

IMPROVING THE INITIATION OF ACQUISITION ACTIVITIES FOR AUTOMATED INFORMATION SYSTEMS

 ***Chiang H. Ren, Col Stephen Busch, USAF (Ret.),
and Matthew Prebble***

The success rate for acquiring automated information systems (AIS) continues to be a source of considerable interest to Congress and the Department of Defense. Building upon Defense Science Board recommendations, the authors' research identifies improvements for initiating information systems acquisition. Adding a soft start period prior to the Materiel Development Decision allows the acquisition community to negotiate with end users regarding system concepts that can satisfy the *materiel solution concept* and better manage the flexibility of AIS concepts to lower risks. This management is enabled through a newly formulated reference frame that maps the materiel solution concept to the system concept and allows the system concept to be reorganized and optimized prior to the analysis of acquisition approaches.

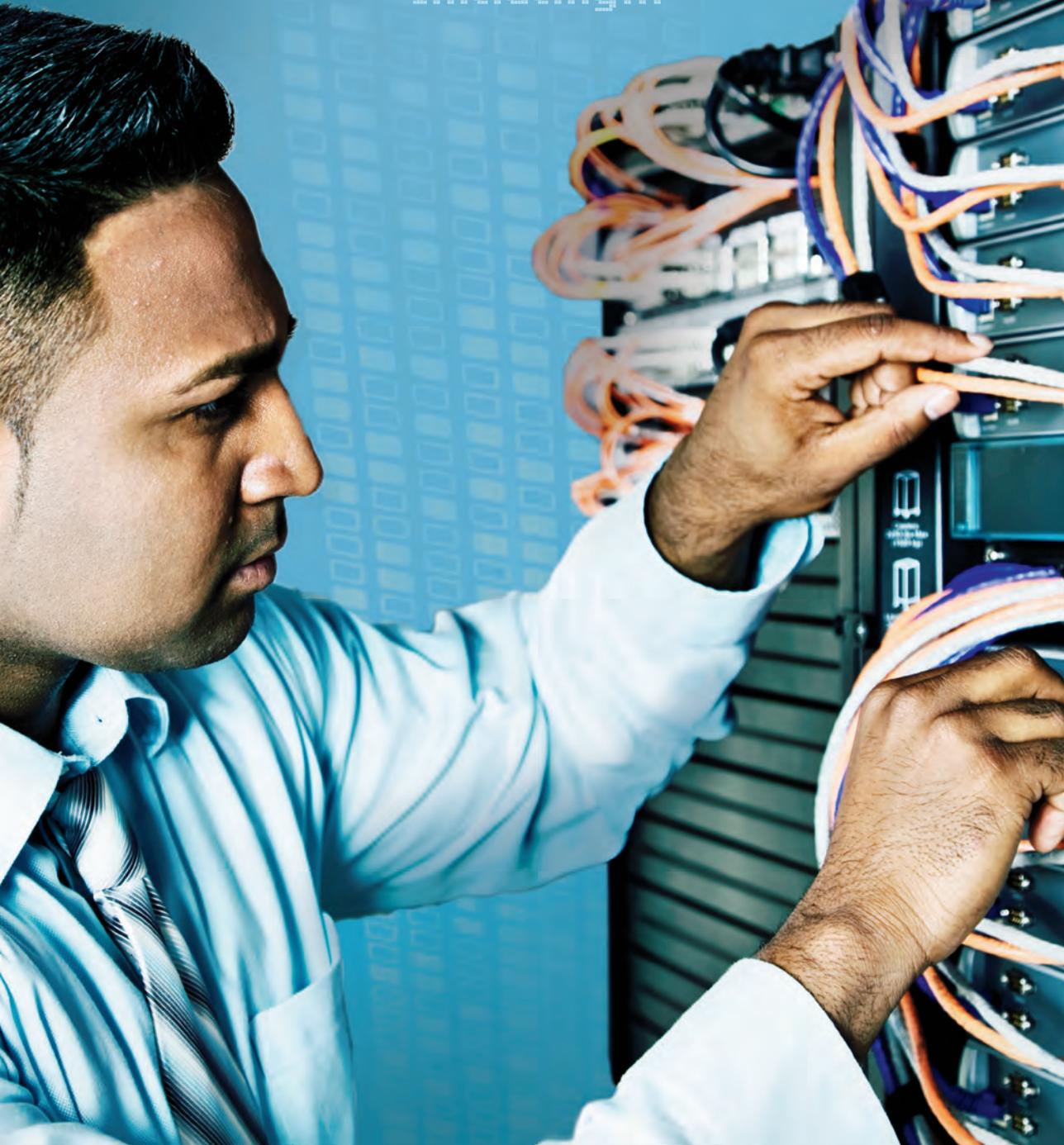
Keywords: *Major Automated Information System (MAIS), Information Technology (IT), Defense Science Board (DSB), Materiel Solution Concept, Materiel Development Decision (MDD), IT Acquisition Process, MAIS Initiation, Section 804 FY10 NDAA*



DoD:>begin improved acquisition_



Initiating...



