Software Engineering Institute

Multi-view Decision Making (MVDM) Workshop

Christopher Alberts
James Smith II
Carol Woody

February 2009

SPECIAL REPORT
CMU/SEI-2008-SR-035

Acquisition Support Program

Carnegie Mellon

This report was prepared for the

SEI Administrative Agent
ESC/XPK
5 Eglin Street
Hanscom AFB, MA 01731-2100

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

This work is sponsored by the U.S. Department of Defense. The Software Engineering Institute is a federally funded research and development center sponsored by the U.S. Department of Defense.

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.

For information about SEI reports, please visit the publications section of our website (http://www.sei.cmu.edu/publications).

Executive Overview

The problems with the DoD system acquisition process have been well documented; most of these problems stem from an inadequate understanding of the technological and programmatic risks within programs [GAO 08, Smith 06a, Smith 06b]. Just as important, given the prevalence of systems of systems as the preferred approach to fielding operational capability today—and for the foreseeable future—are the risks arising from the interactions between different programs that comprise a given system of systems [Meyers 08]. The continued problems experienced by DoD acquisition programs point out the limitations of conventional program management and engineering practices in this regime. A major hurdle for complex development efforts is the lack of effective insight into problems early enough so solutions can be addressed without jeopardizing critical project and organizational constraints. An approach is needed to address the breadth and depth of activities, decisions, and products that must come together to successfully address complex development. For example, mission goals must align with secure quality technology that can be acquired within reasonable cost and schedule and produce a highly integrated system of systems. Effectively addressing these challenges requires a set of programmatic and engineering practices that reflect the realities of system-of-systems development, acquisition, fielding and support: multi-view decision making (MVDM).

MVDM addresses the satisfaction of mission goals with secure quality technology, within cost and schedule, for systems of systems within today's complex, highly dynamic operational environment. Risks arising from multifaceted interrelationships and dependencies can affect the chances of achieving a successful outcome when development and operations are spread across multiple organizations and systems. Few analysis techniques go beyond the limitations of a single system, and many are also limited to a finite set of stakeholders. System-of-systems analysis techniques must consider a range of problem areas including mission risk, interoperable acquisition, and operational security and survivability. Hence, a multi-view approach to decision making is needed. This approach—MVDM—combines techniques that delve into each of these three areas, providing a systems-of-systems roadmap to help you find where you are and get where you want to go.

Task 2.3.3 of the Army Strategic Software Improvement Program (ASSIP) FY08 Strategic Software Improvement Plan (SSIMP)[1] is structured to address improving confidence in software and software-intensive systems. In support of this task, a workshop was conducted on 9-10 July 2008 at the Arlington offices of the Software Engineering Institute (SEI).This workshop introduced three analysis techniques—(1) Mission-Oriented Success Analysis and Improvement Criteria (MOSAIC), (2) Interoperable Acquisition, and (3) Survivability Analysis Framework (SAF)—and how they can be combined to address the challenges of managing complexity. Through the use of examples, the limitations of decision making from any one single view and the value provided by a multi-view approach were presented and discussed.

[1] Signed by LTG N. Ross Thompson, Military Deputy, Assistant Secretary of the Army (Acquisition, Logistics, and Technology) and Dr. Paul Nielson, Director, Software Engineering Institute on 3 September 2007

Abstract

A major hurdle for complex development efforts is the lack of effective insight into problems early enough so solutions can be addressed without jeopardizing critical project and organizational constraints. Effectively addressing these challenges requires a set of programmatic and engineering practices that reflect the realities of system-of-systems development, acquisition, fielding and support: multi-view decision making. In support of the Army Strategic Software Improvement Program FY08 Strategic Software Improvement Plan, the Software Engineering Institute conducted a workshop to introduce three analysis techniques—(1) Mission-Oriented Success Analysis and Improvement Criteria, (2) Interoperable Acquisition, and (3) Survivability Analysis Framework—and how they can be combined to address the challenges of managing complexity. Through the use of examples, the limitations of decision making from any one single view and the value provided by a multi-view approach were presented and discussed. This report provides a recap of the workshop.

Workshop Attendees

Name	Organization	Email	Phone
Tom Coradeschi	LLOSE Combat Systems	tom.conadeschi@us.army.mil	**973-724-4344**
Judith Dahmann	MITRE	jdahmann@mitre.org	**703-983-1363**
Dave Fersch	Army, ASA-FM, ODASA-CE	david.fersch@us.army.mil	**703-601-4160**
William Messer	ASPO	william.messer@us.amy.mil	**703-428-8818**
Joe Pronti	Army, ARDEC	joseph.pronti@us.army.mil	**973-724-7835**
George Rebovich	MITRE	grebovic@mitre.org	**781-271-8503**
Marti Roper	Army, ASA-FM, ODASA-CE	Martha.roper@hqda.army.mil	**703-601-4140**
Robert Schwenk	Army ASA (ALT)	Robert.schwenk@us.army.mil	**703-697-6901**
Ceci Albert	SEI	cca@sei.cmu.edu	**703-908-8215**
Christopher Alberts	SEI	cja@sei.cmu.edu	**412-268-3045**
Angela Llamas-Butler	SEI	albutler@sei.cmu.edu	**412-268-2701**
Jim Smith	SEI	jds@sei.cmu.edu	**703-908-8221**
Carol Woody	SEI	cwoody@cert.org	**412-268-9137**

Workshop Agenda

Objectives

Multi-view decision making is needed to manage the complexity of addressing mission goals with secure quality technology within acquisition cost and schedule for a highly integrated operational environment. Attendees were introduced to three areas of SEI expertise that combine to provide a systems-of-systems (SoS) roadmap.

Date of Meeting	Facilitators	
9-10 July 2008	Chris Alberts, Jim Smith, Carol Woody	

Topics Day 1

Time	Topic	Discussion Leader	Activities
0830	Introduction	Carol Woody, Ceci Albert, Bob Schwenk	Welcome and ASSIP overview
0900	Army example of technology change in a highly complex mission	Carol Woody & Jim Smith	Presentation
1000	**Break**		
1015	Current Army approaches to address complex technology changes	Jim Smith	Discussion
1100	View 1 – Mission Survivability focus	Carol Woody	Presentation and discussion
1200	**Lunch**		
1300	View 2 – Interoperable acquisition and programmatics	Jim Smith	Presentation and discussion
1500	**Break**		
1515	View 3 – Mission success	Chris Alberts	Presentation and discussion

Topics Day 2

Time	Topic	Discussion Leader	Activities
0830	Recap Day 1 – summary of example and three views	Carol Woody, Jim Smith, Chris Alberts	Presentation with Q&A
0915	Impact of decisions made from only one view	Carol Woody, Jim Smith, Chris Alberts	Presentation & discussion
1000	**Break**		
1015	Multi-view – how the pieces should work together	Carol Woody, Jim Smith, Chris Alberts	Presentation
1130	Example use with complex change	Carol Woody, Jim Smith, Chris Alberts	Presentation & discussion
1200	**Lunch**		
1300	Changes needed for current Army approaches to consider mulit-view management	Carol Woody, Jim Smith, Chris Alberts	Discussion
1430	Summary and future actions	Carol Woody, Jim Smith, Chris Alberts	

Workshop Materials

Each participant was provided with copies of the presentation slides for each of the three views (summary slides are included in this report) along with activities worksheets distributed during each presentation. In addition attendees received copies of two SEI Technical Reports:

- *Survivability Assurance for System of Systems* by Robert J. Ellison, John Goodenough, Charles Weinstock, & Carol Woody; CMU/SEI-2008-TR-008, May 2008
- *Interoperable Acquisition for Systems of Systems: The Challenges* by James D. Smith II, & D. Mike Phillips; CMU/SEI-2006-TN-034, September 2006 http://www.sei.cmu.edu/pub/documents/06.reports/pdf/06tn034.pdf

1 Introduction

The workshop provided a forum for discussing how the Army addresses complex technology change today and ways this could be improved through the use of multi-view decision-making. An example of a complex technology change the Army is already facing, infrastructure technology changes that impact the participants in Close Air Support (CAS), was presented at the start of the workshop to provide a concrete basis for comparing current with proposed approaches. The key elements of this example are summarized in Section 1.1.

