and better testing. No doubt all those can be improved. However, I suggest the motivation to improve them needs to be put in place first. Think about it. Most of these failures have been fully paid for! We not only pay well for failure, but the bigger the failure, the more people get paid!

My suggestion is simple. Pay only when defined results are provably delivered. This requires several things:

- Contracts that release payment only for meaningful results;
- The ability to define those results, particularly qualitative ones, and particularly the organizational ones;
- The ability to deliver those results incrementally, thus proving capability at early stages and continuously.

Note: This paper specifically addresses the software problem, but I am sure that the ideas here apply to the wider systems engineering problem to some interesting degree as well.

6.3.3 Application of Patterns to Systems Engineering and Architecting

R. J. Cloutier, J. Boardman, D. Verma,
Stevens Institute of Technology

A system architecture pattern constitutes a high-level structure that has been used multiple times in the same or multiple system architectures, appropriate to the design of the major components of a system. It expresses the relation between the context, a problem, and a solution. It documents attributes and usage guidance. It should be time-proven in solving problems similar in nature to the problem under consideration. The existence of patterns is almost universal, and their use is evident in many domains. The human mind seems to perceive patterns without conscious thought - we notice an individual's personal habits because they form patterns. Patterns are also used in a number of engineering disciplines - software engineering, requirements engineering and mechanical engineering to name a few. The purpose of this paper is to discuss motivations for using patterns in architecting complex systems. Further, research relating to this subject is examined and summarized.

6.3.4 Impact of Embedded Software Technology on Systems Engineering

L. J. Doyle, *ITT Industries*;
M. C. Pennotti, *Stevens Institute of Technology*

Systems such as aircraft and consumer electronics have included embedded computers in their designs for several decades now. In these systems, software was usually treated as a component inside a box much the same way you would treat a circuit card assembly or a power supply. Requirements would flow down to the box specification and the box specification would then flow down to the components inside the box. But recently, several trends are changing the relationship between embedded software and the system as a whole. Many embedded computer systems now use network protocols and infrastructures similar to information systems deployed on commodity hardware. Systems with embedded computers now have expectations and requirements for extensibility similar to those of information systems where methodologies for extensibility are well established. But unlike information systems, systems with embedded software often have complex non-software elements and the software is not loosely coupled with the hardware. These systems require a synthesis of approaches.

SESSION 7

Session 7 Track 1: Complex Systems

Session 7 Track 2: Modeling & Tools

7.2.1 The FAR Approach : Functional Analysis/Allocation and Requirements Flowdown Using Use Case Realizations

M. Eriksson, K. Borg, *BAE Systems Hägglunds AB*;
J. Börstler, *Umeå University*

This paper describes a use case driven approach for functional analysis/allocation and requirements flowdown. The approach utilizes use cases and use case realizations for functional architecture modeling, which in turn form the basis for design synthesis and requirements flowdown. We refer to this approach as the FAR (Functional Architecture by use case Realizations) approach. The FAR approach is currently applied in several large-scale defense projects within BAE Systems Hägglunds AB and the experience so far is quite positive. The approach is illustrated throughout the paper using the well known Automatic Teller Machine (ATM) example.

7.2.2 Performing Functional Analysis/Allocation and Requirements Flowdown Using Use Case Realizations: An Empirical Evaluation

M. Eriksson, K. Borg, *BAE Systems Hägglunds AB*;
J. Börstler, *Umeå University*

The FAR approach (Functional Architecture by use case Realizations) is a use case driven methodology for functional analysis/allocation, and requirements flowdown. The FAR approach is tailored towards the development of long lived software intensive defense systems, for example vehicles. In this paper, we present an empirical study where FAR is applied and evaluated in two large-scale defense projects. Our results indicate that the FAR approach performs better than the previously used approach in the organization.

7.2.3 Tying Requirements to Design Artifacts

H. C. Briggs, *California Institute of Technology*;
M. Sampson, *UGS*

A new generation of Product Data Management and Product Lifecycle Management tools offer a combination of systems engineering and design engineering capabilities with a common database. This provides the opportunity to tie requirements to elements of the design such as CAD models, drawings and analyses. Connecting the currently disjoint systems engineering and design engineering modelling domains provides a new opportunity for a rich set of relationships and the promise of alleviating much of the tedious checking of requirements usually performed by design engineers. This paper explores the use cases for joining systems engineering and design engineering modelling and proposes a set of relations that provide meaning to the new links.

SESSION 8

Session 8 Track 1: Systems Science

8.1.1 Application of Systems Engineering to Industrial Supply Chains

C. Haskins, *NTNU*

This paper discusses the application of systems engineering alongside industrial ecology, systems dynamics and soft systems methodology to the study of industrial supply chains. As supply chains are ultimately a human activity, a multidisciplinary approach must include human-centric investigation and knowledge-sharing techniques. Systems engineering is seen as a proper foundation for defining a process for which the characteristics are unknown and for solving the problems found in cross-organizational supply chains. Building on (Haskins 2003, 2004), Christopher Alexander's pattern language approaches are considered as a mechanism for sharing best practices. The paper closes with suggestions for the direction of further research.

8.1.2 A Study of Applying Game Theoretic Concepts on Distributed Engineering System Design

H. Wang, D. Frey, *MIT*

This paper considers the application of game theoretic concepts to the dynamics of interactions among the designers in an engineering systems design problem under a distributed decision making environment. Among all the possible distributed frameworks, we focus on two types of frameworks in the system design field, the simultaneous and the sequential response frameworks. In the investigation of the simultaneous response framework, we apply the concepts of learning theory to construct insights into the dynamics among the designers and to address convergence and other challenges of the decentralized design problems. We use a special class of games, the potential game, to demonstrate the effectiveness of the sequential response framework which points out a prospective application of game theoretic concepts to future distributed engineering systems problems.

8.1.3 Identification of Real Options "in" Projects

T. Wang, *Morgan Stanley*; R. de Neufville, *Massachusetts Institute of Technology*

The concept of real options - initiated in the field of finance - has extended into engineering systems to model design flexibility in the realistically uncertain environment. However, whereas financial options are well-defined traded contracts, real options "in" engineering systems are a priori undefined, complex, and interdependent. Moreover, systems involve many more options than designers could consider. Therefore designers need to identify the real options most likely to offer good flexibility and the most value.

This paper proposes a procedure to identify real options "in" engineering systems. It consists of a screening and a simulation model. The screening model is a simplified, conceptual, low-fidelity representation of the system that reflects its most important issues. As it is inexpensive to run, it is used to test extensively designs under dynamic conditions. The following simulation model is used to validate critical considerations, such as the robustness and reliability of the designs, which are omitted from the screening model in order to expedite its operation.

The paper first establishes the concepts of the options identification model, and then resorts to examples to detail its application. The case of a hydro power system formulates the screening and simulation models, and presents the specific steps needed to search systematically for the interesting real options.