The U.S. Congress established new reporting requirements and performance constraints for *Major Automated Information System (MAIS)* acquisition programs in section 816 of the FY 07 National Defense Authorization Act (NDAA, 2006). These requirements, codified in 10 United States Code (U.S.C.), chapter 144A, were later modified to include pre-MAIS programs and a completion standard of 5 years from first obligation of funds to full deployment decision.

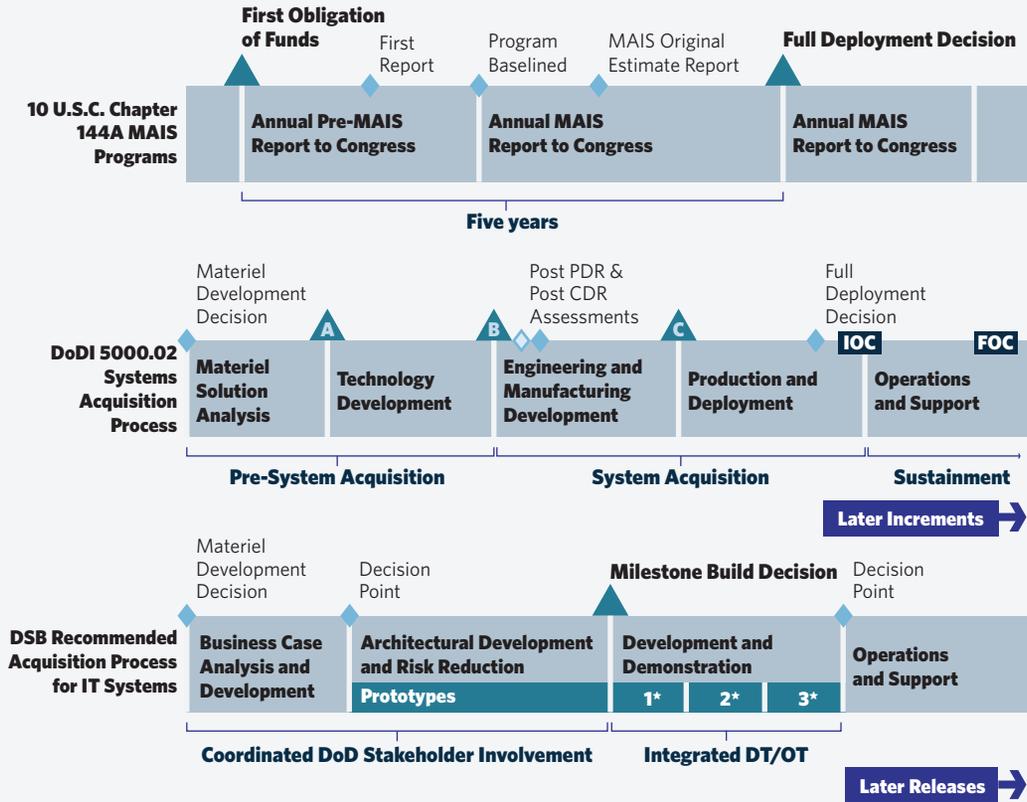
Defense Science Board and Acquisition of Information Technology

At the request of Congress, the *Defense Science Board (DSB)* then studied the DoD policies and procedures for the acquisition of *information technology (IT)* and recommended in their March 2009 report a new acquisition process for IT systems based on commercial worldwide best practices (Department of Defense [DoD], 2009a). This more streamlined process, primarily for stand-alone software development, progressively refines the software product based on continuous stakeholder participation and multiple iterations leading to a major release. The DSB task force further recognized that the current acquisition process for all DoD systems, as presented in the December 8, 2008, release of Department of Defense Instruction (DoDI) 5000.02, is still valid for IT systems acquisition, with substantial trade-offs in design, development of nonsoftware technologies, and integration with major weapon systems (DoD, 2008). The Congressional reporting process for MAIS programs and the current, as well as proposed, acquisition processes for satisfying congressional expectations are presented in Figure 1.

In Section 804 of the FY 10 NDAA, Congress officially called for the Secretary of Defense to develop and implement a new acquisition process for IT systems based on the DSB recommendations (NDAA, 2009). Regarding the selection of which acquisition process to adopt for a specific IT program, the DSB members in their report remarked, "One could argue that if the leadership and program managers cannot sort out this high-level decision, they have no chance of effectively managing or overseeing the programs."

