Socorro Systems Summit–Collaborative Knowledge Exchange–Oct 28-29, 2016

Co-sponsors: INCOSE Enchantment Chapter and New Mexico Tech Electrical Engineering Department.
Practitioners gathering for knowledge exchange and development on issues of mutual interest.
Location: New Mexico Tech (NMT), Socorro, New Mexico.

Day-1: Choose four from eight ¼-day topic introductions, collaboratively setting objectives for Day-2.
Day-2: Choose two from eight ½-day topic workshops, for developing collective knowledge.

Organizers

- Mary Compton, Sandia National Laboratories, mlcompt@sandia.gov
- Rick Dove, Event Chair, Paradigm Shift International, dove@parshift.com
- Dr. Aly El-Osery, New Mexico Tech Electrical Engineering Department, aly.elosery@nmt.edu

Program Committee (session moderators)

- Ed Carroll, Sandia National Laboratories, ercarro@sandia.gov
- Rick Dove, Paradigm Shift International. dove@parshift.com
- Dr. Celeste Drewien, Sandia National Laboratories, cadrewi@sandia.gov
- Dr. Kevin Forsberg, OGR Systems, kforsberg@ogrsystems.com
- Dr. Regina Griego, Sandia National Laboratories, griegor@sandia.gov
- Jack Ring, OntoPilot, jack@ontopilot.com
- Bill Schindel, ICTT System Sciences, schindel@ictt.com
- Dr. Scott Workinger, Workinger Consulting, scottworkinger@gmail.com

Friday October 28, 2016				
08:15	**General Session**: Welcome and What Will Happen – Aly El-Osery and Rick Dove			
08:45	**Keynote:** Garry Roedler, INCOSE President Elect			
09:15	**Adjust Room Walls for Break-Outs – Coffee Break**			
09:30	**Systems Engineering Cultural Transformation** Topic Intro and Objective Setting for Saturday Workshop Moderator: Ed Carroll		**Agile Security Adaptable to Attack Evolution** Topic Intro and Objective Setting for Saturday Workshop Moderator: Jack Ring	
11:00	**SE as Multidiscipline Enabler / Art / Science** Topic Intro and Objective Setting for Saturday Workshop Moderator: Regina Griego		**Organizational Teaming for Joint Project Pursuit** Topic Intro and Objective Setting for Saturday Workshop Moderator: Kevin Forsberg	
12:30	**Lunch on Your Own**			
13:30	**High Performance Teaming** Topic Intro and Objective Setting for Saturday Workshop Moderator: Celeste Drewien		**Agile HW-Development Infrastructure / ConOps** Topic Intro and Objective Setting for Saturday Workshop Moderator: Rick Dove	
15:00	**Systems of Systems Evolutionary Integrity** Topic Intro and Objective Setting for Saturday Workshop Moderator: Scott Workinger		**Fail-Fast Rapid Innovation Concepts** Topic Intro and Objective Setting for Saturday Workshop Moderator: Bill Schindel	
16:30	**Break**			
17:00	**Reception with Refreshments and Saturday Workshop Posters: Objectives and Choices**			
18:30	**Dinner on Your Own**		**Optional Dinner Gathering** (Separate Advanced Ticket): Empowering Women as Leaders in Systems Engineering	
Saturday October 29, 2016				
08:00	**Systems Engineering Cultural Transformation** Moderator: Ed Carroll	**Fail-Fast Rapid Innovation Concepts** Moderator: Bill Schindel	**Systems of Systems Evolutionary Integrity** Moderator: Scott Workinger	**Agile Security Adaptable to Attack Evolution** Moderator: Jack Ring
11:30	**Lunch on Your Own**			
12:15	**High Performance Teaming** Moderator: Celeste Drewien	**Organizational Teaming for Joint Project Pursuit** Moderator: Kevin Forsberg	**SE as Multidiscipline Enabler / Art / Science** Moderator: Regina Griego	**Agile HW-Development Infrastructure & ConOps** Moderator: Rick Dove
15:45	**General Session:** Eight Brief Outs of Results @ 10 Minutes Each			
17:15	**General Session:** Wrap Up and Open Discussion			
18:00	**Adjourn**			

Collaborative Exchange at the Socorro Systems Summit

Rick Dove, Paradigm Shift International

Workshop abstracts below are suggestions by the moderators, but **workshop participants will own the agenda**. Moderators will instigate discussion with a brief introduction on the first day, and then turn the floor over to participants for convergence on objectives for the second day. Moderators welcome pre-Summit communication – see Program Committee listing for email addresses.