Several discussions were scheduled at key points in the workshop flow to capture ideas from attendees. These discussions covered the following:

- Current Army approaches to address complex technology changes (structured into two groups, managerial and technology related)
- Challenges specific to a managerial issue (KPPs)
- Challenges specific to a technology issue (software blocking)
- Ways in which a multi-view approach could be considered (pilot opportunity)

The detail material identified in the first three bullets is assembled in the Appendix of this document. A proposed summary analysis prepared by the workshop facilitators is included in Section 5. The opportunities for considering the multi-view approach (Next Steps) are provided in Section 6 of this document. It is the intent of the facilitators that the material in these two sections will support further discussions after the workshop.

In Sections 2-4 of this document each of the three views (mission survivability, interoperable acquisition and programmatics, and mission success) is summarized to remind attendees of the key values each view provides and provide some background information for those readers unable to attend the workshop.

1.1 DESCRIBE THE EXAMPLE

To illustrate the value that multi-views bring to analysis and decision-making, a scenario drawn from Close Air Support (CAS) provided a way to frame discussions and identify issues related to mission, survivability, and interoperable acquisition. Figure 1 shows the platforms participating in a CAS mission and the connectivity expected between each.

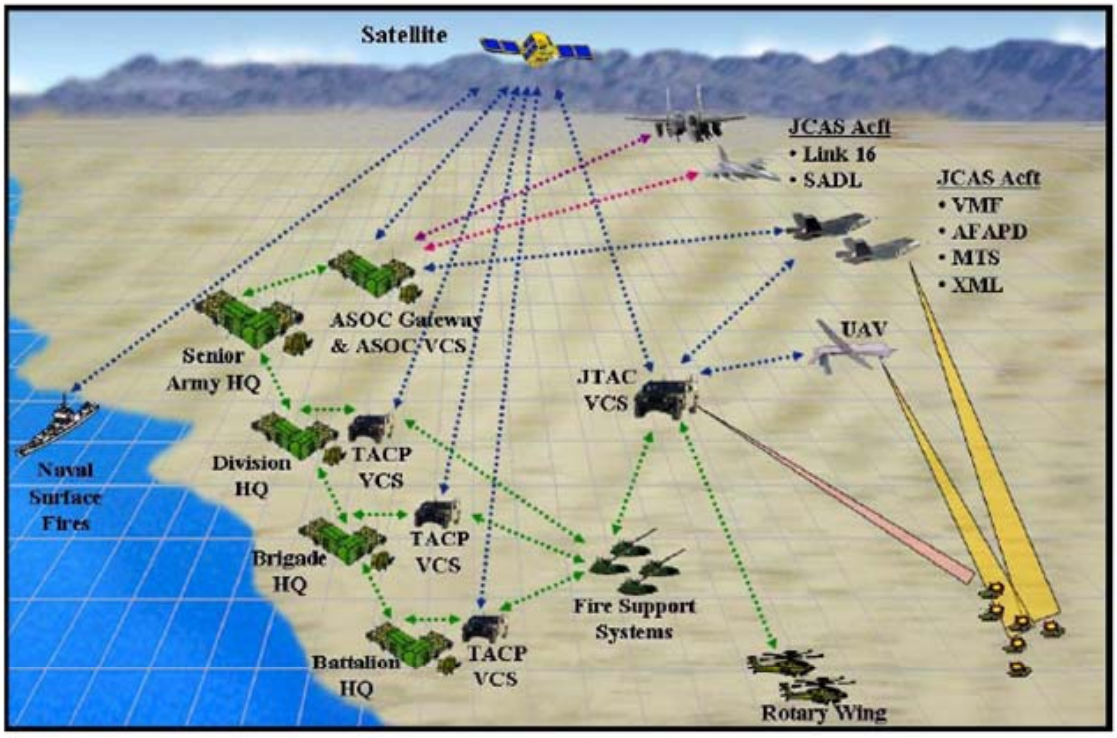

Figure 1: TACP/ASOC VCS Communications OV-1 from TACP-M VCS TRD dated January 2008[2]

To analyze survivability of a mission requires the selection of participants and connecting them in an execution sequence as shown in Figure 2. The activities to be performed at each step and the order in which they must be executed isolates a subset of the potential roles shown in Figure 1 and connects them through activities and responsibilities to establish an information flow as laid out in Figure 2.

[2] Blue = BLOS/LOS voice/data; Green = LOS Ground C2 voice/data; Purple = LINK16 (via ASOC gateway); Pink=SADL

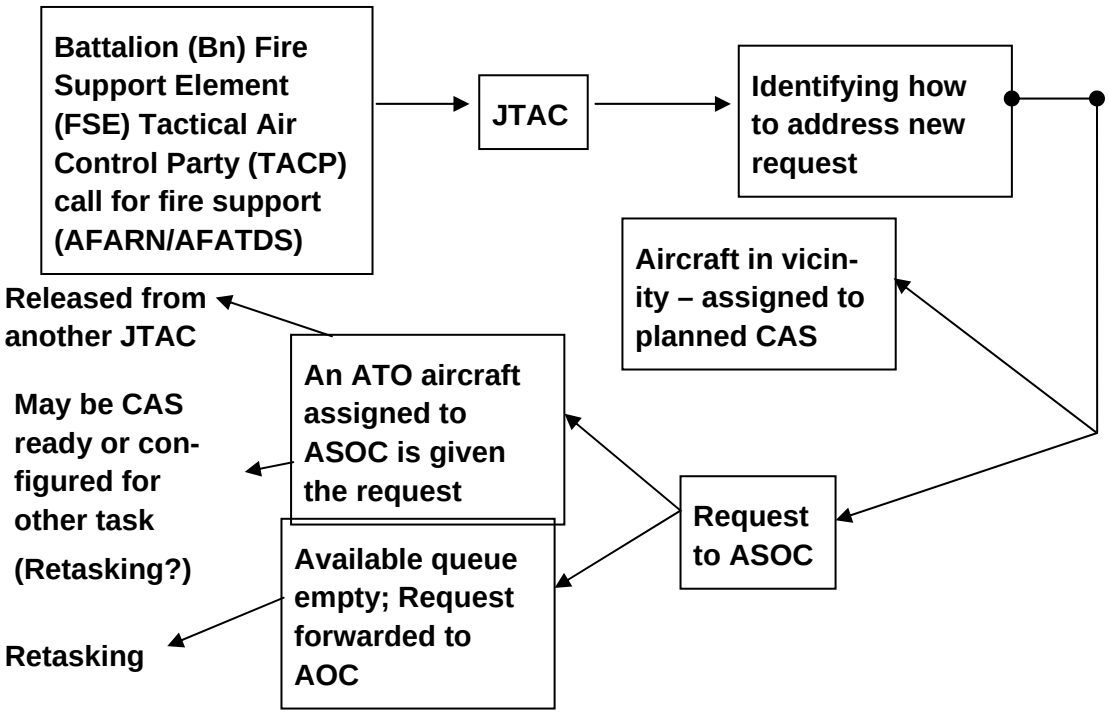

Figure 2: Close Air Support Mission Activity Flow

By identifying a single role (JTAC) and articulating both the information flows and the sequence in which they should occur to support a mission, the view in Figure 3 is developed. If this were the only view provided, a great deal of information about the capabilities available to support CAS would be lost. However, in considering and tuning the interaction level for the JTAC, the view in Figure 3 removes a great deal of visual clutter that is not critical to the specific analysis need and provides a way of considering isolated steps in the CAS sequence.