Our research presupposes that this DSB remark actually strikes at the heart of the problem as to why so many IT programs have faced developmental problems in recent years. Identifying which IT programs involve nonsoftware technologies and embedding software into weapon systems, as governed by interoperability standards, are relatively straightforward. But, understanding which IT programs do not involve substantial trade-offs in design and approach can be

FIGURE 1. MAIS REPORTING, DoDI 5000.02, AND DSB RECOMMENDED PROCESSES



Note. CDR = Critical Design Review; DT = Developmental Testing; FOC = Full Operational Capability; IOC = Initial Operational Capability; OT = Operational Testing; PDR = Preliminary Design Review; * = Iteration

overtly or even covertly complex. Much of this complexity starts with how one defines the IT system to be developed.

The IT system can be one small application, a suite of applications, or a total solution for the enterprise because of the integrative nature of modern IT capabilities. Over the past decade, overly ambitious IT programs involving multiple integrated or flexible components have ultimately: (a) decoupled into independent development paths, (b) collapsed due to escalating baselines, (c) breached schedules and costs, and/or (d) failed to fully meet user expectations. At the same time, dozens of small IT programs have been initiated across the DoD with: (a) redundancies in development activities, (b) overlaps in functionality, (c) barriers toward integration or federation, and/or (d) limitations in scalability. Defining the IT system concept correctly at the initiation of acquisition activities is therefore critical to both the

decision to use a streamlined *IT acquisition process* and the approach for leveraging the rigors of the current DoDI 5000.02 process to manage developmental risks. This article will explore methods for improving the initiation of acquisition activities through better concept definition of the IT system.

Analysis Methodology

The objective of this research endeavor is to determine acquisition process improvements for future automated information systems without dwelling on the program deficiencies and failures of the past. Therefore, a methodology of process decomposition and functional advancement is adopted in lieu of case studies. The first step in decomposition is to identify the point of disconnect between the IT materiel solution concept and IT system concept formulation within the current process. This point then permits the insertion of a transformation function, which maps the user materiel solution and the associated system concept to a common reference frame. This reference frame then enables the reorganization of the system concept for more effective acquisition while preserving connectivity to user requirements. An inductive analysis of current terminologies in the IT community is used to create the reference, and a deductive analysis of the relationships within the reference frame is used to establish the application methodology. Finally, recommended changes to the acquisition process are formulated using the analytical results.

Analysis Results

Currently, the initiation of acquisition activities for major IT systems within the DoD is based on a *Materiel Development Decision (MDD)* by the Milestone Decision Authority after consideration of requirements generated through the Joint Capabilities Integration and Development System (JCIDS) and the acquisition community's ability to satisfy the requirements (Joint Chiefs of Staff, 2007). The JCIDS process of capabilities-based assessments, functional area analysis, and functional needs analysis ensures that the established need for a materiel solution (physical system) results in an essential capability for the warfighter. The resulting Initial Capabilities Document must accurately address operational gaps, align with the integrated operational concepts of the regional and functional combatant commanders, and present validated requirements.

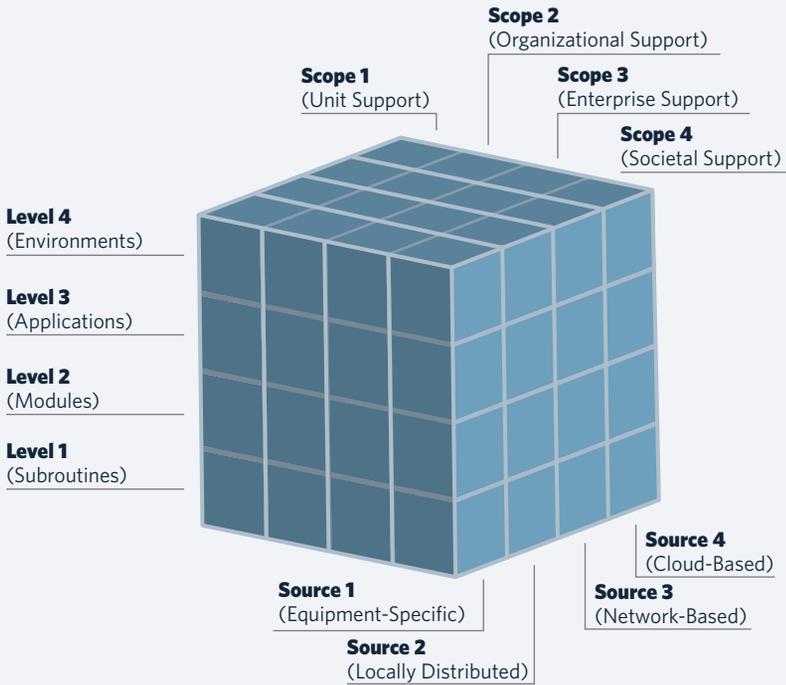
The analysis of the acquisition community's ability to satisfy the requirements must take into account technologies from all sources and explore solution options in the context of the DoD Architectural Framework (Wisnosky & Vogel, 2004). The DoD Architectural Framework used in capabilities-based analysis starts with a high-level operational concept graphic as the first operational view and then quickly progresses down levels of detail in operational understanding, systems definition and configuration, and technical standards.

