THE SYSTEM ARCHITECT ROLE IN ACQUISITION PROGRAM INTEGRATED PRODUCT TEAMS

Ronald R. Luman and Professor Richard S. Scotti

The appointment of overall system architects for Department of Defense (DoD) acquisition programs, preferably as leaders of Integrated Product Teams, would ensure design flexibility, provide for rapid insertion of advanced technology, enhance system functionality, and make more effective tradeoffs between cost and performance. It is especially critical in developing weapons or other systems that must be integrated into an existing system of systems to achieve an enhanced synergistic effect. This approach also facilitates development processes for current and future user needs, consideration of a full range of design alternatives, and testing throughout the full operational envelope.

America's military–industrial complex evolved over five decades to support the defense needs of a nation engaged in a Cold War. With the end of that war, government and industry have been forced to reorient their strategies, priorities, overall industrial base, and weapon systems to meet the military requirements of a "Hot Peace." Complicating their efforts, the American public remains wary of distant low-intensity conflicts and exhibits little tolerance for American casualties. Also, Congress increasingly sees the Pentagon as an obstacle astride its path to a balanced budget. These concerns demonstrate the need to develop and field advanced technologies to increase America's warfighting effectiveness and ultimately minimize the number of American combat casualties.

Indeed, the program manager faces new and complex challenges for systems acquisition: to accelerate the development cycle, deliver affordable systems, and minimize risks by integrating new technology when it arises. Not surprisingly, the challenges of *faster, cheaper, and better!* do not always nicely dovetail. Integration of existing, readily available components into "new" systems is being encouraged by industry, but at a price. Many defense firms are now shortchanging long-term technology research to invest in prototypes or system components they hope

will meet some immediate military requirement. A barrage of information about commercial-off-the-shelf (COTS) components or nondevelopment items (NDI) bombards the decision maker every day. Program managers, sorting through these "solutions" developed with private sector dollars, attempt to formulate an optimal combination that will meet their needs in the most cost-effective manner.

Acquisition reform is under way to meet these challenges while leveraging industry investment. Government attempts at accelerating the usual acquisition cycle include such innovative and complementary measures as the Advanced Technology Demonstrations (ATDs) and the Advanced Capability Technology Demonstrations (ACTDs); often described, respectively, as "technology pushes" and "military need pulls" (Lynn, 1994). Although these initiatives promote the quick fielding of new, militarily useful technologies, they also operate outside of the normal acquisition process and have not yet addressed the issue of effectively transitioning these advanced technologies into either category of large, on-going weapons procurement programs or the many existing, complex "systems of systems" that support entire warfare areas (Eisner, 1991, 1993).

Compounding this, the private sector's investment in staff, facilities, and technology for research is increasingly restricted to the perceived niche markets of each firm. Today's potential system developers are more apt than not to identify problems that can be accommodated by those solutions that are already on their shelves or technologies integral to their own Independent Research and Development (IRAD) investments when peering through this lens. Although well-intentioned, the resulting products are usually marketed as advanced technology insertion to complex systems of systems—without due regard for the appropriate system of systems architecture.

A program manager can cope or even thrive in this new environment by using the system architect to achieve a level of design flexibility that will allow for the rapid insertion of advanced technology, and also enhance system functionality and performance. The system architect is independent of the developer or contractor, and is well-positioned to monitor the marginal utility of the new or upgraded system as it relates to the effectiveness of any larger "system of systems." Other benefits of this approach include the facilitation of (a) development processes for both current and future user needs, (b) consideration of a full range of design alternatives, and (c) representative testing throughout the full operational envelope to ensure low risk.

A system architect naturally complements the system developer, much like the commercial architect complements a builder. The system architect concept also fits naturally into the Integrated Product Team approach directed by the Under Secretary of Defense (Acquisition and Technology) as a key tenet of acquisition re-

Ronald Luman is with the Applied Physics Lab of Johns Hopkins University and **Professor Richard Scotti** is with The George Washington University.

form (Under Secretary of Defense, 1995; Secretary of Defense, 1995). Further implementing guidance regarding IPT structures (USD[AT] and ASD[C^3I], 1995) suggests that the "Integrating IPT" chairman functions as the system architect. To understand the role of the system architect and its potential for accelerating the acquisition cycle and advanced technology insertion, it is necessary to review the current DoD system engineering process.

