26th Annual INCOSE International Symposium (IS 2016)
Edinburg, Scotland, UK, July 18-21, 2016

Agile Systems Engineering Process Features Collective Culture, Consciousness, and Conscience at SSC Pacific Unmanned Systems Group

Rick Dove
Paradigm Shift International
rick.dove@parshift.com

Bill Schindel
ICTT System Sciences
schindel@ictt.com

Chris Scrapper
SSC Pacific Unmanned Systems Group, US Navy
chris.scrapper@navy.mil

Abstract. SPAWAR System Center Pacific (SSC) in San Diego has established an Unmanned System Integration, Test, and Experimentation (UxSITE) capability to facilitate agile development of unmanned systems. This capability utilizes a unique agile systems engineering process with 6-month overlapping “waves” consisting of four phases: development by multiple-subcontractors, and operational management of systems architecture evolution, capability integration, and validation testing. The UxSITE capability supports a portfolio of projects, and has three years of respected and effective results. Most notably, the process puts a prime emphasis on enabling and facilitating team effectiveness: creating an embraced culture of engagement, a collective consciousness emerging from comprehensive real-time information support, and a team conscience on a mission for the customer.

Introduction

An INCOSE project-in-process is seeking a generic Agile Systems Engineering Life Cycle Model (ASELCM), and is doing this by analyzing and building case studies of agile systems engineering in a multifaceted variety of applications, collectively covering agile software, firmware, and hardware systems engineering processes in experienced practice. The objective of the project is to discover and justify process principles as repetitively employed patterns, which are necessary and sufficient for any system engineering process that must contend effectively with an unpredictable, uncertain, and evolving engineering environment; and to document case studies that show how those principles are employed in the context of different engineering process environments.

The first 3-day workshop, held August 5-7, 2015, analyzed a unique and highly effective agile SE process employed by the SPAWAR System Center Pacific Unmanned Systems Group in San Diego to establish an Unmanned System Integration, Test, and Experimentation (UxSITE) capability. It is the uniqueness and effectiveness of the UxSITE evolutionary system engineering approach that prompts this paper. The process to be described was created to replace a process plagued by cost overruns, missed schedules, inadequate development achievement, uncooperative teaming, and poor status visibility. New-process effectiveness has been demonstrated consistently over 3 years in lower and predictable costs, on-time capability deliveries, and continual advancements on the overall performance of the systems under development. The effectiveness of the UxSITE capability is marked by SSC Pacific's interests in migrating this approach to other programs and operational domains.

Notable process concepts that will be discussed include:

- Common process spanning a portfolio of projects.
- Retained architecture ownership.
- Systems engineering structured as a Wave-model-inspired evolutionary process.
- Continuous integration with comprehensive regression testing.
- Clear unambiguous roles and responsibilities.
- Common culture embracing development contractors.
- Ubiquitous real-time shared awareness.
- A sense of personal mission.
- Quality-of-engagement sensitivity.
- Distributed test threads and continuous risk management.
- Comprehensive stake-holder involvement.

The portfolio of projects employing UxSITE span U.S. DoD RDT&E 6.1–6.3 categories; where 6.1 and 6.2 explore and develop new technologies for their potential in military applications, and 6.3 covers advanced technology development meant to help the transition from the laboratory to the field and includes Advanced Technology Demonstrations (ATDs) and Advanced Concept Technology Demonstrations. "Taken together, these first three activities (6.1-6.3) constitute what is called DOD's Science and Technology (S&T) program. The S&T program does not support development in a formal acquisition program, although as one goes from 6.1 to 6.3, the connection to a specific military operational capability becomes more important and apparent." (Moteff 1998).

The UxSITE project portfolio involves hardware and software development, and deployment testing, for unmanned expeditionary autonomous ground vehicles. Unlike self-driving cars, these vehicles have to maneuver over and through rough, obstacle-laden terrain, for which suitable technology does not exist. The project context is a platform-based system of systems consisting of sensor processing systems, world modeling systems, behavior generation systems, collaborative control systems, and platform specific control systems.

The project began in US Government fiscal year 2010 with a typical waterfall approach, which exhibited difficulties in achieving sufficiently demonstrable progress. Inspired by an agile systems engineering approach called the SoS Wave Model (Dahmann 2011, 2012), a new evolutionary systems engineering process went live in fiscal year 2013 specifically for the UxSITE project context, one that would put an emphasis on testable capability development in 6-month increments of capability advancement.

A paragraph from the U.S. Defense Acquisition Guidebook (DAU 2013) explains the general concept of the new process well: "The backbone of SoS SE implementation is continuous analysis that considers changes from the broader environment as well as feedback from the ongoing engineering process. The results of that analysis provide the basis for developing and evolving the SoS architecture, identifying or negotiating changes to the constituent systems that impact the SoS, and working with the constituent systems to implement and integrate those changes. This view of SoS SE implementation provides structure to the evolution of the SoS through changes in constituent systems that are typically on different life-cycle timelines, adapting as systems come in and move out, and as concept of operations (CONOPS) adapt and change. Hence the need for continually updating the SoS analysis and adapting the architecture and updating systems on an ongoing basis."

For purposes of describing the relevant systems engineering process issues unambiguously, we recognize three distinct systems of interest, distinguished as Systems 1, 2, and 3. System-1 is the target system under development, an autonomous ground vehicle. System-2 includes the basic systems engineering development and maintenance processes, along with their operational domain that produces System-1. System-3 is the process improvement system, called the system of innovation that learns, configures, and matures System-2.

System-1 will not be described in any detail, but will be explicitly referenced as such, when necessary, to avoid confusion in the discussion. System-2 is the principle focus of this article, but as it is shaped through time by experiential learning, the nature of System-3 that manages the maturation of System-2 will be dealt with explicitly and most particularly in the section that models the UxSITE agile systems engineering process. Figure 1 shows a general representation of the relationships between Systems 1, 2, and 3.

