Western States Regional Conference 2024 Technical Program

Presentation and Tutorial List

As of August 14, 2024

Table of Contents

#2 Applying Formal Methods to SysML Models to Prove Correctness and Enable Hallucination-Free LLMs, Jamie Smith

SysML (Systems Modeling Language) is a widely-used formalism and international standard for designing and communicating complex system designs, and is especially relied upon in the development of safety-critical autonomous systems. SysML v2 is a major new version of SysML currently undergoing finalization, scheduled to be released mid-2024. Among many key enhancements, SysML v2 introduces a formal semantics for specifying the precise meaning of SysML models, which makes it possible to apply formal verification and automated reasoning to analyze safety and correctness properties of SysML designs automatically.

At Imandra, we are applying our Imandra Automated Reasoning System to the design and analysis of SysML v2 models. Key to this work is a formal semantics for SysML v2 models expressed in Imandra's logic, and corresponding verification automation for answering deep questions about the possible behaviors of SysML models. In addition to verification, we can then leverage Imandra's "proof-carrying prose" integration with LLMs to obtain conversational assistants (chatbots) which can rigorously answer complex questions about SysML v2 designs, backing every answer up with an independently verifiable logical audit trail produced by Imandra reasoning about the SysML v2 model.

During this talk, we will demonstrate how to use Automated Reasoning with SysML v2 models to:
- Mathematically prove properties of system correctness.
- Add a hallucination-free natural-language interface (chatbot) to empower more stakeholders to gain insights and leverage the SysML v2 model.

#4 Designing Affordable Resilient Systems, Scott Jackson

This work presents the case for the affordability of systems. These cases include transportation, such as commercial aircraft and railway systems. space shuttle, power generation systems. Case studies are presented for each of these categories showing cost to system modification ranging from no cost to moderate cost. Case studies include system type, scenario, adversity, resilience technique, outcome, and estimated cost impact. The conclusion is that these systems and the associated scenarios and techniques demonstrate that system resilience can be achieved with little or no cost impact.

#5 Understanding and Applying the Comprehensive System Design Language (CSDL), Sarah Rudder

This professional development tutorial will explore how the MBSE with CSDL is accomplished and how the integrated system data repository is created and subsequently utilized to create diagrams. CSDL is a structured Entity-Relationship-Attribute (ERA) language with well-documented semantics for each entity, enhancing collaboration by reducing ambiguity within most modeling languages.

The tutorial will go through the construction of an Unmanned Aerial System (UAS) using CSDL, starting with importing requirements through several SE activities. Attendees will learn how to use CSDL for modeling structural decomposition, use cases, and system functionality.

The proposed time for this tutorial is four hours and the topical outline includes:

- Introduction to CSDL
- Benefits of an ERA Language for MBSE
- Importing Requirements
- Grouping Requirements
- Setting up Requirements for Successful Verification
- Modeling System Structure
- Modeling System Use Cases
- Modeling System Functionality
- Modeling Relationships
- Creating Diagrams from the Data Repository
- Report Writing

The UAS design is most relevant to the aerospace and defense industry, but the underlying MBSE principles can be applied to any system.

#6 Mapping CSDL to ISO 15288 Ontology for Model Validation, Sarah Rudder

This talk will show the ongoing research into mapping modeling languages to an ISO 15288 ontology by showcasing the Comprehensive System Design Language metamodel. This metamodel will include each CSDL entity, attributes, and relationships and be mapped to the ISO 15288 ontology to determine if a system model is compliant with the standard.

The ISO 15288 ontology focuses on the processes identified in the standard with the overall goal of seamless data transfer/integration between MBSE languages and tools by relying on a machine readable semantic language.

#8 Systems Engineering Innovation Through Higher Education, Paul White, Nicole Falkenberg and Rainie Ingram

In the INCOSE Vision 35, it states that the digital transformation, innovative technologies, and virtual environments will lead to substantial changes at universities. In response to these trends, Weber State University has developed a Systems Engineering program. The university delivers the program virtually and features coursework that offers INCOSE academic equivalency, senior projects, and partnerships with companies. The university is seeking ways to strengthen our curriculum by incorporating Dassault products, such as Cameo, into coursework.

In this presentation, we will cover the unique aspects of our program and how our program serves as a pioneering model for others to follow. Utah's strong economy, welcoming business environment, and natural beauty provide unique opportunities for Weber State University to provide a high-quality, affordable, and accessible education for all students.

#9 Leading and Taking Charge of a System Engineering Human Integration Team, David Shostak

Human Integration for System Engineering (SE) considers many factors and challenges. This presentation will explain in fine detail and completely lay out all the factors needed for the Human Integration of a System Engineering team.

Here are the key topics that will be presented and discussed:

The System Engineer and the People

The lead System Engineer needs to understand how to plan for the people the SE needs on the project and what tools will be needed to acquire the best people for the SE team. The SE needs to know how to find the people needed. The SE needs to organize a baseline technical core team and structure. What will each person on the SE team's technical responsibility be? What project resource management tools are needed?

How will the SE team be organized and developed? What methods and tools will be used to develop the team? What if virtual teams are involved? How will the virtual team be organized and communicated, too?

System Engineering Fundamentals and Skills

There are System engineering fundamentals the lead SE will need to know and use. What characteristics and roles of a System Engineer will need to be followed? What are the functions of a lead System Engineer in running a team of people? What challenges does a lead System Engineer face? There are many things to consider and think about.

What will make the System Engineer succeed? What leadership style is needed? What kind of leadership styles are there? What conflicts do they have to deal with? What are the successful motivation factors?

In Summary, Leading the Charge and Contributions

Regarding this abstract and presentation, I will present and demonstrate the skills a lead System Engineer needs to consider when leading the charge of his team and what his contribution would be to the project to make it a success. I will explain in fine detail what they are.

Take Aways:
1. How to plan and what to consider in a System Engineering Human Integration team
2. What are the lead System Engineering Responsibilities
3. How to conduct Team Organization
4. What are the System Engineering fundamentals and skills needed for Human Integration?

#10 Time-Delta Method for Measuring Software Development Contribution Rates, Vincil Bishop and Steven Simske

The Time-Delta Method offers a new framework for assessing individual contributions in software development projects by analyzing Commit Time Delta (CTD) and complexity metrics. This method utilizes statistical analysis to measure developer efficiency and effectiveness, proposing a reliable estimator for assessing development efforts and optimizing project management. Our approach integrates established metrics such as Cyclomatic Complexity with innovative techniques for estimating unobserved work durations, providing a pragmatic tool for real-world applications. The validation of this methodology through extensive data analysis confirms its potential to enhance resource allocation strategies in software development.

Key Takeaways:
Understanding of the Time-Delta Method for evaluating developer contributions.
Insights into using statistical techniques to enhance software project management.
Practical approaches for applying complexity metrics to assess software development efforts.

Presenter Background:
Vincil Bishop, a seasoned software engineer and is currently a Cloud Application Architect with a cloud provider. He is a PhD candidate in Systems Engineering at Colorado State University, with a track record of leading innovative software development projects. His expertise includes data science, cloud architectures, and the implementation of complex software systems in various industries.

Dr. Steve Simske is a professor of Systems Engineering at Colorado State University. His research spans AI, machine learning, and systems optimization. Dr. Simske has authored over 500 publications and holds more than 200 patents, primarily in security, imaging, and text analysis technologies. His experience in academia and industry, including a long tenure at HP where he led global teams in developing innovative solutions, is focused on simultaneous academic understanding and practical applications in diverse sectors including healthcare, security, and sustainability.

Relevance to Industry:
This presentation is highly relevant to professionals in software development and systems engineering, providing them with a novel tool for measuring and optimizing developer productivity. The method's application across different industries demonstrates its adaptability and broad utility.

#11 Realizing the potential of SysML V2 with SysON: the Fundamental Role of Open-Source for Enabling the Digital Engineering Transformation, Stephane Lacrampe

Over the past two decades, the field of Systems Engineering has undergone a transformative journey, marked by the emergence of Model-Based Systems Engineering (MBSE). Initially introduced over 20 years ago, MBSE aimed to elevate the role of models as a primary mean for communication and documentation in Systems Engineering, as a way to better manage the growing complexity of our systems.
Still nowadays, the INCOSE SE Vision 2035 underscores the unprecedented stakes faced by the Systems Engineering community. Solving global challenges such as climate change, sustainability in a context of technological explosion and exponential growing complexity demands a fundamental shift in our Systems Engineering practice, with Digital Engineering emerging as a pivotal pillar in this revolutionary journey.

A collaborative effort led by INCOSE and OMG in 2005 resulted in the creation of SysML v1, setting a standard for systems engineering modelling and MBSE. Despite growing interest and adoption of MBSE over the last 15 years, challenges highlighted by studies like the SERC 2020 MBSE Maturity survey underscored the need for a major evolution of the SysML standard to extend its capabilities and overcome limitations. In this context, SysML v2 emerges as a completely reshaped standard and a critical technology, resulting from an extensive collaborative development effort involving over 80 organizations spanning five years.

This forthcoming standard is poised to be a game-changer, promising heightened precision, expressiveness, consistency, usability, interoperability, and extensibility. The Systems Engineering community is now awaiting a new era of tools, enabling seamless and trusted collaboration in a digital engineering environment based on this new standard. The race is on, with major tool vendors striving to bring their SysML V2 tool to the market, each working on their distinctive solution.

Looking ahead five years, the Digital Engineering transformation mandates the integration of SysML v2 tools into every Systems Engineer's daily toolbox, transcending specialized utilities to become as ubiquitous as the Words and Excel of the past—a true commodity. Accessing a SysML v2 environment should be as common as turning on a tap, ensuring accessibility for every System Engineer at minimal cost and great ease of use. At Obeo, we believe that open-source has a major role to play to realize this vision.

Over the last 20 years, the software industry has transitioned to open source for foundational capabilities in software development tools (Python, Java, JavaScript, Git, Jenkins, Docker, SonarQube, to name a few). Pioneering technologies like Jupyter Notebook provide insight into how cutting-edge and open-source innovations can revolutionize the Systems Engineer's experience in terms of accessibility, interoperability, and costs of software tools.

This presentation aims to demonstrate why and how, given the evolving landscape and challenges ahead, it is crucial for the systems engineering community to build a robust, high quality open-source SysML V2 tool delivering the expected foundational capabilities for a new era. This will be illustrated through the SysON open-source project (https://mbse-syson.org/), a web based SysML v2 modelling tool for authoring and visualizing models.

The presentation covers the following topics:
- Exploring the evolving tool landscape with the advent of SysML V2 and why modelling with SysML V2 should become a commodity.
- Explaining why industry-led collaborations should shape the future of Systems Engineering tools, elucidating our vision on why industry participation and commitment are indispensable for the success of our Systems Engineering community amid the development of the next generation of tools.
- Introducing the governance, organization, and dynamics of collaborations in open-source projects and communities, outlining how end-users, academia, and tool vendors can actively participate in shaping product road-maps within community-driven software development.
- Exploring the underlying business models and economic frameworks that sustain these ecosystems
- Sharing insights and lessons drawn from 15 years of experience in developing globally-used open-source modelling tools and communities, exemplified by open-source projects such as Capella or SysON.
- Proposing strategies for how the Systems Engineering community can actively engage, collaborate, and contribute to shaping the future of the SE tools ecosystem.

Author Information:
Stéphane LACRAMPE co-founded Obeo in 2005 in France. Obeo is an independent software vendor with a global reach, leading in open-source modelling software for system and software engineers, enterprise architects, and domain modelling experts, and supporting the open-source MBSE tools Capella and SysON.
Stéphane LACRAMPE acted as the company's CEO until 2018 and is now the director of Obeo Canada. Stéphane LACRAMPE is in charge of developing the Capella and SysON ecosystem in North and South America as well as in Asia. Stéphane LACRAMPE is also the co-chair of the INCOSE Systems Engineering Tools Database Working Group and board member of the INCOSE Canada chapter.

#12 FULL-DAY TUTORIAL: Navigating the Future: Exploring SysML V2 with SysON- A Hands-On Tutorial, Stephane Lacrampe

This submission is made under the category "Tutorial" as a half day tutorial and as an "Advanced Technology Tutorial".

SysML V2 is a general-purpose modelling language, standardized by the OMG, for specifying, analyzing, designing, and verifying complex systems that may include hardware, software, information, personnel, procedures, and facilities (source: https://www.omgsysml.org/). It is the result of an extensive collaborative development effort that began in 2015 and involved more than 80 organizations. It is also the eagerly anticipated successor to SysML V1, addressing some of its limitations, and poised to play a key role in the Digital Engineering transformation in the years to come.

SysML V2 represents a significant evolution compared to SysML V1, including:
- It is no longer based on UML but on a new metamodel (KerML).
- It provides both textual and graphical syntax.
- It defines a standard API for accessing the model.
- It systematizes the concepts of definition and usage and provides variability capabilities.
- It provides enhanced extension capabilities and includes a large set of predefined model libraries.

Anticipating the adoption of the final specification in 2024, this tutorial aims to provide participants with a first experience in using SysML V2. No previous knowledge in SysML is required.

After an initial overview of SysML V2 key concepts and innovations, this tutorial will consist of an interactive hands-on experience where participants familiarize themselves with modelling in SysML V2. They will develop their own SysML V2 model through a system example used throughout the tutorial, covering the key features and functionalities of SysML V2.
This hands-on session will include Structure and Requirement modelling using the General view and Interconnection View, Behavior modelling using the Action Flow View and the State Transition View. We will also cover topics such as extending SysML V2 with your own library and how such extensions can be used. While we will provide an overview of the textual syntax, most of the exercises will be done using graphical representations. This tutorial will not focus on comparing or transitioning from SysML V1 to SysML V2 but rather on learning SysML V2 itself.

Join us on this journey to navigate the future of systems engineering with SysML V2.
Participants should come with their laptops as well as an internet connection (provided by the conference). The SysON open-source project (https://mbse-syson.org) will be used to support the

modeling activities conducted in this tutorial. No installation is required.

Author Information:
Stéphane LACRAMPE co-founded Obeo in 2005 in France. Obeo is an independent software vendor with a global reach, leading in open-source modelling software for system and software engineers, enterprise architects, and domain modelling experts, and supporting the open-source MBSE tools Capella and SysON.
Stéphane LACRAMPE acted as the company's CEO until 2018 and is now the director of Obeo Canada. Stéphane LACRAMPE is in charge of developing the Capella and SysON ecosystem in North and South America as well as in Asia. He is a regular speaker in Systems Engineering conferences like INCOSE IS or in events organized by local North American chapter, and has already delivered online and on-site tutorial on Capella. Stéphane LACRAMPE is also the co-chair of the INCOSE Systems Engineering Tools Database Working Group and board member of the INCOSE Canada chapter.