Systems Engineering Cultural Transformation
Ed Carroll, Sandia National Labs.

- A systems engineering culture is an umbrella of shared values and behaviors that transcends the individual cultures of teams, departments, and disciplines—rooted in the appreciation of overarching system concepts and system relationships. Engineering is an ancient discipline, but systems engineering has a history of only a few decades. The primary benefits of systems engineering have been stated as the ability to control complexity, improve communication, and prevent defects. Systems engineering and particularly model-based systems engineering, is often touted as the approach to ensure high reliability from systems that are at the same time becoming more automated, adaptable, agile, and interoperable. These systems tend to also become more complex system-of-system solutions.

  If systems engineering is the approach to control this explosion of complexity and assurance of reliability, then why is the transformation to a systems engineering culture so difficult? It has been said that determining the return-on-investment for a transformation to a systems engineering approach is practically impossible to determine. What, then, is the paradigm shift that needs to happen to implement a successful systems engineering culture? What is required for an effective transformation? What impedes the recognition and realization of value here?

  This workshop will explore individual and organizational challenges that need to be overcome to effect a transformation toward a successful systems engineering culture.

Organizational Teaming for Joint Project Pursuit
Kevin Forsberg, OGR Systems, INCOSE Fellow, ESEP.

- Joint project teaming brings together different organizations with diverse capabilities to satisfy a customer need competitively. An "A" team covers all the project bases with specialty expertise, capability, and experience, presenting no weak spots. Appreciating and seeking the values of joint-team strength can be inhibited by organizational culture, tradition, and politics. Finding appropriate team members that can fill the technical gaps, improve proposal reception, and/or deliver superior results can be problematic under time pressures and the hurdles of new-relationship trust and respect development. There are awesome resources available for A team configurations.

  What are the values of joint-project teaming that can outweigh the obstacles? What are the obstacles? What requirements must be satisfied to encourage and realize beneficial teaming relationships? How might joint-teaming opportunities be enabled and facilitated to compelling benefit?

  This workshop will open the dialog, explore the opportunity, and identify means for advancing the pursuit of organizational joint-project teaming.

High Performance Teaming
Dr. Celeste Drewien , Sandia National Labs.

- A *high performance* team is a group of people committed to a common purpose, who consistently show *high* levels of collaboration and innovation. A high performance team produces superior results and a sense of personal joy in every participant - it takes the work out of work. There is plenty written on the characteristics of high performing teams and high performance teaming. So why isn't high performance teaming very prevalent? Why isn't it a compelling behavior that draws all of us in naturally? Is it a fault of leadership? Or is it a collection of personal issues and systemic organizational conflicts?

  What are the compelling personal values for working in a high performing team and what inhibits an irresistible pull in that direction for all of us? What motivates people to create a team culture of high performance? What personal issues stand in the way, no matter how much it is wanted? Claiming lack of enlightened leadership and corporate strategic imperative is an excuse to live with the status quo. High performance teaming is fueled most effectively by personal desire, personal motivation, and personal initiative, coupled with a trust-filled team environment. Many of us have had the occasion to be a member of a high performing team, but all too occasionally. If you've ever experienced it you know it's fun, rewarding, and memorable.

  This workshop will explore the personal and organizational inhibiting barriers, requirements for a personally-compelling solution, and personal initiatives to make a difference.

Fail-Fast Rapid Innovation Concepts
Bill Schindel, ICTT Systems Science, CSEP.

- Innovation delivers new stakeholder value, and includes discovery of new system configurations—including those which are insufficient or inadequate. The value of well-organized exploration efforts is that they will, on the average, produce higher-value results for a given investment of resources than other approaches. But "Fail-Fast Rapid Innovation" cannot simply mean quickly producing a series of rejected options. The discovery and experimental aspects of engineering are sometimes overshadowed by a belief that engineering proceeds only by syllogistic reasoning from a known place and first principles to a new place, but that is not the nature of innovation, which is itself not always so well understood.

  If we must organize and direct resources into completely unknown territory, what roadmap can we use for planning, budgeting, and scheduling? How can we optimize use of our resources so that these investments are well-justified and understood?

  This workshop will explore the nature and properties of the innovation process as related to effectiveness of experimentation and discovery as key parts of innovation.

Continued on next page

Systems of Systems Evolutionary Integrity
Dr. Scott Workinger, Workinger Consulting.