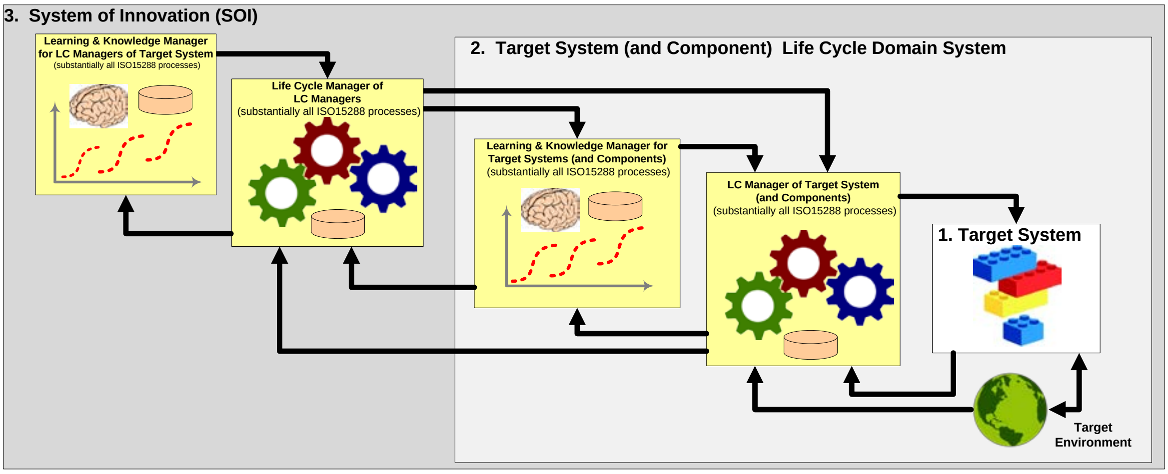

Figure 1. Notional Relationships of Systems 1, 2, and 3. (Schindel and Dove 2016)

In the new and current evolutionary process, roles, responsibilities, and working relationships of the 60 some engineers and managers were restructured and supported by an infrastructure that enables and facilitates coherent close-knit team work. Under the new approach contractors have responsibility for technology analysis and development of System-1 components, and UxSITE personnel have responsibility for evolving System-1 architecture, integrating System-1 capability-enhancement developed by the contractors, and validating the latest wave of System-1 performance. Shown in Figure 2, these four activities happen in overlapping System-2 waves of incremental System-1 enhancement, with each wave approximately six months in duration. Importantly, UxSITE personnel also evolve the System-2 process concept and supporting infrastructure with attention in System-3.

This article will focus on the unique aspects of the UxSITE evolutionary systems engineering process that are most responsible for the demonstrated and respected effect of the new process. The Wave structure and process makes a unique contribution, in that the UxSITE SE process is the first known effective implementation of the SoS Wave Model; which is better done justice in a separate detailed article (Scrapper, Halterman, and Dahmann 2016). This present article will focus principally on the enabling and facilitating infrastructure, which supports team coherency and coordination.

Subsequent sections will lead off with an overview of the five-element Integration Strategy, placing the Evolutionary Systems Engineering approach in context with four complimentary elements; the focus will then shift to the enabling and facilitating infrastructure in some detail,

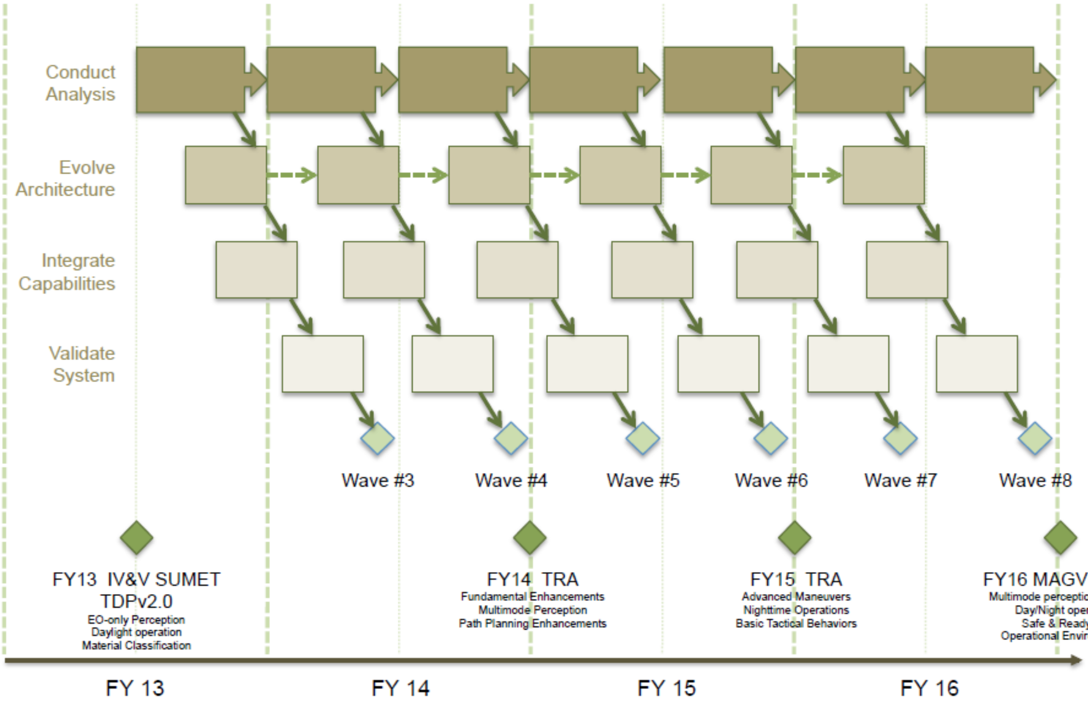

Figure 2. Integrated Master Plan: System evolution in six-month capability-enhancement waves.

and a final section before concluding remarks will provide a high-level pattern-based model view of the unique operational aspects.

Overview

Agile systems engineering processes are justified and effective when it is expected that the engineering environment and activities will be subject to changes that affect the ongoing development effort throughout the project. A UURVE framework (Dove and LaBarge 2014) for characterizing the general/high-level dynamic nature of the engineering environment provides guidance for what must be accommodated.

Agile SE processes are necessary and justified when the engineering environment has characteristic of unpredictability, uncertainty, risk, variation, and evolution (UURVE). The general UURVE key characterizations of the UxSITE engineering environment, justifying and agile SE approach, are as follows:

Unpredictability (unknowable situations):

- Strategic realignment by sponsor.
- Changes in and/or availability of key personnel and development contractors.

Uncertainty (randomness with unknowable probabilities):

- Feasibility of technical approach and initial designs.
- Contracting issues. funding gaps, and budget short falls.

Risk (randomness with knowable probabilities):

- Failure to meet technical performance measures.
- Maturation and integration of required component technologies.

Variation (knowable variables and variance ranges):

- Availability of test ranges and test support, and obtaining requisite approvals.
- Reliability, Availability, Maintainability (RAM) of vehicle test-beds (vehicle, sensor, computing hardware, cables, connectors).

Evolution (gradual successive developments):

- Changes in technical landscape and insertion of emerging technology.
- Changes in programmatic objectives and stakeholder's requirements (scope creep).