#13 Leveraging AI for Enhanced Systems Engineering, Rick Hefner

The integration of Artificial Intelligence (AI) into Systems Engineering processes has emerged as a pivotal paradigm shift. Through real-world case studies and discussions, this presentation will provide systems engineers with the knowledge, tools, and insights necessary to harness the transformative power of AI for enhanced Systems Engineering.

Attendees will learn how incorporating AI components into products can enhance functionality, efficiency, and user experience. Examples include:
1. Personalized Recommendations: E-commerce platforms like Amazon and streaming services like Netflix use AI algorithms to analyze user behavior and preferences.
2. Enhanced Customer Service: Many companies integrate AI-powered chatbots and virtual assistants into their products to provide round-the-clock customer support. These AI components utilize natural language processing (NLP) and machine learning to understand and respond to customer inquiries, resolve issues, and provide relevant information promptly and efficiently.
3. Predictive Maintenance: In industries such as manufacturing, aviation, and energy, companies deploy AI-based predictive maintenance systems to monitor equipment health in real-time. By analyzing sensor data, historical maintenance records, and environmental factors, these systems predict potential equipment failures before they occur, enabling proactive maintenance and minimizing downtime.
4. Improved Healthcare Solutions: Healthcare companies are integrating AI components into medical devices, diagnostic tools, and healthcare systems to improve patient care and outcomes. AI algorithms are used for medical image analysis, disease diagnosis, personalized treatment recommendations, and drug discovery, accelerating the development of innovative healthcare solutions.
5. Autonomous Vehicles: Automotive manufacturers are incorporating AI components such as computer vision, sensor fusion, and deep learning into autonomous vehicles. These AI systems enable vehicles to perceive their surroundings, navigate complex environments, and make real-time decisions to ensure safe and efficient autonomous driving.
6. Financial Services: In the financial industry, companies utilize AI components for fraud detection, risk assessment, and algorithmic trading by analyzing transaction patterns and user behavior to identify suspicious activities in real-time. Machine learning algorithms predict market trends and optimize investment strategies for better returns.
7. Content Creation and Curation: Media and entertainment companies employ AI components to automate content creation, curation, and personalization.

AI methodologies, techniques, and tools can also be used to improve Systems Engineering practices:

1. AI-Enabled Requirements Engineering: AI-powered techniques such as natural language processing (NLP) and machine learning (ML) facilitate the elicitation, analysis, and validation of system requirements, ensuring alignment with stakeholder needs and objectives.
2. Intelligent Design Synthesis: AI algorithms, including genetic algorithms, neural networks, and evolutionary computation, expedite the iterative process of system design synthesis, enabling rapid exploration of design spaces and identification of optimal solutions.
3. Predictive Analytics for System Performance: AI-driven predictive analytics and simulation modeling is being used to forecasting system performance, reliability, and resilience under diverse operating conditions and unforeseen scenarios.
4. Autonomous System Verification and Validation: AI is being used in automating verification and validation processes, encompassing techniques such as model checking, automated testing, and anomaly detection, to enhance system safety, security, and compliance.
5. Cognitive Decision Support Systems: AI-powered decision support systems leverage data-driven insights, expert knowledge, and probabilistic reasoning to assist systems engineers in making informed decisions throughout the system lifecycle, mitigating risks and optimizing resource allocation.

By leveraging AI algorithms and technologies, companies can deliver personalized experiences, improve operational efficiency, and drive innovation in their products and services.

#15 HALF-DAY TUTORIAL: Requirements: A Comprehensive Overview, Rick Hefner

This presentation delves into the use of systems engineering tools for defining and enhancing business processes. By drawing parallels between systems engineering in product development and process design, it explores the commonalities in the artifacts and methodologies employed across both domains.

In the realm of business processes, adopting a systems approach enables a thorough grasp of organizational dynamics, facilitating the identification of inefficiencies, bottlenecks, and avenues for enhancement. The systems engineering (SE) methodology emerges as a versatile framework for structuring, analyzing, and optimizing systems, whether they pertain to tangible products or operational workflows. Key artifacts like stakeholder needs, functional requirements, operational behavioral diagrams, and system architectures find leverage through established methodologies and tools.

Furthermore, the integration of systems thinking with Lean Six Sigma methodologies, particularly the DMAIC (Define, Measure, Analyze, Improve, Control) and DMADV (Define, Measure, Analyze, Design, Verify) frameworks, offers additional avenues for refinement. Through practical examples, this presentation illustrates how this integration enriches the process design paradigm.

#16 HALF-DAY TUTORIAL: Mastering Your Systems Engineering Competencies, Rick Hefner

To excel in systems engineering, professionals must possess a diverse set of competencies. This tutorial provide a thorough and practical guide for professionals looking to develop the competencies outlined in the INCOSE Systems Engineering Competency Framework. It offers a structured approach that encompasses both theoretical knowledge and practical application, catering to individuals at various stages of their careers, from novice practitioners to seasoned experts.

The framework identifies a wide range of competencies, organized in the following categories:
- Core Competencies: Systems Thinking; Lifecycles; Capability Engineering; General Engineering; Critical Thinking; Systems Modelling and Analysis
- Professional Competencies: Communications; Ethics and Professionalism; Technical Leadership;

Negotiation; Team Dynamics; Facilitation; Emotional Intelligence; Coaching and Mentoring
- Technical Competencies: Requirements Definition; System Architecting; Design for…; Integration; Interfaces; Verification; Validation; Transition; Operation and Support;
- Management Competencies: Planning; Monitoring and Control; Decision Management; Concurrent Engineering; Business & Enterprise Integration; Acquisition and Supply; Information Management; Configuration Management; Risk and Opportunity Management
- Integrating Competencies: Project Management; Finance; Logistics; Quality

Participants will assess their own skills against the framework and identify areas for improvement, for both the current roles and career goals. Then practical guidance will be provided on how to acquire new skills and improve existing ones. By the end of the tutorial, participants will be equipped with the knowledge, tools, and mindset necessary to enhance their systems engineering capabilities. Whether embarking on a new career path or seeking to advance in their current role, participants will emerge from the tutorial empowered to tackle the most challenging systems engineering problems with confidence and competence.

#17 Model Based Document Generation, Dennis Shen, Hadi Malik and Michael Topolski

Model-Based System Engineering (MBSE) is an applied systems engineering methodology aimed at intended to replace traditional document-centric systems engineering approaches with a data-centric model that serves as the "single source of truth". Wide-scale adoption of MBSE still faces opposition due to users' inexperience with the Systems Modeling Language (SysML) and client software, access to the MBSE model, or confusion about how a model adds value throughout the life-cycle for each functional area. These issues, when coupled with the lack of a data-centric approach and human error, lead to inaccurate or incomplete information being delivered. This style of documentation leads to problems over the lifecycle of the system, and these problems are amplified when documentation is not routinely updated, leading to unplanned costs over the lifecycle of the system. These problems often culminate in systems having issues with unknown functions, capabilities, and maintenance. This paper will address these problems by providing model-based approaches to document generation.

As a means to address the aforementioned concerns modern approaches to model-based document generation have been created. These approaches provide a recognized format, such as interface control documents, system requirements documents, or drawing packages, that can be easily understood and are familiar to stakeholders. These documents output from the model allow stakeholders to get more involved in the modeling process by giving them a means to interact with the model elements without the barriers to entry associated with setting up and accessing a modeling environment. In addition, the process allows the stakeholders to familiarize themselves with the model structure so they can ease into SysML and eventually gain enough experience to interact with the model directly. The document generation process can also be integrated into organizational processes, such as an agile workflow, to ensure that documentation reflects the most up-to-date material. Provided that the model is updated over the course of the system lifecycle, minimal effort is required to ensure that documentation is concurrently updated. This benefit is not the only reason to adopt this approach; organizational processes can be built to include the structure of document generation in the model workflows to save time and cost. The structure that goes into document generation can be reused for future efforts, streamlining the process. This benefit is especially true of regulated formats such as Department of Defense (DoD) Data Information Description (DID) templates, and it opens up the possibilities of future contractual DID requirements being satisfied using a model-based approach.

GTRI has leveraged Cameo Systems Modeler's (CSM) built-in interfaces to implement multiple methods to generate documents from MBSE models. The first approach is through Apache's Velocity Template Language (VTL) in conjunction with CSM's Report Wizard to pull the data from the model into pre-made templates. This approach has been best utilized in the DID format, and leverages pre-existing model structure. The second approach is the use of CSM's Java API's to access the model and output it to any desired format. This has been utilized in more flexible templates to achieve reusable code that aims to provide updated test documentation. As consistent with systems engineering, both of these methods require some initial upfront investment of time and effort to build. This initial effort will pays dividends in the long run, as the reuse of either method can be as quick as pressing a few buttons for a new document to be generated. The time to generate documents can be spread out in the system life cycle with the proper organizational and technical processes in place, such as (but not limited to) DevOps workflow to automate processes. These two methods, while very different, both carry benefits and have situations where they are applied best. In this paper, GTRI will discuss the experience in using these methods to generate documentation along with the lessons learned from the process to help streamline inefficiencies.

While the MBSE approach was originally created to improve upon the inefficacies of document clutter, a real world need for documents still exists. In this new era, documentation is quickly transitioning from being the "single source of truth" to a fractional representation of system architecture. This need does not detract from MBSE, rather it offers a unique and agile opportunity for MBSE to fill. In this space CSM provides multiple routes to work within an integrated system that can generate documents to fulfil that need while still advocating a model-based approach for systems engineering. These avenues allow the MBSE model to remain the "single source of truth", while the documents can be generated iteratively throughout the lifecycle of the system to reflect the truth. This analysis of these benefits allows for better leveraging of the document generation process and ensures that the process can become organic within broader organizational and technical processes.

#18 Digital Transformation Challenges: Real-World Observations and Mitigations, Stephen Guine

(Hint: The digital part is not the hard part)

As industry presses onward with the transformation to digital-based operational frameworks in engineering, project management, and operations across domains, one feedback signal is clear: it's not quite as easy as the advertisements make it look. Yes, the value propositions are real and obtainable. The challenge is that organizations must have the right frames of reference and be prepared to tackle not only the technology, but also the leadership, program management, organizational development, and process evolutions required to be truly successful.

The purpose of this presentation is to present and discuss a series of often overlooked challenges, based on real-world examples, that can thwart organizations on the road to transformation ad posit potential solutions and mitigations to be considered prior to embarking on the journey or deployed when trouble is encountered.

#19 Detecting Defects in Sequential Inputs to Digital Twins Using Machine Learning, Nathaniel Brown and Steven Simske

This research presents a method for detecting defects in sequential data inputs for digital twins (DT) during simulation runs, emphasizing the importance of input validation for ensuring the accuracy and reliability of the simulation results. By thoroughly validating input data, researchers and practitioners can have confidence in the validity of their models, ultimately leading to better decision-making processes and outcomes that are more successful. The proposed framework for validating inputs in real time offers a way to improve the quality and credibility of DTs, guiding future research in the evolving field of modeling and simulation (M&S).

The case study described in this presentation uses second-order polynomial regression to detect defects in rocket trajectory data streams, highlighting the effectiveness of validation techniques. A DT consuming trajectory data during simulation execution could read in the data directly or receive it live from an external data source. In particular, live data from external sources present a challenge to DTs, as they cannot verify the veracity of the data prior to execution and data defects may introduce faulty DT modeling. Using this technique and suggesting future research with other advanced machine learning methods, this research demonstrates a step towards robust data input management in M&S systems.

This work proposes a method to detect defects in sequential data inputs for digital twins in real time during a simulation run. Validating M&S inputs is crucial for ensuring the accuracy and reliability of simulation results. As demonstrated in the case study, proper validation techniques can identify defective data and lead to accurate and realistic simulation outputs, which ultimately contribute to better decision-making processes and results that are more effective. It is essential for practitioners to prioritize the validation of simulation inputs to improve the quality and credibility of digital twin results. This research contributes potentially valuable insights to the field, emphasizing the significance of input validation for enhancing the quality and accuracy of simulation models.

#20 Enhancing Onboarding in QA Teams: Object Detection Approach for Test Case Generation in Exploratory Testing, Ricardo Reyna and Steve Simske

Software testing is a pivotal stage in the Software Development Life Cycle (SDLC), being crucial for ensuring the quality and reliability of new products. Quality Assurance Engineers (QAEs) play a vital role in this process, diligently identifying and rectifying errors to achieve error-free systems. To accomplish this, precise software testing techniques are indispensable. Within the diverse landscape of software testing methodologies, notable examples include white box, black box, and grey box testing. These methodologies can be employed individually or in tandem during the test execution phase. White box testing provides QAEs with in-depth knowledge of the application by granting access to its source code and design documents. Conversely, black box testing focuses solely on validating software functionality based on specified requirements, remaining agnostic to internal workings or implementation details. Bridging these approaches, grey box testing combines elements of both with and black box techniques, offering a balanced perspective on system testing.

Numerous test case generation techniques have been developed to elevate the quality of test cases, many of which are customized for manual testing in order to optimize execution efficiency while broadening test coverage. These techniques are invaluable for QAEs in isolating crucial test conditions vital for comprehensive software evaluation, particularly those that may be elusive to identify. Moreover, in conjunction with manual testing, a plethora of software techniques and tools exist to streamline the creation of automation scripts for testing purposes.

Exploratory testing is a dynamic form of software evaluation wherein QAEs diverge from predefined methodologies, embracing an unscripted approach. While typically associated with QAEs, developers also implement this method, leveraging their expertise, insights, and skills to unearth potential bugs within their codebase. The objective behind exploratory testing is to optimize and refine software comprehensively, often aligning with the principles of black box testing methodologies.

The benefits of exploratory testing are manifold, ranging from the discovery of bugs that may elude structured testing phases to its utility in shaping user stories. QAEs can annotate defects, incorporate assertions, and record voice memos during exploratory sessions, laying the groundwork for subsequent test case creation. Additionally, this approach aids in formalizing and automatically documenting findings, fostering collaborative engagement from all team members in application exploration. Insights gleaned serve as a basis for creating automation scripts, thereby facilitating functional testing. Leveraging pre-existing automation scripts enables cross-browser deployment, allowing for comprehensive observation of application behavior. Exploratory testing is particularly advantageous for onboarding new QAEs, providing a swift avenue for familiarizing them with the application and offering timely feedback.