The current acquisition initiation process, as presented, extends from a tradition of preserving the purity of user/warfighter needs, with the acquisition community responding to these needs. The concept of evolutionary acquisition allows the acquisition community response to evolve in increments driven by stages in technology maturation. However, the assumption is still that the user-defined materiel solution concept can easily extend to a system concept for acquisition. This assumption remains quite valid for mechanical systems. Obviously, it would be ridiculous if the warfighters want a plane and the acquisition community tells the warfighters that what they really want is some wings, avionics, and jet engines. Even when the warfighters want a system of systems, such as missiles, radars, and control centers, the nature of the systems is very clear.

The assumption of clear system concept is not necessarily true in IT acquisition. For example, if the warfighters want an integrated suite of applications to create an enterprise solution environment, it would not be silly for the acquisition community to tell the warfighters that the enterprise can be better supported by more loosely coupled applications developed as multiple programs. Essentially, when users have an IT solution concept, that single concept could equate to a single system, multiple systems, a system of systems, or even a mixture of systems and services. An eagerness of the acquisition community to rush down one path when given validated requirements can lead to great program risks. The complexity of the initiation process cannot be ignored.

A hierarchical architectural framework may not be sufficient to map the complex understanding of requirements into a paradigm where effective system or systems definitions can be formulated. To create a complementary framework, we examined the ways in which IT is described in industry and concluded that interlinked dimensions of characterization can be used to better define IT systems. These dimensions are not clearly elucidated in literature, in part due to the competing approaches and schools of thought in commercial software development. As a result, a better standardized and organized set of parameters for capturing the dimensions of DoD IT system characterization is still merited.

FIGURE 2. REFERENCE FRAME FOR MAPPING MATERIEL SOLUTION CONCEPT TO IT SYSTEM CONCEPT



Our research suggests that once a materiel solution concept with operational requirements has been formulated by the users, the acquisition community can place that concept into a three-dimensional reference frame as shown in Figure 2. Based on initial market research and technology assessment prior to MDD, the solution concept can be mapped across a range of blocks within the reference frame and be defined through the identification of activities in each block. The scalar parameters for establishing each of the three dimensions are proposed and explained in the discussion that follows. The methods of defining IT systems using this mapping of the solution concept are also presented.

Dimension 1: Mapping the Magnitude Level of the Concept

Software/computer codes, unlike mechanical devices, have a greater ability to be both decomposed into progressively smaller individual units and integrated into progressively larger overarching units. Therefore, defining a system based on bounding software in the modern era of distributed computing is almost as hard as finding boundaries within a continuous body of water. The current state of software technology does suggest that four levels of boundaries

can be established. These boundaries are four parameters on a scale that indicate the levels and total magnitude of software organization. When mapping a solution concept to this scale, software organization can reach the point of being an application (Level 3) or creating an environment (Level 4) without first having an extensive number of modules (Level 2) or subroutines (Level 1).

Also, software modules and subroutines can sometimes become stand-alone applications, and software applications can sometimes be recategorized as being able to establish an environment. The distinctiveness of these four parametric levels, therefore, requires integration with the other two dimensions—Scope and Source. Otherwise, the description of activities at each level can become inconsistent from one acquisition effort to the next.

LEVEL 1 (SUBROUTINES)

Portions of code that perform specific tasks or functions within the context of an overarching program/executable file. Subroutines can be reused, and popular subroutines can be stored as a task library for easy access. In a program, subroutines can be within subroutines to form a nested structure and/or be integrated through commands and shared parameters (Fischer, 2001, pp. 1–8).

LEVEL 2 (MODULES)

Independently executable files/programs that perform functions capable of being organized and integrated through interfaces to satisfy the design capabilities of an application. Modules built on the principles of component-based engineering can be reused and rearranged to achieve multiple capabilities (Szyperski, 2002). Also, modules of foundational utility can be offered as a service through Service-Oriented Architectures (Bell, 2008).

LEVEL 3 (APPLICATIONS)

IT products that meet the performance of defined capabilities. Applications will generally have a functional architecture, and the architecture can be designed to be open to the reorganization, updating, and upgrading of constituent modules/components.

LEVEL 4 (ENVIRONMENTS)

IT infrastructures consisting of a network of applications including the core applications that sustain the environments. Environments are

generally designed to satisfy the total mission needs of the entity they support. And, environments can be network-centric, highly restricted and compartmentalized, or centralized.

Dimension 2: Mapping the Utility Scope of the Concept

Once a concept has been mapped to levels of software organization, each level of the software can have different scopes in utility as defined in the following discussion. The most straightforward alignment is that: (a) a software environment can cover the entire society impacted by a DoD mission, (b) a software application within the environment can support a Service or joint enterprise, (c) a software module can address the functional needs of one or more organizations within the enterprise, and (d) a software subroutine can execute a specialized task for a specific unit of the organization. However, the scope of utility does not have to match the level of software organization in a parallel manner. A software environment or application can serve only the mission of a specific unit. A software subroutine or module can alternatively serve the mission of an entire enterprise or society.