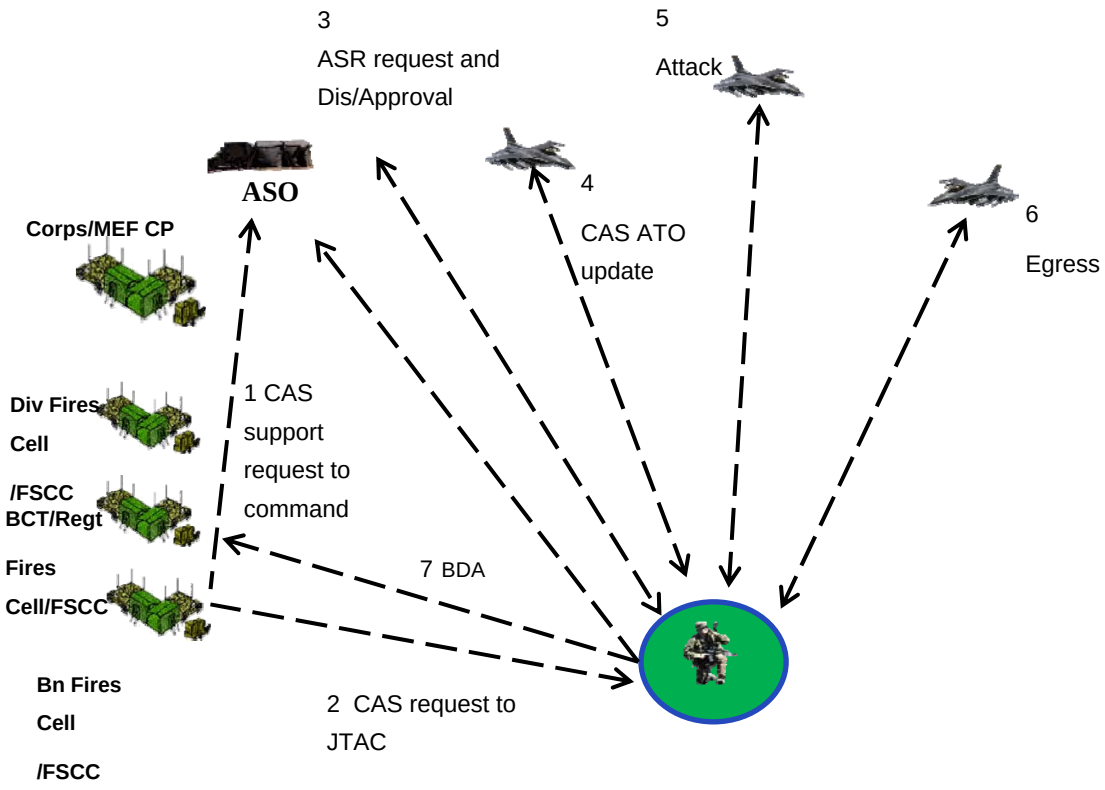

Figure 3: JTAC Interactions for CAS

In considering the impact of a technology change, each view provides a range of information useful to the problem. However, it takes the range of views plus additional information to evaluate the impact of a major technology infrastructure change on each role and on the CAS mission.

2 Mission Survivability Focus

The Survivability Analysis Framework (SAF), a structured view of people, process, and technology, was developed to help organizations characterize the complexity of multi-system and multi-organizational business processes. Survivability gaps arise when assumptions and decisions within one organizational area are inconsistent with those of another, resulting in differences and conflicts among the systems developed and used to support each organizational area. SAF provides a structure for capturing information about a business process so that gaps are readily identifiable. The SAF is designed to address the following:

- identify potential problems with existing or near-term interoperations among components within today's network environments
- highlight the impact on survivability as constrained interoperation moves to more dynamic connectivity
- increase assurance that the business process can survive in the presence of stress and possible failure

Operational activities increasingly require end-to-end completion of a series of steps (mission threads) across systems and components that are independently designed and developed to optimize local needs (local threads). SAF[3] is performed as follows:

- identify a mission thread-specific example
- describe critical steps required to complete the process (end to end)—sequenced activities, participants, and resources
- select one or more critical steps within the mission thread for detail analysis.
- identify the mission-critical resource(s)
- identify stresses relevant to critical resources within the context of the mission
- evaluate threats relevant to the selected mission-critical resource

This view of survivability is needed because of the growing complexity of systems and technology components that must work together to execute a mission thread. Current certification and accreditation approaches such as DITSCAP and DIACAP have worked well for relatively isolated systems. However, the move to a highly interoperable environment is unprecedented for the Army. SAF provides a framework for evaluating the quality of linkages among roles, dependencies, constraints, and risks for critical technology capabilities. SAF provides a structure that supports the connection of analysis to mission success.

[3] See *Survivability Assurance for Systems of Systems* (distributed to attendees) for further details and examples.

2.1 WIDER SPECTRUM OF FAILURES

Technologies such as Web services make it easier to assemble systems, but ease of assembly may only increase the risk of deploying systems whose behavior is not predictable. Fairly simple computing architectures that could be understood have been replaced by distributed, interconnected, and interdependent networks. Business requirements increase the likelihood of failure by bringing together incompatible systems or by simply growing beyond the ability to manage change. As we depend more on interdependent systems, failures are not only more likely but also harder to identify and fix.

DoD must consider the following trends that increase complexity and interoperability, increasing the importance of survivability analysis.

Increasing scope and scale of systems

- Diversity of users
- Dynamics of usage

Increased importance of non-functional requirements

- Systems of systems (SoS) issues
- Operational independence of elements
- Managerial independence of elements
- Emergent behavior
- Geographic distribution

Continuous evolution and deployment

- Heterogeneous, inconsistent, and changing elements
- Erosion of the people/system boundary
- Normal failures

An increasing number of failures are caused by unanticipated interactions between SoS constituents. Failures may be the result of discrepancies between the expected activity and the actual behavior that occurs normally in business processes. The overall success of a business process depends on how these discrepancies are dealt with by staff and supporting computing systems. Changes in business processes and systems often introduce these kinds of discrepancies.

Dealing with discrepancies becomes much more difficult as the number of participants—people and systems—increases. Each participant has to deal with multiple sources of discrepancies, and a single discrepancy can affect multiple participants. There is increased likelihood that a poorly managed discrepancy will result in additional discrepancies affecting additional participants. Failures are frequently the result of multiple, often individually manageable errors that collectively become overwhelming. This pattern is shown graphically in Figure 4.

Individual failures can be identified and mitigated using a range of available analysis and development techniques, but recent experience, such as the Northeast power failure in 2003, has shown that consideration of individual failures is insufficient for complex technology environments. SAF provides a structure for considering a range of mission-critical resources and evaluating the ways in which a failure in one resource can stress other resources to lead to cascading failures.

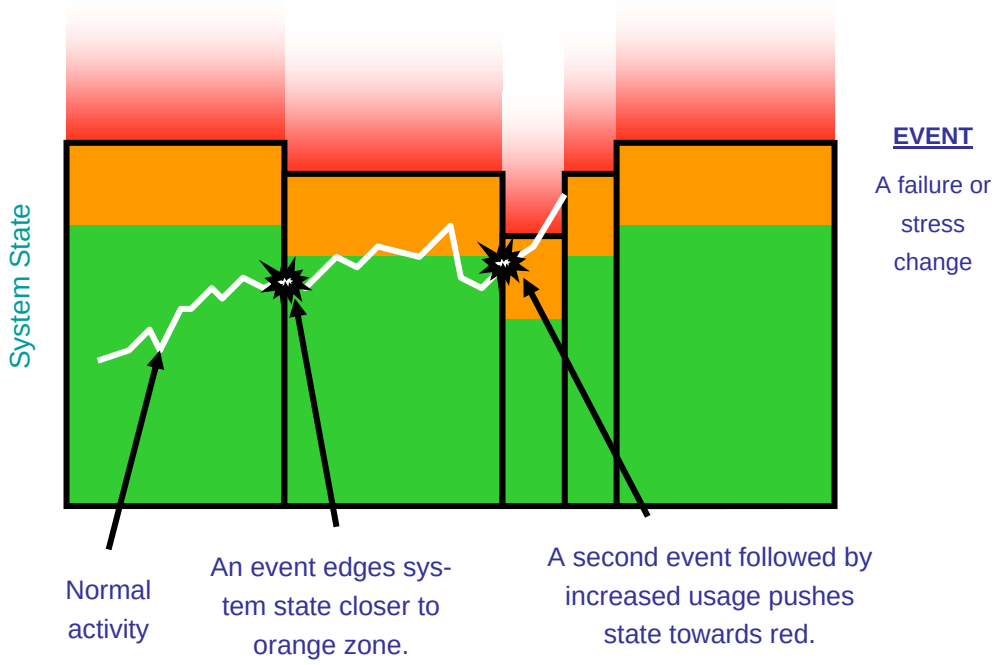

Figure 4: How Complex Systems Fail – A Combination of Problems

Stresses are the normal variations that occur constantly in the course of performing a business process. Some are expected variations that the process is constructed to accommodate, such as higher volumes. In addition, unexpected errors and variations that the business process is not designed to accommodate can occur, leading to potential failure of a critical step and subsequent impact on the successful completion of the business process. In building the failure outcomes for each critical step of a mission thread, a range of stress types and potential failures should be considered. Stresses may include

- interaction (data) triggered stress—missing, inconsistent, incorrect, unexpected, incomplete, unintelligible, out of date, duplicate
- resource triggered stress—insufficient, unavailable, excessive, latency, inappropriate, interrupted
- people-triggered stress—information overload, analysis paralysis, distraction (rubbernecking), selective focus (only looking for positive reinforcement), diffusion of responsibility (for example, “it’s not my job”), lack of skills or training

Large distributed systems are constructed incrementally. The functionality of the initial deployment of a system may suggest other applications that were not anticipated in the initial design. Users frequently exploit system functionality in unanticipated ways that improve the business processes but that may also stress the operation of components that were not designed for the new usage.