Review of the Systems Engineering Process

The discipline of systems engineering is an integral element of DoD acquisition. Although the DoD initiative to adopt the best commercial practices has resulted in cancellation of the old MIL-STD-499 (1974) on systems engineering, its intended successor (MIL-STD-499B, 1994) has been converted to a commercial standard, EIA/IS-632 (1994). Managers of major government acquisition programs are required to take the five-month Defense Systems Management College (DSMC) systems engineering management curriculum at Fort Belvoir, VA (Department of Defense, 1991 and 1995). Systems engineering will remain as the foundation of acquisition and development, lending standardized quality to the process, and providing

> ... a comprehensive, structured, and disciplined approach for all life-cycle phases, including new system product and process developments, upgrades, modifications, and engineering efforts conducted to resolve problems in fielded systems. (EIA/IS-632, 1994, p. 1)

The basic systems engineering process contains four activities, applied iteratively as illustrated in Figure 1 (reproduced from MIL-STD-499B, 1994, nearly identical to that in EIA/IS-632): requirements analysis, functional analysis/allocation, synthesis, and systems analysis and control. This process is generally executed by agreement between two parties: the tasking activity as the organization requiring the technical effort (i.e., program manager), and the performing activity as the organization doing the technical effort (e.g., system developer or prime contractor).

Systems Engineering Process Input

Generating and assembling the information necessary to effectively develop a system is an iterative process. In theory the tasking activity provides the performing activity with all the information relevant to its needs, objectives, requirements, measures of effectiveness (MOEs), operating environment, constraints, etc. It then directs the performing activity to consolidate this information for the government's review and approval during the requirements analysis phase. Obviously, the systems engineering process requires sufficient detail for a system developer to generate a realistic proposal. However, it is no exaggeration to say that the government's initial statement of mission need, for example, can consist of a one-sentence statement. Hence the generation of a comprehensive set of the necessary inputs is an iterative process involving both the tasking activity and the potential developers themselves.

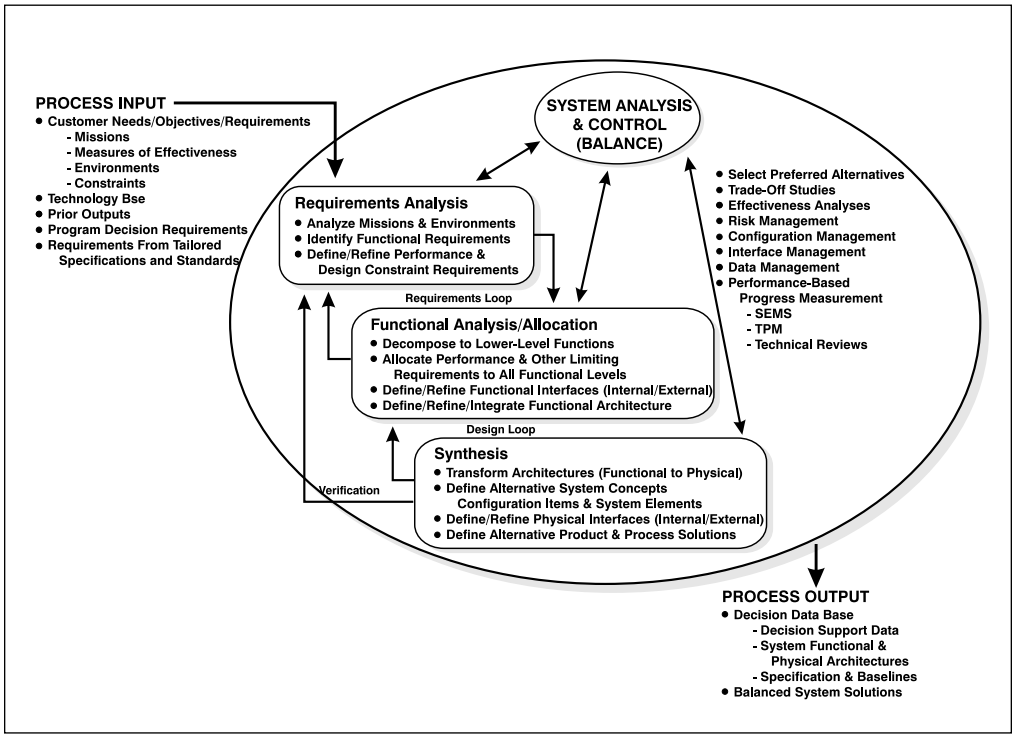

Figure 1. The Systems Engineering Process

The system architect should be a key player in this first phase of system development as part of the tasking activity, offering both systems engineering and program management perspectives. Discipline is required to avoid a natural tendency to solve the problem before it is fully formulated (i.e., premature movement to particular solutions must be resisted in the interest of solving the correct problem).