The UxSITE SE process encompasses research, development, integration, test, and evaluation of deployable system and component technologies that can provide new capabilities for military autonomous ground vehicles. The integration strategy is central. Contractors provide the research and development, with UxSITE government personnel providing the architecture, integration, test, and experimentation.

Depicted in Figure 3 are the five elements that constitute the UxSITE integration strategy. Each will be briefly discussed for key points in this section, with some emphasis put on the Continuous Integration Environment in the next section.

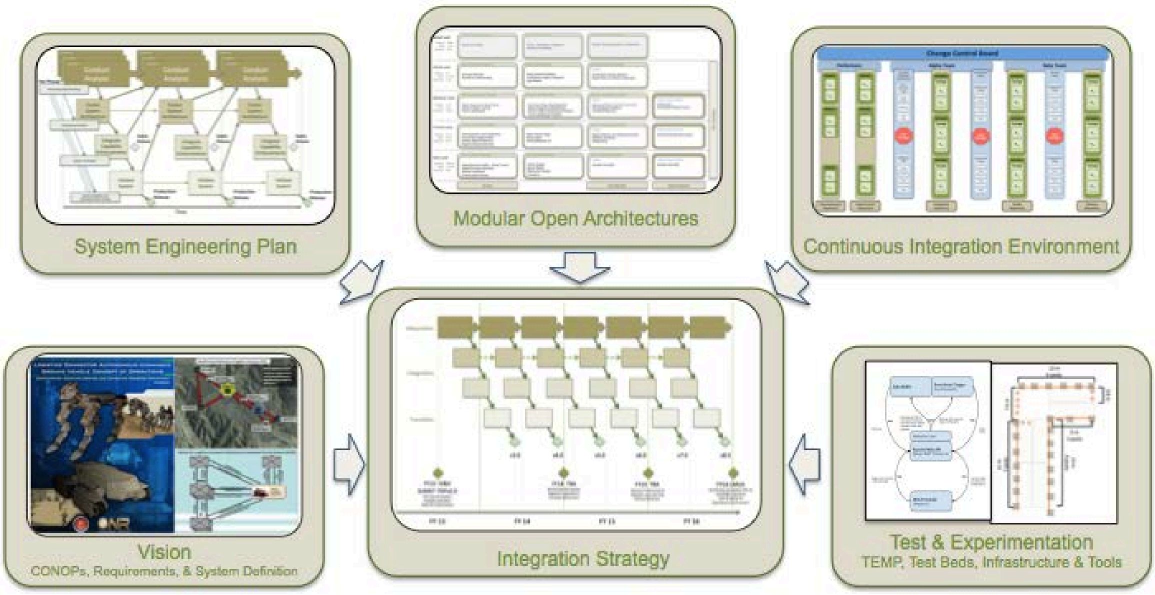

Figure 3. The five elements contributing to the Integration Strategy

Vision – This is the stable driving element for the process that establishes a common understanding of success, both technically and programmatically. It is repeatedly reviewed throughout the entire SE process as meeting-opening positioning for the various and frequent team interaction sessions, with the featured cornerstone of taxpayer and warfighter as the customers. It includes the concept of operations for the System-2 integration strategy, and the system definition and requirements for System-1. The clarity of vision and objectives provides common criteria so that everyone can bring their knowledge to bear within a common relevant context. Disagreements caused by different criteria are avoided, and interactions are decisive rather than unresolved. This approach drives closure among multiple perspectives.

Systems Engineering Plan – The Systems Engineering Plan embodies the SoS Wave Model approach, with four phases in approximately-six-month overlapping evolutionary Waves, as depicted in Figure 4. As depicted at the top, contractors lead in analysis and development of System-1 component technologies, with Government (UxSITE) support. In the middle, UxSITE personnel lead in System-1 architecture development and evolution, with contractor support. UxSITE personnel also lead in integrating the component technologies, with contractor support. At the bottom, UxSITE personnel lead in validation, with end-user

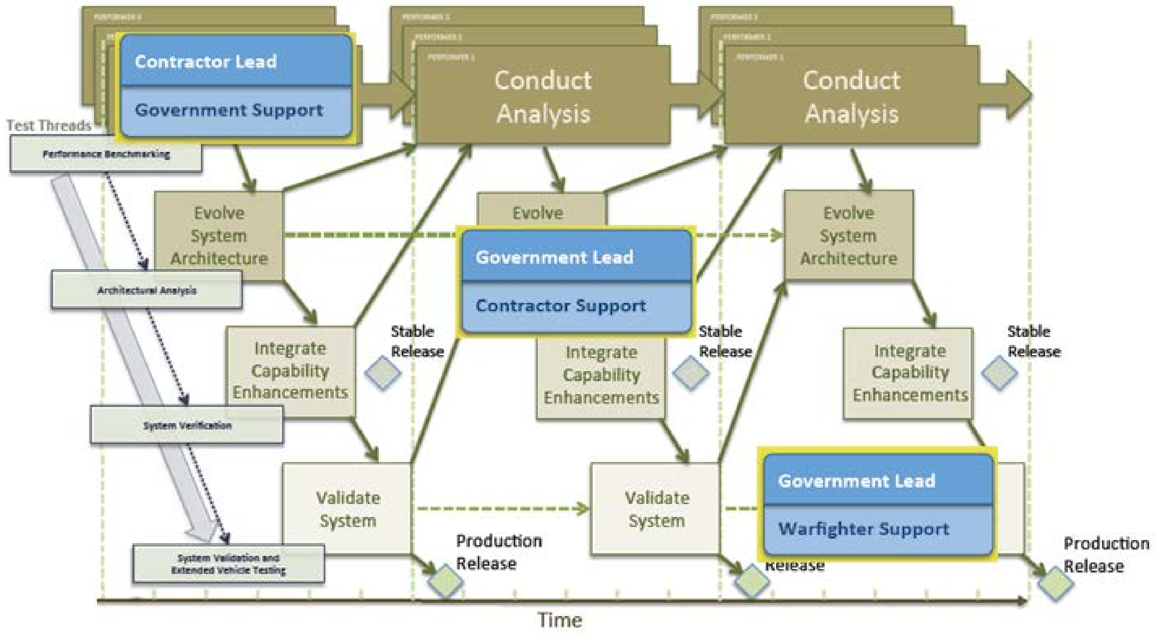

Figure 4. Systems Engineering Plan

warfighter support. Each Wave has test threads that run through each of the four phases: performance benchmarking, architectural analysis, system verification, and finally system validation and extended vehicle testing. The details of the Evolutionary Systems Engineering approach developed by SSC Pacific are beyond scope for this article, but are available in (Scrapper, Halterman, and Dahmann 2016). Notably, risk analysis and management is distributed and integrated throughout all phases at all time, is a shared responsibility of all team members, and is not centralized in a risk manager role or a life-cycle process task.