Discrepancies (stresses and errors) arise normally in business processes. The overall success of a business process depends on how effectively discrepancies are accommodated through the people, resources, and actions that comprise the end-to-end process. Changes in business processes and systems can introduce new types of discrepancies. Inconsistencies must be assumed as we compose systems:

- Systems developed at different times exhibit variances in technology and expected usage.
- A system will not be constructed from uniform parts; there are always some misfits, especially as the system is extended and repaired.

Human interactions may be necessary to bridge between systems, eroding the boundary between people and system and establishing critical business process dependencies on people interacting with multiple systems.

2.2 SAF SUMMARY

The current certification and accreditation (DIACAP) approach used by the Army is necessary but not sufficient for establishing information assurance and mission survivability for systems of systems. Requirements for system development and validation are incomplete without an analysis and evaluation of failure potential and mission survivability. Failure analysis must move beyond considering the failure of systems and resources individually to identifying the impact of interacting failures. SAF provides a means of establishing context for connecting components to mission to identify faulty assumptions, critical interactions, and missing requirements.

3 Interoperable Acquisition and Programmatics

Interoperable acquisition consists of practices that enable acquisition, development, and operations organizations to collaborate more effectively and field interoperable systems. These practices are distinguishable from the processes used to acquire individual systems. The purpose of interoperable acquisition is to influence acquisition behavior of the constituents to maximize the likelihood of a successful system-of-systems outcome, as opposed to maximizing the outcome for any individual system.

Traditional system acquisition focuses on achieving functional requirements along with specific performance objectives such as throughput and transactions per second, maintainability, and reliability within cost and schedule constraints. Each system is developed in response to a perceived operational mission need with individual funding, community of interest, etc. This system-centric approach results in a proliferation of stovepipe systems that can only be integrated with difficulty—if at all.

3.1 CONSIDERATIONS FOR INTEROPERABILE ACQUISITION

Conventional notions of "point-to-point" interoperability are insufficient in a system of systems context: interoperability requirements cannot be completely specified during construction, only becoming apparent at run time … and they may change each and every time!

While the DoD interoperability goals are laudable:

- Seamless interoperability
- Dynamic, flexible, responsive to changing environment and threat, survivable
- Programs will "cooperate and graduate"

the reality is lacking in effectiveness:

- Interoperability is either insufficiently defined (e.g., "interoperable with *XYZ* system") or too narrowly constrained (e.g., "interoperable with *XYZ* system, using TDMA over 5 KHz UHF DAMA SATCOM …" etc.
- Integration results in brittle systems that are fragile and difficult to sustain
- Stovepiped programs/systems not incentivized to engage in altruistic behavior

A different perspective of interoperability—the system of systems interoperability (SOSI) model— is shown in Figure 5. To understand this model, one must recognize that programmatic, constructive, and operational interoperability are different *aspects* of interoperability. In conventional practice, interoperability is usually defined as a characteristic of fielded systems. However, using a more sophisticated definition for interoperability (i.e., the ability of fielded systems to exchange information, and act upon that information according to some shared understanding), we can see that realizing interoperability in the fielded systems—operational interoperability—requires that

we consider the implications to the acquisition and development activities, as well as the interrelationships between all three aspects. To be successful, acquisition programs must address these interactions. For example:

- *Cost* (programmatic) may affect the *architecture* (constructive), which affects the capabilities of an *operational system* (operational).
- *Technology maturity* (constructive) may affect the *risk* of holding *schedule* (programmatic) to deliver an *operational system* (operational).
- The ability to reflect evolving *doctrinal concepts* (operational) into a *system of systems acquisition strategy* (programmatic) and *system of systems architectural concepts* (constructive)

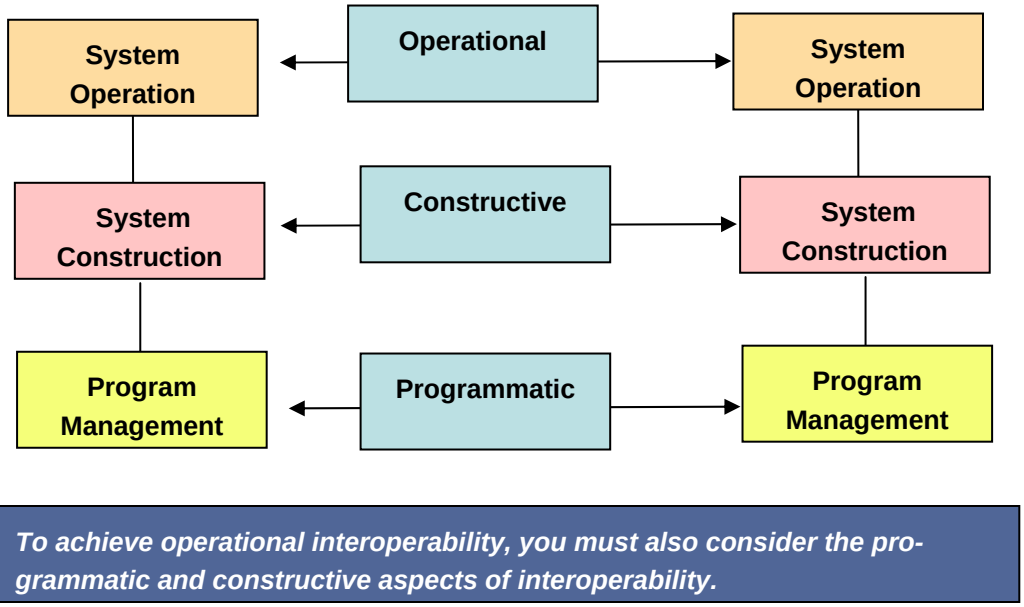

Figure 5: System of Systems Interoperability Model (SOSI)

This is achieved through sharing relevant information and performing necessary activities that enable the collective behavior of these organizations to successfully deliver system-of-systems capabilities.

3.2 CONSIDERATIONS FOR SYSTEM OF SYSTEMS PROGRAMATICS

As shown in [Smith 06], rigid, hierarchical management and organizational patterns are inappropriate in systems of systems. Similarly, practices and processes cannot be determined solely by a single program: they must reflect the reality that they are influenced by **everything else** in the system of systems [Meyers 08]:

- Causes and effects become a combination of known, unknown, and unknowable

- Dependencies arise in context of the particular program associated with other programs. Changing dependencies may need to be managed.
- Influences with other organizations and effects they could have on a particular acquisition program
- Control of information and decisions require negotiation to develop a shared understanding.

Specific recommendations for various practices are provided below.

Influence Management

Influence management consists of two phases: first, you need to identify existing influence relationships:

- Explicit understanding—and agreement—on the semantics of the relationship
- Define how things ***actually happen*** rather than how they are supposed to happen; accuracy is important, but precision is not; comprehensiveness is important, but completeness is impossible.

Second, it is necessary that one maintains the current state of influence relationships.

- The nature of—and the extent, parties, etc.—interdependency relationships will change.
- Be aware of emergent, cascading, and multi-dimensional influence relationships.

Schedule and Risk Management

Schedules should take into account external influences, constraints, etc.

- Make interrelationships explicit (tacit knowledge or assumed agreements may have drastic implications).
- Identify relationship chains that "cross over" from one aspect to another or have "downstream" effects.