Session 8 Track 2: SE Management Process

8.2.1 Establishing a Verification and Validation Process in Automotive Development: Increasing Product Quality while Reducing Costs

M. de la Cruz, A. Vollerthun, *3D Systems Engineering GmbH*;
J. Meisenzahl, *BMW Group*

In spite of continuous improvements in maturity levels of the electric/electronics development process, the structured verification and validation of automobile electric and electronics in all product phases takes on increasing importance. The methodology presented in this paper is based on the “5 dimensions” of verification and validation: what to “verify”, how, wherewith, when and by whom and focuses on a) the systematic planning and specification of verification and validation activities as a link between the requirements engineering and the testing itself, and b) the results of its successful user-oriented implementation in an actual vehicle project of the BMW Group.

This paper describes a proven methodology that contributes to handle the increasing complexity (e.g. caused by the multitude of variants) of automotive systems while still enabling innovations and premium quality. Further the organizational tasks, which must be met in order to customize the methodology to the specific project situation, are addressed.

8.2.2 Applying System Engineering to Naval Shipbuilding

B. H. Wells, *Raytheon Integrated Defense Systems*

The processes for naval architecture and ship design evolved before the systems engineering processes used today were defined and documented. The standard systems engineering processes based on MIL-STD-499 and now ANSI/EIA-632 have been extensively applied to the development of military and commercial products. While the designs of the naval ship combat system payloads have used systems engineering methods, the ship design, and development has traditionally applied processes that are unique and not included in the standard systems engineering activities. The development of many ship subsystems has followed an evolutionary approach that has not required many of the systems engineering process steps. This paper defines some of the unique challenges of applying the systems engineering process to the design and development of Navy ships. Some of the traditional naval architecture and ship design processes are presented and contrasted with the systems engineering standards. The challenges that required the amalgamation of these two cultures are discussed. Finally, processes and methods that provide an integrated approach are presented.

8.2.3 Giving the Integrator Role a Sporting Chance

J. R. Armstrong, *Systems and Software Consortium*

As systems become larger and more complex, there are more opportunities for playing the role of integrator in an expanding 'food chain' of layered development. Many organizations are seeking to increase their integration role and move up the ladder. This paper provides guidance on what that change means and requires. It also does so with an analogy to levels of play in the world of sports.

Session 8 Track 3: SE Support Process

8.3.1 Quantitative Analysis: Clawing your Way to the Top of the Maturity Pinnacle

P. J. Frenz, A. C. Gurvin, *General Dynamics AIS*

Quantitative Analysis is discussed at length in text books and articles. Statistical Process Control (SPC) articles address how you apply these principles, but real-world application is the ultimate challenge. This paper describes tools, techniques and methods used to institutionalize Quantitative analysis in a Capability Maturity Model Integration (CMMI) Level 3 environment. General Dynamics Advanced Information Systems (GDAIS) Mission Systems achieved CMMI Level 3 rating June 30, 2004 using the CMMI-SE/SW/IPPD/SS, V1.1, staged model.

The goal: achieve a CMMI Level 5 rating before the end of 2005.

8.3.2 Use of Technical Business Practices by Geographically Separated Teams to Facilitate Concurrent Engineering of Ultraquality Systems (LA-UR-05-8500)

W. E. Neff, *Los Alamos National Laboratory*;

C. H. Dagli, *University of Missouri-Rolla*

In 1997, the National Nuclear Security Administration (NNSA) undertook the development of an architecture of standards and requirements to standardize the engineering design and manufacturing of ultraquality weapon systems. This architecture, known as the Technical Business Practice (TBP) system, was implemented in 1999. The TBP system is centered around a concurrent engineering philosophy and based on a Product Realization Process methodology which is broken into four process steps: Definition, Development, Delivery, and Support.

The four-step concurrent engineering approach is accomplished through multi-disciplinary teams known as Product Realization Teams (PRT) which are comprised of managers and subject matter experts from across the Nuclear Weapons Complex. Depending on the breadth and complexity of the engineering tasks being executed, PRTs can be subdivided into sub-PRTs or Task Teams. Both sub-PRTs and Task Teams tend to have a "component-focus" versus the "system focus" of the PRTs. Using the architecture of TBPs, focused around a concurrent engineering philosophy employed by PRTs, execution of nuclear weapon programs has become more efficient, consistent, and comparable across the Nuclear Weapons Complex.

8.3.3 Use of Architecture for Engineering Systems: The Good, The Bad, and The Ugly

G. Osvalds, *Wells Landers Inc.*

Many programs have created architectures yet they have not been effective in producing viable results. The primary reason for this the ineffectiveness is derived from how programs are managed. While System Engineers perform engineering activities by developing artifacts; the complex relationships among engineering processes, the final products, and the creation of systems that meet customer expectations have not been well-managed. Resolving this issue can be difficult. Even though Systems Engineering processes can provide guidance in Engineering Systems, it requires customer support and effort to make architecture actionable. To be successful in providing solutions, the Systems Engineer must apply Systems Engineering concepts to the customer's problems by educating the customer and assisting them to drive the system development effort. The author will present examples of "The Good," "The Bad," and "The Ugly" he has observed in U.S. Government Federal Agency and Department of Defense programs as it relates to the application of architectural engineering. Based on these tendencies, recommendations are presented to improve architectural effectiveness by urging a return to fundamental systems engineering principles and insuring that the customer is a solid stakeholder in upholding, supervising, and executing the architecture as an essential element of the new system or program.

SESSION 9

Session 9 Track 1: Specialty Engineering

9.1.1 Trade Study Cost Analysis Model

E. J. Casey, D. M. Davis, *Raytheon Missile Systems*

The basic goal of a Cost-Performance Trade Study is to find the optimum solution between several options that best balances cost, performance, risk and schedule given an agreed upon weighting.

The Trade Study Cost Analysis Model was developed to support the CAIV trade study process by providing a means for the user to quickly and easily analyze alternative options. The model computes a rough order Life Cycle Cost (LCC) estimate for each study alternative, which can be compared giving the user a powerful input to the decision making process. The output should be considered a comparative estimate that is not the bottom line number, but rather the relative cost relationships between the alternatives. This model is not a bidding tool.

The Trade Study Cost Analysis Model is primarily a predictor of Life Cycle costs and particularly Operation and Support (O&S) costs. Development and production costs are inputs to the model but not derived using the model. The Trade Study Model provides an estimate of the costs for the full life of the system/item being studied. While primarily hardware oriented, there is a software module embedded. The model has been designed to be used at the sub-system and Lowest Repairable Unit (LRU) levels.

Life Cycle Cost (LCC) Analysis is helpful to engineers in making decisions for trade study alternatives. They (LCC Analysis) are extensive and often require a great deal of time and resources to prepare.

Included in this technical paper is the framework by which this model was created.

9.1.2 A Basic Primer in Life Cycle Cost Analysis

F. Q. Redman, A. T. Crepea, *Raytheon Missile Systems*

Life cycle cost (LCC) has become increasingly important as systems are designed with longer useful technological lives. In addition, deferred modernization due to limited budgets is causing many aging weapon systems to be extended, thus incurring greater support costs. This paper will present an overview of the process of life cycle cost analysis (LCCA). It will address the basic concepts and applications, along with a review of methodologies and some aspects of engineering economics.