With the rapid evolution of Artificial Intelligence (AI), an increasing array of companies is seamlessly integrating this state-of-the-art technology into their products and services. One notable application is within the automotive industry, where Computer Vision (CV), a pivotal component of AI, is driving the development of autonomous vehicles. By leveraging CV, these vehicles adeptly recognize pedestrians, interpret road signs, navigate obstacles, and interact harmoniously with other vehicles on the road. Beyond transportation, AI has significantly transformed everyday life, simplifying tasks like food ordering through Virtual Assistants (Vas) such as Alexa. These advancements are made possible by breakthroughs in Natural Language Processing (NLP), Large Language Models (LLMs), and Convolutional Neural Networks (CNNs). As AI continues to evolve, QAEs are embracing innovative approaches to keep pace with the dynamic technological landscape, ensuring that testing processes remain robust, adaptable, and resilient in safeguarding software quality and cybersecurity integrity.

It is understood that when new QAEs join a team, they may possess knowledge of various testing techniques or software testing activities. Some QAEs may find themselves unfamiliar with applications, ranging from simple website browsing to complex ones utilizing artificial intelligence. In certain instances, QAEs may struggle to discern the specific elements requiring verification or validation to ensure the application functions as intended. This challenge is often encountered by new team members who are still acquainting themselves with the product or application they are tasked to evaluate. Acclimating to the application can be time-consuming and may result in additional costs for the company. While some teams provide documentation outlining key testing points, others may lack such resources, necessitating close collaboration between new QAEs, developers, and business analysts to grasp the nuances of the application and gather essential information for testing. We believe that furnishing QAEs with documentation or relevant information upon joining the team can expedite their familiarity with the application under test (AUT). This approach enables QAEs to focus on testing tasks rather than spending time deciphering which aspects of the application require testing.

Our research aims to utilize a computer vision (CV) algorithm as a foundation for detecting UI components from websites or mobile apps. The documented results will be presented in a structured format, such as documentation suitable for use in JIRA or even training materials like virtual reality applications, providing QAEs with invaluable insights. This documentation will offer a comprehensive view of the UI components, their interactions, potential testing paths, and other pertinent details. By leveraging this approach, QAEs can explore the application more effectively and integrate seamlessly into the team. This initiative is poised to boost their confidence in the testing process by equipping them with a deeper understanding of the application’s functionality and purpose.

Ricardo Reyna is a seasoned QAE with a strong dedication to continuously seeking innovative methods to meticulously validate software system functionality. He holds a bachelor’s degree in computer information systems from Texas State University and a master’s degree in software engineering from Pennsylvania State University. Currently pursuing his Ph.D. in Systems Engineering under the guidance of Dr. Steve Simske, Ricardo brings over nine years of experience in software testing to his role as a Senior QA Engineer at Dun and Bradstreet. His research interests span AI, ML, and Software testing, reflecting

his commitment to staying at the forefront of technological advancements in the field.
Ricardo, a dedicated QAE, innovates to validate software functionality, improve performance, and enhance user experience, surpassing expectations in software quality assurance.

#21 Strategies and Best Practices for Managing Complex System Architectures, Sean Densford and Chris Klotz

As the paradigm shifts towards the adoption of Model-Based Systems Engineering (MBSE), the Change and Configuration Management of digital models and handling of complex digital project model architectures emerges as critical area of focus as traditional document-based methods fall short in addressing the unique demands of digital model management. While change and configuration management is a widely recognized procedure in conventional document-based systems engineering, its implementation in digital models is comparatively less established. New capabilities and tools that simplify the management of digital models have been provided with the implementation of MBSE. The increased capacity of these skills makes digital processes and procedures more suitable for management than their traditional counterparts.

Currently, the industry is saturated with many approaches for managing digital models coming from a traditional document-based perspective. Many questions emerge when none of the approaches are suitable for the project model architecture and leading to a perceived gap in capability that prevents MBSE adoption. The goal of this paper is to demystify complexities surrounding digital model management, communicate considerations for management of digital models, and propose model architecture patterns that support the effective handling of models throughout their lifecycle.

Sean Densford is a MBSE with a decade of experience working in the aerospace and defense industry. He holds a Masters Degree in Systems Engineering from Johns Hopkins University and is an OMG-Certified Systems Modeling Language™ (SysML®) Professional (OCSMP). He currently works as an Industry Process Expert for MBSE helping all industries using systems engineering to better utilize engineering tools to develop their systems architectures.

Chris Klotz is an Industry Process Consultant for Dassault Systems. He joined No Magic 8 years ago as a developer working on plugin development. He then transitioned into a leading support for both the deployment and usage of the modeling tools and the server based products. The last 5 years he has been a SysML trainer, and worked directly with customers applying MBSE techniques to solve complex system engineering problems. He has also become a leader the deployment, migration, and usage of the Teamwork Cloud. He is an OMG Certified Systems Modeling Professional.

#23 Integration of System Data Requirements in Stuttering-Aware Speech Recognition Systems, Ibibia Altraide and Steve Simske

Purpose: This study investigates a formalized, but refined, Systems Engineering approach to data requirements for the automatic recognition of stuttered speech in AI-enabled systems (SE4AI).

Focus: Stuttering is speech that is characterized by the repetition or prolongation of sounds, syllables, words, and hesitation or pauses that disrupt the rhythmic flow of speech. People who stutter (PWS) want to use artificially intelligent automatic speech recognition (AI-ASR) systems but are frequently misunderstood and cutoff because AI-ASR models are optimized on data from people who do not stutter. A primary reason for the deficiency in current AI-ASR models is the lack of large, diverse, and specified data on stuttered speech. To remedy this problem, this study proposes a refined Systems Engineering

(SE) approach to data specification and modeling of stuttered speech for AI-ASR. While traditional SE lifecycle and principles have been successful in building heretofore complex systems, current AI-enabled systems have introduced new paradigms that do not fit SE traditions. Using an ad-hoc approach, AI-ASR systems are capable of sophisticated behavior that allow them to learn and evolve during operations, making their prior specification difficult or impossible. Despite the difficulty, this study advocates a refined SE approach in establishing design integrity, artifacts, and configuration baselines for such systems.

Methods: This study proposes to assemble the largest dataset of diverse stuttering speech data to date. The dataset assembly and specification will be done in a predetermined way, using a systemic approach to planning, analysis, design, implementation, verification, validation, deployment, and maintenance. Firstly, data requirements (data type, size, nature, quality, distribution, complexity, annotation, classification) will be elicited from all relevant stakeholders (people who stutter, researchers, foundations, data banks, etc.). The elicited data will be analyzed and prioritized according to an agreed-upon scheme. Next, a Data Baseline Architecture composed of a Requirements Traceability Matrix, Verification and Validation Cross-Reference Matrices, and Systems Engineering Data models (conceptual data model, logical data model, physical data models) will be created as configuration baselines within an MBSE environment to manage and maintain the data. Complementing the collection and labeling of the dataset, this study proposes the use of Systems Engineering modeling approaches to create models for the specific classes of stuttering, including repetition, prolongation, hesitation, and pauses. It concludes with examples of verified and validated AI-enabled ASR systems that correctly recognize and transcribe stuttered speech because of this approach.

Results: The benefits include well-formulated, complete, and stable data requirements for the automatic identification and classification of stuttered speech using AI-ASR models and devices; a centrally managed and maintained data repository that provides an enduring, authoritative source of truth so that stakeholders have current, authoritative, and consistent information for use over the lifecycle; open and better collaboration in the research community.

Implications/Conclusion: The systematic specification of data requirements and systems engineering models for stuttered speech when deployed in AI-enabled ASR systems is poised to improve the recognition and transcription of stuttered speech and help reclassify low-confidence “normal speech” ASR outputs into high-confidence “stuttering” classifications.

#24 HALF-DAY TUTORIAL: Integrating System Architecture in SysML with Hardware for Rapid Prototyping and Validation and Verification, Saulius Pavalkis

Faster time to market and more and more software intense systems requires higher level of integration and faster decisions that are more informative. This is accomplished leveraging the best practices of model-based systems engineering (MBSE), digital engineering, engineering disciplines integration enabling faster prototyping and V&V.

This hands-on tutorial explores the integration of system architecture in SysML with hardware through the Internet of Things (IoT) protocol and other means. Connecting SysML with hardware, specifically Arduino, allows for real-time and rapidverification and validation (V&V) and prototyping of systems. The tutorial covers the vocabulary, technology stack, architecture of the connection solutions and actual connection libraries. It provides the integration of the connection in the tool of choice - CATIA Magic, and required library for connecting to other hardware using IoT protocol and other means. The demonstration showcases the seamless interaction between system architecture and hardware, emphasizing the importance of model based systems engineering (MBSE) enabling bridging the gap between system requirements in the model and hardware implementation for fast prototyping and (V&V).

• Structure and Format:
**Hands-on
**4hours - half-day session, providing adequate time for hands-on activities and in-depth coverage of concepts and solution.
**Products, licenses, samples and hardware provided:
***CATIA Magic
***Arduino IDE
***Hardware
***Library for connection to hardware using IoT protocol and other means.
• Knowledge Level: Assuming familiarity with systems engineering (intermediate), but new to IoT and Hardware integrations or concepts (beginner).
• Position within Systems Engineering: The tutorial positions itself within the realm of systems engineering, focusing on system architecture and its practical application in real-world environment for fast prototyping and V&V.
• Target Audience: Targeted at systems engineers interested in digital engineering, IoT applications, Prototyping, V&V.
• Practical Uses:
** Participants can directly apply the knowledge to real-time verification of engineering designs.
** Provided library, sample models, and hardware will create experience and abilities to reproduce integration with other model, and hardware.
• Organizational Improvement: The skills taught can lead to enablement and improvement how systems architecture is used for V&V and prototyping, and how it is integrated with hardware.
• Professional and Personal Value: Offers significant value to attendees by enhancing their professional capabilities in systems engineering and digital engineering.
• Demand: due to faster time to market, and more and more software intense systems digital engineering, engineering disciplines integration, faster prototyping and V&V integrating SysML, IoT and hardware is a high-demand area.
• Attractiveness: The hands-on aspect, provided solution enabling skill, and practical implementation focus make this tutorial attractive.
• Educational Nature: Highly focuses on educational aspect, hands-on, sharing method, tools and samples, not on selling a product, despite mentioning specific solutions.

#25 MBSE Digital Engineering Ecosystem, Saulius Pavalkis

Integration between tools and disciplines is major direction of MBSE and digital engineering.

In this session we will present SysML model of all integrations covering MBSE ecosystem / integrations (internal and external), use cases, and integrations maturity. We have a large MBSE ecosystem of internal, partner, and third-party integrations. Clients, sales, consultants, and partners are leveraging this information every day. In this session, we present updated architecture of hundreds of integrations we have currently and in progress. In addition, we present major integrated MBSE ecosystem use cases applied by major clients in major adoption cases. This model will help to move the MBSE ecosystem forward, as proof of concepts we have world-wide.

After general overview we will concentrate on newest integrations: MBSE ecosystem application for design reviews, PLM as data backbone, MBSE for electrical engineering, MBSE and hardware integratin, MBSE and IoT, and model based acquisition.

#26 Enabling MBSE through Function-Based Requirement Synchronization using SysML, Theodor Behrens and Dr. Ingo Stolpe

This PhD project aspires a novel approach in the development of hardware-in-the-loop (HiL) testbenches for software-based vehicle functions. The focus lies on deriving testbench requirements from vehicle function algorithms. Assuming that every vehicle function can be decomposed into atomic entities, we propose to design equivalent atomic testbench entities for any given atomic vehicular entity. Integrating and allocating the designed testbench entities in accordance to the architecture of the related vehicle function provides a requirement model for the testbench. This requirement model is described in SysML by means of activity diagrams and other language elements. This standardizes the product development process and guarantees superposition of verification & validation for both testbench and vehicle. The introduced approach is illustrated by the example of a well-known vehicle function.

#27 Justifying Resilience, John Brtis

During Q&A after a presentation on system resilience at WSRC 2022, the author was asked for thoughts on how to justify the investments required to achieve system resilience. Although a few ideas were discussed, it was clear that better answers were needed for this important question. The author has studied this issue, performed a literature search, and facilitated a workshop on the topic at INCOSE IW24. This presentation presents the findings; it explores why justifying resilience is such a challenge, discusses how these challenges are shared with many of the other quality characteristics, and offers a set of strategies for dealing with the problem along with specific guidance on how they fit into SE activities.

Establishing resilience requirements for a system can be a challenge. Stakeholders often focus on their personal needs that systems must satisfy, and resilience is seldom top of mind. Further, stakeholders and decision makers may not understand resilience or its value. Adding resilience may cost time, money, and performance, and project managers are primed to resist “gold plating,” arising in the engineering team. But the situation is not hopeless. Studies show that there are a number of stakeholders in any organization that may be allies in advocating for resilience. Examples include: risk and insurance managers, strategic planners, company lawyers, public relations managers and financial leaders. Also, resilience advocates can find common cause with those working on other quality characteristics (security, safety, etc.). The presentation offers specific strategies for advocating for resilience and other quality characteristics.

#28 FULL-DAY TUTORIAL: Risk, Safety, and Reliability Analysis in Model Based System Engineering (MBSE) [A Tutorial], Ron Kratzke, Brian Pepper and Bill Bentley

The Object Management Group (OMG) released a standard providing standard processes for Risk Analysis Assessment Modeling Language (RAAML). The analysis techniques and requirements discussed in this standard are an integral and important part of engineering today’s complex systems. This is particularly important in the aerospace, automotive, energy, and defense industries.

The RAAML standard includes: Failure Mode and Effects Analysis (FMEA); Fault Tree Analysis (FTA); System Theoretical Process Analysis (STPA) methods; and Functional Safety Analysis in accordance with ISO 26262.

These analysis techniques have generally been accomplished using unique tools by subject matter experts. This tutorial explores these applications in the context of Model Based System Engineering (MBSE). In the RAAML standard, each of these analysis techniques are defined as extensions to the general System Modeling Language.

Students in the tutorial will explore the extension for each of the techniques and construct examples based on a common system design model.

Specifically we will cover the following in the tutorial:

1.) Introduction to the RAAML Standard
2.) Development of a Failure Mode and Effects Analysis (FMEA)
3.) Development of a system fault tree using Fault Tree Analysis (FTA)
4.) Application of System Theoretical Process Analysis (STPA) methods for system risk management
5.) Conduct of a Functional Safety Analysis using ISO 26262
6.) Conclusion and summary by understanding the commonalities and differences in the methods

This tutorial is design to take 8 hours to accomplish, but can be condensed to 4 hours by streamlining the hands-on development of each analysis.