Control of schedule information and decisions require negotiation to obtain a consensus -- global optimization may require sub-optimization at the individual project level.

Schedules will continually change and evolve.

Risk mitigation needs to factor in the broader SoS context—some mitigations may result in sub-optimal results for individual systems.

System of Systems Engineering

Systems of systems engineering is about continually understanding and negotiating what capability can be provided that is sufficient to meet current and evolving operational or business objectives:

- continuous characterization of needs, of drivers of change, of evolving systems
- control is no longer the purview of each individual project, but must be shared

Solutions are formed that encompass the full operational life—implications of funding, schedule, capability on transition to operations, support, repairs (and evolution).

Solutions need to be designed with uncertainty and change in mind, including different operational contexts than those initially envisioned, less-than-perfect communications (e.g., latency, robustness), and unanticipated patterns of use.

3.3 SUMMARY FOR INTEROPERABLE ACQUISITION AND PROGRAMATICS

Key system-of-systems concepts include

- Unboundedness—or uncertainty about the precise boundaries—of systems of systems
- Uncertainty/fluidity of system-of-systems constituents, operational context, requirements, etc.
- Emergence as a fundamental (or perhaps dominant) behavioral mode in systems of systems

To deal effectively with these, you need to

- Identify, understand, and maintain knowledge of influence relationships.
- Balance competing interests across program management, development, and deployment/operations.
- Utilize processes/procedures that leverage emergence—don’t assume it away!

Systems of systems require a holistic interoperable acquisition approach—everything is connected to everything else.

4 Mission Success in Complex Environments (MSCE)

Although most software development programs and organizations implement some type of risk management approach, preventable failures are commonplace. A variety of issues are at the root of these failures, including the following:

- **Uneven and inconsistent application of current risk-management practice is common.**
- **Point solutions that focus on a specific life-cycle phase and type of risk (e.g., program, security) are prevalent, which results in significant gaps in risk-management practices across the life cycle.**
- **Traditional risk management practices do not readily scale to today's multi-enterprise, multi-system environments (e.g., collaborative software development programs, systems of systems).**

The problem will only get worse as the size, scope, scale, and complexity of software-intensive systems increases and the interconnectedness programs increases. New approaches to risk management are needed. Based on this need, SEI chartered the Mission Success in Complex Environments (MSCE) Special Project. MSCE's mission is to develop practical and innovative risk-based methods, tools, and techniques for ensuring successful operation of software-intensive systems and systems of systems.

MSCE is currently focusing on integrated risk and opportunity management, which is a disciplined, holistic approach for striking the proper balance between the potential for loss and the potential for gain across the life cycle and supply chain. The project team is developing the Mission-Oriented Success Analysis and Improvement Criteria (MOSAIC), a suite of protocols, methods, tools, and techniques for integrated measurement, assessment, and management of risk and opportunity during the development and operation of software-intensive systems and systems of systems.

4.1 APPLYING MOSAIC

MOSAIC's underlying analysis approach comprises the following four steps:

1. Clearly articulate key objectives.
2. Determine the current probability of success.
3. Establish the uncertainty range (i.e., best- and worst-case scenarios for the current probability of success).
4. Examine how events could affect the current probability of success (i.e., sensitivity analysis).

A key objective is a vital outcome intended to be achieved in the future. The set of key objectives for a program provides a benchmark against which success will be judged. Examples of the types of key objectives analyzed in a MOSAIC assessment include performance, technical, cost, schedule, operational, and business/mission objectives. Key objectives are used to set the scope when

assessing distributed programs and operational processes. They are also used to define the conditions and events that must be assessed during an assessment.

The probability of success is defined as the likelihood that a key objective will be achieved (also called the potential for success). A program's current probability of success is established by analyzing a set of drivers, which represent conditions that influence a program's eventual outcome. During a MOSAIC assessment, drivers are used to determine the degree of momentum toward key objectives and estimate the overall chances of achieving key objectives (i.e., the probability of success).

Using the program's key objectives, an appropriate set of drivers is selected from a catalog of software development drivers. The set of drivers address the program, product, and operational conditions that can affect the likelihood of achieving the program's key objectives. Examples of driver categories include the following: organizational conditions, stakeholder sponsorship, technical objectives, funding objectives, schedule objectives, plans, customer requirements, capability, coordination, event management, integration, support of operations, user preparedness, maintainer preparedness, data security, system availability, contingency plan, infrastructure support.

Driver questions are phrased from the success perspective. Probability is incorporated into the range of answers for each driver. Driver data is collected through interviews and evaluated based on predefined criteria. A scenario-based analysis of drivers is used to determine the current probability of success.

Uncertainty is generally defined as having doubt or being unsure of something. Some degree of uncertainty will be associated with the probability of success based on having doubts about or being unsure of current conditions (i.e., driver values). As drivers are analyzed, key uncertainties are identified. MOSAIC's uncertainty analysis then examines how the probability of success will be affected when uncertainties are resolved. In some instances, the probability of success will increase when conditions end up being better than expected. In others, the probability of success will decrease when conditions end up being worse than expected. These best- and worst-case scenarios define the uncertainty range for the current probability of success.

An event is an occurrence that changes the current state (i.e., status quo). Events can have a positive or negative effect on the outcome. For example, a decrease in funding would likely produce a negative consequence that might adversely affect a project's outcome. In contrast, an increase in funding would likely produce a positive consequence that might put a project in better position for success. MOSAIC's sensitivity analysis examines an event's likely effect on the potential for success. An increase in the potential for success resulting from the occurrence of an event is called a tactical opportunity, while a decrease in the potential for success resulting from the occurrence of an event is called a tactical risk. A range of events are typically analyzed during a MOSAIC assessment. The management goal related to events is twofold: (1) to minimize the potential, negative consequences of events and (2) to maximize the potential, positive consequences of events.

4.2 MOSAIC SUMMARY

Most traditional risk-analysis approaches are point solutions focused on a specific entity (e.g., organization, project, or asset), a particular life-cycle phase, and a single type of risk (e.g., secu-

rity, project, operational). In contrast, MOSAIC defines a broad assessment of a program's key objectives based on key drivers of success. When a driver indicates potential concern, mitigation actions or solutions might be recommended based on the results of the MOSAIC assessment. In some cases, additional in-depth analysis might be recommended to reduce uncertainty and better characterize the value of a driver. SAF and Interoperable Acquisition and Programmatics are two examples of follow-on, in-depth analysis that might be recommended as follow-on analyses. This concept is illustrated in Figure 6.

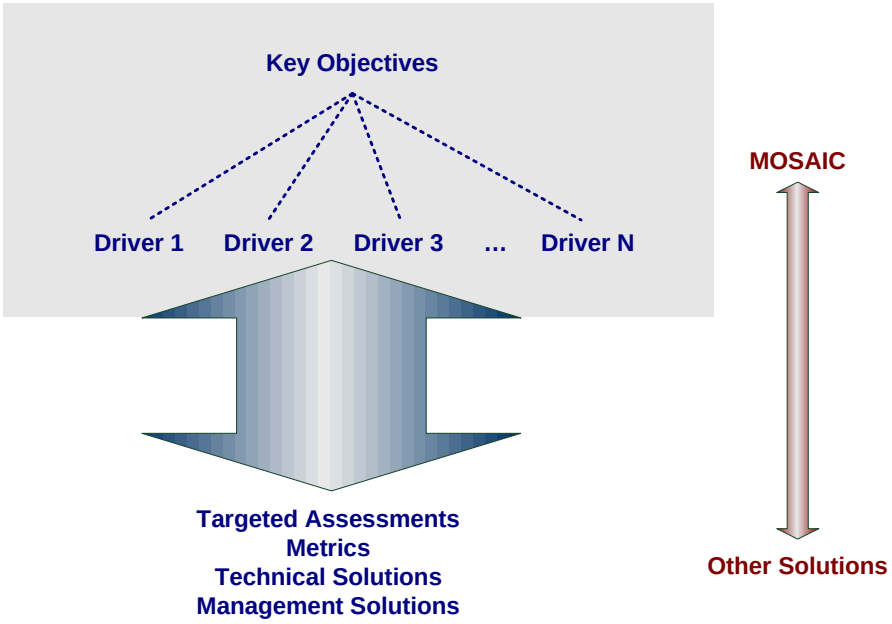

Figure 6: MOSAIC Context

Finally, most traditional risk management approaches focus on preventing failure, which fosters a mindset of playing not to lose. Managers following this philosophy can easily lose sight of what needs to be accomplished to achieve a successful outcome, with failure often the unintended result. In contrast, MOSAIC is based on a philosophy of playing to win. With MOSAIC, managers can strike a balance between risk and opportunity across the life cycle and throughout the supply chain, which puts them in better position to succeed.