LCC is defined as the sum total of the direct, indirect, recurring, non-recurring, and other related costs estimated to be incurred in the design, research and development (R&D), investment, operation, maintenance, and support of a system over its life cycle, i.e. its anticipated useful life span. It is normally organized into four phases: research, development, test, and evaluation (RDT&E); procurement (or acquisition), operations and support, and disposal.

Economic Decisions are often made solely on initial investment or RDT&E and procurement cost. But this is only the proverbial tip of the iceberg – a portion of the total cost of ownership. More and more customers (especially government) are emphasizing and requiring an LCC perspective. To make intelligent acquisition decisions to meet a specific need, it is necessary to look beyond acquisition. In defense systems O&S costs can encompass up to 80% of the total LCC.

LCC allows the evaluation of competing system proposals on the basis of total ownership cost and allows more effective budgeting of future funds such as O&S costs and disposal costs. A LCC perspective maximizes the benefit of applying strategies such as Cost as an Independent Variable (CAIV) and Design to Cost (DTC). Most LCCAs require a mix of estimating methodologies. The methods may be applied individually or in combination. Analogy and parametric tend to be most useful in the early stages of product development. As program and system design stabilizes parametric estimating becomes more useful. A detailed design facilitates engineering estimate and projection of actuals.

In summary, this paper gives an overview of LCC as well as addressing many of the complexities of LCC analysis. More and more customers (especially government) are emphasizing and requiring an LCC perspective making LCC an increasingly important engineering design parameter. Total cost impact, not just initial near-term cost, must be considered.

9.1.3 Screening for Real Options “In” an Engineering System: A Step Towards Flexible System Development PART 1: The Use of Coupled Design Matrices to Create an End-to-End Representation of a Complex Socio-Technical System

J. E. Bartolomei, D. E. Hastings, R. de Neufville,
D. H. Rhodes, *MIT*

The goal of this research is to develop an analytical framework for screening for real options “in” an engineering system. Real options is defined in the finance literature as the right, but not the obligation, to take an action (e.g. deferring, expanding, contracting, or abandoning) at a predetermined cost and for a predetermined time. These are called “real options” because they pertain to physical or tangible assets, such as equipment, rather than financial instruments. Real options improve a system’s capability of undergoing classes of changes with relative ease. This property is often called “flexibility.” Recently, the DoD has emphasized the need to develop flexible system in order to improve operational, technical, and programmatic effectiveness. The aim of this research is to apply real options thinking to weapon acquisitions in order to promote the ability of weapon system programs to deftly avoid downside consequences or exploit upside opportunities.

The practice of real options in systems engineering is a nascent field of inquiry. One of the most significant challenges in applying real options to engineering systems is the problem of identifying the most efficacious points within the system to create options. In order to identify the points of interest, systems engineers require knowledge about the physical and non physical aspects of the system, insight into sources of change, and the ability to examine the dynamic behavior of the system. We propose a two-phase process to perform this analysis. The first phase is a system representation phase that seeks to create an end-to-end representation of engineering system that includes endogenous interactions across system views and interactions with a systems environment. The next phase is an analysis phase that models the evolution of the engineering system in order to identify the real options in the system. This paper presents the system representation phase and proposes a methodology for creating an end-to-end representation of an engineering system.

The methodology for representing an engineering system extends existing systems engineering and architecting methods in two dimensions. First, the framework couples traditional architecting

views to represent traceability and endogenous interactions within an engineering system. Second, the framework includes views of the system not represented in traditional engineering frameworks that includes social networks and environmental interactions. The framework uses coupled Design Structure Matrices (DSM) to represent the traditional and new architecting views. The coupled DSMs are organized into an Engineering System Matrix (ESM), which is a holistic representation of an engineering system that captures the critical variables and causal interactions across architectural elements. The result is an analytic framework that captures the qualitative understanding of the system into a single view that is conducive for deep quantitative inquiry.

This paper presents a discussion of pertinent literature, an overview of the ESM framework and underlying theory. In addition, this paper previews ongoing research using the ESM to identify options for a mini-air vehicle (MAV) weapon development system.

Session 9 Track 2: Systems Science

9.2.1 A Merlin Perspective Shines Light on Tough Issues

J. W. Carl, *Retired*

Shining light on tough issues requires leaders, including technical leaders, to commit to creation of a future that cannot reasonably be extrapolated from the place they are now standing. Fortunately, about a dozen years ago Dr. Charles Smith published an important paper (Smith, 1994) that has not received enough attention on the part of leaders responsible for solving the toughest issues of our time – affordable renewable energy, environmental health of the planet, failures of global cooperation, among others perhaps more mundane but of interest to systems engineers in their technical leadership roles. This paper reviews the advice in (Smith, 1994), and applies it to the establishment of visions for possible futures that are desired by a significant portion of the public, although the envisioned futures seem impractical to achieve. And that requires leaders to decide to identify with a future for which the way must be opened.

9.2.2 A Framework for a National Undergraduate Systems Engineering Stream of Studies in Discipline-centric Degrees: Proposal Analysis

T. L. J. Ferris, *University of South Australia*;

Y-S. W. Peng, *Overseas Chinese Institute of Technology*

The authors have developed a framework proposal for undergraduate systems engineering education as a stream of studies forming part of a Bachelor of Engineering program in any discipline of engineering. This proposed framework has been developed with the intention of use broadly across many universities, colleges, and in all engineering disciplines in Taiwan. The paper discusses the pedagogical background to the proposal and analyses the proposed structure from several educationally significant viewpoints. The broad fronted approach to curriculum presented in this paper includes Bloom's taxonomy of educational objectives, the concept of graduate outcomes, a contrast of the capability of engineers educated traditionally and through the proposed curriculum, the current cultural context of undergraduate students, and the national objectives to be served through the educational activity.

9.2.3 Architecting Synthetic Environments to Support the Systems Engineering of Capability

D. J. Battersby, *BAE Systems (SEIC)*

Validating the link between an architecture framework representation of a capability and the systems which provide that capability is a substantial challenge. Synthetic environments are suggested as a method whereby this can be achieved. A traceable method for supporting this is provided through the mapping between architecture framework views, synthetic environments and the design lifecycle.

The use of synthetic environments is also highlighted as a key method for the management of capability, through the ability to integrate human factors, process and technical issues. An iterative use of synthetic environments, with increasing levels of modeling fidelity, is suggested as an approach to supporting engineering and acquisition throughout the whole project lifecycle.

Session 9 Track 3: SE Technical Process

9.3.1 Trade Studies with Uncertain Information

D. G. Ullman, *Robust Decisions*; B. P. Spiegel, *Honeywell*

During every stage of the design process, designers trade off performance, cost, and risk in an evolutionary process whose goal is to find a satisfactory solution. This paper explores a recent method to manage the trade study process especially when uncertainty is pervasive and decisions are a mix of quantitative and qualitative information. We believe that it is possible to support a trade study process that is sensitive to the uncertainties in evolving system information, a key ingredient in managing risk, robustness, changes and spiral development. In this paper we explore what is needed to support such activities. To do so we follow an example as it gets increasingly complex and realistic. As the issues addressed increase in computational need, we make use of Accord, a decision support system base on Bayesian Team Support methods.