Modular Open Architecture – The modular open architecture developed for System-1 is a key enabler for sustaining an agile System-2 capability, which depends upon the ability to replace and upgrade individual System-1 components in successive evolutionary waves. Four different System-1 vehicle platforms[1] are currently under evolutionary development, which can share common functional capability components. This need is underscored in the stated project objectives: “Low-cost, platform and mission agnostic applique’ with modular, reconfigurable executable mission modules.” SSC Pacific retains design, control, and ownership of the System-1 architecture, ensuring that developed components conform to a plug-and-play system-1 architecture. Contract developers retain proprietary ownership of component technology, but must design and develop to the Government owned system-of-systems architecture. The open architecture is well planned before the evolutionary systems engineering process begins, under the principle that architectural refactoring incurs time and cost that should be avoided.

Integration Test and Experimentation Master Plan – This plan defines the overall test strategy and objectives for the continuous test and experimentation required to support the maturation and integration of new capabilities. It is structured to accumulate evidentiary

---

[1] RaDER (Reconnaissance and Detection Expendable Rover) and three different-mission expeditionary vehicles: EV-1, EV-3, EV-4.

information for feedback into the development cycle, and ensures critical capability objectives and functional requirements are being met. Regression testing is done to ensure that new development doesn't impact prior accomplishments. All testing of new capability added during each iteration of the evolutionary SE process is preceded with a standard initial suite of test methods, which first verifies that the system can do 10-of-10, 19-of-20, and 28-of-30 repeatable performance actions, in both the lab environment and in the real world environment. Then and only then is new added capability tested.

Continuous Integration Environment – This is a core element of the enabling and facilitating operational infrastructure. It is all about orchestrating the interaction of the 60-some engineers and managers on the project: 17 engineers working the technical integration, with six external organizations of 4-5 engineers each working on development projects for development of functional capabilities to be integrated into a federated system. Physically the CIE is a federated system of software applications that receive, disseminate, and present operational data. Operationally it is an orchestration and collective-consciousness mechanism that includes principles of management and personnel interaction. It is an evolving infrastructure that can already be credited with the great leap forward over the prior process. This element will be discussed in more detail in the next section.

Enabling, Facilitating, and Sustaining Agility

The discovery and description of a common Agile Architecture Pattern (AAP) for systems and processes that successfully deal with UURVE operational environments is detailed in (Dove and LaBarge 2014). Briefly, the architecture contains three principle elements: a pool of resources that can be configured to address the process-activity of the moment, a passive enabling infrastructure that establishes common rules for readily interconnecting these resources, and an active facilitating infrastructure with responsibilities for sustaining the agility of the SE process by maintaining and evolving the resources, the interconnection standards, and the agile SE Concept of Operations. Figure 5 depicts the UxSITE systems engineering process as an instance of the AAP, and provides a relevant framework for discussing key points.

The UxSITE evolutionary SE process architecture is structured to span multiple System-1 vehicle development projects that can share common component resources. Component resources can be dragged and dropped into a variety of configurations, interfacing with each other according to plug-and-play rules and standards.

The principle intent of this section is to discuss the passive enabling infrastructure and the active facilitating infrastructure.

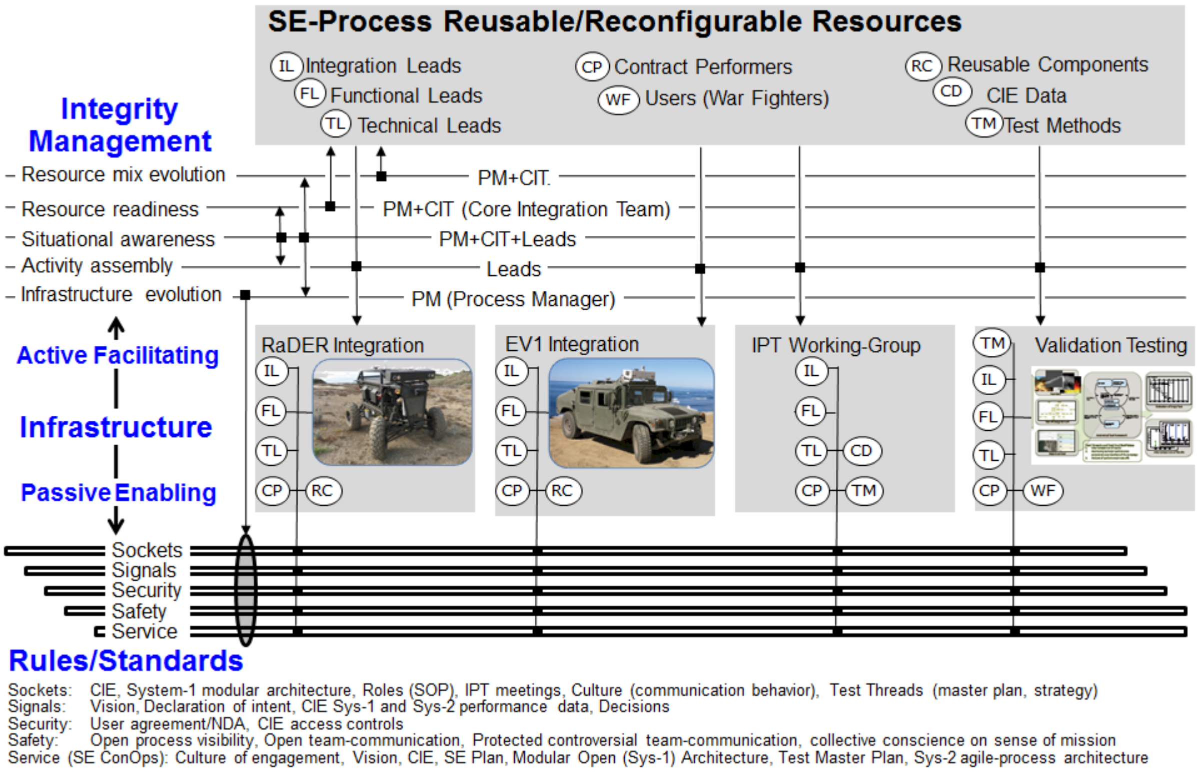

Figure 5. Agile-process architecture depicting four example process-activity configurations assembled from drag-and-drop assets in conformance with the rules and standards of the plug-and-play passive enabling infrastructure.