#29 Data Markings and Classification in MBSE, Ron Kratzke and Sean Densford

Many models contain information, which is restricted or classified in nature. The engineering teams need a way to individually designate information classification on individual elements in a system design model. This presentation will demonstrate the basic functionality of a Data Marking and Classification plugin in an MBSE environment. The demonstration will include how to mark information in a model and how to redact the model to show information at certain levels. We will also explore how to make custom data markings tailored to the needs of the organization. The demonstration will also include the functionality of setting and managing restricted access for model management. And, will discuss methods for marking models with a restriction or warning notice prior to opening up a model that contains sensitive/classified information.

#30 Digital Twins Platform Systems Engineering to Optimize Astronaut Physiology During Human Space Exploration, Caleb Schmidt, Tom Paterson, Michael Schmidt and Steve Simske

Spaceflight presents complex engineering and human factors challenges that are rivaled by few other human endeavors. To reach extraterrestrial locales, the engineering spaceflight community has solved, and continues to solve, myriad complex problems utilizing systems engineering (SE) methods for system realization and life cycle management. In certain embodiments, digital twin platforms (DTPs) that represent physical systems virtually, mimic these systems quantitatively, and capture their variation have been exceedingly helpful to SE efforts. Similarly, when the human body is taken beyond the 1g, low radiation, open environment of earth, the spaceflight hazards presented are multiple and constant, and lead to significant perturbations to the human system. Thus, there is a necessity for continued deep study, rapid innovation, and technical prowess to address and mitigate how humans can effectively thrive in this milieu. The Astronaut Digital Twin (ADT) is a biological DTP specifically designed to emulate the behavior of the astronaut entering the spaceflight environment by leveraging an agile SE approach. It can incorporate real time data, assess state, simulate outcomes, and alter the trajectory of the human system through countermeasure application. A spacefaring civilization will require considerable advancements in the ability to enhance astronaut performance, health, safety, and survival. Within this context, the ADT is one critical tool to shorten the development cycle, pose complex questions about any unique environment, and test hypothetical countermeasures before entering those environments. This is expected to provide an advanced capability to the spaceflight medicine community and help them keep pace with the spaceflight engineering community, as they collectively grapple with these highly complex operations.

To date, the spaceflight medical and earth-based medical and biological sciences communities have not optimally used engineering formalism and the tools that SE affords to predict and manipulate human systems and have been heavily grounded in reductionist science. Realizing this gap, disciplines such as systems biology and systems medicine have attempted to categorize the volumes of data generated towards dynamic and causal understandings with some successes. The approach described here advances that goal utilizing a hybrid agile SE/computational systems physiology approach to realize and deploy a biological DTP (i.e., the ADT). Utilizing a mechanistic model based in ordinary differential equations and Bayesian inference, the ADT uses real-time sensing information on structure, dynamics, and failure mode topologies as inputs into a virtually represented astronaut that evolves over time to simulate, monitor, diagnose, predict, and optimize system behaviors.

The problem of construction of the ADT is primarily about data aggregation and dimensionality reduction to support a real time, bidirectional digital twin. Through an SE process that is most akin to the agile framework, the ADT is constructed iteratively in six structured yet evolving stages, with three main objectives layered within this process. The agile component of this framework is vitally important because the modeler must follow what narrative evolves from the data and be nimble towards answering the chosen question (i.e., how the human body responds to space) to fully represent system behavior and its variability adequately. A major strength of this approach is its ability to capture fragmented data and aggregate multiple high-quality mechanistic, epidemiological, and interventional studies into a common framework. Within this gestalt, there are three sources of data that are necessarily incorporated into the ADT: 1) data of terrestrial origin, 2) spaceflight historical and Earth-based space analog data, and 3) novel data from prospective missions, with multiple data classes layered within these types.

DTPs and the emergence of powerful artificial intelligence and machine learning (AI/ML) modalities should be viewed as complementary, not competitive. Where successful technologies such as GPT-4 and AlphaFold2 utilize large sets of training data (e.g., text and protein sequences) and code (i.e. prompts and sequences without noise and “fully observed”), biological data is noisy, partially observed, and longitudinal. This underlying fact necessitates trade-offs across multi-modal solutions to provide decision support. In short, AI/ML can identify correlative patterns in high-dimensionality datasets, while the ADT deciphers those patterns, creating testable hypotheses of the causal processes underlying them.

Once the ADT is developed, tested, evaluated, and deployed, there are multiple clinical and research applications for which it can be utilized. For clinical purposes in professional and commercial astronauts, this promotes five major application characteristics: 1) risk assessment, 2) prediction, 3) precision and personalization, 4) clinical decision support, and 5) recommendations and countermeasures. For research, there are four classes of activities that are pertinent to its utility: 6) iterative model development, 7) experimental design, 8) hypothesis generation, and 9) hypothesis filtering. This presentation will focus on A) how biological DTPs can be leveraged to support rapid advancement in human spaceflight through the construction of an ADT, B) how an agile systems engineering methodology is a powerful tool to support this development, and C) how the ADT is complementary to AI/ML methodologies and is a core framework to further leverage those technologies.

About Caleb M. Schmidt: Caleb is the Vice President of Research for Sovaris Aerospace, a spaceflight medicine company widely regarded as one of the leaders in the field of precision medicine in human spaceflight and high-performance operations on Earth. Sovaris has gained this status by utilizing molecular analytics, pattern analysis, pattern recognition, systems engineering, computational and digital twin modeling, and countermeasure development. Sovaris' work covers a spectrum from NASA, the NFL, the NBA, US Olympic teams, Nike, SpaceX, Axiom Space, Corvette Racing, NASCAR, US Special Forces, SWAT, the Naval Submarine Medical Research Lab, the Mayo Clinic, Cornell Weil Medicine, Baylor College of Medicine, Johns Hopkins School of Medicine, George Washington University School of Medicine, and others. Caleb is also a professional doctoral student in the Department of Systems Engineering at Colorado State University under Steve Simske, where he has leveraged his work on digital twin platform systems engineering as a major component of his dissertation work. He has a Bachelor of Science in Biochemistry, with a minor in Chemistry, and a Master of Science in Microbiology, Immunology, and Pathology. Caleb resides in the mountains of North Carolina with his wife and young daughter.

#32 System Interface Clustering using Design Structured Matrix (DSM), James Hummell

Design Structured Matrix DSM is a simple, compact, and visual representation of a system or project in the form of a square matrix. It is used in systems engineering to model the structure of complex systems to perform system analysis, project planning and organizational design. The presentation will show how to define interfaces, itemFlows, and structure to be compatible with MIT OpenCourseWare Design Structure Matrix (DSM) methodology. I will also demonstrate how to perform clustering and grouping of sub-systems and coupling of those systems based on the definition of the interfaces and itemFlows between systems.

#33 HALF-DAY TUTORIAL: Use a Framework for SE in Early-Stage R&D to Build Your Bridge that Spans the Chasm Between Research and Engineering, Ann Hodges

Abstract: Researchers and funding organizations often do not understand the value of systems engineering in early-stage projects, defined as technology readiness levels TRL 1-5, during which systems engineering may be viewed as an unnecessary cost, and as a process-heavy effort applicable only for mature technologies. This may result in a relative lack of engineering rigor and of understanding of innovation context which often contributes to failures leading to the "valley of death" between fundamental research and applied development. There is more than one pathway for crossing the valley of death, and relevant application of systems engineering implemented at an appropriate level of rigor provides a foundation for transition and use of technical innovation. This tutorial, updated from IS2024, provides an overview of the valley of death associated with technical and product incubation, the principles and foundational elements necessary for transitioning research projects to engineering development that bridges this valley of death, and presents a framework for systems engineering applicable in early-stage research and development (ESR&D), including tailoring considerations associated with TRL, stakeholder roles, and relevance to the use of MBSE and Digital Engineering. Associated framework metrics are presented to enable evaluation and practical implementation of the framework for systems engineering innovation management at this phase of technology development.

Outline:
1. Introductions – instructors and participants, ask participants to share {name, organization, their domain(s) of experience (e.g., academia, communications, IT, etc.), problems experienced with applying SE in research
2. Problem statement for systems engineering (SE) in early-stage R&D - issues and impacts
3. Framework elements overview – emphasize the framework is applicable to a range of research project types and scale (single project, to program [set of projects] to enterprise level), use critical thinking to

include other relevant and useful approaches (e.g., design thinking, agile methodology)
a. Value proposition expressed in terms understandable to stakeholders. Exercise: Break into groups, participants provide their perspectives given their domain experiences, debrief with larger group
b. Framework principles overview – Group Q&A: ask participants for feedback
c. Standards basis overview – Group Q&A: as a group discuss standards that are relevant for participants' domains
d. Risk-based graded approach overview – Exercise: Break into groups, discuss whether participant case studies are low or higher rigor (e.g., "moon shot", grand challenge)
e. Present TRL roadmap for SE activities, artifacts, assumptions concerning the roadmap – group Q&A: discuss assumptions
f. MBSE & Digital Engineering – Group Q&A: as a group discuss models that are useful for the SE activities and artifacts. Discuss relevance of tool choice on integration, usability, and "buy-in" of others.
g. Research domain types – present overview of the layers of practices/artifacts: common core of practices/artifacts; tailored extensions (e.g., methodology, organizational); domain specific
h. Training/coaching – Principal Investigator/research team and Systems Engineer coach each other so that each has sufficient domain knowledge to apply the TRL roadmap
i. Measures and metrics overview – present overview and core set of measures/metrics
j. Continuous improvement – present how measures/metrics are used to provide insight for improvement, gather lessons learned during execution of planned activities/artifacts from TRL roadmap, turn lessons learned into lessons applied
k. Present suggested flowchart for key decisions in using the SE in early-stage R&D framework elements
4. Using the SE in early-stage R&D framework elements – Exercise: Break into domain groups. Each group:
a. Discuss changes to value proposition and principles for the domain's specific culture
b. Discuss domain-specific standards to consider in the TRL roadmap for SE activities/artifacts
c. Discuss appropriate rigor for the domain – may evolve during TRL maturity
d. Tailor TRL roadmap for domain-specific standards, processes, practices, deliverables – select at least 2 process areas in the roadmap
e. Tailor measures and metrics for the domain
f. Each group presents their results to the wider group

Primary learning objectives - what the participants will gain:
1. Participants will learn about challenges associated with transitioning research to engineering development.
2. Participants will learn about the framework elements that support technical planning for transitioning research to engineering development.
3. Participants will apply the framework elements to a domain-specific case study of their choosing.
4. Participants will learn about an approach that address challenges highlighted in the SE Vision 2035: use of multi-disciplinary analysis collaboratively (researchers/systems engineers), an analytical framework for planning SE activities and deliverables in early-stage R&D, explore domain patterns for SE activities/deliverables.
5. Participants will learn about an approach that helps to normalize relevant SE practices and deliverables that support successful transition of research to engineering development, and that can provide a basis for reuse of SE assets at a domain level – steps in the practices facet of the SE Vision 2035 top-level roadmap.

Presenter biography:
Ann Hodges retired after 48 years of service at Sandia National Laboratories (SNL) and was a

distinguished member of technical staff. She was the Mission Services Division's systems engineering lead for the systems engineering part of the project and product delivery system (PPDS) at SNL and was a project manager and systems engineer for a complex exploratory-phase project. She is a primary author of the risk-informed graded approach to the application of project management, systems engineering, and quality management which is one of the key aspects of the PPDS. She collaborated with the Laboratory Directed R&D program office to tailor the application of PPDS to SNL's research portfolio.

Tutorial experience and other relevant background:
- Co-presented a tutorial on "Integrating SE, Project Management and Quality Management" to the INCOSE Enchantment Chapter in 9/2017 and INCOSE IS2018.
- Co-presented this submitted tutorial to INCOSE IS2024.
- Was project manager and SE for a complex exploratory-phase project and collaborated with the SNL Laboratory Directed R&D program office to tailor the application of PPDS to SNL's research portfolio.
- Co-developed PPDS instructional materials, and taught PPDS concepts to over 200 management and staff members.
- She co-chairs the SE in Early-Stage R&D Working Group and was co-editor and co-author of several papers in INSIGHT volume 26 issue 3, "SE in Early-Stage R&D: Bridging the Gap."

#34 Overcoming Barriers to Smart Home IoT Security: The Impact of Manufacturer Guidance on the application of User-controlled security features, Kelvin Shorts and Steve Simske

As Internet of Things (IoT) devices allow our homes to become more interconnected, the importance of privacy and cybersecurity best practices becomes increasingly paramount. This study aims to explore the level of support provided by Smart Home IoT manufacturers to consumers in applying user-controlled cybersecurity features. User-controlled features are settings within an IoT device that only the end-user can adjust (e.g. passwords, multi-factor authentication, data backup, etc.). Structured surveys and proficiency test were used to evaluate the number of user-controlled security features identified based on the clarity and comprehensiveness of security guidance provided through an IoT device setup manual. This research contributes valuable insights for both IoT manufacturers and IoT users on the difference clear and comprehensive guidance makes in assisting users with the implementation of user-controlled cybersecurity features.

#35 Rapid Risk Management, Thomas Duerr

OVERVIEW OF THE TOPIC

The traditional risk management process poses challenges for a lean, rapid-acquisition Government Program Management Office that is not resourced to support the requisite rigor. This briefing explains an alternative approach to risk that mitigates those challenges and generates actionable decision-support products. The presentation explains the root causes of the challenges, the revised process modeled on the OODA loop, calibration of risk effects in terms of threat to mission, and more suitable risk mitigation decision aids than the traditional 5x5 risk matrix, including a pragmatic, resource-driven definition of risk tolerance. Results from a case study illustrate the differences between traditional and Rapid Risk Management. Rapid Risk Management has been in use on multiple government programs in Albuquerque NM since 2023.

SPECIFIC RELATED INDUSTRIES
Space systems

WHAT THE AUDIENCE WILL TAKE AWAY FROM THE PRESENTATION

The audience will learn an alternative, proven risk management process that was developed for the particular needs of small teams managing rapid-acquisition programs. They will be introduced to new risk management decision aids that provide more complete and defensible information than a 5x5 risk matrix when resource constraints limit the team to qualitative analysis. Finally, they will be shown a practical, resource-driven definition of risk tolerance that can be applied regardless of how risks are prioritized.