9.3.2 Real Options and Value Driven Design in Spiral Development

J. W. Dahlgren, *The MITRE Corporation*

This paper proposes that the failure of many programs stems from inaccurate requirements development and failure to design for spiral development. The tremendous growth in the complexity of our society has led to a proportional growth in the number of requirements and requirements changes for many systems. This paper proposes the use of the Real Options and Value Driven Design paradigms to the system engineering, design, spiral development and program management processes. The paper also discusses how the use of the Pareto frontier with the purpose of optimizing the performance/cost ratio may actually lead to problems in fielding complex systems. This author has recently begun a research project in Real Options to determine the design tenets that make some systems flexible, adaptable, upgradeable, and scalable, while still reliable; thereby enabling these systems to last many more years than comparable systems. Research results will be discussed in a follow up paper.

9.3.3 Defining, Finding, and Hiring REAL Systems Engineers

E. P. Arnold, *BAE Systems L.P.*

Demand for skilled, knowledgeable, Systems Engineers in government, industry, and academia is increasing around the world. How would a hiring manager know if they are hiring a “real” Systems Engineer, unless the Manager is one, or is thoroughly versed as to what a real Systems Engineer is and does? There are thousands of open requisitions for Systems Engineers, yet little guidance as to how to identify, attract, and hire Systems Engineers. *Defining, Finding and Hiring REAL Systems Engineers* provides guidance as to how to acquire Systems Engineers. It first defines what is meant by a “real” Systems Engineer, discusses the writing of good quality requisitions (Seven Elements of a Great Requisition), provides insight into where to find “real” Systems Engineers, captures what to look for on resumes (resume keywords), provides favourable and less favourable examples of requisition content, and provides interview questions within the framework of Interview Question Steps, with tips as to what to listen for during the interview. Finally, a summary of how to retain the Systems Engineers you found and hired is included.

SESSION 10

Session 10 Track 1: SE Management Process

10.1.1 An Index to Measure and Monitor a System of Systems' Performance Risk

P. R. Garvey, C-C. Cho, *The MITRE Corporation*

This paper extends an earlier published methodology [1] for measuring the technical performance risk of a system to that of a system-of-systems (SoS). The earlier work established an approach for combining an individual system's Technical Performance Measures (TPMs) into an overall measure of performance risk, defined as the Technical Risk Index (TRI). This paper extends this approach so a similar index can be developed for a system composed of many interdependent or connected systems that come together as a whole to provide an SoS capability.

10.1.2 Using Earned Value to Track Requirement Progress

P. Solomon, *Performance-Based Earned Value®*

It is necessary to track the status of each requirement as it moves through engineering life cycle activities. Measures that reflect the status of the requirements are essential to monitor program status and serve as a scorecard to indicate that requirements are being implemented on schedule.

This paper provides guidance to use the tools of requirements traceability to plan and measure the progress of the requirements management activities. The requirements traceability matrix (RTM) can be used as a scheduling source and as a set of base measures of Earned Value (EV). Finally, the importance and value of comparing the schedule variances of the requirements management and tracing activities with the variances of other project activities is discussed.

10.1.3 Reaching CMMI Level 5 is More Than Just Having Adequate Metrics

R. O. Lewis, J. F. Duckworth, *The Boeing Company*

At first glance, the Capability Maturity Model Integration (CMMI) appears to be clearly divided into two advanced levels of maturity, Level 4 dealing with quantitatively managing products and processes using statistical methods, and Level 5 dealing with comprehensive corrective and preventive actions. However, underlying the four process areas, two of which belong to each level, is a closed-loop system that encompasses both levels that if left unrecognized and unused makes reaching and sustaining Level 5 very difficult. This paper reveals and describes this closed-loop system with its intricate feedback loops to the degree necessary for all who aspire to Level 5 to benefit. The paper is based on the actual experience of one of the larger Boeing sites and is written by two of the team that developed this system and then used it to reach CMMI Maturity Level 5 in September of 2005. Those implementing CMMI at advanced levels of maturity must be able to recognize and make good use of this embedded system and the high degree of integration that exists across the model at all levels.

Session 10 Track 2: SE Support Process

10.2.1 The Dual Vee - Illuminating the Management of Complexity

H. A. Mooz, K. Forsberg,
The Center for Systems Management

The Waterfall, Spiral, and Vee models are reminder models that guide us to less perilous paths when developing solutions to problems. Dr. Royce's Waterfall (Royce 1970) provides an orderly approach to software development. Dr. Boehm's Spiral (Boehm 1988) provides emphasis on solving known software risks before proceeding with Royce's Waterfall. The Forsberg/Mooz Vee Model (Forsberg 1991) embraces full systems development by including details of integration, verification, and validation and opportunity and risk management in the symmetry of the Vee development sequence. However, all of these single solution development models fail to address the necessary concurrent development of a system's architecture with the entities of that architecture. The Dual Vee Model introduced in *Visualizing Project Management*, 3rd Edition in 2005, does just that and illuminates the necessary interactions and sequences recommended for orderly maturation of a system and systems of systems. This paper explains the power of the Dual Vee Model when applied as a reminder model for development of complex systems.

10.2.2 IDEF0 Lessons Learned

D. K. Smith, *UGS Corporation*

The IDEF0 methodology is becoming widely used as a tool all types of systems. If a program is considering using IDEF0, there are serious pitfalls to avoid to assure that applying the IDEF0 leads to the desired accurate and complete system definition.

These are illustrated with three case histories to support the lessons learned described. Some lessons apply to other methodologies beyond IDEF0. Although IDEF0 has been in use for many years, it is still being applied to new programs the lessons learned still apply.

10.2.3 On the Use of Semantic Web Technology for Requirements Satisfaction, or How Do I Find a Good Bike?

D. Price, R. Bodington, *Eurostep Limited*

This paper reports on an investigation into the use of Semantic Web technology in the Systems Engineering domain. In 2004 the World Wide Web Consortium standardized the Web Ontology Language (OWL). Software tools called Reasoners are available that support formal, logical analysis of concepts in a subset of OWL called OWL DL (for Description Logic). An approach to the use of this technology for the Systems Engineering task of identifying Systems that satisfy a set of Requirements is described.

Session 10 Track 3: Modeling Tools

10.3.1 The "Big Navy" Meta-model as a Framework for Major Defense Development Projects

C. M. Ryder, *Johns Hopkins University APL*

Patterns, frameworks, meta-models: the utility of these tools for many systems architects and engineers is to avoid "heavy lifting". Others will say that they use meta-models to keep from "reinventing wheels." Builders and structural engineers have utilized patterns, frameworks and meta-models which include building codes and construction standards for thousands of years.