A potential mistake in acquisition planning is overlooking the fact that the subroutines and modules used to create applications and environments may either have a more individually unique scope or a broader scope than the application to which they belong. An individually unique scope implies the need for greater fidelity in stakeholder participation, while a broader scope implies latent potential for DoD reuse or co-development. To better understand the varying scopes associated with the elements at each level, the relationship of levels and scopes can be captured in a four-by-four matrix.

SCOPE 1 (UNIT SUPPORT)

Tasks, functions, and/or capabilities are designed around supporting specific divisions within DoD. A high degree of specificity, characteristic of Scope 1, is tailored to the unique needs of small user groups.

SCOPE 2 (ORGANIZATIONAL SUPPORT)

Tasks, functions, and/or capabilities are designed around supporting major commands and agencies within DoD. Scope 2 places emphasis on meeting needs associated with the specific mission of the organization.

SCOPE 3 (ENTERPRISE SUPPORT)

Tasks, functions, and/or capabilities are designed around supporting the entire DoD community or a military service within the DoD. Scope 3 places emphasis on bringing the community into using a single conformance standard.

SCOPE 4 (SOCIETAL SUPPORT)

Tasks, functions, and/or capabilities are designed around supporting activities and awareness among all societal stakeholders in a DoD mission. Stakeholders can be other government agencies, state and local government organizations, international organizations, corporate entities, foreign governments, and/or U.S. as well as foreign citizens with a need to respond.

Dimension 3: Mapping the Delivery Sources for the Concept

After the solution concept has been mapped into the matrix of levels and scopes, each box in the matrix can then have a unique source for providing service to the user. As with scope, the source for elements at each level can be from one to all four of the following access methods. Further, the nature of access at higher levels may be different than access at lower levels. Software modules and subroutine libraries, for example, can be deployed at specific user sites and from user servers. However, once these modules or subroutines are integrated with other modules, the total resulting application can be accessible from DoD networks. Alternatively, software modules can be acquired as a cloud computing-based service or network-based tool. Then, these modules could be offered to users as a part of local or even site-specific applications. Sometimes, users may not even be aware that their application, with personalized features, may rely upon capabilities that are delivered across the Internet.

SOURCE 1 (EQUIPMENT-SPECIFIC)

IT system element resides within a single hardware associated with a specific user or group of users. For example, software loaded onto user desktops and portal devices.

SOURCE 2 (LOCALLY DISTRIBUTED)

IT system element resides within multiple, interconnected hardware equipment across the user facility (peer-to-peer connectivity) or is accessible by multiple hardware equipment

through connection to central hubs (local area networks). For example, software with multiple user licenses offered through a local Intranet.

SOURCE 3 (NETWORK-BASED)

IT system element hosted at a DoD recognized facility that is accessible across an entire DoD validated network (wide area networks, router networks across the Internet, etc.). For example, DoD developed tools offered to the enterprise through DoD portals.

SOURCE 4 (CLOUD-BASED)

IT system capability accessible as a scalable service across the Internet (Knorr & Grumman, 2008). For example, commercial tools from multiple redundant sites that are available to DoD users in a device- and location-independent manner.