REQUIREMENTS ANALYSIS

The primary outputs of this phase are the overall functional architecture and associated performance requirements that have been built on MOEs provided or approved by the sponsor. As noted above, MOEs are often not known by the sponsor at the level of detail necessary to determine even overall requirements, and must be developed with full consideration given to mission need, operating environment, achievable technology, and sponsor objectives and constraints. This effort requires expertise in a range of disciplines, from concepts of operation through state-of-the-art technology, and, increasingly, a knowledge of the performance and robustness of available commercial systems as well. Requirements analysis is conducted iteratively with Functional Analysis/Allocation to ensure that the system's objectives are achieved within the limits of available technology and resources.

Moreover, system performance require-

ments "shall consider the full life cycle envisioned and must be characterized in terms of degree of certainty in their estimate, the degree of criticality to system success, and their relationship to other requirements, in order to facilitate prioritization of requirements during trade studies and/or final evaluation of alternatives and selection of the system design" (DoD, 1991). Considerable controversy exists as to the degree of flexibility with which "requirements" are to be treated, with design-to-cost approaches requiring maximum possible flexibility.

Functional Analysis and Allocation

In this phase, the system's functional architecture is developed in detail sufficient to support a synthesis of alternatives. The overall functional architecture is analyzed and logically sequenced, with inputs, outputs, and interface requirements clearly defined. Levels of performance are either assigned or derived for each functional requirement and interface so that overall performance requirements may be traced throughout the functional architecture. This division and allocation is continued until the resulting set of requirements is defined in quantifiable technical performance measures (TPMs) or go/no-go criteria, as appropriate, and in sufficient detail to be used as design criteria.

Functional analysis and allocation generally does not have one "right" solution, and "optimal" is hard to define, let alone achieve. Hence this phase requires the exercise of judgment when initially allocating performance requirements to functional elements. Moreover, flexibility must be maintained to allow the different approaches arising in this phase of development to be considered and costed out by each subsystem activity during the subsequent synthesis phase. For example, the allocation of an overall budget of accuracy errors for a strategic missile system across such subsystem activities as initial conditions, in-flight guidance, re-entry body deployment, geodesy, and re-entry flight dynamics will require an understanding of the disparate technologies and costs associated with maximizing subsystem performance. Successes and failures at innovation must also be accommodated through an iterative process that can reach all the way back to requirements analysis. Of course, it gets harder to adjust performance (especially in functional allocations) as the development cycle progresses. Not only is it more costly to accommodate design changes, but the system developer grows increasingly reluctant to consider changes affecting its own costs (and profits), even when it may be clear that the system would better meet mission objectives were changes made.

Synthesis

Synthesis is that phase of development in which complete alternative system designs are generated in an iterative fashion with functional analysis and allocation. Synthesized designs will describe the entire system, including the interfaces between internal subsystems or components and the external environment. The system designer must verify that alternatives will satisfy functional and performance requirements and that they are attainable within estimated risk levels.

As previously discussed, there is cur-

rently pressure to find ways of using COTS components or technology. A COTS solution may appear to be so simple that a sponsor will wish to interject it into the process as his own system alternative. Unfortunately, an often-overlooked and certainly unappreciated risk factor is the cost required to integrate various COTS components from different vendors into a cohesive system that meets requirements. Especially misunderstood are unpredictable software costs required to achieve effective interfaces between hardware, software, and human operators, as well as to produce displays and features customized to the needs of the operational user.

Systems Analysis and Control

Systems analysis and control is an overarching activity that operates concurrently with the iterative processes spanned by requirements analysis, functional analysis and allocation, and synthesis (Figure 1). It covers a variety of analyses: tradeoff studies, effectiveness assessments, and system or subsystem design analyses and simulations to estimate progress in achievement of TPMs and overall requirements. It also employs several control mechanisms: risk management, configuration management, data management, and various technical reviews.

The analyses are conducted by the performing activity. The control activities are generally joint endeavors involving the tasking as well as the performing activity (MIL-STD-499B, 1994; EIA/IS-632, 1994). System analysis is considered more generally in a later section.