The AAP calls out the principle resources that are employed in assembling process-activity configurations:

- Integration Lead – Develops the Vision for System-1 and oversees the technical execution and coordination of activities and processes in System-2.
- Technical Leads – Oversees technical execution and mitigation of technical risk associate with a specific phase in System-2.
- Functional leads – Provides in-depth technical expertise in each designated functional area to support the research, design, implementation, operation, maintenance, and assessment of new capability enhancements.
- Performers – Leads the development of desired functional capability for System-1.
- Users (War Fighters) – Validates the operational concept for System-1 and provides feedback into System-2 regarding utility of current and planned capabilities.
- Reusable Components – Functional capabilities and tools to support the integration and specification of System-1 capabilities for different vehicle types and mission sets.
- CIE Data – Artifacts and evidentiary information produced by System-2 and shared across extended team to enable the rapid and agile development of System-1.
- Test Methods – Tools, procedures, and metrics for quantifying the performance of System-1 to enable the rapid assessment, characterization, and inter-comparison of experimental results.

Process Enabling and Facilitating Infrastructure

Infrastructure consists of passive and active sections. The passive section is in essence the systems engineering Concept of Operations, complete with resource interconnection standards.

The principle intent of the UxSITE passive infrastructure is to enable effective process-activity assembly and to specify the process operational intent and methods. The active section designates responsibilities for maintaining agile process capability and effectiveness. The principle intent of the UxSITE active infrastructure is to facilitate sustainable process agility.

Passive Enabling Infrastructure

Figure 5 at the top shows the principle SE-process resources that can be assembled into process-activity configurations for specific situations. The ability to drag-and-drop these resources into plug-and-play configurations is enabled by the passive infrastructure, so called because it encompasses the fairly stable rules that enable effective resource interconnection. Many of the elements below have already been described in the Overview section, with a brief tabulation here.

Sockets – physical interconnect:

- CIE custom-view interfaces for human access.
- System-1 modular architecture for connecting components.
- Roles describing how people fill positions.
- IPT/Working-Group meeting format.
- Culture of full-team engagement behavior.
- Test Threads interconnecting testing with strategy and Test Master Plan.

Signals – data interconnect:

- Vision reiterated in all meetings.
- Declarations of intent for accomplishing the vision.
- CIE System-1 and System-2 performance data, serving as collective Team consciousness.
- Decisions, openly communicated unambiguously.

Security – trust interconnect:

- User agreement and Non-Disclosure Agreements for managing performers IP of all technical data produced and stored in the CIE.
- CIE access controls partitioned to prevent sensitive cross-interest performer and functional area leakage.

Safety – of SE-process users:

- Open process visibility with CIE access to all project status and information affecting collective ability to meet project objectives and individual ability to perform.
- Open communication on all team issues (no private conversations).
- Protected communication to encourage controversial open communication without fear of subsequent reprisal.
- Collective conscience in personal mission to do what is right for the user.

Service – the SE-process Concept of Operations:

- Culture of Engagement, embraced by the full team.
- Vision, unambiguous and stable.
- Continuous Integration Environment (CIE), enabling and facilitating a collective consciousness.
- SE Plan, the Evolutionary Systems Engineering approach based on the SoS Wave Model.

- Modular Open Architecture, for System-1 component development and integration, enabling and facilitating reuse of components in multiple vehicle types for multiple mission needs.
- Integrated Test and Evaluation Master Plan, enabling and facilitating comprehensive standard practices.
- System-2 agile-process architecture, enabling and facilitating reconfigurable SE-process resources.

Three elements are central to the process enabling infrastructure and the effectiveness of the UxSITE SE process, which warrant further discussion: the full-team culture of engagement and the collective consciousness enabled by the CIE.

Culture of engagement – Most pronounced during the analysis activity of the UxSITE capability was the pervasive nature of the culture, its thoughtful development, and its continual reinforcement. This is done with a combination of soft skills and supporting infrastructure.

Culture is shared expectations for behavior and an environment that enforces that behavior. Under the UxSITE capability culture isn't written like a mission statement, but is rather practiced by process leadership, shaped by consistent reinforcement, and enforced by dealing openly with infractions detrimental to the team and at odds with a pervasive collective agreement to work together toward total success.

Full and active engagement with the SE process intent and the SE project objectives is the expectation. All team members are on a shared mission, and all team members need to support and be supported by all other team members, at all times. The nature of the UxSITE SE process, its leadership, and the transparency of comprehensive real-time project status provide team-engagement sensitivity. When a team member's engagement falls off, it becomes apparent, and the culture responds with empathy and mitigation of the causal issues. Team members that can't embrace this culture are not tolerated for long, which serves to strengthen the culture.

Collective consciousness – A software construct developed to support UxSITE is known as the Continuous Integration Environment (CIE), The CIE is a data-driven repository of knowledge, with customized viewing templates for different needs. It is distinguished as data driven rather than document driven. A Model-View-Controller (MVC) concept is employed for data exploitation. Wikipedia provides more detail under *Model View Controller*[2], but the intent of an MVC is to provide user interfaces that separate internal representations of data from the ways that information is presented to users. A *view* can be any output representation of information, such as a chart, a diagram, or tabular information, with multiple views of the same information custom suited to meet requirements of different stakeholders; such as sponsors, management, technical integrators, performers, testers, or whomever.

CIE serves much the same purpose intended by commercial Product Life-Cycle Model (PLM) software, but is homegrown at SSC Pacific's Unmanned Systems Group to provide a federated structure now being called for by the PLM user community. CIE is structured as a federation of independent capabilities, mostly off the shelf, and is being evolved to provide real-time relevant and comprehensive views of history and current status to all team members. This home-grown federated approach enables capability evolution suited to timely UxSITE SE process needs, rather than commercial software product strategies and release schedules. The intent is to facilitate a real-time collective consciousness, where all team members are plugged in to all information associated with full project success, as well as to the information of

[2] More details on Model-View-Controller on Wikipedia: https://en.wikipedia.org/wiki/Model–view–controller

relevance to their specific responsibilities and tasks. New data, new decisions, new issues, new test results, wherever and whenever they occur, ripple through the relevant federation of CIE components and CIE user views immediately. This collective consciousness manifests for the team much like it does for musicians in a symphony orchestra, where off notes and bad timing are immediately sensed by all.

Collective Conscience – Meeting openings consistently remind everyone that the customers are taxpayers and warfighters. Superficially this can sound like hollow, politically-correct cheesiness. But these reminders don't stop with a simple statement. They are rooted in emotional image and story that elevates them to personified walking needs with faces. The warfighter needs tools that are effective, timely, and affordable for self preservation and for mission achievement. The tax payer needs tools that are effective, timely, and affordable for national/homeland security. With context each of us can walk in the shoes of a warfighter and a tax payer, relate to being the customer, and evaluate everything we are doing with that critical internal customer voice. A collective conscience to do what is right is developed.