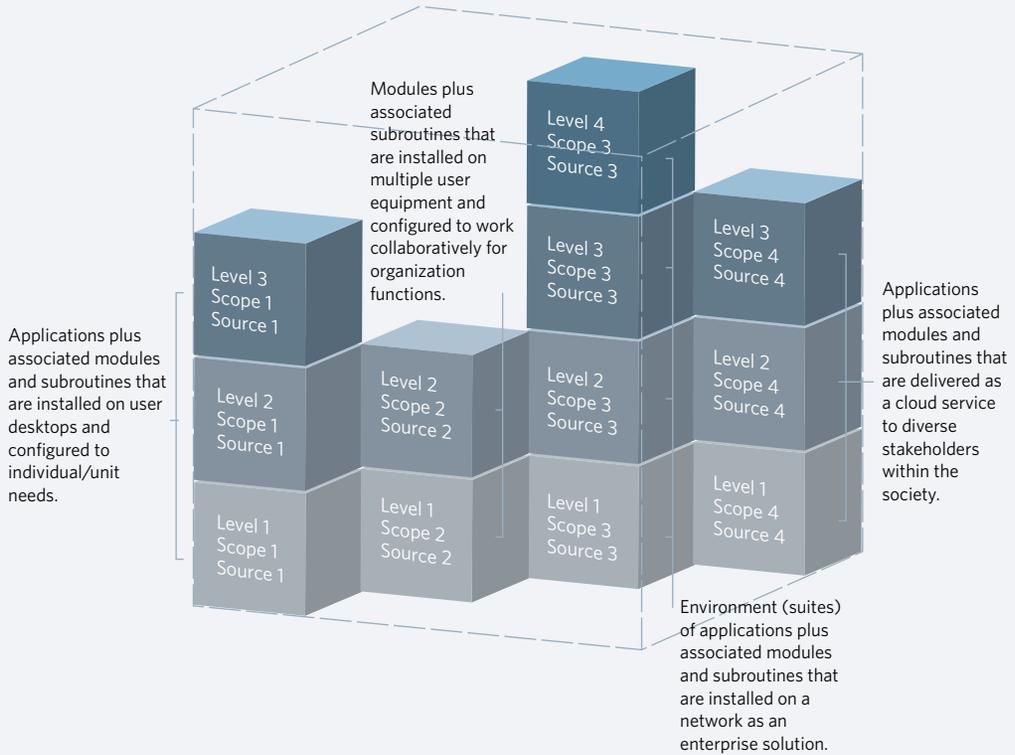
Application of the Reference Frame

The definition of the three-dimensional reference frame (Figure 2) permits a materiel solution concept to be mapped/decomposed into a set of blocks from which a system or multiple system concepts can be formulated. Figure 3 presents a notional configuration of blocks that represents a concept. The goal in describing the activities in each block is not to achieve overwhelming detail but to understand the relationship of blocks to one another. As the relations of activities across the blocks are all achieved through codes, the correction and optimization of relationships through systems formation are feasible and could be critical to acquisition success. Four methods of systems analysis, using this reference frame, can support the organization of acquisition activities.

Method 1: Forming Multiple Systems or Reduced Systems Based on Decomposition of the Mapped Concept

The relationships between blocks from a mapped concept may have natural weak points where forced integration yields acquisition risks. These weak points may merit the breakup of the materiel solution concept into separate systems for acquisition. Weak points could be places where there will be: (a) high risks or low benefits for integration in terms of schedule, cost, and performance, (b) high stress in sustaining integration because of varying pace in technology advancement, and/or (c) extreme challenges in achieving

FIGURE 3. NOTIONAL EXAMPLE OF BLOCK CONFIGURATION MAPPED FROM SOLUTION CONCEPT



integration due to technology scaling and new technology requirements. The breakup can be along any of the three dimensions and can follow a complex path of weak points. Some attempts at creating an integrated software environment may be better pursued as the development of separate applications. Some attempts at trying to provide capability to the society or enterprise may be better pursued by providing separate capabilities to smaller sets of key organizations. In addition, some attempts at delivering services through the Internet may be better pursued as a delivery of more controlled services through secure networks and portals. The objective is an end state characterized by a stable configuration of blocks to create system concepts before the start of the acquisition process.

*Method 2: Forming a Single System or Reduced Set of Systems
Based on Integration of Multiple Mapped Concepts*

The configuration of blocks from a mapped concept may lend itself to being integrated with configurations of blocks associated with other material solution concepts. If so, an opportunity may exist to

develop a single system that can satisfy multiple sets of requirements. Factors to consider when studying the opportunity for integration include: (a) the ability to consolidate the integrated configuration to reduce schedule and cost, (b) the ability to enhance performance through cross-leveraging of activities, (c) the challenges in achieving integration, and (d) the risks in integration. The simplest way to integrate is to identify all the overlapping block definitions along the three dimensions and create the hybrid configuration based on the overlaps. Alternatively, one can study the definitions of the blocks on all sides to see whether a redefined set of blocks can satisfy all the configurations to be integrated.

Method 3: Establishing Co-Development Activities Between Multiple Mapped Concepts

Some blocks in a mapped concept may exhibit common characteristics with blocks in other materiel concepts even though the separate configurations will not be integrated. Such commonalities suggest that co-development or technology sharing opportunities are still available. Given the complexity of software structures, commonalities at the lower levels, scopes, and sources may not always be obvious when comparing overarching concepts. Therefore, comparing blocks along each of the three dimensions is important. A commonality of software elements at any level clearly encourages coordination between systems acquisitions. However, even a commonality of users at any scope suggests a coupling of funding priorities. Also, a commonality of delivery mechanisms from any source suggests a coupling of infrastructure and supporting hardware.

Method 4: Forming New Systems Based on Reorganization or Redefinition of Activities for Mapped Concepts

The most optimal sequencing and timing of developmental activities for the blocks may not be that suggested by the initially mapped concept. Optimal timing may require grouping the blocks into evolutionary increments or iterations within one major release. Optimal sequencing may require some blocks to be redefined and others to be eliminated to facilitate the efficiency of acquisition.