- Evolutionary integrity in System of Systems is concerned with upgrades to constituent systems during operation and mitigating disruptions that arise from asynchronous and unpredictable changes when independent constituent systems change without warning. Effective integrity management seeks seamless upgrades to constituent systems and the SoS as a whole. Yet, in practice, service-outage windows often don't accommodate major upgrades, and lengthening the outage can create an unacceptable disruption to SoS capabilities. Moreover, self-serving changes in constituent systems can interfere with total-SoS functionality. Even with well-meaning efforts to manage constituent systems, emergent behavior from constituent system interactions can arise unpredictably, creating serious disruptions.

  In general, what are the barriers to integrity management in an SoS composed of independently-owned systems? What inhibits sustained integrity in a complex collection of interacting systems and how can we define integrity for an SoS with no central authority to approve changes? What are the characteristics of a workable integrity management approach? Are these characteristics of effective integrity management represented in examples that we can share? Are there general principles that we can identify and apply to achieve robust integrity management?

  This workshop will explore these questions and others that participants have, with the objective of profiling the issues, converging on a set of general needs that an effective integrity management approach must satisfy, and sharing knowledge and experience on approaches that show some effectiveness.

SE as Multidiscipline Enabler/Art/Science
Dr. Regina Griego, Sandia Nat'l Labs, INCOSE Fellow.

- The branding of systems engineering in many companies and with too many systems engineers is that systems engineering is about developing good process and enabling that process in an organization to achieve systems that are delivered on time, within schedule, and meets requirements. While process is an enabler, it is like the score of music that a good conductor interprets with talented musicians and instruments to deliver a system that not only meets customer expectations, but indeed delights the customer and has an enduring quality. The conductor (systems engineer) knows how to adjust the score for the effect they are trying to achieve and integrates the musicians effectively based on their unique abilities.

  Think about the systems that you are most proud of, or the times that you have been a part of a system development effort that felt exciting, even exhilarating. Would you say they are works of art, or simply science, process, and project management? Systems architecture and design are the most obvious areas where the art of systems engineering is applied, but it is equally important to apply the art of understanding people and teaming. In systems engineering you are working with at least two systems, the system you *are* delivering and the system that *is* delivering. When have you experienced the flow as a systems engineer? How would you characterize systems engineering in your organization: process and project management or a blend of art and science?

  This workshop will explore the art and science of systems engineering and the notion of the systems engineering brand.

Agile Security Adaptable to Attack Evolution
Jack Ring, OntoPilot, INCOSE Fellow.

- Agile security must be reactively resilient and proactively composable at the pace of unpredictable and evolving adversarial attackers and their attack methods. The adversarial attack may originate from outside the system or, particularly in system of system scenarios, from inside the system. This idea encompasses information systems, cyberphysical systems, physical systems, infrastructure systems, and national defense systems. Success demands close collaboration and co-learning by system engineering and security engineering interests. System engineering seeks sustainable systems. Security engineering seeks sustainable system defense. It takes both to succeed against agile adversaries. The respective practitioners march to separate drum beats. Security engineering must educate systems engineers on the kinds and sources of threats and needs for detecting and defeating them. System engineering must satisfy new demands on system architecture, system design, systems engineering, and security engineering. All need to better understand their requisite interoperability.

  What stands in the way of synergistic engineering cooperation? What are the requirements for an effective engineering-team approach? What can systems engineering do to enable and facilitate the needs of agile-security engineers? What can security engineering do to enable and facilitate engagement with systems engineers?

  This workshop will explore values and needs for cooperative agile-security engineering, identify the inhibiting barriers, suggest concepts that any effective solution must address, and open a dialog on potential solutions.

Agile HW-Development Infrastructure/ConOps
Rick Dove, Paradigm Shift Int'l, INCOSE Fellow.

- An agile development infrastructure provides an architectural framework for component interconnect that enables asynchronous, incremental, and iterative component development. An agile hardware-development infrastructure would facilitate asynchronous component testing, alignment with agile software development, demonstrable and testable work-in-process of mixed component releases/prototypes/simulations, and operational system evolution. But hardware development is very different than software development. Agile software development relies on object-oriented infrastructure and web-page hyperlink couplings as architectural underpinnings. Software developers are simultaneously designers and fabricators, and incremental development lends itself to incremental test and demonstration. In contrast, hardware development has issues of tooling; communication between designers, fabricators, and assemblers; and costly re-work.

  An agile approach is beneficial when development occurs under uncertainty, unpredictability, and situational evolution – requiring the application of incremental learning during development. What are the barriers to incremental and iterative hardware development? Can concepts from product-line engineering, open system architecture, or live-virtual-constructive approaches offer guidance? Are proprietary approaches the only avenue, or is there opportunity for affordable common development-platform tools?

  This workshop will explore the values, the issues, and the requirements for possible solutions. ∞