Systems Engineering Process Output

The systems engineering process should produce balanced, feasible system alternatives or solutions and a decision database that includes decision support data, system architectures, and specifications and design baselines from which the key decisions made during the process can be reconstructed and justified (MIL-STD-499B, 1994; EIA/IS-632, 1994). A framework and procedure for evaluation should also be established from which the final system design can be selected by the sponsor (Eisner, 1988).

What's Missing?

Is the systems engineering approach which we have just reviewed still viable in the current acquisition reform environment? Certainly it is central to the current acquisition process, which is universally regarded as needing reform and acceleration. Today, the acquisition program manager's challenge is to accelerate the development cycle, reduce costs, and maintain the capability to insert the most advanced technology appropriate for the stated need. Is the system engineering process part of the problem?

The key to meeting this challenge is found in the system analysis and control process, in which complex, highly visible, and continuing technical evaluations are conducted. These evaluations guide decisions regarding the design, capabilities, or selection of system alternatives. It is here that the systems architect may provide objective judgment and perspective to ensure a successful development process.

Technical evaluations may be grouped by function into four categories:

1. Evaluations that determine basic parameter values. For example, a technical evaluation may be necessary to determine the variation of an inertial navigation system's (INS) heading gyro drift bias as a function of platform azimuth. This evaluation may involve calculations, computer simulations, field tests, and measurements (Pace, 1986).

2. Evaluations that determine system performance. A system performance evaluation may determine the overall INS error growth as a function of time. The resulting tool might be a covariance simulation. Performance analyses are arguably the most visible type of technical evaluation, and can be the primary factor in discriminating between system alternatives (Atallah, 1993).

3. Evaluations that determine operational effectiveness by considering mission-level MOEs as a measure of the system performance. In our example, the ballistic missile system accuracy that is initialized by the INS depends on the accuracy of the INS, which may be dependent upon platform azimuth.

4. Evaluations that address concepts of operation, tactics, or strategy, and consider how the system will be used to satisfy mission objectives. Increasingly, weapon systems must be integrated into a larger, extant system of systems. The difficult analysis that predicts the marginal utility of the new system to the larger system of systems is often overlooked.

These technical evaluations use a variety of modeling and simulation methods. Figure 2 displays the full range of such

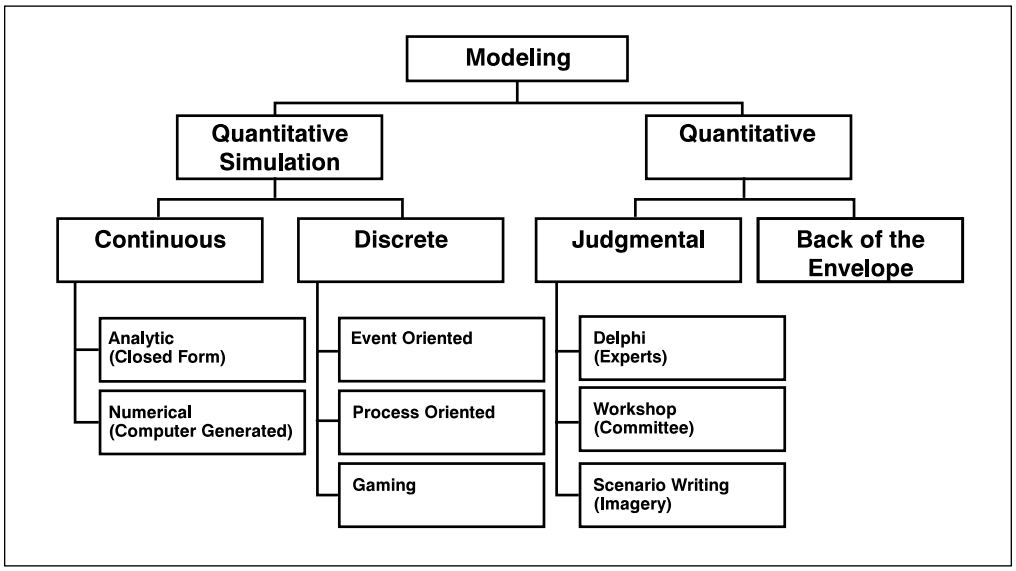

Figure 2. Modeling Taxonomy

techniques available to the systems analyst (Eisner, 1988; Scotti, 1994). However, it is the quantitative modeling methods that are the most commonly used in the systems engineering process. Laboratory experiments, field tests, and operational data from military exercises are important sources of validation data for the modeling and simulation processes.