In the realm of defense systems engineering, the “Mystical Land of DoDAF,” systems engineers have been slow to realize the full potential of meta- model applications. Instead, functional analysis teams performing work prescribed by the Joint Capabilities Integration and Development System (JCIDS) end up doing that “heavy lifting” in the course of developing JCIDS functional area, needs and solutions analyses. Analysis teams performing JCIDS Functional Area Analysis (FAA) assess hundreds of activities listed in the Universal Joint Task List (UJTL) for relevance to their functional domain. This paper does not intend to discount the importance of the review of the tasks and activities required in the FAA, yet it is highly desirable to utilize the precedent foundations provided by those functional area analysts who blazed the trail for predecessor systems currently operating within their operational domain. A systems engineering meta-model provides such a foundation for a living engineering model from operational needs analysis through production and deployment. The framework described below, the “Big Navy” Meta-model, is an example of one pattern that can be applied to facilitate the development of complex defense-related systems. Systems models derived from Big Navy can be further refined over its life cycle to include requirements capture, functional analysis, architecture development, concept and engineering design and design validation which includes modeling and simulation.

10.3.2 An Enterprise Systems Engineering Model

R. S. Swarz, G. Rebovich, Jr., J. K. DeRosa,
The MITRE Corporation

The INCOSE (ANSI/EIA 632) systems engineering model may not be adequate for an enterprise environment, in which environmental influences and the interaction of people, processes, and technology are not taken into account. This paper considers some “systems thinking” concepts and applies them to develop an expanded model that may be more appropriate in the enterprise environment.

Session 10 Track 4: Specialty Engineering

10.4.1 Certification & Accreditation: The Role of Security Engineering in the Systems Development Life Cycle

J. S. Tysenn, *Harris Corporation - GCSD*

Security is most effective if it is planned and managed throughout an organizations Systems Development Life Cycle (SDLC). Many security risks, analyses, and events occur during a systems or applications lifetime and these issues should be dealt with from the initial planning stages and continue through all phases of the SDLC. Many systems today must also address Certification and Accreditation (C&A) before going operational. C&A facilitates including security as an element of the SDLC assuring that a clear set of Security Requirements is developed and implemented, residual risk is minimized and clearly understood, and all security controls developed and deployed in the final operational system are documented in a System Security Plan. This paper examines the role of security engineering and their activities throughout the SDLC and the C&A process, the guidance that helps define the Security Requirements, and the roles of the people involved to provide a basic understanding of security engineering as it applies to C&A throughout the SDLC. One of the objectives of this paper is to help Program Manager’s understand and take into account the C&A process into program plans and schedules.

10.4.2 Managing Compatibility Throughout the Product Life Cycle of Embedded Systems: Definition and Application of an Effective Process to Control Compatibility

F. H. Bornemann, S. Wenzel, *3D Systems Engineering GmbH*

Due to the fact of increasing complexity in modern embedded systems, problems associated with compatibility have become an important issue when managing product life cycles. The question of how to maintain and restore the system wide compatibility throughout the entire product life cycle is yet to be answered. So far, compatibility management has not been established as a standard procedure among the traditional and established business areas. Even the CMMI (Capability Maturity Model Integration) [Chrissis, M. B.; Konrad M. and Shrum S. "CMMI - Guidelines for Process Integration and Product Improvement" Carnegie Mellon Software Engineering Institute, (2003 by Pearson Education, Inc.)] does not classify compatibility management as a specific single process area.

This paper gives a first impression of what an effective compatibility management needs to deliver and how it should be integrated into the product life cycle process.

Furthermore, in the following paper process are being defined and applied of how to manage and remedy problems associated with compatibility. Procedures will be described and their functionality will be explained. With the help of that process incompatibilities can be removed and system-wide compatibility can be maintained throughout the entire product life cycle of an embedded system. The paper also shows that without clearly defined processes that describe and help how to remove incompatibilities, maintain and restore system wide compatibility, especially during system updates and upgrades, the time, money and effort spend to secure the functionality of complex embedded systems will reach uneconomical levels.

SESSION 11

Session 11 Track 1: SE Technical Process

11.1.1 Technical Performance Measures

J. Oakes, R. Botta, *BAE Systems*;
A. T. Bahill, *University of Arizona*

Technical performance measures (TPMs) are tools that show how well a system is satisfying its requirements or meeting its goals. This paper presents the BAE National Security Solutions TPM process and a TPM template. It gives lessons learned from piloting it on a program.

11.1.2 Innovative Systems Engineering: A Creative System Development Approach

R. A. Powell, *U S Military Academy*;
D. Buede, *Innovative Decisions, Inc.*

The top-level requirements for a system address the needs and objectives of the stakeholders and serves as a key input to the design process for a system's concept and detailed architecture. In today's systems engineering environment, the process of defining and choosing from multiple concepts is driven by top-level (mission-based) requirements. The use of requirements in engineering systems has gained such importance that an enormous amount of research focuses on various topics related to formulating, defining, and managing requirements. Recently, the US Department of Defense (DoD) and a few other organizations have moved to capability-based systems engineering, which focuses on defining and trading off valuable capabilities rather than requirements.

We have not always relied on requirements to initially define systems. Beginning with goals or defined system capabilities instead of requirements has proven as effective, if not more effective in the design of systems. The use of goals has been proven to produce 1) requirements that can be implemented, 2) minimal changes to requirements throughout system design since requirements are not yet fixed, and most importantly 3) a successful system. The purpose of this paper is to explore the use of goals as a precursor to requirements in developing a DoD military system. Case study analyses, of system development projects occurring as early as the late 1940s, reveal the use of goals as preeminent in developing successful operational systems. Although these cases represent DoD systems, specifically aerospace and missile systems, the concept of the use of goals as a precursor to requirements may generalize more broadly. This paper explores one case: the development of the Sidewinder missile.

11.1.3 An Introduction to Network Centric Warfare

A. Gastelum, *The Boeing Company*;
C. Dagli, A. Miller, *University of Missouri-Rolla*

This paper gives a short explanation of Network Centric Warfare, some background information and technology that makes this a reality. It also provides an example of how network centric solutions can be used to do a kill Chain and its advantages over platform centric scenarios. NCW faces a lot of challenges to System Engineers interoperability, increased complexity, fault tolerance, distributed computing are all needed to be understood to make NCW a reality. The nature of network centric makes us focus also on wireless networks and how they can be used, its advantages, protocols, and later on two frameworks implemented by Sun, Jini

an Jxta. Each has its own set of protocols, procedures and ideology, and both are being used to create very reliable networks with changing topology.

Session 11 Track 2: SE Management Process

11.2.1 Challenges in the Application of Systems Engineering Principles to the Design of Appealing Consumer Products

V. Agouridas, J. R. Longstaff, T. Childs, A. McKay,
University of Leeds

This paper reports results of research carried out in the context of a co-ordination action funded by the European Commission that dealt with the gathering and evaluation of state of the art tools, best practice methods and techniques deployed by leading manufacturers in developing appealing consumer products. Data was gathered through questionnaires and semi-structured interviews. Analysis of the data showed that the design of appealing products involves an intricate web of relationships amongst understanding consumer needs, determining and verifying product characteristics, and validating products that excite consumers. The need for a systems engineering approach in rationalising such an intricate web of relationships is described based on the research findings. A number of challenges associated with the implementation of such an approach are discussed.