Active Facilitating Infrastructure

The active infrastructure is what sustains the agility of the SE process. In order for new activity-configurations to be facilitated when needed, five responsibilities are required: the roster of available resources must evolve to be always what is needed, the resources that are available must always be in deployable condition, the assembly of new activity configurations must be effectively accomplished, and both the passive and active infrastructures must evolve in anticipation and/or satisfaction of new needs. These five responsibilities are outlined in standard role descriptions, assigned to appropriate personnel, and embedded within the process to ensure that effective process-activity is possible at unpredictable times.

- Resource mix evolution – ensures that existing resources are upgraded, new resources are added, and inadequate resources are removed, in time to satisfy needs. This responsibility is triggered by situational awareness, and dispatched by the personnel shown in Figure 5.
- Resource readiness – ensures that sufficient resources are ready for deployment at unpredictable times. This responsibility is ongoing, and dispatched by the personnel shown in Figure 5.
- Situational awareness – monitors, evaluates, and anticipates the operational environment in relationship to situational response capability. This responsibility is ongoing, and dispatched by the personnel shown in Figure 5.
- Process-activity assembly – assembles process-activity configurations. This responsibility is triggered by Wave phases and situational events, as and when needed, and dispatched by the personnel shown in Figure 5.
- Infrastructure evolution – evolves the passive and active infrastructures as new rules and roles become appropriate to enable evolving needs. This responsibility is triggered by situational awareness, and dispatched by the personnel shown in Figure 5.

Pattern-Based Model View of Unique Operational Aspects

The ASELCM Pattern (Schindel and Dove 2015) helps us ask and understand how the UxSITE approach effectively addresses the UURVE environment. In the framework of the ASELCM Pattern, this can be seen as a " System-3 question". Using the language of Agile User Stories (Leffingwell, 2011), the following describe how (System-2) stakeholders might state their own value interests:

As a <Sponsor>, I want <timely project incorporation of emerging technologies> so that <I obtain a best-in-class autonomous vehicle system>.

As a <Functional Lead>, I want <to obtain timely project status> so that <I direct in a timely manner>.

As a <Project Performer>, I want to <obtain timely project directional awareness> so that <I contribute responsively to the overall project>.

Agile User Story: As a <stakeholder role>, I want <system behavior> so that < value statement>.

The expeditionary autonomous navigation capability developed by UxSITE for System-1 vehicles is an emergent vehicle level capability, reflecting systemic attributes that arise from multiple interactions between different vehicle components and the components of the vehicle environment. In the same way, UxSITE System-2 project level execution capabilities such as Status Awareness and Direction Awareness arise from multiple interactions between different System-2 components. The ASELCM Pattern captures these using a subset of the S*Model, shown in Figure 6.

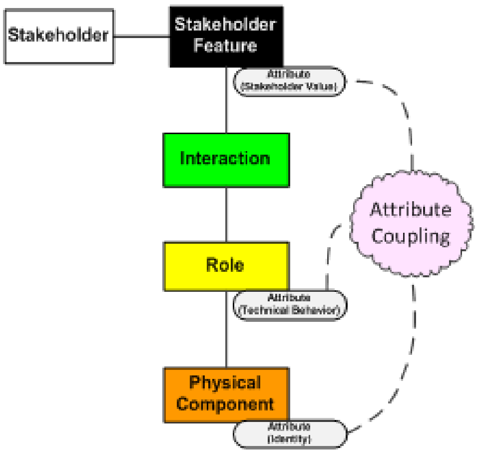

Figure 6: Selected Subset of ASELCM Features, System-2

ASELCM Pattern Features describe the (subjective) stakeholder perspective on what outcomes are valued. Figure 7 shows a subset of the ASELCM System-2 Features representing stakeholder interests such as the above User Stories. (The ASELCM Pattern includes many other Features, outside the scope of this paper, including the basic systems engineering technical process Features.)

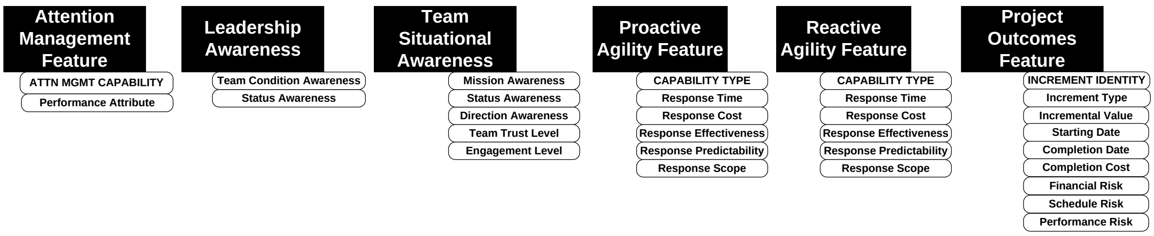

Figure 7: Selected Subset of ASELCM Features of System-2

These Features represent capabilities important in many human processes, not limited to systems engineering, and are particularly critical when the environment is highly dynamic or uncertain. Like the sensor sampling rates and signal processing parameters of a System-1 autonomous vehicle, we can expect that the parameters of System-2 "sensor" accuracy, sampling rates, data communication networks, and interfaces contribute to the emergent capabilities of the UxSITE SE process. Actual UxSITE projects outcome capabilities of these Features emerge as a consequence of multiple System-2 Interactions that occur between the people, information systems, equipment, infrastructure facilities, and other components of UxSITE System-2, including customers and System-1 domain components. Interactions are real physical exchanges of information, energy, mass, or forces, and are modeled as part of the more objective technical aspect of the ASELCM Pattern. Many of those interactions are understandably about technical processes relatively familiar to the engineering world, but they also include the five more general Interactions shown in Figure 8:

The Interactions summarized in Figure 8 include both (1) elements of leadership, management, and other human skills, along with (2) non-human operational information systems and other engineered infrastructure. Indeed, some of the well-known practices popularized by the Agile Software community (Fowler and Highsmith 2001) are focused on key human skills and interactions, and there is likewise a significant history of COTS and other information systems concerned with facilitating one aspect or another of the above interactions in various business environments (Schmidt 2015).