Effective execution of the system analysis and control activity through judicious technical evaluations brings the following benefits to the systems engineering process:

- generation of system alternatives (including concepts of operation) that may more effectively or efficiently achieve the sponsor's objectives;
- explicit consideration of assumptions, uncertainties, costs, consequences, etc.;
- an objective framework and common basis for evaluating system alternatives and selection of a preferred alternative;
- improved understanding of the issues and hence better understanding on the part of the sponsor and ultimately the system users; and
- improved managerial capabilities for planning and administration of the system life cycle (Miser and Quade, 1985, pp. 25–26).

There are, however, adverse consequences that can arise from dependence on systems analysis beyond the level appropriate to the scale of the problem:

- unwanted delays in system development;
- undesirable centralization and concentration of decision making in top-level staff;
- increased dependence on complex processes (e.g., simulation) that require expensive talent to operate; and
- loss of risk mitigation capability through elimination of apparent inefficiency and redundancy.

A system developer may seek to avoid detailed technical evaluations, citing the potential for schedule impact as a justification. This position may also be motivated by a lack of qualified people to do the evaluations, or concern that an analysis will encourage reconsideration of an alternative already effectively discarded. However, a vibrant, objective, ongoing systems analysis is essential to the maintenance of systems perspective, especially in regard to the "system of systems." The standard systems engineering process does not address this overall architecture question, generally considering the system under development as an isolated entity.

Systems Architecting

A broad role growing out of systems analysis and control in the systems engineering process has recently been characterized as *system architecting* (Rechtin, 1991, 1994). The function of a system architect is to act as the system development agent of the program manager, to create and manage the design, to maintain sys-

tem integrity, and to help achieve user satisfaction with the procured system. Hence it is a role that exists at the level of the acquisition process, yet may also contribute to the engineering of a system.

The discussion of systems analysis presented above has focused on tradeoff studies, quantitative technical evaluations, system integration, and interface management—all within the context of the systems engineering process. However, the role of the system architect goes beyond systems engineering to include the comprehensive synthesis, certification, and qualitative satisfaction of user needs—all of which are goals of the Integrating IPT (USD[AT] and ASD[C^3I], 1995. System architecting applies systems analysis methodology to the acquisition process, rather than operating strictly within the confines of the single systems engineering process, per se.

Another way to understand system architecting is to contrast its tasks with those generally performed as part of systems engineering. Architecting is working *for* the program manager and *with* a system developer; engineering is working *with* an architect and *for* a system developer. In short, the core of architecting is system integration and a continuing verification that the desired product is being obtained. Figure 3 illustrates the scope and potential value added by the architect throughout the life cycle of the system.

The military has long recognized the need for system architects in the acquisition process, though that terminology is not widely used outside of the software engineering specialty. The term technical direction agent (TDA) reflects a role similar to what we have discussed for the system architect:

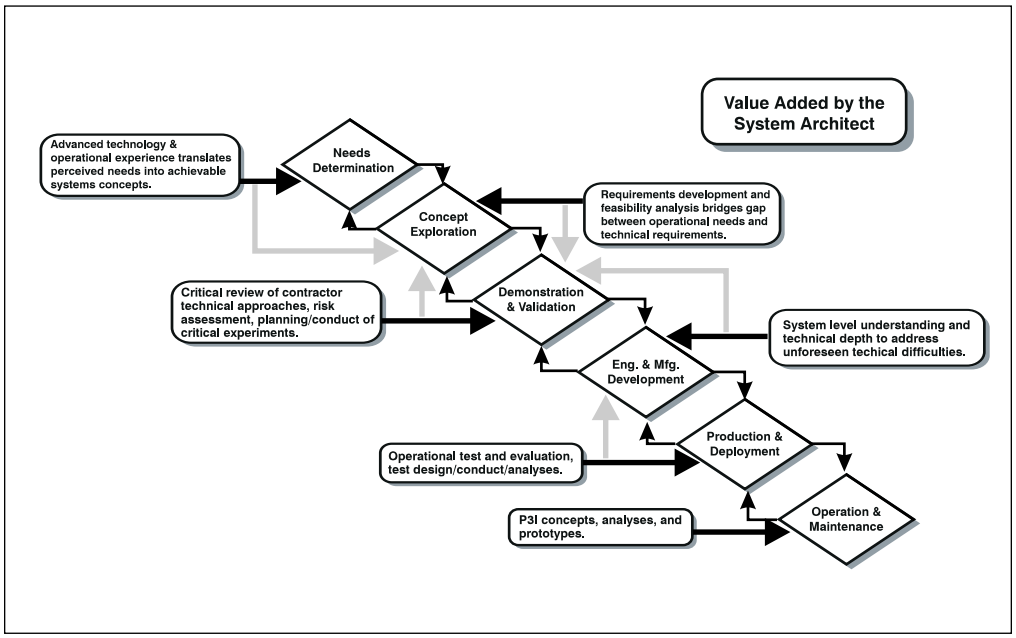

Figure 3. System Development Process

> The TDA assists the level III System Manager in the establishment of initial program concepts, performs system engineering, develops performance specifications, and performs or directs research, development, test and simulations to investigate problems, probe alternative approaches, and to evaluate design agent achievements. (Department of the Navy, 1985)

To summarize the role of the system architect, it will help to define some terms:

- The "system" is what is built.
- The "model" is a description of the system to be built.
- The "system architecture" is the structure of the system.
- The "overall architecture" includes the structure not only of the system, but of its functions, the environment within which it will operate, and the process by which it will be developed and operated (Rechtin, 1991, p. 75).

The systems architect is then concerned with the overall architecture, not just the system architecture, the model, or associated technical evaluations.

The System Architect in the Integrated Product Team

Increasingly in this era of joint and combined operations, systems must interoperate with other systems, forming a "system of systems" that offers a true warfighting capability. Hence many acquisition programs are justified and judged on their marginal utility to the effectiveness of this larger "system of systems." This is the perspective of the system architect. Furthermore, the Integrated Product Team is an excellent, cooperative vehicle that can easily accommodate establishment of the system architect as the integrating IPT chairperson, acting as the design agent of the program manager. Depending on the particular program and the technical knowledge required, the system architect role may be satisfied by one of the following.

1. The program manager or his or her staff.
2. A government laboratory.
3. A university laboratory.
4. A nondevelopment division of a systems contractor.

Recalling our earlier discussion about industry investments and COTS technology, it is critical that DoD sponsors have the best possible perspective and knowledge available when buying in the technological marketplace. This must be provided with a competence, integrity, and objectivity beyond question. It is not reasonable to go to the free enterprise marketplace from which the government will buy its systems and ask for advice on which systems to buy. Conversely, the system architecting organization has no potential for conflict of interest, as it is not a candidate to develop the operational system, though it may well build prototypes as part of the technical evaluation

processes described above. This independence from the business of operational system development and production means that the system architect doesn't come to analysis or architecting with a particular technology or system solution in mind. Indeed, the architect is obligated to table the sponsor's initial agenda or solution (typically, the sponsor has one in mind, stated or not) until the problem is sufficiently understood and structured.

The in-house institutions that perform this role are sometimes referred to as "research and development centers" and take the form of either military or university laboratories. These laboratories, and the government, realize that they must have full knowledge of military operations and the implications of technology, which cannot be gained by mere observation. In short, they train to be system architects, or independent systems analysts operating at the acquisition level, whose role spans the full spectrum from research and development through operational performance evaluation in the field or at sea.

Perhaps the strongest argument for this independent architecting in the DoD acquisition process is that it can reduce cost and improve the product in spite of government contracting procedures. To illustrate: The government frequently punishes a contractor more severely for underrunning than it does for overrunning (Kershner, 1981). This is an almost inevitable consequence of the DoD contracting practice for systems being developed as opposed to those in production. In the former case, the contract is typically of the "cost plus" variety because the costs of development are too uncertain for either the contractor or the government to agree on a fixed price. Nevertheless, the cost and schedule estimates generated at the start of a program will form the basis for significant allocations of resources and staff to which the contractor and the government are then committed. Consider, however, how this may affect an ongoing program in which an innovative application of advanced technology might halve the remaining cost and schedule of development. This innovation may create a conflict of interest for the contractor, who was counting on the initially agreed-on funding and schedule to gainfully employ significant numbers of staff. In this case an independent system architect could be relied on to uncover the new technology application and present it to the system developer and sponsor.