BACKGROUND ON THE PRESENTER AND THEIR QUALIFICATIONS TO PROVIDE THE TALK

The author has 44 years of project and technical leadership experience in acquisition, decision analysis, architecting, systems engineering, risk management, and systems analysis. His projects have ranged in scope from enterprise- to component-level, covering the domains of indications and warning (I&W), intelligence, surveillance, and reconnaissance (ISR), satellite communications (SATCOM), and theater ballistic missile defense (TBMD).
Recent customers have included the Space Rapid Capabilities Office (SpRCO) and the National Reconnaissance Office (NRO).
He is currently the lead engineer on two government rapid-acquisition programs.

PREVIOUSLY PRESENTED ELSEWHERE

The approach was introduced in the presentation "Risk Control for Rapid Acquisition," Adapting Mission Assurance Conference, Albuquerque NM, 18-19 October 2022. Improvements due to feedback from that event and others led to its adoption on a first program in early 2023.

#36 Mission Assurance and the Enterprise Lifecycle: A Systems Thinking Approach, Sian Terry

About Presenter

Sian Terry is a PhD student at Colorado State University and a Senior Systems Engineer with Peraton. Throughout her career she’s worked with many systems across the aerospace industry, including commercial and government systems, utilizing system-of-systems and product line systems engineering approaches. With her 10+ years of experience, B.S. in Aerospace Engineering and M.S. in Systems Engineering, she is prepared to tackle her current topic of interest, the evolution of legacy systems into an enterprise.

Abstract: Mission Assurance and the Enterprise Lifecycle: A Systems Thinking Approach

This presentation continues the enterprise lifecycle model (ELM) research presented at the 2023 WSRC titled “Development of a Model to Support Legacy System Evolution into an Enterprise” [7], a summary of the novel article by Terry and Chandrasekar that introduces the ELM. A model that partially addressed a new issue with the development of today’s systems: legacy system evolution. Traditionally, requirements were periodically added to legacy systems to meet the continued needs of the customer. Over time, system size, longevity, and complexity has increased; necessitating the creation of an enterprise (“a purposeful combination of interdependent resources that interact with each other to achieve business and operational goals” [9]). This combination results in a system that is continually supported and is thus able to meet the ever-evolving needs of the customer and sustain functionality for decades. This change in the system development process results in an unexpected strain on traditional systems engineering (SE) techniques, such as the Vee model which ends with the delivery of the system-of-interest (SOI) [9]. The ELM seeks to ease this strain by addressing the major needs of an enterprise that are not embraced by legacy systems. These needs may be numerous, depending on the source, and heavily tied to certain industries. To address this, the presenter embraces systems thinking methods that mandate the use of abstraction to find points of greater leverage, which in this case is system complexity management. As such, the ELM was designed to incorporate multiple levels of perspective, from the external environment and stakeholders down to the SOI entities. Here, feedback loops in causal loop diagrams were utilized to display the causal links between the interdependent piece-parts of the enterprise. Fostering a greater understanding of the interactions (direct and indirect) between these interdependent resources, and thus reducing complexity.

This initial version of the ELM defined the use of the three-system concept (SOI, system context, and enabling systems) to provide a balance between the following aspects of the SOI: technical, managerial, and environmental. The causal relationships between the elements of the agile and iron triangles were depicted and used to define the meaning of value, quality, and constraints (scope, schedule, and cost) in this context [7]. In this presentation, titled Mission Assurance and the Enterprise Lifecycle: A Systems Thinking Approach, the ELM will be further developed by including mission assurance. Mission Assurance focuses on the mission (read: scope) and allows the ELM to avoid the potential pitfalls of the 'cost is king' and 'schedule is king' mentality that is common with commercial and large-scale government systems, respectively [5,8]. By adopting Hahn and Hodges's practices for project management (PM) and SE integration [1,2], the presenter was able to create a more abstract version of the ELM level 2 diagrams [7] and draw connections between the aforementioned agile triangle elements. Resulting in a model that crosses the boundary between PM, quality management (QM), and SE efforts to assure mission success. This boundary crossing allows for the inclusion of topics previously considered to be out of scope, namely risk management (which will be addressed through the management of uncertainty) and PM. As well as the addition of capability maturity levels and the resources needed to support system planning, development, and execution (hardware, software, and wetware), key concepts to the evolution of legacy systems. Expanding the ELM's ability to address system complexity and further SE knowledge by creating a model that reduces complexity to a critical point. Here a balance must be struck between reducing complexity to the point that the system can be depicted and understood in its complex environment and ensuring that intricacies needed to embrace the emergent properties (flexibility, adaptability, scalability, and changeability [4]) are maintained [3].

The audience will be presented with SE focused updates to the ELM that describe the SE/PM and SE/QM boundaries, addressing key areas in the SE Vision 2035 [6]. It should be noted that the ELM has been specifically developed with the evolution of large-scale aerospace ground systems in mind, however, it is ideally applicable to multiple industries. As such, system evolution will be addressed in general, industry and tool agnostic, terms. This presentation does not complete ELM development. Future tasks will include defining the three environments (planning, development/test, and external) and maturing this model from one that is solely based on published works to a model that has been validated. As such, the advantages and disadvantages of this approach will be discussed along with these next steps.

References

[1] Hahn, H.A. and Hodges, A.L. (2019) 'Integrating Program/Project Management and Systems Engineering in practice', INCOSE International Symposium, 29(1), pp. 69–85. doi:10.1002/j.2334-5837.2019.00590.x.
[2] Hodges, A. (2013) '7.2.1 bricks for a lean systems engineering Yellow Brick Road', INCOSE International Symposium, 23(1), pp. 606–616. doi:10.1002/j.2334-5837.2013.tb03042.x.
[3] INCOSE. 2016. A Complexity Primer for Systems Engineers. TP-2016-001-01.0.
[4] Rebovich G Jr,White BE. Enterprise Systems Engineering: Advances in the Theory and Practice

(Complex and Enterprise Systems Engineering). 1st ed. CRC Press; 2019.
[5] Stribling, R. “Hughes 702 Concentrator Solar Array.” Conference Record of the Twenty-Eighth IEEE Photovoltaic Specialists Conference - 2000 (Cat. No.00CH37036). IEEE, 2000. 25–29. Web.
[6] Systems engineering vision 2035 (2021) INCOSE. Available at: https://www.incose.org/publications/se-vision-2035 (Accessed: 29 April 2024).
[7] Terry, S. and Chandrasekar, V. (2023) ‘Developing a model that supports the evolution of Legacy Systems into an enterprise’, Systems Engineering, 26(6), pp. 859–873. doi:10.1002/sys.21700.
[8] Whelpton, P. (2015) Understanding the success factors of a large-scale system implementation in an emerging market. dissertation. ProQuest LLC.
[9] Wiley J. INCOSE Systems Engineering Handbook. 4th ed. John Wiley & Sons; 2016.

#37 Understanding INCOSE's Systems Engineering against CISA's Secure by Design (SbD) and DOE's Cyber Informed Engineering (CIE), Susan Ronning

Cybersecurity is the current buzzword considering on-going ransomware and phishing attacks, yet underlying the human aspect of secured systems are the networks and devices that support the IT/OT and iOT "tech" that we use and rely on.

There are two major players that are defining how critical infrastructure systems should be secured; (1) the US Department of Energy's (DOE) Cyber-Informed Engineering (CIE) and Cybersecurity and Infrastructure Security Agency's (CISA) Secure by Design (SbD). In addition, a new National Institute for Certification in Engineering Technologies (NICET) Systems Software Integration (SSI) certification is being released and the INCOSE Information Communication Technology (ICT) Working Group members are in the middle of them all.

Learn the similarities and differences between CIE and SbD, as well as how INCOSE SEP relates to NICET's SSI certifications and how all of this relates to your own industry and roles within it.

#38 Systems engineering and systems thinking to implement a Crisis Intervention Team in a rural town, Paul Havis and Dr. Steve Simske

Crisis Intervention teams (CIT) were created throughout the USA to increase police awareness of, and in so doing alleviate the shootings of, mentally ill individuals. Studies have shown that individuals with mental health issues are more likely to be shot and killed by responding officers. A case in September of 1987 in Memphis, TN, served as the catalyst to the creation of the CIT in Memphis. In the case, an individual with mental illness was shot and killed by officers due to their not being trained in how to handle individuals displaying symptoms associated with mental illness. CITs being implemented into rural areas are very important. The Fort Collins CIT Director suggested that all police departments train officers in de-escalation and provide resources for the mentally ill. Officer training should start from the academy and continue on a monthly basis as new strategies arise. Most mentally ill people who have gone off their medication can be irrational, but with the correct approach they can be calmed down. Through research we have concluded that officers across the USA need to be trained to de-escalate situations when dealing with the mentally ill.

Systems engineering and systems thinking principles are valuable in suggesting how to get started. Our analysis suggests that systems can be put in place to allow for trainings to be implemented in rural towns such as Abbeville, AL, in order to change the culture from “shoot first, ask questions later” to one focused on the prevention of violence and explicit de-escalation. Implementing CITs in the rural areas gives the officers insight into how to approach and help the mentally ill when they come into contact with them, and focusing on rural communities is a systems-based approach to providing ubiquity of the skill set. Geographical dispersion of the training reduces the odds that a crisis will arise at a long distance from appropriate expertise for crisis remediation. In this paper, we will describe how such a "rural-first" approach can be rolled out and how it can make optimal use of training resources.

#39 Effective Integration of Diverse Engineering Competencies In the Development of Complex STEM (Science, Technology, Engineering, or Mathematics) Projects: Optimizing Efforts and Investments in Student-Led Research Projects, Marco Rosa

Our work proposes an Interdisciplinary approach to student-led research applying innovative techniques of systems engineering and project architecture. This unique interdisciplinary approach will enable students to develop more efficient, resilient, and scalable answers to a variety of outcomes. Those outcomes include addressing project requirements, successful initial completion of long multifaceted projects and further development of projects.

This work seeks to demonstrate the relevance of utilizing this approach in academia so students can themselves develop their teams and their projects in a manner that will better prepare them to serve as professionals who will address the "big picture" and not only the focus on their specialization. This methodology will empower upper division students to develop their leadership and problem-solving skills. While also enabling Junior students the opportunity to learn and develop with mentorship of their seniors. This operational architecture will also ensure contribute of efforts and resistance within the Organization. These aptitudes are best encouraged in an environment where they will use interdisciplinary tools, concepts, knowledge, experience, and metrics. Such will purposedly enable them to be effective mentors to peers and lower division students. At the same time, this approach will embolden these students to use resources in the most productive manner and produce reliable, consistent, and cost-effective results that would benefit all stakeholders.

Applying this methodology, we can also benefits industry by providing degree conferred students that know how to work in complex engineering research teams. Time, energy, money, and materials will be saved because these new employees have already demonstrated dependability while working on a project with diverse engineering competencies and other areas such as business and integration.

Imagine a university that invests several thousands of dollars in student-led projects, only to have the project restarted from the beginning once the previous student leadership graduates. Now, visualize the same institution instead applying our proposed methodology, the project will not only have a higher success rate, but it will also be set up for further development and growth the next academic year. This approach will facilitate relevant research and development to more efficiently address project requirements.

These applications will help exponentially multiply the investment that the University would have made. Stable projects will also indisputably elevate the quality of experience and enhance practical education students graduate with. Furthermore, this will also be a particularly useful tool to highlight the innovative and resilient culture that the academic institution fosters. Such proficiency will support economic efforts of industry, and governmental projects, that depend on prompt, reliable, cost-efficient operations associated with complex engineering research, development, and production.

Integrating concepts and techniques from system engineering and project management into complex projects involving research, development, and manufacturing is a crucial aspect for success . An innovative approach where Graduate Students develop "Cadre" teams to assist other student-led projects will provide an essential foundation to establishing more successful teams. The investment in student-led teams will be better utilized, managed, and developed utilizing all resources that this proposed methodology will provide. Achieving this fluent integration will ensure that measures are in-place for an organized, adaptable, and consistent set of techniques. Interdisciplinary engineering specialties integrated in-step with a methodology intrinsically guided by systems engineering and project management will inevitably result in better return on investment now and in the future.

#41 Digital Engineering Tool Evaluation Criteria Template (DETECT)) Selection and SysML v2 Transition Guidance, Frank Salvatore

In today's dynamic engineering landscape, the integration of digital tools and processes is pivotal for organizations striving to achieve innovation and operational excellence, particularly within the U.S. Department of Defense (DoD). Digital engineering necessitates increased utilization of digital technologies and an interconnected infrastructure of tools, data, processes, and people. To address this, the Digital Engineering, Modeling & Simulation (DEM&S) Office within the Office of the Undersecretary of Defense, Research and Engineering (OUSD R&E) has begun developing a Digital Engineering Tools Evaluation Criteria Template (DETECT) model which considers various factors such as ecosystem architectures, tool categories, company size, industry sector, and technological maturity. DETECT aims to provide guidance for tool selection, based on organizational scale, digital engineering ecosystem scope, model and simulation interoperability needs and technology compatibility.

Additionally, the DoD recognizes the importance of robust engineering practices and modeling tools in developing superior weapon systems, with a particular focus on implementing standards to promote interoperability within digital environments. With the impending release of SysML v2 in late 2024, questions arise within the community regarding the transition from SysML v1.x to SysML v2. DEM&S, in collaboration with the systems modeling community, is developing SysML v1 to v2 transition guidance including FAQs, transition planning guidance, model conversion examples, a starter model, and more. This effort aims to support organizations, projects, modeling teams, and tool vendors in preparing for the transition by describing steps and considerations to take when transitioning modeling tools and practices to SysML v2, as well as connecting members of the model-based systems engineering community from industry, academia, and the DoD. By combining the DETECT emphasis on tool interoperability and organizational fit-for-purpose use with the SysML v1 to v2 Transition Guidance, organizations can strategically select and integrate digital engineering tools to enhance their engineering processes and maintain superiority against global threats.

Biography: Frank received his Bachelor of Science degree (BS) in Electrical Engineering (EE) from the New Jersey Institute of Technology (NJIT) and his Master of Science (MS) in Computer Science (CS) from American University (AU.)
Frank has been a consultant for SAIC since 1999 providing various degrees of Systems Engineering support to the U.S. Army, the US Navy, and the Office of the Under Secretary of Defense. Frank also worked as an employee for the U.S. Army and ITT industries providing a variety of systems engineering support for the development of Army Smart Munitions, GPS navigation payload, and a Software Programmable Radio. Frank is an Expert Systems Engineering Professional (ESEP) an Object Management Group (OMG) Certified Systems Modeling Professional (OCSMP), and Six Sigma Green Belt. Frank is also a member of the International Council on Systems Engineering (INCOSE,) is the Past President of the Liberty Chapter, the Chair of the Decision Analysis Working Group, and a Co-Chair of Digital Engineering Information Exchange Working Group (DEIX WG.)