The ability to organize IT acquisition activities into all manner of partial product releases under the spiral development concept is a two-edged sword. When done effectively, the timing of partial releases can take advantage of technology maturation stages, availability of supporting commercial products, user feedback for product refinement, and early mission support opportunities. However, identifying the right pieces for release, timing of release, and extent of release can be difficult. Too many releases may eat away the time

periods for refining products. Too few releases may lead to greater adjustment of developmental activities. Further, poorly timed releases may result in less beneficial user feedback, missed opportunity of mission impact, and wasting of test and certification resources.

Understanding the natural breakpoints in a system concept could reduce the level of error in sequencing acquisition activities and determine conditions for partial product release. The proposed reference frame could aid in this understanding throughout the acquisition process.

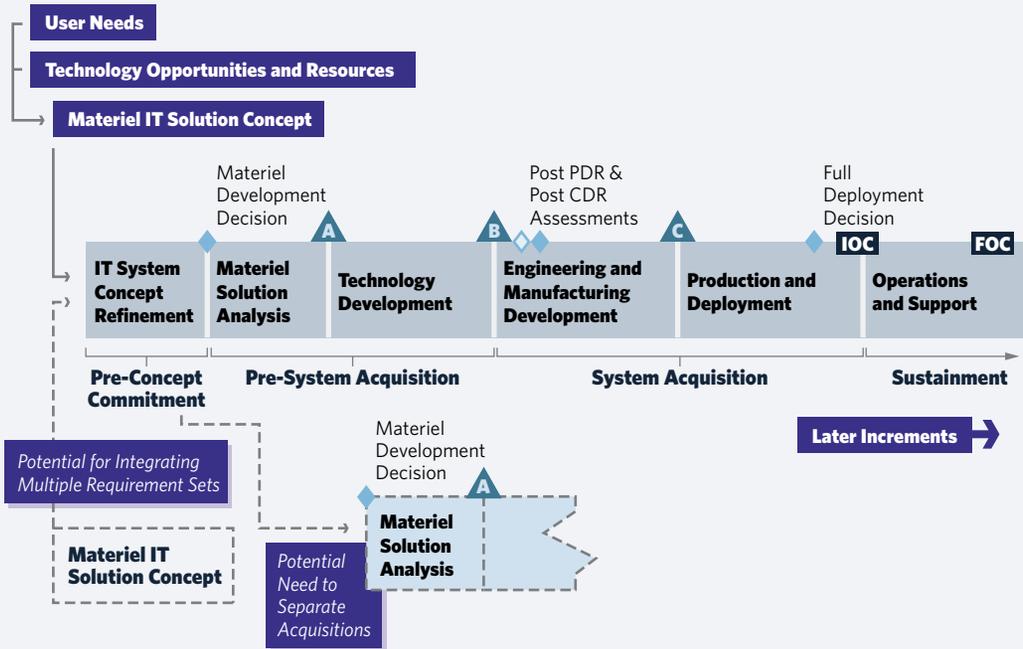
Conclusions

The systems acquisition process in DoDI 5000.02 and the tailored IT systems acquisition process proposed by the DSB both have a hard starting point in the form of the MDD. Our research suggests that the IT systems acquisition process can benefit from a soft start period where the acquisition community can negotiate with the user community regarding system concepts that can satisfy the materiel solution concept resulting from JCIDS. This period also allows the proposed acquisition to be considered within the context of all acquisitions, paralleling the user community effort to ensure that the materiel solution concepts fit within the total concept of joint warfighting. An MDD Review can then be held after the system concept has been appropriately established.

The risk in committing to materiel development without an accurately defined system concept is that the acquisition process can quickly advance to focusing on the user materiel solution concept as the default path to a system concept. Although the Analysis of Alternatives process could, and maybe should, question the materiel solution concept, the analyses in many cases have been directed toward comparing different acquisition approaches instead of questioning what is to be acquired. Even with the alternatives of “maintaining the status quo capability” or “do nothing about achieving capability,” the requirements and associated materiel concept are still typically used as the measurement baseline. If the concept of what is to be acquired has not been optimized for efficient system acquisition by the MDD point, the resulting challenges could continue uncorrected throughout the acquisition life cycle.

The drive to satisfy user requirements may push the acquisition community along paths of overwhelming integration, overly ambitious expectations for use, and overestimation of commercial capabilities. When the complexity of acquisition activities exceeds the government’s ability to understand, some program offices may

FIGURE 4. UTILITY OF THE SOFT START PHASE FOR IT SYSTEMS ACQUISITION



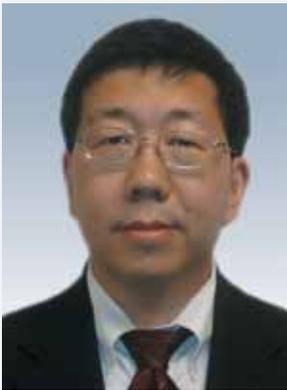
attempt to acquire the capability as a service, anticipating that commercial entities will assume the risks of development. The shifting of risks through a Service Level Agreement (SLA) does not eliminate the risks. If commercial entities cannot master the nature of government system needs, then all an SLA can do is to reduce the government's financial risks through nonpayment for an undelivered service. However, the lack of service capability to support missions will remain a problem.