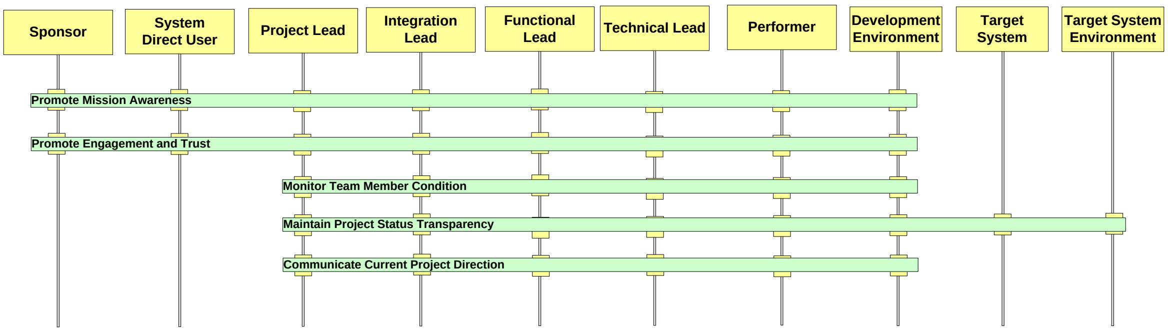

Figure 8: Selected Subset of ASELCM Interactions, System-2

The SSC Pacific UxSITE discovery workshop made it evident that both the human and IT aspects of those interactions were important to the successful outcomes reported. Although they are not uniquely important to Agile Systems Engineering, they are among the most basic necessary elements of agility in any organization, so their characterization within the ASELCM Pattern is appropriate. Many of the most important performance aspects emerge not as a result of the characteristics of a single component role, but instead emerge as systemic attributes of System-2. Figure 9 illustrates the idea that certain quantitative attributes describe individual component roles, but others describe emergent attributes of the system as a whole.

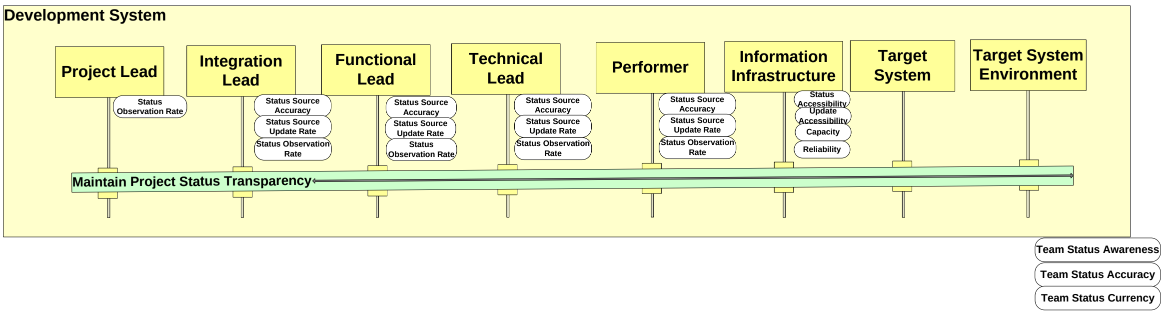

Figure 9: Attributes of Individual Component Roles, and Emergent Systemic Attributes

Just how are these system parameters related quantitatively? Integrating the above model segments, Figure 10 shows that the values of Feature Attributes of Figure 7 are impacted by the emergent system technical attributes of Figure 9, and those are likewise impacted by the performance attributes of the individual interacting components shown in Figure 9. The parametric relationships between the stakeholder Features, emergent technical performance attributes, interacting Functional Role attributes, and underlying Physical Components performing those roles, are all illustrated by the Attribute Couplings of Figure 10.

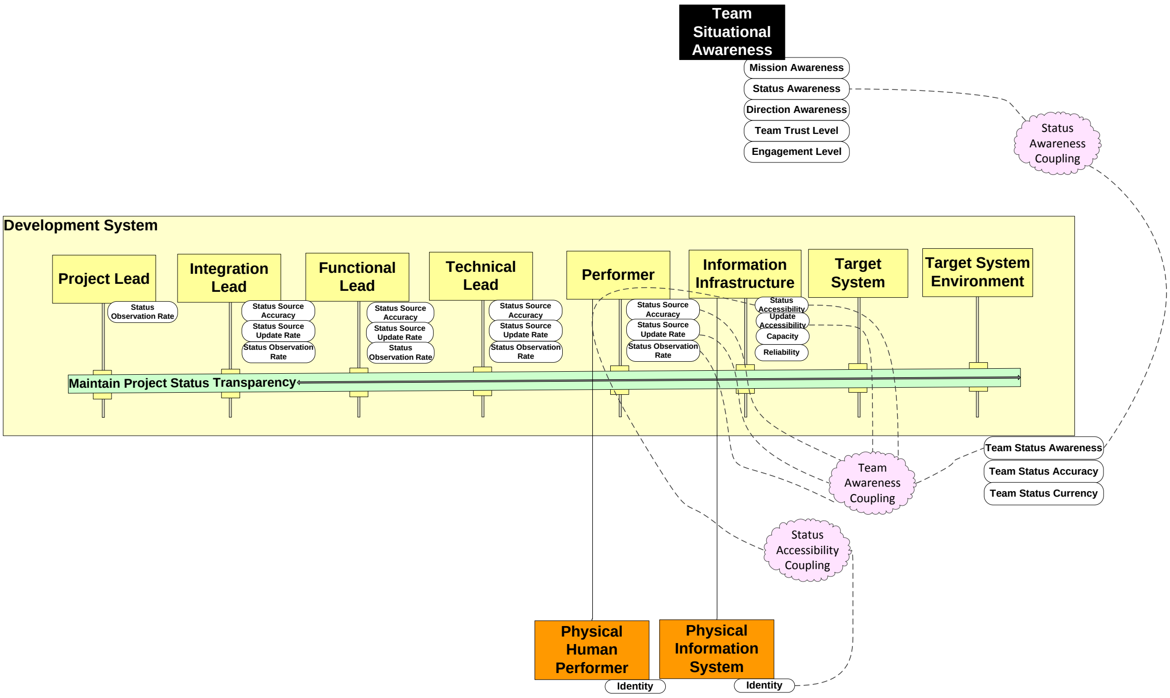

Figure 10: Modeled Parametric Couplings of ASELCM Features, Functional Roles, and Physical Components

Such Attribute Couplings are at the heart of this project’s assertion of causality principles in agility, and the calibration of those couplings (as in Figure 11) is among the goals of the referenced project. The capability of the System-2 components to perform accordingly is the responsibility of System-3, which selects them and configures System-2.