#42 Adaptation Requirements for Department of Defense Contracts: A Systems Engineering Perspective, Afia Rahman

The US military's yearly expenditures are growing rapidly. As a result, the number of contractors hired and therefore, the number of contracts created by the Department of Defense (DoD) are drastically increasing. Contracts, especially those used for international affairs, are highly susceptible to fraud, waste, and abuse due to their complexity. The Afghan War was the longest war the U.S had initiated and was known for its exorbitant cost. Special Inspector General for Afghanistan Reconstruction (SIGAR) was created by the US as a response to provide oversight on the billions of dollars the US had provided to reconstruct Afghanistan. Systems engineering, specifically the development of requirements, can be used to modify, understand, and develop requirements that can improve the implementation of these contracts and mitigate financial waste. This paper performs a requirements decomposition on Request for Proposal (RFP) W56HZV-15-R- for Contract No. W56HZV17C0117, which was awarded by the Department of Defense Army Contracting Command (DOD-ACC) for National Maintenance Strategy – Ground Vehicle Support (NMS-GVS). Understanding the requirements written for the RFP will give insight into what was potentially missing from the contractual requirements, and, from a systems perspective, what needs to be considered for the future. With the creation of Adaptation Requirements, contracts written by the DoD will be better suited for nation development. This report proposes criteria needed to write Adaptation Requirements because the traditional approach to writing requirements may not be the most suitable for this scenario. The Adaptation Requirements in this report are not complete but are suggestions on how to start writing them. Requirements writing is a difficult process and requires the engineer to understand the environment, in this case an environment of another country. Advocating for further research in Nation Development and writing requirements cognizant of the nation will be beneficial for the US both economically and politically.

#43 Incremental MBSE : Deliver MBSE Value Faster, Randall Satterthwaite

Adoption of MBSE is often hindered by the steep learning curve which overwhelms users with a massive amount of information in a short period of time. This then becomes a hurdle to adoption as the user struggles to find out where to start and often gives up.

Approaching the learning through an incremental approach greatly improves the adoption from users. Focusing on delivering MBSE value faster to the user and building fidelity over time eliminates the overwhelmingness seen with traditional approaches. This serves to build a desire from the user to learn and adopt because of achieved value instead of pushing "This will help you in the long run" rhetoric that while true, requires users to go on faith rather than experienced value.

This new approach has demonstrated clear improvements in adoption and retention of new MBSE practitioners as well as enable non-system engineers to also learn and gain value from MBSE.

#45 Optimization of Requirements Management for complex Systems: An Innovative Approach with Product Classes and Attribute, Antonio Cristiano and Ernesto Barone

The aerospace industry, characterized by complexity and rapid technological developments, requires advanced solutions to effectively manage the design and development of complex systems. This study presents an in-depth analysis of the implementation of Model-Based Systems Engineering (MBSE) by AVIO, a leading aerospace company responsible for the development and production of launchers for the European Space Agency. Through the adoption of MBSE, the company is seeking to revolutionize the way it manages and understands the development of complex systems, the definition and verification of requirements, significantly improving the operational efficiency and overall quality of its projects. Avio is now implementing the MBSE methodology on a pilot programme involving all departments and corporate functions with the aim of developing knowledge to be subsequently applied on all projects. A complex SysML model is being developed in Dassault's Magic Cyber Systems Engineer currently containing hundreds of diagrams and thousands of requirements applied to the System of Interest and all its sub-systems and components. The whole requirement lifecycle is performed in SysML without use other requirements management tools: from initial requirement definition to verification, ensuring complete traceability, change management, and compliance verification.

The presentation will also focus on how this approach has been implemented within a complex industrial setting, such as a company with over a thousand employees, addressing all ensuing challenges like task and responsibility division.

This work explores the faced challenges, implemented methodologies and achieved results, with a particular focus on the innovative management of general (multi-product) specifications, offering a valuable perspective for companies seeking to optimize requirements management in continuously evolving aerospace environment.

A 'General Specification' refers to a type of requirement that is broadly formulated and can be applied to multiple products or systems within a business or project context. Unlike detailed specifications that are closely linked to a particular product or process, general specifications provide broader guidelines that can be adapted and implemented in various contexts.

For example, a ‘General Specification might address the overall performance of a product class rather than specific details of a specific item. It might include requirements for safety standards, compatibility with certain technologies or industry regulations that must be met by all products in that class. General specifications are useful when you want to establish basic guidelines or common criteria that must be met by a range of related products.

The presentation will detail the process of defining project-independent product types called ‘Classes’ represented in the model with the SysML entity ‘Block’. Classes are defined and linked to each other in a Block Definition Diagram (BDD) by means of a tree structure using the ‘Generalization’ relationship so that each product class is defined in greater detail than the higher-level classes. The approach is based on general SysML rules and is applicable to any environment utilizing this standard.

Empty value properties have been defined for each “Class”; these properties act as identifiers of essential characteristics that must be defined specifically for each product belonging to a given class. This approach allows extraordinary flexibility, making it possible to indicate the specific values of the properties according to the peculiarities of each real product (through SysML redefinition), without compromising overall consistency. These value-free properties represent a conceptual map of the fundamental characteristics of a product class, indicating the essential parameters that need to be detailed in later stages of the development process. This modular approach facilitates the product design evolution.

To optimize the management of general requirements, the company introduced an efficient mechanism that allows the gradual application of these requirements to product classes. This strategic approach allows each product belonging to a class to automatically inherit the general requirements of that class, greatly simplifying the definition of specific requirements at product level.

Thus, this approach has two advantages, it allows products to inherit both the properties already defined for the classes and the general requirements that the parent classes must fulfil.

For example, let us consider the product class 'Liquid Propellant Engine'. Each product within this class, such as the ‘M10 engine’ (engine of one of the launchers developed by AVIO), automatically inherits the general requirements defined at class level. This includes, for example, specific safety requirements for liquid propellant engines, energy efficiency standards, and other fundamental criteria. However, the flexibility of the structure also allows further special requirements for the M10 engine to be specified and adapted to its specific characteristics.

The scalable approach of this methodology is further demonstrated by considering the Product Class ‘Antenna’, which in turn belongs to the broader Class "Avionic Equipment." Within this hierarchical structure, the specific product ‘Telemetry Antenna’ inherits not only the general requirements defined for the Product Class "Antenna," but also those established at a higher level for the entire Class ‘Avionic

Equipment'.
In this context, the 'Antenna Class' could include general requirements relating to communication specifications, resistance to environmental conditions and specific safety regulations for antennas. The 'Avionics Equipment' Class, in turn, could define broader requirements that apply to all avionics equipment, such as interoperability with other on-board systems and electromagnetic compatibility.
In this way, the company benefits of an efficient process in which general requirements are naturally allocated and specific requirements are clearly defined using attributes as a guide. This methodology not only optimizes the time and resources spent on defining requirements, but also helps to build a cross-cutting and reusable structure that is totally independent to the specific project, ensuring better consistency, traceability and control in the overall management of requirements in the company's aerospace projects.

#47 Systems Engineering Agility – Guide Book Foundations for Systems Engineers, Rick Dove

Agile systems engineering can design, build, sustain, and evolve purpose-fulfilling creations when knowledge is uncertain and operational environments are dynamic. That is the promise – but what does it look like and how does it do that?
INCOSE is producing a 4-page Systems Engineering Agility Primer as part of its Future of Systems Engineering (FuSE) initiative, with publication expected mid-2024. The Primer focuses on the what (behavior) and why (needs) of 8 strategic aspects that enable and facilitate agile systems engineering. The Primer is intended as an introductory and motivational overview suitable for individual reading as well as support for group discussions, workshops, or tutorials that want to explore the strategic aspects in more detail with the aid of a succinct desk-top reference.
This presentation will include a quick review of the Primer content and then focus on how that material is being expanded into a more detailed 50-60 page Guide. The Guide is a current work in process with an INCOSE production target date of mid-2025. This presentation will show some finished treatment as well as expose some open questions about alternatives and offer opportunities for involvement in review and completion of the work.
This industry-agnostic Guide is currently developed as a five-section document, each of which will be discussed in the presentation. The Guide opens with sections on Purpose and Context, and then a third section devotes a few pages to each of the eight aspects – Adaptable Modular Architectures, Iterative Incremental Development, Attentive Situational Awareness, Attentive Decision Making, Common-Mission Teaming, Shared-Knowledge Management, Continual Integration and Test, and Being Agile: OpsCon. A fourth section discusses Design and Employment Considerations for each of the aspects, and a fifth section closes with a series of Case Stories on how each of the aspects is employed in a diversity of project domains.

Bio: Rick Dove is an independent researcher, systems engineer, and project manager generally focused in the system security and system agility areas. He chairs the INCOSE working groups for System Security Engineering, and for Agile Systems and Systems Engineering; and leads INCOSE's Future of Systems Engineering (FuSE) project areas for both systems engineering security and systems engineering agility. He is an INCOSE Fellow, and book author of Response Ability – the Language, Structure, and Culture of the Agile Enterprise.

#48 Toward an Anti-Security Security Primer for Systems Engineers, Rick Dove

In pursuing the Future of Systems Engineering (FuSE) initiative, INCOSE's working group for systems security has taken its mission from INCOSE's Vision 2035: "Security will be as foundational a perspective in systems design as system performance and safety are today." In examining the situation it appears that the current approach to systems security is itself systemic in nature – systems engineering's attitudes, processes, and actions remain consistent with tradition: security is a non-functional requirement, necessary to satisfy stakeholder compliance requirements and Authorization to Operate needs. Why and how it is time for this systemic tradition to change needs illumination.

Calling it like it is.
• Predatory hostility is an active characterization of a system's operational environment that eclipses passive characterizations that use words like threat, adversary, and cyber contested environments. Damage and destruction are the intended or ransomed outcomes.
• Complexity of attack and defense continuously increases as iterative incremental attack evolution makes yesterday's defense approach insufficient and obsolete.

Predatory hostility is not new activity, but featuring it as the bottom-line issue can change the way we think and deal with it. Increasing complexity is not a new situation, but understanding its cause and continuance can change the way we think and deal with it. The nature of predatory hostility constantly evolves ahead of systems not designed or supported for functional perseverance. With these thoughts in mind a different way of looking at things can lead to a different goal, with a different set of objectives, strategies, and requirements. That's not to say what is being done should be stopped; rather what's being done should be repositioned within something completely new and practical that more directly addresses situational reality.

This presentation will make the industry-agnostic case for a change in mind set and goal, advance a framework of strategies, and articulate a vision of acceptance.
Mindset: Hostile predatory environment.
Goal: System functional perseverance in a hostile predatory environment.
Strategies: Protect, Defend, Recover, Evolve.
Vision (of a sustainable outcome):
• SEs are comfortable and natural with security as an obvious and necessary first design priority.
It is not perceived as a burden or distraction.
• SEs intuitively recognize, feel a sense of threat, and feel a need to correct when this isn't the prevailing situation.
A sense of wrongness prevails.

We will show that security is not simply a functional requirement, but rather a prerequisite of system's functionality and performance. We will show that SEs don't need to learn new fundamental skills, only how to apply those skills to:
Security requirements development, verification, and validation while sustaining a continual sense of relevant awareness.

This foundation is guiding a Security Primer for Systems Engineers, in final stages of development by INCOSE's Systems Security Engineering working group.

Bio: Rick Dove is an independent researcher, systems engineer, and project manager generally focused in the system security and system agility areas. He chairs the INCOSE working groups for System Security Engineering, and for Agile Systems and Systems Engineering; and leads INCOSE's Future of Systems Engineering (FuSE) project areas for both systems engineering security and systems engineering agility. He is an INCOSE Fellow, and book author of Response Ability – The Language, Structure, and Culture of the Agile Enterprise.

#49 Towards Reliable Embedded Systems: A Review of Hardware Reliability Challenges, Ryan Aalund and Vincent Paglioni

Ensuring the reliability of (cyber-physical) hardware is a critical aspect of building any reliable system, but perhaps even more important to creating reliable embedded systems, wherein multiple pieces of electronic hardware communicate and interconnect to accomplish a larger system objective. In practice, embedded systems are critical to many common and critical industries, from “smart” home appliances and wearable devices, to medical devices and mission-critical aerospace applications. The sheer pervasiveness of embedded systems, across a range of diverse fields, underscores their importance in the technological landscape.

As embedded systems power a revolution in industry, ensuring the reliability of these systems has become an increasing challenge. This is due in part to the increasing complexity of embedded systems and growing portfolio of interconnected applications, and partly to foundational gaps in our understanding of software, cyber, and cyber-physical reliability. Additionally, although hardware reliability concerns have been the subject of significant research and techniques for assessing hardware reliability are widely available, there is a lack of focus on the integration into larger embedded systems. This has further complicated our ability to understand and assess the reliability of embedded systems architectures. This lag is particularly concerning as connected systems and the “internet of things” (IoT) architectures are rapidly being introduced into more critical systems.

In response to these issues, we provide a review of the different layers of hardware involved with embedded systems, including processors, memory, power, communications, and peripherals. Furthermore, for each layer, we assess the purpose, reliability challenges, and current mitigation strategies as well as the techniques, tools, and methods available to assess hardware reliability. This review summarizes the state of the industry regarding embedded system hardware reliability, discusses the current research developing novel methods to address the identified challenges, and creates a path toward improving our understanding of the applications, problems, and potential solutions in embedded systems reliability.

This review is the first piece of a larger research project aimed at taking a higher-level systems approach to appropriately assess the reliability of embedded systems, including hardware, software and layer interactions. We propose that a systems-level view of reliability will expose more challenges and identify additional mitigation strategies, in addition to ensuring that the objectives of these systems are maintained. Understanding reliability from a systems perspective will ensure that all layers of embedded systems are supporting the function of these critical systems.