Situations of this magnitude are rare, but a independent system architect, working within the constructive atmosphere of the Integrating IPT, can make significant, consistent contributions toward integration.

SUMMARY

Systems analysis applied to the acquisition process, sometimes described as system architecting, is vital to successful development of complex systems and systems of systems. The architect role is best performed by an agent of the government program manager (and, thus, independent of the system developer) who can be relied on to ensure sponsor satisfaction with the final system. The accelerating pace of technology, the aggressive investment in and marketing of components and systems by the private sector, and the increasing complexity of military needs all mandate an ever more sophisticated government

consumer. The system architect must offer a broad expertise in the state-of-the-art technology, information systems, and a knowledge of mission operational needs, as well as the skill to apply this knowledge in a cost-effective manner.

The acquisition program manager can maintain this vision and focus by appointing a system architect, independent from the system developer or contractor, to chair an essential Integrated Product Team. This is especially critical in the context of a "system of systems" development environment, wherein program success will ultimately be judged on the marginal utility of the new or upgraded system to the entire system of systems' effectiveness.

REFERENCES

Atallah, G. C. (1993). Systems Engineering Performance Analysis Throughout the Systems Acquisition Lifecycle. *Proceedings of the Third Annual International Symposium of the National Council on Systems Engineering* (August).

Department of Defense. (1991). *Acquisition Management Policies and Procedures* (DoD Directive 5000.1 and DoD Instruction 5000.2). Washington, DC.

Department of Defense. (October 1995). *Acquisition Management Policies and Procedures* (Draft DoD Directive 5000.1 and Draft DoD Instruction 5000.2). Washington, DC.

Department of the Navy. (1985). *Delegation of Technical Responsibility and Authority to Engineering Agents* (NAVSEA Instruction 5400.57A). Washington, DC: Author.

EIA/IS-632. (1994). *Systems Engineering*. Engineering Industrial Association Interim Standard.

Eisner, H. (1988). *Computer-Aided Systems Engineering*. Englewood Cliffs, NJ: Prentice Hall.

Eisner, H., Marciniak, J., & McMillan, R. (1991). Computer-Aided System of Systems (S2) Engineering. *Proceedings of the 1991 IEEE International Conference on Systems, Man, and Cybernetics*. Charlottesville, VA.

Eisner, H., McMillan, R., Marciniak, J., & Pragluski, W. (1993). RCASSE: Rapid Computer-Aided System of Systems (S2) Engineering. *Proceedings of the National Council on Systems Engineering* (26–28 July). Washington, DC.

Kershner, R. B. (1981). Where Have All the Underruns Gone? *Johns Hopkins APL Technical Digest*, *2* (4).

Lynn, L. (1994). The Role of Demonstration Approaches in Acquisition Reform. *Acquisition Review Quarterly*, *1* (2).

MIL-STD-499A. (1974). *Systems Engineering*.

MIL-STD-499B. (1994). *Systems Engineering* (Draft). Washington, DC: Joint OSD/Services/Industry Working Group.

Miser, H. J. & Quade, E. S. (Eds.). (1985). *Handbook of Systems Analysis*. New York: North Holland.

Pace, D. K. (1986). Use of Scenarios in Technical Evaluations. *Johns Hopkins APL Technical Digest*, *7* (1).

Rechtin, E. (1994). The Systems Architect: Specialty, Role and Responsibility. *Proceedings of the Fourth Annual International Symposium of the National Council on Systems Engineering.* San Diego, CA.

Rechtin, E. (1991). *Systems Archi-tecting: Creating and Building Complex Systems*. Englewood Cliffs, NJ: Prentice Hall.

Scotti, R. (1994). *Systems Analysis Class Notes*. Unpublished material, George Washington University, Washington, DC.

Secretary of Defense, Perry, W. J. (May 1995). *Use of Integrated Product and Process Development and Integrated Product Teams in DoD Acquisition*. Washington, DC: Office of the Secretary of Defense.

Under Secretary of Defense (Acquisition & Technology) Kaminski, P.G. (1995). *Reengineering the Oversight and Review Process.* Washington, DC: Office of the Under Secretary of Defense (Acquisition and Technology).

Under Secretary of Defense (Acquisition & Technology) and Assistant Secretary of Defense for C^3I. (November 1995). *Rules of the Road: A Guide for Leading Successful Integrated Product Teams.* Washington, DC: Office of the Under Secretary of Defense (Acquisition and Technology).