11.2.2 The Systems Project: Life Cycle Development/ Management of as Many as Four Interrelated Systems

A. S. Paul, C. Owunwanne, *Howard University*

Work presented at this Conference previously proposed the Producing System (or enabling system) as a mechanism for organizing and managing the enabling resources in the development of a Product System (Alias, required system, system-of-interest, intervention system, end product, operational system, etc.). It presented the producing system as separate from, but closely related to, the product system. Specifically, the relationship was equated to that of a parent: child (producing system: product system) relationship. (Paul and Yerace 2004)

This paper explores current practices regarding the identification and processing of systems that interact with, or have some form of involvement in the life cycle of, the product system. The primary focus is, of course, the interrelationship between the product system and the enabling resources or producing system; however, studies have identified diverse interactions with competing systems, context systems, collaborating systems, containing systems, and others. This paper identifies the similarities and differences among the various perspectives.

Finally, the paper expands on these concepts by introducing the Systems Project as a family of up to four closely related systems whose interactions are focused on the life cycle (birth to death) of the product system. Specifically, the Systems Project is a framework for packaging and conducting all of the systems engineering activities associated with developing/managing the product system, the producing system, the Existing System (if there is one in place, if applicable [I/A]), and the Maintenance and Support System through the life cycle of the product system.

11.2.3 Integration Challenges of Complex Systems

B. R. Haskins, J. M. Striegel, *The Boeing Company*

Integration of complex systems normally requires many different components, subsystems and systems. Commonly referred to as system of systems (SoS), these many different systems are too complex for any one company to develop. Current examples are the international space station and network centric operation programs. Since complex systems require many different companies, challenges arise in effective communications and understanding of requirements, interfaces, validation and verification of integrated systems. This paper addresses the foundation of integration, which is a clear understanding of validation and verification activities at all levels of development, including the key interfaces between systems, subsystems and components. Equally as important is the ability to work key issues during the development process. Attention to detail in both of these aspects is the key to having a successful integration of complex systems.

Session 11 Track 3: SE Support Process

11.3.1 How Good Is A Process? Evaluating Engineering Processes Efficiency

T. S. Gilb, *RPL*

What is 'best practice' for an engineering process? How good is your current set of development, maintenance and service processes? How can we decide exactly which processes we are going to adopt in our organization, for example in a CMMI implementation?

It is the assertion of this paper that such questions are often dealt with without explicit and quantified regard to the full set of real, and well-defined business needs, as well as often not taking into consideration the current processes and the issues of changing them. We too often carry out and change processes because we are told to, not because there is a clearly defined need to do so.

11.3.2 Using Fuzzy Decision Support to Compare Systems Modelling Tools

E. Rajabally, S. Whittle, *Systems Engineering Innovation Centre*

The use of modelling in systems engineering is growing both in terms of model complexity and the importance attributed to model results. In accepting a model as fit-for-purpose, there are numerous considerations or criteria to be fulfilled and this acceptance is complicated by a number of issues, as reviewed in this paper. Consequently, model evaluation, comparison and selection may benefit from formal Multiple Criteria Decision Support (MCDS) and this benefit is explored here through application to a systems modelling case study. Specifically, an existing MCDS technique based upon fuzzy logic is used to compare five aerodynamics tools across fifteen key criteria in a typical context. An overview of the case study is presented in this paper and several benefits of fuzzy MCDS are identified. Despite the considerable effort required from model experts to conduct such MCDS, a more rigorous means of comparing models is provided than typical ad hoc efforts.

11.3.3 Enabling Measurement-Driven System Development by Analyzing Testing Strategy Tradeoffs

R. Selby, *Northrop Grumman Space Technology*

Measurement-driven system development focuses on using quantitative data to evaluate capabilities, benefits, progress, and tradeoffs as well as identify improvement opportunities. This paper describes a controlled study that addresses software testing effectiveness and focuses on the combination of individual testing techniques into team-based testing strategies. This analysis is intended to enable measurement-driven process improvement by characterizing how testing effectiveness relates to several factors, including testing strategy, software type, and developer expertise. In this study, a representative group of software development professionals applied common testing techniques to different types of software. This study compares the six possible team combinations of three testing techniques: (1) code reading by stepwise abstraction, (2) functional testing using equivalence partitioning and boundary value analysis, and (3) structural testing using 100% statement coverage criteria. Thirty-two professional developers applied the techniques to three unit-sized programs in a fractional factorial experimental design.

The major results of this study are the following.

- The six combined testing strategies detected 17% more of the programs' faults on the average than did the three single techniques, which was a 35% improvement in fault detection.
- The highest percentages of the programs' faults were detected when there was a combination of either two code readers or a code reader and a functional tester. However, a pairing of two code readers detected more faults per hour than did a pairing of a code reader and a functional tester.
- The pairing of two individuals of advanced expertise resulted in the highest percentage of faults being detected.
- The most cost-effective (number of faults detected per hour) testing strategy overall was when code reading was applied by an individual. The most cost-effective combined testing strategy was when a code reader was paired with either another code reader or a structural tester.
- Both the percentage of faults detected and the fault detection cost-effectiveness depended on the type of software being tested.

In conclusion, we outline future research directions that build on these strategies and ideas.

Session 11 Track 4: Modeling & Tools

11.4.1 Generating Predictive Models Using Decision Trees and Neural Networks for Large-Scale Systems Engineering

R. W. Selby, *Northrop Grumman Space Technology*

Systems engineering must tackle the challenges of computational systems that are increasingly large-scale and software-intensive in terms of system size, component breadth and maturity, and development heterogeneity. This research describes and empirically evaluates techniques for generating predictive models for enabling large-scale system development and management. We describe two types of metric-driven decision models, decision trees and neural networks, which classify software components in large systems according to their likelihood of having user-specified properties such as high fault-proneness or high development effort. The metric-driven decision models enable

coarse-grain analysis of large-scale multi-component heterogeneous systems, and they identify high-payoff areas for directing the application of fine-grain analysis techniques for fault detection or redesign. The decision models serve as metric integration mechanisms that enable the synergistic use of numerous metrics simultaneously and integrate measurements collected by development tools or infrastructure. Model generation techniques automatically generate the decision models to calibrate them to new projects and organizations.

We evaluate the predictive effectiveness of the decision models in terms of correctness, consistency, and completeness using fault and effort data from large NASA systems. Correctness is defined as the percent of components correctly identified, consistency is defined as 100% minus the percent of false positives, and completeness is defined as 100% minus the percent of false negatives. On average, the decision models had 83.44% correctness, 71.96% consistency, and 65.25% completeness in predictions of high fault and high effort software components. The network models had 89.63% correctness, 79.49% consistency, and 69.09% completeness, while the tree models had 77.25% correctness, 64.42% consistency, and 61.40% completeness. Non-parametric ANOVA comparisons showed that the network models were statistically more accurate than the tree models ($\alpha < 0.0001$).