Soft starts in the acquisition process are already occurring to reduce technology risks. In many cases, technology projects such as Joint Capability Technology Demonstrators are used to accelerate program execution to capture technology opportunities or delay program initiation to manage technology risks (Department of Defense, 2009b). The IT soft start period proposed through this research (Figure 4) will require far less resources than technology projects. Further, the resources should come from a general pool of existing funds to: (a) permit early acquisition community involvement, (b) allow integrated examination of concepts across multiple materiel solution needs, and (c) delay the first obligation of funds for a specific system acquisition until after MDD. Through the soft start, the risks in

achieving a full-deployment decision 5 years after the first obligation of funds, as measured by Congress, is dramatically reduced.

In DoDI 5000.02 (DoD, 2008), the Concept Refinement Phase was redefined and renamed the Materiel Solution Analysis Phase. This change may more accurately reflect the activities of the acquisition community after MDD. However, the notion of concept refinement may still be valid. Our proposed soft start period can be named the IT System Concept Refinement Phase, which leads to a development decision and the analysis of the adopted concept. Once IT acquisition activities have been more accurately initiated, perhaps the debate regarding which tailored acquisition process is best suited for a specific system acquisition will not be as critical because all processes can be more easily refined.

Author Biographies



Mr. Chiang H. Ren is the chief technical officer of Kepler Research, Inc., and is a senior consultant to the Defense Information Systems Agency. Mr. Ren is an associate fellow of the American Institute of Aeronautics and Astronautics and has published numerous journal articles on systems analysis. He holds an MS in Aeronautics and Astronautics from MIT and a BSE in Mechanical Engineering from the University of Pennsylvania.

(E-mail: ChiangRen@KeplerResearch.com)



Col Stephen Busch, USAF (Ret.), is manager for the acquisition business area at Kepler Research, Inc., and is a senior consultant to the Defense Information Systems Agency. Col Busch retired as an Air Force colonel and is Defense Acquisition Workforce Improvement Act (DAWIA) Level III-certified in Contracting. He holds a Juris Doctorate/LLB from the University of Memphis as well as a BA in Business Administration from the University of Memphis.

(E-mail: Steve.Busch@KeplerResearch.com)



Mr. Matthew Prebble is an operations research analyst at Kepler Research, Inc. His consulting experience includes assignments to the Office of the Secretary of Defense and Air Force Headquarters. Mr. Prebble holds an ME in Mechanical and Aerospace Engineering from the University of Virginia and an MS in Operations Research from George Mason University.

(E-mail: Matt.Prebble@KeplerResearch.com)

REFERENCES

- Bell, M. (2008) *Service-oriented modeling: Service analysis, design, and architecture*. Hoboken, NJ: Wiley & Sons.
- Department of Defense. (2008). *Operation of the defense acquisition system*. Department of Defense Instruction 5000.02. Washington, DC: Office of the Under Secretary of Defense for Acquisition, Technology, and Logistics.
- Department of Defense. (2009a). *Report of the Defense Science Board task force on Department of Defense policies and procedures for the acquisition of information technology*. Washington, DC: Office of the Under Secretary of Defense for Acquisition, Technology, and Logistics.
- Department of Defense. (2009b). *Joint capability technology demonstration practical operating guide*, Ver. 2.0. Washington, DC: Office of the Deputy Under Secretary of Defense for Advanced Systems and Concepts.
- Fischer, G. (2001) The software technology of the 21st century: From software reuse to collaborative software design. *Proceedings of ISFST2001: International Symposium on Future Software Technology*, November 5-8, 2001, Zheng Zhou, China.
- Joint Chiefs of Staff. (2007). *Operation of the joint capabilities integration and development system*. Chairman of the Joint Chiefs of Staff Instruction 3170.01F. Washington, DC: Author.
- Knorr, E., & Grumman, G. (2008). What cloud computing really means. *Cloud Computing*. Retrieved from *InfoWorld* at <http://www.infoworld.com/d/cloud-computing/what-cloud-computing-really-means-031>
- National Defense Authorization Act for Fiscal Year 2007, 10 U.S.C., Pub. L. 109-364 § 816 (2006).
- National Defense Authorization Act for Fiscal Year 2010, 10 U.S.C., Pub. L. 111-84 § 804 (2009).
- Szyperski, C. (2002) *Component software: Beyond object-oriented programming* (2nd ed.). Boston, MA: Addison-Wesley Professional.
- Wisnosky, D. E., & Vogel, J. (2004). *DoDAF Wizdom: A practical guide to planning, managing and executing projects to build enterprise architectures using the Department of Defense architecture framework*. Naperville, IL: Wizdom Systems.