#50 Resident Pathogens in Systems Engineering: Case Study of Accident Analysis of Boeing 737 Max-8 Crashes, Sanjeev Appicharla

. The aim of the presentation is to present “resident pathogens” identified in Systems Engineering (SE) practices in the aviation industry that led to the two fatal accidents of Lion Air Flight 610 and Ethiopian Airlines Flight 302 involving Boeing 737 Max- 8 airplanes. The fatal accidents led to the tragic loss of 346 lives. A Cybernetic Risk Management model drawing upon Nobel Prize winning Heuristics and Biases (H and B) approach is used to study the accidents and for the audience extend the results published to raise awareness of and improve SE Professionalism. (Appicharla.S, 2023a) may be consulted for the results already published.

The Systems Engineering (SE) Handbook version 5 noted,thus: Professionalism can be summarized as a personal commitment to professional standards of behavior, ethics, obligations to society, the profession, and the environment. SE practitioners are trusted to apply reasoning, judgment, and problem solving to reach unbiased, informed, and potentially significant decisions because of their specialized knowledge, skills, abilities, and behaviors. SE professionalism includes consideration of personal behaviors beyond using methods and tools. SE practitioners recognize the benefits of behaviors and outcomes related to professional competencies from ethics, professionalism, and technical leadership to communications, negotiation, team dynamics, facilitation, emotional intelligence, coaching, and mentoring." (section 5.1.2)(pp.263).

In the section on Ethics, The SE Handbook version 5 noted, thus: "As stated in the INCOSE Code of Ethics (2023), The practice of SE can result in significant social and environmental benefits, but only if unintended and undesired effects are considered and mitigated. Part of the role of the SE practitioner as a leader and professional is knowing when unacceptable risks or trade-offs are being made, knowing how to in-fluence key stakeholders, and having the courage to stand up for stakeholders, the community, and the profession when necessary (International Council on Systems Engineering (INCOSE), 2023)(section 5.1.4), (pp. 264).

However, in this context or risk management, the available counter-factual evidence shows that organisations exhibit a tendency to silence engineers who raise such concerns, indicating a resident pathogen in the SE communication, decision making and risk management processes. For example, this tendency is demonstrated through the NASA Columbia accident case studies where engineers became silent after making several tries (Starbuck, W., et al, 2005)(pp.251). Organisational routines followed by managers and engineers in the NASA Case Study of Columbia disaster were different despite the same signal of foam strike led them different perceptions of the event (Starbuck, W., et al, 2005)(pp.251).

From the perspective of classification of information processing in accordance with the Skills -Rule-Knowledge based Human Performance Framework and the eight stage decision Ladder Frameworks, managers were acting in accordance with the rules based behaviour and engineers were acting according to Knowledge based behaviour (Rasmussen, J, 1983), (Rasmussen,J, Pejtersen, A.M , Goodstein L.P, 1994). The technical routines followed by the engineers suggested that the proper course of action for responding to the debris strike was to acknowledge the uncertainty of the event, attempt to gather data, and proceed with a technical inquiry (Starbuck, W., et al, 2005)(pp.251). Affected by the same stimulus, the responses of the engineer, chief structural engineer Rodney Rocha and of the manager, Mission Management Team chair Linda Ham, manager led them to deny certainty of the data to formulate the hypothesis that there is a safety of flight issue by the former and accept confirmation of the hypothesis that there is no safety of flight issue in the case of latter.

Milliken, F.J., Lant, T.K., Bridwell-Mitchell (Chapter 13) note thus: " if Ham and Rocha were seeking to test their respective hypotheses, they would have both sought to find counter-evidence for their theories. What is the counterevidence for flight safety? When Ham searched she found none. This result was due, however, to the bias of her sample. Her decision not to obtain flight imagery effectively censored her sample so that possible counter-evidence was not included in the pool. Yet not observing counter-evidence does not mean it does not exist. This is a basic, though often ignored, principle of scientific inquiry. Unfortunately, the Columbia accident was not the first time NASA managers fell prey to the fallacy of making conclusions even though there was missing data. Similar use of censored data resulted in NASA managers incorrectly predicting O-ring safety on the cold January morning of the Challenger accident (Starbuck and Milliken, 1988). What about the counter-evidence for unknown flight safety? Rocha's insistence on obtaining outside imagery demonstrates his hope of strengthening the sample of data on which he could base his conclusion. There were no data to challenge the claim that flight safety was unknown. In fact, even if the requested shuttle images had been received, flight safety could only be constructed from probabilities and would remain partially unknown. The decisions of the NASA managers were bounded by past routines, frames of reference, and operational pressures, leading to differences in their interpretations of the same stimulus and, in some cases, biasing their interpretations, both of which made it difficult to proceed with the next steps in the learning process." (Starbuck, W., et al, 2005)(pp.253).

(Wikibooks contributors, 2015) discuss the professionalism concern of the above individuals and conclude that the organisational culture and hierarchy environment at NASA was the major contributor to the tragedy. (Rocha R. Aerospace Engineer, NASA Johnson Space Center, 2011) noted the non - technical factors of Emotional & attitude factors (anger, upset, distress, arrogance, denial,…) involved in the case study – Personalities clashing; Personalities clashing; the "Rocha filter" as a perception of overstating a problem Urgent concern is so weak in evidence, it then "drops through the crack" with no further action to investigate or resolve. – Examples: "I can't work with an angry person." "You're not a team player." Emotional interaction factors are real and still can color what should be technical, rationale discussion is one of the cautions issued by Rodney Rocha for future high risk programmes. (Appicharla.S., 2023b) may be consulted for more details on the NASA Accident studies.

#51 Using SysML v2 to Define an MBSE Methodology, J. Simmons and Tony Davenport

One of the first lessons of Model-Based Systems Engineering (MBSE) is that it takes three things to execute MBSE: a descriptive modeling language, a compatible modeling tool, and an MBSE methodology. While the need for the first two is obvious (one cannot develop a model of any kind without a language and compatible tooling), the need for an MBSE methodology is not always so clear.

A well-defined methodology goes beyond the specification for the language and the features of the tool to formalize the use of the architecture model to perform systems engineering activities. By formalizing the use of the architecture model with complete and unambiguous guidance, an enterprise can realize tangible improvements in its MBSE practices. These improvements include ensuring architecture models include sufficient detail without giving into the temptation to "over-model", developing self-consistent, reusable architecture models with teams ranging in size from 5 to 500, and providing a shared basis of model-based communication between systems engineers and stakeholders.

This presentation will describe an architecture for a well-defined MBSE methodology in SysML v2 that includes verifiable data structures and approaches as well as templates and work instructions for end users to follow, derived from the verified data structures and approaches. The presentation will then provide an overview of a model-based approach for developing and customizing this methodology and concludes with a detailed description of an example process from the presented MBSE methodology and a demonstration of executing the process.

While the call to formalize the use of models is specifically called out by the US Department of Defense in its 2018 Digital Engineering Strategy, the use of a well-defined MBSE methodology is critical to any Digital Transformation effort. Industries that will benefit from applying the lessons of this presentation include US Aerospace and Defense, Automotive, Energy Production and Storage, and Medical Devices.

During this presentation enterprise leaders will learn why adopting a well-defined MBSE methodology is an essential part of a digitally transforming an enterprise and how to recognize when an MBSE methodology is well-defined. Systems Engineering department leaders will learn how they can plan and lead a model-based approach for developing a well-defined MBSE methodology in SysML v2. And Systems Engineering practioners will learn how they can leverage a well-defined MBSE methodology to perform familiar tasks directly within a SysML v2 architecture model.

Dr. J. Simmons is a Digital Engineering Consultant with nearly 20 years of experience in Digital and Systems Engineering. He holds a Ph.D. in Space Systems Engineering from the Air Force Institute of Technology. Dr. Simmons specializes in the development and execution of MBSE process, with an emphasis on model integration and the use of digital threads to support analytically driven decision making. His industry experience covers all corners of US Aerospace and Defense including US government, its contractors, and national labs. Before working as an independent consultant, Dr. Simmons served as a Digital Engineering manager and an internal MBSE consultant at Northrop Grumman. He is presently co-leading the development of the MBSE methodology described in this presentation.

Mr. Davenport is the director of the North American Systems team at Ansys. He has spent over 30 years focused on working with organizations to perform "smart" digital transformation. Most recently, the integration of system engineering with mission, physics, embedded software, digital twins, safety, and cost models has enable new technologies that allow organizations to connect with their suppliers and their customers for making complex product decisions and win more projects. Mr. Davenport and his team of system engineering experts at Ansys are helping customer put open architectures in place that allow them to exceed their organizations goals and connect system of systems for all parts of the product life-cycle.

#52 Examining The Impact of Prompting GAI: A Comparative Analysis of Testing Strategy, Jennifer Giang and Steven Simske

Generative Artificial Intelligence (GAI) is a class of AI that generates data from human-provided parameters, data, and input. GAI has significantly advanced in the past decade. The parameters act as constraints and guidelines to the model with the relative flexibility of the GAI approach being contextual and algorithm dependent. GAI can provide generative novel content more complex than traditional predictions and pattern identification. A concern with AI is the level of bias within the training dataset that leads to unfair outcomes and societal inequalities. This study focuses on a comparative analysis of how adapting GAI prompting and testing strategy fosters inclusivity and equality, from a system engineering perspective. We explore how iterative interventions can reshape the GAI job recommendations between males and females. Specific related industries include technology, healthcare, business management, and human resources.

The experiment was conducted in two phases: control and preliminary. The control was prompted by a GAI model, ChatGPT 3.5, on 10 job recommendations for both women and men, 10 tests each. The preliminary phase added guidance to the system prior to executing the same 10 prompts for each women and men job recommendations. The guidance focused on encouraging job diversity, focusing on skills and qualifications, avoiding gendered language, considering intersectionality, promoting fair presentation, promoting gender-neutral policies, and including a reminder to be mindful of bias in training data.

In the control phase testing, the job recommendations showed variations in alignment with gender stereotypes and societal norms. Some recommendations were traditionally associated with specific genders such as women being teachers and nurses, and men working in trade or emergency services. Some recommendations were more diverse such as a variety of engineering, physician, financial analyst, and marketing manager recommendations.

In the preliminary phase testing, the guidance that encouraged diversity and mitigating gender bias aimed to guide the model towards recommending diverse jobs not related to gender and demographic factors. This testing phase revealed significant shifts in the recommendations compared to the control phase. The tests exhibited a more equitable distribution of jobs across genders, with a reduction in gendered language and stereotypes.

Comparing the two phases, job recommendations for women are reduced after guidance is applied in the education, healthcare, and business industries, and significantly increased in STEM. For the men, it shows a significant decrease of trade job recommendations from 37% to 1% after guidance is applied,

with a concomitant increase in the recommendations in the STEM and Healthcare industry.
From a system engineering perspective, this study presses the importance of iterative testing and adaptation of testing approaches of GAI systems. By incorporating prompts and interventions at various stages of testing, developers can refine their model algorithms to align with principles of fairness, inclusivity, and responsible practices. Integrating feedback loops and continuous improvement mechanisms aid in the refinement of GAI systems.
In conclusion, this study demonstrates how iterative testing, feedback mechanisms, and proactive interventions can improve the testing approach of GAI for a more effective and accurate system. Future research will explore other factors that can address the ethical concerns of GAI systems and using other GAIs to test prompt outputs.

#53 Systems Engineering for Developing Tech Standards: Lessons Learned, Artis Riepnieks and Kaustav Chatterjee

In the modern technology ecosystem, documentary standards are the key enablers in ensuring technology interoperability, interchangeability, and consistency. Ideally, these documents should strive for the perfect balance between two seemingly contradictory objectives -- specificity of application and generalizability of design. Deviations from this perspective might limit the usability and adoption of an otherwise well-written standard. A formal approach built on the elements of systems engineering can significantly improve the technical standards writing process and enable working groups to deliver on these expectations. To this end, with this presentation, we share our experiences of working with an electrical measurement standard -- the IEEE 1459 “Standard Definitions for the Measurement of Electric Power Quantities Under Sinusoidal, Nonsinusoidal, Balanced, or Unbalanced Conditions”. The presentation covers lessons learned from the applications of systems thinking, systems approach, and elements of systems engineering (such as requirements engineering) in the drafting of this documentary standard. The presentation cites specific instances where the systems engineering approach exposes the gaps in standardization processes and provides suggested steps for improvements. The presentation shares insights drawn from the application of the approach and the lessons learned within the field of electrical engineering. The broad theme of the presentation is systems engineering with slightly more benefit to those with measurement instrumentation, electrical engineering, or standard development background.

#54 [WITHDRAWN] Integrating BOM Evaluation for Enhanced Validation of SysML Models, Chandrima Ghatak, Rik Chatterjee and Jeremy Daily

In the field of systems engineering, ensuring the fidelity of system models to their real-world implementations is paramount. System Modeling Language (SysML) serves as a pivotal tool in this domain, offering a formal framework to document, analyze, and visualize complex system interactions, requirements, and architectures. However, aligning theoretical SysML models with their actual implementations poses a significant challenge. Discrepancies often arise from variances between the model's expectations and the actual execution within engineering practices. Traditional validation methods, which predominantly rely on theoretical assessments and testing, may not fully capture these discrepancies.
To bridge this gap, our research introduces a strategy that utilizes the integration of Bills of Materials (BOMs) to critically evaluate and empirically validate SysML models against their real-world implementations. BOMs are essential artifacts of the engineering design process, developed and refined throughout the product lifecycle. They provide detailed inventories of all necessary components, both hardware and software required for building a system. Our approach begins by generating enhanced BOMs, which are comprehensive and include annotations that detail the interconnections and

dependencies between components. These enhanced BOMs are derived from initial BOMs produced by hardware design tools and software development environments. For hardware, we assemble detailed descriptions of each component, including part numbers, specifications, and supplier information through an automated process. For software, using Clang tools, we generate an Abstract Syntax Tree (AST) to extract detailed data about functions, libraries used, and extensive implementation details from the source code. These elements are then integrated into a unified BOM that provides a complete view of the system's physical and software components. Simultaneously, we extract data from SysML models, focusing on block definition diagrams that represent the architectural layout of the system. These diagrams are converted into XML format, which facilitates the extraction of component hierarchies, relationships, and properties. The next step involves translating both the enhanced BOMs and the SysML-derived data into a graphical block diagram format. This visualization enables a direct, side-by-side comparison of the diagrams, providing a clear visual juxtaposition that serves as an effective cross-reference. This method allows for an empirical validation of the SysML model against the actual components of the system.

The practical application of this method is demonstrated through its implementation in the design of an embedded system for a vehicular data logging device currently under development. This application not only showcases the method's effectiveness in enhancing the accuracy and reliability of system designs but also underscores the critical role of BOMs in fostering a robust, integrated, and iterative systems engineering process. By providing a tangible, systematic method to ensure model fidelity, this approach addresses a crucial gap in traditional systems engineering practices.