11.4.2 Top 40 Systems Engineering Work Products from Phrase Lists and Self-Organizing Maps

D. G. Beshore, *The Aerospace Corporation*

A computer-automated method using semantic web algorithms is described that ranks the best systems engineering work products on websites and corporate repositories. The process is both server-based and PC-based to automate the capture and ranking of knowledge from industry experts within The Aerospace Corporation. A virtual Process Assets Library system (vPALs) economically collects and self-organizes documents through long lists of phrases (phrase vectors) using next-generation web engines. Ranking documents implements scoring algorithms with phrase vectors in the development of 1-D, 2-D, and 3-D self-organizing maps (SOMs). vPALs is being deployed in 1) real-time decision making during launch vehicle systems testing and day-of-launch at Cape Canaveral and Vandenberg, 2) proposal technical evaluations, 3) data mining and maintenance of corporate document repositories, 4) formal design reviews with customers, and 5) web-based systems and software engineering books of knowledge. Nine websites are used in this paper to rank order the most relevant documents related to SE workproducts: NASA-PBMA FAA-ASD, FAA-IPG, SEI, SEPO-SPAWAR, SMC, DAU, STC and Aerospace's AeroLink corporate document repository.

11.4.3 Advancing the Canadian Capability Engineering Approach

C. Lalancette, M. Lizotte, C. Nécaille,
Defence R&D Canada-Valcartier;

W. Robbins, B. Waruszynski, *Defence R&D Canada-Ottawa*

The Collaborative Capability Definition Engineering and Management Technology Demonstration (CapDEM TD) project is investigating Capability Engineering (CE) in order to support the Capability-Based Planning (CBP) decision-making process. This paper presents the evaluation effort being applied to the Capability Engineering approach, a methodology based on the systems engineering paradigm and articulated around three axes: People, Process and Materiel. Originating from the laboratories of Defence R&D Canada, the CapDEM effort is now evolving from theory into practice, based on an on-going evaluation strategy concretely realized via three validation exercises that consist of realistic simulations of people applying the process and materiel to resolve capability gaps. This paper details the evaluation strategy and includes lessons learned from the first and second validation exercises. The tenets and expectations from the final exercises will form the final part of the paper, with the intent of providing insight into the advancement of capability engineering.

KEY RESERVES

KR.01 Universal Electronic Health Record: Just what the doctor ordered

R. Shahi, *CTIS, Inc*

The time has come for health informatics to unify; this call for integration will help in better patient care across various systems spread globally. This feat requires standardization of health related information. The implementation of Electronic Health Record (EHR) is such standard which will facilitate collaboration of disparate systems. This emerging standard proposes to digitally accumulate patient's entire healthcare information and allow real time effortless exchange between different care providing sites or individuals. This will result in non redundant patient data accessibility (Any time, any where, any device) resulting in improved efficiency and quality of patient care. The realization of this effort requires different standards organizations like Health Level 7 (HL7), Clinical Data Interchange Standards Consortium (CDISC) and European Committee for Standardization (CEN) to collaborate and produce standards that are interoperable between different organizations. This collaborated effort will result as building blocks for foundation of neural health care network. This state of the art health informatics grid has to overcome barriers in technical implementation, health care domain, globalization, and education and training before realisation of the dream to have an integrated EHR.

KR.02 The Architecture of Enterprise Architecture

T. Blevins, *The MITRE Corporation*

The Architecture of Enterprise Architecture paper answers questions such as; what makes up enterprise architecture? where is enterprise architecture relative to other enterprise processes? and what is necessary to make enterprise architecture work in the complex environments where it is deployed? This paper presents a categorization of architecture offerings that can be used to position and describe the architecture space and provides summarized and supportive information to the main thesis including architecture in the value chain, the social framework in which architecture exists, the skills required for success, and a view of the roles and responsibilities of architects.

KR.03 Foundational Systems Engineering (SE) Patterns for a SE Pattern Language

J. J. Simpson, M. J. Simpson, *System Concepts*

Patterns are classically used to effectively capture large, complex bodies of information and knowledge. This paper outlines the current state of systems engineering (SE) pattern literature as well as proposing a foundational set of SE patterns for use in SE pattern languages. A pattern is defined as a solution of a specific problem placed in a specific context. A pattern language is a collection of interrelated patterns, with specific relationships binding individual patterns together. A collection of SE pattern relationship types are also introduced in this paper. Language is a fundamental aspect of any technical or scientific activity. The technical language used in technical activities impacts the concepts that can be addressed, the effort to express these concepts, and the precision with which these concepts can be expressed. Therefore it is important to consider the language requirements for conceptual expression and precision when designing a new language. In the case of an SE pattern language, the elements to be considered are the individual patterns and the relationships between the patterns.

KR.04 Preparing SEBoK for Korea Defense Systems Acquisition Processes

M-D. Han, *Korea Third Military Academy*

Just as INCOSE tried to set up G2SEBoK, Korea Council on Systems Engineering is now trying to set up its own Systems Engineering Body of Knowledge so that its members and other people related to Korea Defense Acquisition Processes can share a common perspective about what is SE, what is the core of the SE processes, How we do it, etc. Here is the report of the first draft of KCOSE SEBoK list of contents with some description of how we approach this problem.

KR.05 Integrated Project Control Loop Concept - Surviving in the Jungle of Complex Projects with an Advanced Project Management Decision Basis

T. Laudan, A. Mauritz, *EADS Germany GmbH*

The purpose of this paper is to provide a systems engineering approach in project management elaborated in the environment of a complex European aeronautical research project. This paper introduces to actual research activities in the area of preventive risk avoidance and proactive risk handling mechanisms in complex projects based on an Integrated Project Control Loop (IPCL) concept, as a means supporting the Project Management (PM) decision making throughout the project life cycle. First experiences from this European project provide a potential applicable and tool-supported ICPL concept enabling continuous improvements for PM. The IPCL concept consists of uncertainty management in terms of a continuous support to monitor project performance and to steer project functions (cost, time, quality, scope and organization) and the referring requirement profile based on an information integration framework (tool integration concept). The integrated information framework allows the technology based implementation of distributed information to specific PM information models on a centralized level. As a result of the concept outlined above, quantitative and qualitative deciding factors for adequate contingency allowance are exposed and provide a better understanding and situation awareness for PM.

KR.06 Balancing Cost and Performance During Design

D. M. Cronin, *Cognition Corporation;*

K. Ash, *Raytheon Company*

Design and Production can take advantage of new developments in Critical Parameter Management (CPM) to manage the relationships between parts, assemblies, function, and cost. Elements in a WBS/BOM structure can be linked to function, cost, and performance definitions and viewed in a critical parameter tree. Through a series of transfer functions, a dynamic relationship is created to compute and track the results of parameter changes on cost and performance. Design For Six Sigma (DFSS) related techniques are also supported to allow engineers to make these connections and track the cost impact of designs based on performance decisions. This presentation will define the interconnected architecture including customer inputs, system requirements, critical parameters, the effect of parameter variations, transfer functions and the effect of multiple inputs on cost and affordability.