5 Post-workshop Analysis

A process of affinity and causal grouping and prioritization similar to that employed in a workshop "Exploring Programmatic interoperability: Army Future Force Workshop" [Smith 05] was used to analyze the raw issues identified by the workshop participants (enumerated in the Appendix of this document) and identify any causal relationships among the issues, and provide a structure for assessing potential solutions. The analysis was conducted in two stages. First, the issues were analyzed to identify any affinity groupings within a single issue category (i.e., management, technology), as well as across categories. The results of this analysis were used to identify causal relationships between groups of related issues. Second, using the insights thus gained, the Army needs identified by the workshop participants (listed in the Appendix) can be analyzed with respect to the affinity groupings and causal relationships to see how well they addressed the identified issues.

5.1 AFFINITY GROUP AND CAUSAL ANALYSIS

The first part of this analysis consisted of taking the individual issues in each category (i.e., management and technology) and examining them to see if there were any identifiable affinity groupings. Looking just at the management issues, there appeared to be five distinct sets of related issues, plus a small number of issues that don't appear to be related:

1. Disconnect between system-of-systems demands and conventional practices: These issues highlight some of the critical differences between the nature of program management practices (e.g., cost estimating) under conventional (i.e., system- and program-centric) program management and system-of-systems contexts. This appears to have a causal relationship with the other management issue affinity groupings.

2. Requirements: These issues illustrate some of the key differences between defining and managing requirements and specifications in systems of systems, versus individual systems and programs.

3. Compliance versus capabilities: These issues relate to the absence of a direct linkage between adherence to standards, or compliance with key requirements (e.g., NR-KPP) and the actual performance of the system of systems.

4. Testing: While closely related to the previous grouping, these issues address the broader problem of how—exactly—to test systems of systems.

5. Open versus proprietary: These issues are associated with the impacts on various aspects of systems of systems, including sustainment, evolution, and competition from decisions made about the degree of "openness" desired, and the degree to which vendor-specific, proprietary solutions will be allowed/required.

These groupings are shown in Figure 7 (the issue numbers—M1, M2, and so forth—refer to the numbers appended to the issue descriptions in the Appendix).

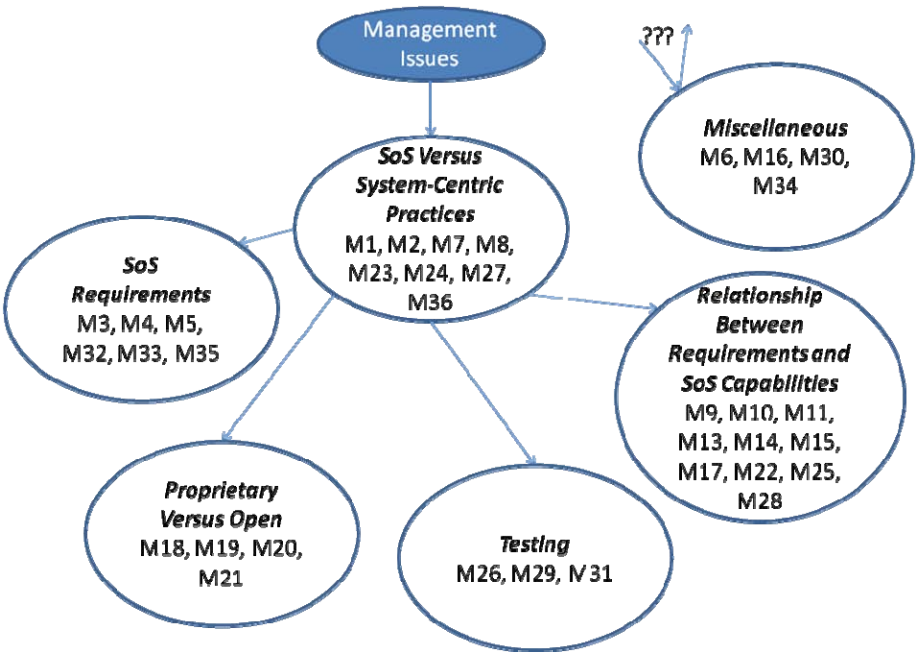

Figure 7: Management Issues Affinity Grouping

It is interesting to note that the affinity groupings obtained for the management issues appear to reflect the close coupling between different aspects of process, product, and acquisition needed for the increased complexity of the system of system environment supporting the need for the type of MVDM approach the facilitators have proposed in this workshop to provide a way of balancing among these competing aspects.

Based on the workshop facilitator's experience, we would expect to see similar correlations between affinity groupings in the technical issues. We would also expect to see a strong correlation between some of the perceived needs and the issues identified by the workshop participants, as well as several cases where the articulated needs don't clearly map to a known problem. There might be several possible explanations for this, including any or all of the following:

- Some of the perceived needs, management and technical issues, or both were influenced by a general sense of frustration (or an opportunity to "vent"), rather than by a purely objective analysis if the problem.
- Perceived needs may be influenced by the most recent caused problem experienced by a particular participant (i.e., "wolf closest to the sled").

- Due to the relatively limited cross-section of workshop participants, their experience—and experiences—may not be truly reflective of the broader Army.

But without further analysis, there is no way to determine the actual cause.

5.2 FURTHER ANALYSIS OF ARMY NEEDS AND ISSUES

This workshop represents a single snapshot of the issues and needs of the Army, based on a particular set of workshop participants. As indicated in the previous section, it is possible that these represent a good cross-section of all major Army system-of-systems acquisition issues and needs, but experience tells us that this is probably not the case. To that end, the Army should consider conducting similar analyses of the data on issues and recommendations that have been developed over the recent series of acquisition workshops to establish a more comprehensive set of issues, and target specific communities of interest within the Army that are missing from all of the workshops.

With the consideration of this reasonably broad range of participants, the confidence that an appropriate set of critical issues (those consistent among a broad range of participants and those critical to an important subgroup) could be selected to define where further analysis would be of value.

6 Next Steps

Each of the three analysis approaches presented in the workshop (i.e., SAF, interoperable acquisition, and MOSAIC) is focused on solving distinct problems inherent in a system-of-systems environment. In addition, each has been piloted with and provided tangible benefits to DoD and federal programs. While all three are focused on managing risk in a system-of-systems environment, each assumes a distinct management perspective. SAF provides information about security and survivability risks; interoperable acquisition examines programmatic interoperability risks; and MOSAIC assesses a system-of-system's overall risk as well as its probability of success.

As developers of these three analysis approaches, we see considerable synergy among them. We believe that an integrated analysis approach will provide managers with a more comprehensive view of their system-of-systems risks. As a result, we have begun to explore the connections among SAF, interoperable acquisition, and MOSAIC. This workshop describes early research efforts in the area of multi-view decision making (MVDM), an integrated analysis and management approach for system-of-systems environments.

The initial analysis of the Army issues and needs identified in the workshop (Section 5 of this document) indicates the need for a new approach to better manage the increasing complexity of the system-of-systems environments. Further analysis options are suggested to refine the preliminary conclusions of this workshop and establish a comprehensive set of management and technical issues that the Army must address to successfully manage system-of-systems efforts.

Participants at the workshop discussed the limitations of and issues affecting management and technical approaches currently being used in systems-of-systems environments. The information collected from the workshop is listed in the Appendix of this document. We believe that the integrated approach underlying MVDM can provide a first step in addressing many of these limitations and issues and warrants further consideration.

During the workshop, we also facilitated a discussion about how MVDM might be applied in the Army. The following three options were discussed:

1. Identify a potential pilot opportunity for one of the approaches (e.g., MOSAIC). Here, SEI could present the selected approach to Army program managers and work with them to identify a candidate pilot opportunity. The goal of the pilot would be to compare the selected approach to the Army's current best practices for managing risk in systems-of-systems environments.
2. SEI could benchmark the use of the three approaches presented in the workshop for managing risk in selected Army programs to answer the following questions:

    - What is the state-of-the-practice for managing risk in system-of-systems environments?
    - Where are the strengths and weaknesses in the Army's risk management practices?