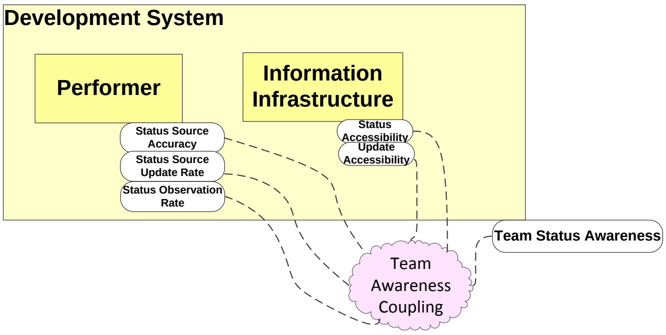

Figure 11: Team Status Awareness Arises from Other Attributes

Concluding Remarks

The UxSITE agile SE process has demonstrated consistent success over three years in lowering costs, meeting schedules, and delivering meaningful technology innovation. The process uses an incremental and iterative approach inspired by the SoS Wave Model concept, with six month overlapping release cycles compatible with the nature of mixed hardware/software innovative-development. The SoS Wave Model fits the UxSITE SE operational environment and project mission very well; better than other candidate, such as Spiral or Scrum.

Put in larger perspective, the SoS Wave Model is one of five elements that comprise the total Integration Strategy. Collectively these five elements are the face of the UxSITE capability.

But that face doesn't reveal the heart and sole of the process, the essence of its unique identity. The power of the UxSITE capability emerges from a core focus on team effectiveness: creating and sustaining a culture of engagement, a collective consciousness, and a shared conscience.

Nobody in the SSC-Pacific Unmanned Systems Group explicitly articulated this core focus on culture, consciousness, and conscience during the three-day analysis workshop conducted there; but as outsiders trying to understand what was happening and why it was working, we could see the forest while they were describing the trees.

These three core concepts, as a foundation for powerful agile systems engineering capability, are not tied to the context or environment of the SSC Pacific Unmanned Systems Group. They are transportable concepts, though the soft skills currently needed for implementation may impede uptake.

References

Dahmann, Judith, Jo Ann Lane, George Rebovich, Jr. and Kristen J. Baldwin. 2011. An Implementers View of Systems Engineering for Systems of Systems. IEEE International Systems Conference, Montreal, Canada, 4-7 April. www.acq.osd.mil/se/docs/ImplementerViewSE-SoS-Final.pdf.

Dahmann, Judith, Rob Heilmann, et. al. 2012. SoS Systems Engineering (SE), and Test & Evaluation (T&E). Final report of the NDIA SE Division SoS SE and T&E Committees. National Defense Industrial Association. March. www.acq.osd.mil/se/docs/SoS-SE-TE-NDIA-FinalReport-28Mar-with-abstractnotes.pdf.

DAU (Defense Acquisition University). 2013. Section 4.2.1.2 – Systems of Systems, Defense Acquisition Guidebook. 15 May. https://acc.dau.mil/CommunityBrowser.aspx?id=638308.

Dove, Rick and Ralph LaBarge. 2014. Fundamentals of Agile Systems Engineering – Part 1 and Part 2. International Council on Systems Engineering, International Symposium, Las Vegas, NV, 30Jun-3Jul. www.parshift.com/s/140630IS14-AgileSystemsEngineering-Part1&2.pdf.

Fowler, M., and J. Highsmith. 2001, "The Agile Manifesto". Dr. Dobb's Journal, August. www.drdobbs.com/open-source/the-agile-manifesto/184414755.

Leffingwell, Dean, *Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise*. Addison-Wesley 2011

Moteff, John D. 1998. Defense Research: A Primer on the Department of Defense's Research, Development, Test and Evaluation (RDT&E) Program. Report for Congress, Congressional Research Service. https://www.fas.org/sgp/crs/natsec/RL34666.pdf

Schmidt, K., ed., *The Journal of Collaborative Computing and Work Practices*, Springer, 2015.

Schindel, B. and R. Dove. 2016. Introduction to the Agile Systems Engineering Life Cycle MBSE Pattern. International Council on Systems Engineering. International Symposium, 18-21 July, Edinburgh, Scotland (submitted).

Scrapper, C., R. Halterman, and J. Dahmann. 2016. An Implementers View of the Evolutionary Systems Engineering for Autonomous Unmanned Systems. IEEE Systems Conference, 18-21 April, Orlando FL (submitted).

Acknowledgements

The UxSITE SE process architect, and the Process Manager referred to in Figure 5, is Chris Scrapper. The SE process analysis described is this article is the collective output of a workshop team that includes (alphabetically): Mike Bruch (US Navy), Rick Dove (Paradigm Shift International), Judith Dahmann (MITRE), Kevin Gunn (MITRE), Ryan Halterman (US Navy), Suzette Johnson (Northrop Grumman), Ken Laskey (MITRE), Tracy Pastore (US Navy), Avinash Pinto (MITRE), Jack Ring (OntoPilot), Garry Roedler (Lockheed Martin), Bill Schindel (ICTT Systems Sciences), Chris Scrapper (US Navy), Rachel TenWolde (US Navy), Michael Yokell (Lockheed Martin).

Biography

Rick Dove is CEO of Paradigm Shift International, specializing in agile systems research, engineering, and project management; and an adjunct professor at Stevens Institute of Technology teaching graduate courses in agile and self-organizing systems. He chairs the INCOSE working groups for Agile Systems and Systems Engineering, and for Systems Security Engineering, and is the leader of the INCOSE Agile Systems Engineering Life Cycle Model Discovery Project. He is and INCOSE Fellow, and the author of *Response Ability, the Language, Structure, and Culture of the Agile Enterprise*.

Bill Schindel is president of ICTT System Sciences. His engineering career began in mil/aero systems with IBM Federal Systems, included faculty service at Rose-Hulman Institute of Technology, and founding of three systems enterprises. Bill co-led a project on Systems of Innovation in the INCOSE System Science Working Group, co-leads the Patterns Working Group, and is a member of the lead team of the INCOSE Agile Systems Engineering Life Cycle Project.

Chris Scrapper is the Integration, Test, and Experimentation lead at SPAWAR System Center Pacific Unmanned Systems Group. Over the past 15 years, Chris has been involved with a variety of innovative programs focused on research, development, integration, test and evaluation of autonomous ground systems. Chris has authored over 35 peer-reviewed publications, and is actively involved in the development of system engineering best practices and performance standards for unmanned ground systems within the Department of Defense, the Department of Homeland Security, and the Department of Commerce. Chris has a B.S. In Computer and Applied Mathematics from American University, and an M.S in Computer Science from the George Washington University.