#55 Space Mission Engineering using Innoslate(R) with example mission, Jim Adams

Mission Engineering is the organization of existing or emerging capabilities into system elements (the architecture) and the element activities (the concept of operations) to effectively achieve a desired goal or objective. Mission Engineering prepares System Engineering teams to develop application use cases and the requirements for the recommended architecture.

A process for Mission Engineering of Space Systems is defined in "Space Mission Engineering: The New SMAD" (Wertz, Everett, Puschell, 2018, Space Technology Library). This presentation reviews an implementation of this process in a Model-Based Systems Engineering (MBSE) tool – Innoslate®. The implementation includes a Dashboard to highlight the key components of the process along with tailored database element types to capture the content of the Space Mission Engineering process. One significant benefit of Innoslate® is the integration of programmatic aspects for full lifecycle engineering. This presentation includes discussion of an example space mission using the process. Mission Engineering for other applications can follow a similar process using similar tools.

#56 System of Systems Engineering and Analysis Nathan Dunson

Overview

We live in a world that is becoming more and more technologically complex. In this technological complexity exist systems that we must interact, work and deal with daily in our lives. On the surface, these systems appear to be singular view of a human interface into a larger capability. As one digs deeper into these systems, it is revealed that these are larger capabilities that are comprised of System of Systems (SoS) in an Enterprise. Based on ISO/IEEE/IEC 21839 (2019) definition of System of Systems as: "A set of systems or system elements that interact to provide a unique capability that none of the constituent systems can accomplish on its own."

In today's world most systems of interest that are being developed, updated (technology refreshed), or modified are part of a larger Enterprise SoS that all must interface, communicate, pass data, share information all while performing to requirements. This presentation will dive into considerations a

Systems Engineer (SE), must take when architecting, designing, developing, and testing in a SoS. Within SoS Engineering (SoSE) there are considerations that must be addressed by the SE that will be included from the following short descriptions.

1.0 Stakeholder needs

Stakeholder needs related to a newly developed system, an updated system or a modified system that is part of a larger SoS will be discussed. This area will cover impacts, relationships and constraints placed in other areas of consideration.

2.0 SoS Architectures

This consideration will review and discuss novel concept approaches that can be used to define and develop logical and physical architectures within a SoS. It will discuss modeling approaches to support these concepts to provide needed views and perspectives for customers.

3.0 SoS Requirements

The requirements for SoS will be reviewed with discussion on how to integrate new functional and non-functional requirement into an existing requirements baseline. These approaches will identify the areas of concern related to relationships into performance, interoperability and integration and test areas of consideration.

4.0 Constituent Systems

When adding a newly developed system, updating an existing system, or making modifications to an existing system the constituent systems must be taking into consideration. This at times can be a difficult task as constituent systems can be Intellectual Property of a third party.

5.0 Interoperability within SoS

One of the key considerations will be interoperability within the SoS of interest. This will have relationships with many of the other considerations list here. An in-depth look into interoperability consideration is paramount in the overall SoSE and analysis.

6.0 SoS Performance

A depth investigation and analysis of the develop, updated or modified system performance and its relationship to other constituent systems is a needed consideration. This requires a more detailed understanding of the existing SoS and of its constituent systems. This area will review and discuss control mechanisms to support the new or updated system.

7.0 Human Systems Integration (HSI) of SoS

The overall consideration of HSI is typical lost within the development of a part of the SoS. The complete SoS and newly developed, updated, or modified system must be considered in the overall architecture and design.

7.0 Cyber Security

In today’s environments Cyber Security is an absolute on requirements and considerations. The overall cyber posture of the SoS is a must to any developed, updated, or modified system within the SoS. This area will dive into implications of Risk Management Framework, Supply Chain Risk Management, and emerging requirements around Zero Trust.

8.0 Integration and Test

The backside of the “Vee” must be considered as part of the overall SoS. Implications on impacts of architectures and designs will be reviewed and discussed.
The unique perspectives and experiences from the author will be shared along with pitfalls in dealing with SoSE at an Enterprise Level. Viewpoints will be provided within the context of the SoS from perspectives of customer community in the likes of stakeholders, funders, users, maintenance personnel and sustainment teams. In addition to the Engineering Systems, hardware disciplines, software, and test disciplines.

Industry Relevance
The overall SoSE approaches and considerations presented in this brief can and should be applied across many industries including banking, transportation, defense, entertainment, and others. These industries all have unique capabilities developed and built to form a SoS that is made up of many constituent systems. The inter-relationships between these constituent systems are key to the overall value and performance provided to stakeholders, customers, and users of these SoS. Each one of these SoS can include all discipline areas including systems, hardware, software, cyber, HSI, reliability and other specialty engineering areas.

References
This presentation will be based on several references that will include but will not be limited to:
- INCOSE Systems Engineering Handbook, Fifth Edition
- Guide to Systems Engineering Body of Knowledge (2023)
- ISO/IEEE/IEC 21839 (2019) Systems and software engineering - System of systems (SoS) considerations for life cycle stages of a system
- ISO/IEEE/IEC 15288 (2023), Systems and software engineering – System life cycle process
- The Art of System Architecting, Third Edition, Maier and Rechtin (2009)
- Architecting Principles for System of Systems, System Engineering, Maier (1998)

About the Author
Nathan Dunson is a proven technical leader and Systems and Software Engineer over his 38-year career. His career has spanned across industries Navigation and Positioning systems used Hydrography, Dredging and offshore oil exploration (6 years), Telecommunications (15 years), and Defense (17 years). Nathan’s primary focus has been architecting, design, development, test, and delivery of these Systems. Most if not all these systems delivered are part of a larger SoS or Enterprise SoS. Thus, providing the foundation for Nathan’s understanding of SoSE and the unique challenges, complications, emerging behaviors and requirements, and system performance that must be considered. Nathan’s experience in SoS and Enterprise SoS capabilities provides a unique perspective the challenges engineers will be faced as systems are added, updated, and modified within a larger SoS. His current position is a Chief Engineer at Collins Aerospace.

#57 Addressing the Upstream Ecological Impacts of Engineering Decisions, Casey Medina and Rae Lewark

Systems Engineering is becoming more aware of the environmental impacts of systems and materials at the distal end of the life cycle. The terms "sustainable, "green," and "environmentally friendly," are being woven into our professional vernacular. As a result, the focus is more often than not on the utilization and end-of-life phases of these systems and materials. Are we addressing the whole ecological picture? Not hardly – as Systems Engineers, we must apply systems thinking across the entire lifecycle of our systems and the materials we select to fuel and sustain them.

This presentation explores considerations that enable us to identify the environmental impacts of upstream processes and engineering design decisions. Through a wholistic approach, we address the need to view the entirety of a system's reach and not only the minimal snapshot required. Through useful insights, we can make informed decisions across the entire lifecycle of a system.

#58 Ultimate Track Hacking Platform (UTHP)- Software Bill of Material (SBOM) Life Cycle Modeling, Teddy Nyambe and Jeremy Daily

Title and Overview
The title of the presentation is Ultimate Truck Hacking Platform (UTHP) Software Bill of Material (SBOM) Life Cycle Modeling. The Ultimate Truck Hacking Platform (UTHP) is an initiative under the Systems Engineering Department at Colorado State University, spearheaded by the System Cyber cybersecurity team. This project is dedicated to developing comprehensive hardware and software tools for ethical hacking of heavy-duty trucks using the J1939 protocol, aiming to identify vulnerabilities and safely report these findings to NIST for onward remediation. The presentation aims to provide an overview of how the System Cyber Research Team leverages the Yocto Linux build environment to produce a Software Bill of Materials (SBOM). Additionally, it explores the use of custom software tools to analyze the SBOM, helping to fully understand the software composition of the UTHP project. This approach allows for a comprehensive assessment of the software components licensing, versioning, and related components, crucial for ensuring software integrity and security. The project is further interested in responding to cybersecurity vulnerabilities reported to Security Vulnerability and Exposures databases with an SBOM as the tool for efficiently identifying affected software components and versions. The SBOM will also provide a source of provenance for stakeholders interested in reviewing licensing software for compliance and standards.

Related Industries
Following the issuance of the Biden Administration's Executive Order 14028 on improving national cybersecurity, contractors supplying software products to the U.S. government are required to include a Software Bill of Materials (SBOM) with their deliveries. This directive has significantly impacted various industries, extending beyond contractors directly contracting with the government. Sectors such as healthcare, defense, space, retail, and manufacturing have all had to adapt to these new requirements, integrating SBOMs into their software development and delivery processes to ensure compliance and manage cybersecurity risks. This highlights the government's strategy to strengthen the software supply chain across critical infrastructure sectors. However, beyond compliance, the production of an SBOM has evolved into a strategic decision to proactively mitigate cybersecurity threats across operations layers of an organization. This shift underscores the broader recognition of SBOMs as essential tools in enhancing software security and managing vulnerabilities effectively. While the strategic adoption of SBOMs has offered significant benefits in mitigating cybersecurity risks, it also presents considerable challenges. These challenges include the potential exposure of a company’s proprietary secrets, which are crucial for maintaining a competitive edge; in such cases, stringent privacy measures may need to be applied. Additionally, the risk of tampering with an SBOM, either while at rest or in-Transit, could compromise its integrity. Such breaches can lead to serious legal consequences.

What the Audience will take away
Following the presentation, attendees will gain a deeper understanding of the Yocto Linux embedded software development environment, enhancing their knowledge and skills in the tool for building customized operating systems. The discussion will also demonstrate the application of truck hacking

tools for ethical hacking of heavy trucks using the J1939 standard. Finally, the audience will gain insight into the specific uses of Hardware Security Modules (HSMs) for enhancing the security of a SBOM. The presentation will cover how cryptographic operations facilitated by HSMs can securely improve the integrity of SBOM data, along with best practices for storage of cryptographic keys and certificates.

Background of the Presenter
Teddy Nyambe, a seasoned software developer and systems administrator with over 15 years of experience from Zambia. He has contributed to both private and governmental sectors, developing software solutions that enhance productivity. Teddy holds an undergraduate degree from the University of Greenwich in the UK and a Master's in Computer Information Systems (MCIS) from Colorado State University, Fort Collins. Currently, he is advancing his expertise as a Ph.D. candidate and Research Assistant in the Systems Engineering department, where he continues to engage in cutting-edge research and development in areas of cybersecurity and application of model-based systems engineering (MBSE).

Conclusion
The presentation will delve into the use of the Yocto Linux environment to craft tailored solutions for embedded systems, showcasing a hands-on demonstration of setting up this environment to build an embedded Linux system. Additionally, attendees will witness the generation of a Software Bill of Materials (SBOM) within this development framework and also how to secure it using HSM to strengthen the integrity of SBOM during rest and transmission to stakeholder.

#60 Systems engineering for scientists and aliens, Pieter Kotze

The National Radio Astronomy Observatory (NRAO) was founded in 1956. The NRAO provides state-of-the-art radio telescope facilities for use by the international scientific community. It is funded by the National Science Foundation (NSF) under a cooperative agreement with Associated Universities, Inc. (AUI), a science management corporation.

Initially, activities were centered at Green Bank, West Virginia, focusing mainly on single-dish radio astronomy, which remained the primary focus for many years.
In 1966, the Very Large Array (VLA) a 27 antenna radio interferometer design work group began its efforts. Congress approved the project in 1972, and construction began in 1973. In 1982, the proposal for the Very Long Baseline Array (VLBA) was made, and by 1993, the first 10 antennas of the VLBA were operational.

In popular culture, the Very Large Array (VLA) is well-known for its appearances in movies, television shows, and other media. Notable examples include: "Independence Day" (1996) and "Contact" (1997). The VLA is prominently featured in these as the facility where scientists receive their first signals from an alien civilization.

Benefiting from technology upgrades through the years the VLA is still the premier centimeter wavelength telescope in the world and one of the most productive telescopes ever built. With such a long history one needs to consider what comes next to stay at the forefront of research.

The next-generation VLA or ngVLA is currently in planning. It will have ten times the sensitivity and ten times the resolution of the VLA, while being constrained in operational costs. The ngVLA is intended to replace both the VLA and the VLBA after its construction

The science community has determined what the instrument should do, identifying five key science goals:

Unveiling the formation of solar system analogues on terrestrial scales.
Probing the initial conditions for planetary systems and life with astrochemistry.
Charting the assembly, structure, and evolution of galaxies over cosmic time.
Using pulsars in the Galactic Center as fundamental tests of gravity.
Understanding the formation and evolution of stellar and supermassive black holes in the era of multi-messenger astronomy.

A brief but broad technical outline of the ngVLA to show the scale of the project:
The instrument covers a frequency range of 1.2 to 116 GHz across six bands or feeds.

The Main Antenna Array consists of 244 offset Gregorian antennas, each boasting a diameter of 18 meters. Meanwhile, the Short Baseline Array comprises 19 antennas, with each having a diameter of 6 meters.

These antennas will be centered on the current VLA site, approximately 114 CORE antennas within a radius of about 2.2 kilometers. From this central location, five SPIRAL arms extend up to 20 kilometers, while an additional 46 MID antennas start at 26 kilometers from the core, extending up to 700 kilometers. Furthermore, 30 LONG baseline antennas are situated at 10 distinct locations, organized in groups of three.

Astronomical data is transmitted back via directly connected array fiber. However, for longer distances, plans include utilizing dark fiber or commercial networks.

Reference timing generation and distribution are centrally managed to supply downconversion oscillator signals, digitizer sample clocks, and pulse-per-second (PPS) timing references to the antennas over fiber. In cases where antennas are not directly connected, local duplicates of the references will be installed.

The instrument captures the RF signal post-downconversion directly at the feed on the antenna. These samples are then transported via Ethernet over fiber to a central electronics building. Within this facility, a Central Signal Processor or correlator, employing FPGAs and GPUs as needed, processes the signals. The resulting output from the correlator then undergoes additional processing utilizing computing resources estimated to peak at around 100 PFLOPS.

The System Engineering approach required to build a complex instrument such as this needs to be specific to the system and the establishing organization. Different organizations and industries have varying opinions on whether the value of Systems Engineering (SE) justifies its cost and effort.

A project-specific Systems Engineering Management Plan (SEMP) was developed, tailored to the unique characteristics of the instrument, while also taking into account the guidance provided by the National Science Foundation (NSF) in its Research Infrastructure Guide (RIG).

The presentation aims to further address the SE approach followed, the tailoring to the SE processes for sub-systems and the current state of the project.