The goal of this option would be to determine whether the SEI integrated approach or any of its components would enhance the Army's ability to manage risk in a system-of-systems environment.

3. The Army could identify a system-of-systems program in which to pilot the SEI integrated assessment approach. The goal would be to identify benefits and lessons learned by applying the MVDM approach with an Army system-of-systems program.

Participants in the workshop identified the above options as potential next steps for MVDM research. They also recommended that SEI continue to engage the Army through workshops, presentations, and discussion with a wide range of participants to socialize the ideas and concepts presented in the workshop. However, no clear action plan for implementing the options and socializing MVDM concepts was developed during the workshop.

Appendix Participant Information Collected at the Workshop

6.1 CURRENT ARMY APPROACHES TO COMPLEX AND HIGHLY INTEGRATED SYSTEM DEVELOPMENT

The following lists were developed during a free-form discussion with participants at the workshop around the topic of current Army responses to complex and highly integrated system development at the start of the workshop. The sub-issues under one selected management area (KPP) and one technology area (software blocking) were assembled during a discussion on the second day of the workshop.

The workshop facilitators provided some affinity grouping for initial analysis (noted with Mx or Tx in the subsections below). The management affinity groups are further discussed in Section 5 of this report.

6.1.1 Management Issues

- Ask vendor for estimate of cost impact on a single-system basis (M1)
- No funding to address vendor costs – "grovel" (M2)
- Change management
  - Change requires a "driver" to push it through (M3)
  - Need sponsor for technology change to push through layers of review to successfully POM (M4)
  - Changes in requirements and specifications have an impact which is not well articulated (M5)
  - Data calls to assess impact *(leadership seeking better information?)*(M6)
- KPP
  - Ownership for capabilities such as net ready KPP is unclear (J6) (M7)
    - CJCS
    - Law behind this?
  - Capability cost estimating (e.g., net ready KPP) is a challenge (M8)
  - CIO G6 sends out a letter of policy to establish mandate (M9)
  - Ease of compliance?
    - Not clear that compliance achieves desired effects (self diagnostic checklist) (M10)
    - Needs interoperability purpose (M11)
    - Works for development but not maintenance (M12)
  - Quantifiability? Verifiability? Documentation exercise (has anyone ever failed?) (M13)
  - Distracts from "real issues" – working to pass the test that does not equate to real interoperability (M14)
  - Networthiness linkage? (M15)
  - ISR emphasis – expansion of network – trade armor for information (M16)
  - Need for end to end aggregate effort to meet goals of interoperability – drive of KPP disconnected from reality of needs (M17)
  - Did end emphasis on proprietary solutions? Puts attention at specification level to assure systems can play together
    - Are proprietary solutions bad? Should be explicit in choices (M18)

        - Have it already – Microsoft, LINUX (FCS) (M19)
        - Sustainment contracts for recomplete – not real competition when current support owns the data (M20)
        - Common formats for interface may be sufficient? Depends on where interaction is handled (M21)
        - Emphasis on emerging standards for commodity opportunity (M22)
    - Makes sense for single system focus? Insufficient specific? (M23)
    - Should this be applied to the aggregate instead of single systems?
        - When have responsible organizations at SoS level – put architecture together to guide membership in family (e.g., SOSCOE for FCS) (M24)
        - Attempting to push up common solutions – may make it so hard at too high a level (M25)
        - How do in meaningful way? (M26)
    - Need to drive conformance validation from contextual requirements
        - Could be owned by group responsible for SoS (M27)
        - Evolution may be at too low a level (e.g., Patriot) (M28)
        - Context overlap and context clashes – currently being studied (M29)
- Everyone "figures out" how to comply
    - Form a community of interest or a community of practice to share information and funding for broad challenges (informal initially and may be formalized) (M30)
- Can't test for everything that can happen in the field (M31)
- Can't specify long-term requirements today (M32)
- Requirements changes through staffing process (M33)
- Management of software blocking is complex and does not work well – delays critical changes to the operational field while meeting interoperability mandates with uncertain operational need
    - Software blocking tells Army what is coming (M34)
    - Requirements are known, but not well executed (M35)
    - Your ability to complete your work in a timely manner becomes more complex because of system dependencies (M36)

6.1.2 Technology Issues

- Software blocking – suite of what must fit together at a point in transition
    - Interoperability validation mission threads for software block are published well in advance (one year) (T1)
    - Block 2 has taken two years (primarily maintenance changes) – started Oct 06 and currently planned for Oct 08 (T2)
    - Requirements well defined for a block, but the timeline to meet requirements not well connected to each program (T3)
- Software blocking as emphasized in Army
    - Fielding too many systems putting too much pressure on operational environment – field block at once to train operational environment once – expanded to validate integration pieces – identified as interoperability certification point can you pass or fail mission threads sufficiently (T4)
    - Block 2 – 69 systems – certification is no level 1 or 2 TIR
        - Struggle to get into test block with mission threads not defined in ORD/CDD (T5)
    - What is the problem being addressed? Concept is good but execution hurts – no code can be changed until full testing completed (CTSF made this choice) once start test of record (T6)
    - Has test-fix-test cycle in advance (T7)

- Mission threads are not prioritized – all considered equal weight (T8)
- Criticality defined as completion of mission thread
  - Alternate paths are not articulated for "work around" (T9)
  - Network is as close to live environment as is available (T10)
- Volume of what is included in a block is extensive (T11)
- G6 certification of systems that work – what is good enough (T12)
- Articulation of system of systems (T13)
- Mission threads can result in incomplete validation of the system of systems (T14)
- Way of holding PMs to a decision point and process (T15)
- After development and acceptance testing (T16)
- OT on the block to be added next year (T17)
- End up doing the "what if" considerations at the PM level – must field something with soldiers assuming block testing completed by point in time
  - Block 1 deployed in 2004 (T18)
  - Block 2 deployment 2009 (T19)
  - 4-5 year cycle to get technology into the field (T20)
  - Is it worth it? Level of trust for program managers? (T21)
- Digital CORPS needed and have to be able to have interoperability (T22)
- Alternative:
  - Existing software ready as base case with suite of regression testing (T23)
  - Each replacement of a module – validate with full regression testing (T24)
  - Replacement for 67 systems with validation of each independently in rolling incremental base line may be too long in cycle time (T25)
- Specific points of insertion with redeployment cycles into operational theater (T26)
- Counterpart in real world – researching this option (T27)
- Alternative:
  - Sufficiently describe boundaries among components to build confidence that changing one will not break something else (T28)
  - Cannot adequately articulate performance requirements (T29)
- Insufficient emphasis placed on interoperability throughout the development (T30)
- 5 % causing 90% of the problems? (T31)
- Navy approach – less interactive relationships; fewer of them; evaluate with capabilities and limitations vs. pass/fail (T32)
- Army development build on interdependencies that require block fielding (T33)
- Effort to get at root causes for problems being experienced after block deployment and link back to block dependencies (T34)
- Establish clusters with focus on validation steps (smaller group) (T35)
- PMs have developers building stuff while block validation comes along and fixing block pieces requires tracking to resources no longer under contract; budget decisions impacting PM based on available resources to meet build window schedules (T36)
- What are interoperability issues and try and to design these in – know two years from now what are the critical issues (T37)
- Is expected capability of flexibility really providing value – would we be better living within limits to gain capability in the field faster (T38)
- Limit number of mission threads to really critical ones – others may be shifted to capability and limitations (T39)

    - Not really stressing systems to determine limitations with block approach (T40)
    - What are limits of test/fix/test cycle? (T41)
- Changes such as IPv4 to IPv6 are huge and overwhelming – over 100 systems on the approved list (T42)
- Cost estimates for the life cycle are based on a concept that is locked in at milestones A/B and never adjusted based on how the program must meet realities (T43)
- Waivers are sought for interim options (e.g., JTRS radio) when expected solutions from dependent programs are not met (T44)
- Primary focus of PM is functionality (T45)
- Potential for emergent behavior is greater as interoperability arrives in the theater (T46)
- Requirements structure not set up to
    - accommodate "spiral approach" with need to fully document each step faster (T47)
    - establish "end state" requirements that do not adjust sufficiently quickly for complex changes (T48)
- JCIDS process creating wrong focus for complexity beyond specific program (T49)
- Buying large quantities of technology too early in the project process for cost savings creates early lock-in that may not be appropriate for the operational timetable and desired result (COTS related problem) (T50)

6.2 ARMY NEEDS FOR ADDRESSING COMPLEXITY IN SOFTWARE DEVELOPMENT

Based on what participants learned from the presentations of the three views aimed to improve mission success, survivability, and interoperability, workshop participants identified the following areas of need for the Army.