KR.07 A Prioritization Process

R. Botta, *BAE Systems*; T. Bahill, *University of Arizona*

A prioritization process has been developed and used at National Security Solutions of BAE Systems. It has been used to derive weights of importance for the criteria in tradeoff studies and to prioritize goals, customer needs, capabilities, risks, directives, initiatives, issues, activities, requirements, technical performance measures, features and functions.

KR.08 Development Program Risk Assessment based on Utility Theory

J. Z. Ben-Asher, *Technion*

A methodology for risk assessment is introduced based on Utility Theory. The main idea is to quantify the risk consequences by using an equivalent lottery where the probability of getting the worst possible risk outcome is taken as the value representing the risk consequences. This improves the evaluation process by making it more systematic, and justifies the use of the expected value for obtaining the risk factor. The idea is compared with other methods and is demonstrated by a case study.

KR.09 Improving the Structure and Content of the Requirement Statement

W. R. Scott, J. Kasser, X-L. Tran, *University of South Australia*

This paper discusses the perennial problem of poor requirements and summarises an attempt to mitigate the problem using an object-oriented approach by developing and using a software tool named Tiger Pro. Next the early results of using Tiger Pro in the classroom are discussed. The major outcome was the transition of the classroom discussion from a focus on the structure of the requirement sentence to a focus on the difficulty of writing a good requirement, or a focus on the content of the requirement sentence.

The paper then considers the distinction of structure and content of a requirement and the levels of examination. This can be shown to classify the attributes of a good requirement. The classification of the attributes of a good requirement reveals that the content of an individual requirement is the focus of examination, but requirement management tools focus on the structure of the requirement set. The paper concludes by proposing a structure to capture the information in a requirement that facilitates the writing of better requirements.

KR.10 Cost Models with Explicit Uncertainties for Electronic Architecture Trade-off and Risk Analysis

J. Axelsson, *Volvo Car Corporation and Mälardalen University*

This paper discusses how the early phases of developing embedded electronic systems can be improved by enhanced modelling of cost and performance that includes explicit handling of uncertainties. The approach is to add cost information to existing UML models, capture uncertainties using probability distributions, and use Monte Carlo simulation to analyze the risk of not reaching the cost targets. It is demonstrated how the information obtained can be used when evaluating different architecture alternatives, while including both development and product cost as well as risk in the trade-off.

KR.11 Capturing Total System of Systems Costs Using SEER-H with Total System Vision (TSV) and SEER-SEM: An Example Application

J. P. Falque, *Galorath Incorporated*

This paper discusses how a parametric life cycle cost model (LCC), SEER-H with Total System Vision (TSV) and SEER-SEM software estimation model, are used to facilitate the cost impact assessment of Systems Engineering labor and materials for a Major System of Systems project containing both hardware and software components. The main purpose is to introduce the underlying methodology and demonstrate the flexibility of the cost model for developing System Level Costs at the System of Systems Level. Readers are introduced to the model, its premise, and how engineers use it to obtain system level engineering labor and material estimates on a Constellation of 18 earth orbiting Optical Satellites. The system includes the launch vehicles and services, mission control and a world wide tracking antenna sites as well as operational and support costs.

KR.12 Enterprise Analysis and Assessment

J. J. Roberts, *The MITRE Corporation*

As military command and control (C2) applications evolve and become immersed within a growing net-centric operational environment, the complexity of interactions among component C2 enterprise capabilities will rapidly increase. The adoption of a Service Oriented Architecture (SOA) approach to C2 development, coupled with the incremental fielding of Internet Protocol (IP) based Global Information Grid (GIG) capabilities, will serve as the principal catalysts for this scenario. As a result of the sheer number of anticipated web service interdependencies, analysis and assessment of C2 behavior at the enterprise scale will be both necessary and challenging. This paper will provide an initial discussion of the emerging need for such analysis and will suggest some important implementation concepts to be considered, each of which could significantly impact future enterprise systems engineering (ESE) processes.

KR.13 Process for ABET Accreditation in a Systems Engineering Undergraduate Program

P. S. Brouse, *George Mason University*

Accreditation of engineering undergraduate programs is very important to both the university and the students graduating from the program. The Accreditation Board for Engineering and Technology, ABET, is an accreditation organization for college and university programs in applied science, computing, engineering, and technology. The quality of engineering programs has become a fundamental issue in the educational system in the United States. Accreditation confirms that the institution or the program meets the minimum quality criteria. ABET has become more important when licensure is required in many states for engineers to practice. According to information provided in ABET website, "Many state board of professional licensure in engineering and surveying require applicants to have graduated from an ABET-accredited program. In states where non-ABET graduates are permitted to be licensed, an additional four to eight years of work experience may be required." (www.abet.org)

The purpose of this paper is to present the process used by the Systems Engineering and Operations Research Department at George Mason University to accredit our Systems Engineering undergraduate program.

KR.14 Process in Enterprise Systems Engineering

K. K. McCaughin, J. K. DeRosa, *The MITRE Corporation*

This paper addresses the importance of process in enterprise systems engineering. It relates process to the balance between complexity and order in a complex adaptive system, and it extends the rationale for process from efficiency to both efficiency and effectiveness in enterprise adaptation. It develops several enterprise principles based on the works of Ashby, Ackoff, Axelrod and Cohen, Barabasi, Gharajedaghi, and others. It makes no attempt to be mathematical or rigorous, but rather postulates several heuristics. It discusses the deep structure of engineering process.

KR.15 Can Systems Modeling Language Impact Systems Engineering?

K. E. Orr, *The Boeing Company*;

S. Ramakrishnan, C. Dagli, *University Missouri-Rolla*

This paper examines the System Engineering processes/applications transforming from being document-centric to being model-centric by examining the benefits and uses of the Systems Modeling Language (SysML). This paper will look at the concept behind SysML and provide a background on existing research to address the Unified Modeling Language (UML) Systems Engineering request for proposal to generate SysML. This paper will identify benefits of using SysML in the Systems Engineering process and makes a recommendation for its use in large scale integration projects and complex hardware and software designs.

KR.16 Heuristics and Genetic Algorithms

M. D. Mobley, *The Boeing Company*;

C. H. Dagli, D. Enke, *University of Missouri-Rolla*

Genetic algorithms are design tools used in generating optimal solutions. While they can often be shown to outperform various heuristic methods and hybrid approaches, using a combination of evolutionary algorithms and heuristic approaches can generate an optimal solution more quickly than either of the two methods independently. Our purpose is to provide an overview of genetic algorithms, to discuss the types of problems that lend themselves to being solved by genetic algorithms, and to identify heuristics that have been shown to aid genetic algorithms in their quest for optimal solutions. While the sample problems discussed in this paper are generally of textbook variety, genetic algorithms can be applied to problems of interest to systems engineers. Such problems include (1) up-front trade studies to look for potential feasible concepts based on combinations of key system attributes within system constraints and (2) resource selection problems. A military example of a resource selection problem is autonomously recommending air attack resources to prosecute evolving targets. The decision space in this problem is bounded by available fuel, available number and types of weapons, current aircraft locations and current target priority rules of engagement.