- Defense architecting – range of behavior instead of specifications without context
    - Key under-determinations – range of behaviors may be more appropriate than KPP approach (context free) (N1)
    - Broad participation drives requirements to be as flexible as possible (e.g. Travelocity/Orbits options – makes participation easy with limited investment to motivate) – industry standards emphasis – evolves over time to find right "sweet spot" – recognition of need to change over time to continue to grow participation (N2)
        - Needs a "keeper" services to provide the shared approach
        - Need to find the "win-win" opportunity
    - Need series of contextual driven capabilities that promote participation – supporting capabilities – finer level of granulation (consider the international banking industry model – consumer driven) (N3)
    - Capability bins based on the "consumer market" being supported (N4)
    - Applying MSCE, SAF, interoperable acquisition to provide "real context" to evolve into (frameworks for articulating drivers, connecting quality and acquisition to drivers) (N5)
    - Difficult to identify fixed anchor points to drive into range of behaviors that become a standard – requires community recognition of value
        - Performance and advance capability emphasis drives to "new" solution (N6)
        - Business model driving commercial incentives does not promote this cooperation (N7)

        - For example – GCSS-AF (SOA framework) provides suite of capabilities attracting participation – need to reach tipping point
            - Suite of APIs participants build to (N8)
            - Different business model for participating vendors (N9)
        - Consider building a capability set instead of constructing individual solutions (N10)
        - Early attempts for common set – lack of trust on base set and performance not up to needs (N11)
        - Services library – reference model project (N12)
- Need to align reward system with key drivers, but these are not well articulated
    - Need for semantic as well as syntactical solutions – requires context (N13)
    - Industry will build what we ask for (N14)
    - Individual key drivers are different from organizational key driver of interoperability (program performance) (N15)
    - Techniques could be used to provide structure to the discussion of options
        - Articulated specific threads to work through competing choices (N16)
        - Establish critical suite of mission needs to drive the discussion of trade-offs (N17)
- Single sign-on concept – bought a solution with vendor and specification
    - Asset library support AKO single-signon (N18)
    - Articulated solution to specific need (N19)
    - Tied to product – risk of support and long-term capability (N20)
    - Architect to be able to change product? (N21)
- Should expect failure in choices and learn from failures
    - Will not get choices right the first time (N22)
    - Need ability to evolve without disruption of current use (N23)
    - Need to have notion of leading indicators to determine what is working and what needs to change (N24)
    - Technically may not be able to mature everything because not everything is backward compatible – over-constrained in current environment (N25)
    - Transition to "better world" will require pain somewhere – change requires ability to "kill" pieces that do not evolve (N26)
- Functionality embedded into software (existing Fortran/COBOL) – pre-documentation, etc. – need different tools to pull knowledge out of earlier systems (N27)
- How do you capture a return on investment for change?
    - Potential for modeling? Evaluating what have today to meet the future needs? (N28)
    - Identified need for interoperability – evolving driver? (N29)
- Can community of interest be an opportunity to drive participation into KPP
    - NCES is an attempt to bring this together (N30)
    - Can it be addressed at lower level
        - Link Net Ready KPP to specific COI – each COI picks standards that make sense for them? (N31)
        - Collaboration needed to initiate group solutions that evolve into broader connections at higher levels (N32)
        - Need for shared semantics along with technology solutions (N33)
        - Conflict of who owns the process (government, contractor, COI) (N34)
- What part of the problem does the government seek to own? (N35)

6.3 WORKSHOP EVALUATION

How well did the workshop meet your expectations (Excellent, Good, Fair, Poor): All reviewers responded … Good

How relevant is the subject matter to your work and your organization (Most, Much, Some, Little): Responses covered the range of Most, Much, and Some

Attendee comments:

- Consideration of factors/dependencies of a program with other programs was particularly useful
- More material on decision making tools and approaches, which respond to the multi-views. A few examples of evaluation criteria, business case, decision making tools, etc.
- Would have been useful to have had participants from a greater cross-section of the Army, as there were a great number of system-of-systems discussions that cross boundaries into areas where the attendees were not SMEs.
- The Wednesday AM presentations & Thursday discussions were very helpful & value-added. For me, the MOSAIC model presentation was much longer than it needs to be … it also strikes me as overcomplicating a relatively simple issue/concept set. The Interoperability/SoS issue is far complex enough without introducing additional complexity through common sense type modeling – just my opinion, but I know some other attendees agree. Overall, good event & excellent hospitality.

Bibliography

[GAO 08] *DEFENSE ACQUISITIONS - Assessments of Selected Weapon Programs* (GAO-08-467SP). GAO, Washington, D.C. March 2008. http://www.gao.gov/new.items/d08467sp.pdf

[Meyers 08] Meyers, B. & Smith, J. *Programmatic Interoperability* (CMU/SEI-2008-TN-012). Pittsburgh, Pa: Software Engineering Institute, Carnegie Mellon University, 2008. http://www.sei.cmu.edu/publications/documents/08.reports/08tn012.html

[Smith 05] Smith, J. & Meyers, B. *Exploring Programmatic Interoperability: Army Future Force Workshop* (CMU/SEI-2005-TN-042). Pittsburgh, Pa.: Software Engineering Institute, Carnegie Mellon University, September 2005. http://www.sei.cmu.edu/publications/documents/05.reports/05tn042.html

[Smith 06a] Smith, J. & Phillips, D. *Interoperable Acquisition for Systems of Systems: The Challenges* (CMU/SEI-2006-TN-034), Pittsburgh, Pa: Software Engineering Institute, Carnegie Mellon University, 2006. http://www.sei.cmu.edu/publications/documents/06.reports/06tn034.html

[Smith 06b] Smith, J. *Topics in Interoperability: Structural Programmatics in a System of Systems* (CMU/SEI-2006-TN-037). Pittsburgh, Pa: Software Engineering Institute, Carnegie Mellon University, 2006. http://www.sei.cmu.edu/publications/documents/06.reports/06tn037.html

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
(Leave Blank)	February 2009	Final

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
Multi-view Decision Making (MVDM) Workshop	FA8721-05-C-0003

6. AUTHOR(S)

Christopher Alberts, James Smith II, Carol Woody

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213	CMU/SEI-2008-SR-035

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER
HQ ESC/XPK 5 Eglin Street Hanscom AFB, MA 01731-2116	

11. SUPPLEMENTARY NOTES

12A DISTRIBUTION/AVAILABILITY STATEMENT	12B DISTRIBUTION CODE
Unclassified/Unlimited, DTIC, NTIS	

13. ABSTRACT (MAXIMUM 200 WORDS)

A major hurdle for complex development efforts is the lack of effective insight into problems early enough so solutions can be addressed without jeopardizing critical project and organizational constraints. Effectively addressing these challenges requires a set of programmatic and engineering practices that reflect the realities of system-of-systems development, acquisition, fielding and support: multi-view decision making. In support of the Army Strategic Software Improvement Program FY08 Strategic Software Improvement Plan, the Software Engineering Institute conducted a workshop to introduce three analysis techniques—(1) Mission-Oriented Success Analysis and Improvement Criteria, (2) Interoperable Acquisition, and (3) Survivability Analysis Framework—and how they can be combined to address the challenges of managing complexity. Through the use of examples, the limitations of decision making from any one single view and the value provided by a multi-view approach were presented and discussed. This report provides a recap of the workshop.

14. SUBJECT TERMS	15. NUMBER OF PAGES
program management, acquisition, risk management, managing complexity, software improvement, survivability, mission-oriented success	36

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL