

The Underlying Keys to Acquisition

Needs, Requirements, Prioritization, Asset Allocation

ALEXANDER R. SLATE

This article is actually intended for three different audiences. The first is the everyday working-level program managers, especially those relatively new to the field who have not yet managed to get a feel for how things are working behind the scenes. For this group, know that an understanding of the entire process of how your programs evolve will enable you to become more effective program managers. Immersing yourselves into the processes described in this article so that you can influence the fate of your programs is even better. But as you do so, understand when it is important to defend your program and when other programs should and do take priority. Each of our individual programs should not be an end unto itself.

The second audience is the requirements community. While many of you may know this information, some of you may be new to the area, without any grounding in program management. Hopefully, this article will help you understand what you are doing in the context of acquisition program management and give you an opportunity to try out a few new ideas.

The third audience is the staff groups within both the Department of Defense and Congress. For you, this article is

simply meant to put the acquisition process into context and perhaps provide some philosophical direction as you help to formulate policies.

Is Acquisition Transformation Doomed to Fail?

Right now, the acquisition community, particularly the U.S. Air Force (USAF) community, is in the throes of transformation. Typically, DoD's acquisition of major weapons systems is characterized as too lengthy, too costly, and inefficiently managed. Regrettably, some truth lies in the charges leveled against the system. For purposes of this article, I will not justify that statement—others (*many* others) have and continue to do so—that simply isn't my intent. To mangle Shakespeare: "I come not to praise or bury Caesar, but lend me your ears."

Frankly, we within the acquisition community itself are not going to solve all the problems and transform DoD acquisition from a supertanker that takes miles to execute a turn, into a speedboat able to turn on a dime. But people are trying.

Within the USAF, two strategies aimed at doing a better job of managing the acquisition process are noteworthy:



- Switching to evolutionary acquisition.
- Reviewing the regulatory controls.

Both of these strategies are important and could potentially improve the process and speed up acquisitions. But without reformation of the title processes (generating needs and requirements, determining program priorities, and allocating assets), the results of evolutionary acquisition and any regulatory rewrites may be disappointing at best, and considered a failure at worst.

Slate is an acquisition advisor for the Acquisition Center of Excellence at Brooks City-Base, San Antonio, Texas. His acquisition career experience includes serving as a test manager, program manager, and team lead for a Systems Program Office. He is Level III-certified in the Program Management; Test and Evaluation; and Systems Planning, Research, Development, and Engineering acquisition career fields; and is Level I-certified in Logistics.

First Among Equals

An old cliché has it that “bean counters rule the world.” Cliché it may be, but like many old sayings, it conveys more than a hint of truth. We can try to improve processes, but ultimately the success or failure of any program will hinge on having a clear ultimate goal, an understandable plan to achieve that goal, and the money to translate the plan into action.

Needs and Requirements

Needs and requirements are two halves of the same whole. In reality, our warfighters have only needs. Requirements are simply statements of how we propose to satisfy those needs. They must be stated clearly and be appropriate to the need. Acquisitions, and thereby acquisition plans, exist to satisfy requirements. If requirements aren't clear, no plan is ever going to satisfy them.

of evolutionary acquisition but would work just as well in a single large straight-line acquisition.

Instead of requirements documents, I would like to see the warfighters tell us the operational deficiencies of their current weapons system, what they would like to accomplish, and how they see operations evolving after delivery of the new weapons system that overcomes the identified operational deficiencies.

Let me give you an example of the way that I would like to see requirements specified.

Currently, nuclear, biological, and chemical protective masks have several requirements related to protection. They must protect against certain agents at a certain agent density for 24 hours. The masks can only leak at a certain rate (this is known as fit factor) for chemical agents and at a different rate for biological agents.

They also have to protect against nuclear radiation (generally accepted to mean radioactive fallout).

Several “problems” exist with stating the requirements in this manner. First, the requirements community took a single concept—*protection*—and broke it into specific areas. What they did was to deliver a design solution as opposed to a need. Second, this process encouraged numbers that may not be meaningful for the sake of numbers.

For example, the entire military population could not possibly obtain the same fit factor, so satisfying the requirements



Frankly, we within the acquisition community itself are not going to solve all the problems and transform DoD acquisition from a supertanker that takes miles to execute a turn, into a speedboat able to turn on a dime. But people are trying.

But requirements also need to be appropriate and to some degree flexible. If not, plans will be too elaborate, too labyrinthine, and too expensive. In my article “Evolutionary Acquisition: Breaking the Mold—New Possibilities from a Changed Perspective,” published in *Program Manager* magazine, May-June 2002, I suggested a process whereby the warfighting community (the users) would develop a statement of need and a concept of operations. Then the users, together with the development (or acquisition) and testing communities, would develop the requirements and acquisition strategy. That concept was treated in the context

In other words, having the clear goal is the *needs and requirements*; the plan is the *acquisition* itself; and determining how money is spent involves both *prioritization and asset allocation*.

All of these processes are important, but the most important, in my mind, is asset allocation. Why? Clearly, without money no program exists. Furthermore, even a clear set of goals and a good plan to reach them are of no use if the money needed to accomplish the plan isn't available.

OPTIMIZED LISTS (NON-COST FACTORS)

It must be remembered that dollars are not the only asset that a program needs in order to function. Manpower, test facilities, manufacturing, and warehouse facilities are just some of the non-dollar assets needed to accomplish the programs we are going to fund. The people responsible for the higher levels of decision making don't have enough information about these assets to do more than understand that the assets also play a role in the accomplishment of the mission. But as the money flows down the execution chain closer to the actual working level, these factors become more important.

To complicate matters, to a degree these non-cost factors are cost factors. Remember that manpower and facility upkeep falls under the heading of Operations and Maintenance (O&M). What if we have put too much money into Development and Production without a sufficient amount of money in O&M to fund

the manpower to oversee the programs?

This is not just an academic question: It is one that is a pressing everyday concern to program offices. For almost 20 years, we have been downsizing the personnel lists of the DoD without decreasing the numbers of programs we have been asking the remaining personnel to accomplish. We have downsized past the point of cutting out the fat. The result is one of two things, and often two of two things: first, inefficiencies, and second the use of different types of money to make up the difference. For instance, if there is insufficient program office staff in the military or civilian service, outside contractors will need to be hired to make up the discrepancy—an additional cost. These contractors don't get paid with O&M funding; they get paid with Research and Development or Production funding.

exactly as written was impractical, if not impossible.

To my mind, the following statement of requirements is a far better description. (Numbers used here are examples only and do not reflect real needs.)

“The mask system must protect the military population under the following conditions. The expected period of threat is four days. Vesicant and nerve (G and V) agents will be present in liquid (at 5 grams per square meter), vapor (density of 20 milligrams per cubic meter of air), and aerosol (20 milligrams per cubic meter of air of agent with mass median particle diameters of 5 micrograms) forms. No greater than 10 percent of the population may exhibit any symptoms related to chemical or biological agents or radiation fallout poisoning, and no more than 3 percent may exhibit inca-

pacitating symptoms. Personnel are expected to work an average of 12 hours a day in the contaminated environment. During the four-day protection period, the warfighters will be exposed to liquid agents a maximum of twice and to vapor or aerosols to 20,000 milligram-minutes per meter cubed.”

Expressing the requirements this way lays out everything the developers need to know. It gives the developers and contractors the ability to balance the different aspects of protection in a way that makes the best design sense.

But the warfighters aren't just laying out needs. They also are writing an initial perceived concept of operations. The initial concept of operations should consist of actual attainable systems, as well as pie in the sky and “if I really had my way” in equal measure. The concept of

operations will need to remain somewhat flexible and allow for capabilities upgrade as the system develops and matures. At this point the requirements document may be generated, although a case might be made for going directly to an initial systems specification.

Prioritization and the Occasional Failure of Reason

To a large measure, prioritization and asset allocation go hand in glove, although no absolute correlation between the two exists. What this means is that funding may not necessarily flow in line with prioritization, though there should be a very strong relationship. (This is a concept more fully explained in subsequent paragraphs).

To begin, we must ask ourselves how we are going to handle prioritization. Now this question itself may have many meanings, but we must first decide whether we want to lump all the efforts in a great big pile and prioritize them en masse, or divide them into discrete groups of programs and prioritize each group separately. For several reasons, I think the answer is the latter. First, and certainly one of the most important reasons, is that it really would be too difficult to deal with everything at once. Second, we have money that has different budgeted uses (generally we call this the “color” of money). Because of that, the different colors of money actually have different users, so the programs for each type of money should be prioritized separately.

Prioritization is the area discussed in this article where the lack of a scientific process causes the least amount of trouble. As in combat—where the right thing to do is intuitive in nature, the result of many different fluid and shifting factors—some measure of a gut feel for what the priorities should be, may be exactly what is needed. Attempting to analyze the situation by its separate parts simply takes too long or actually leads us to the wrong answers.

This is disturbing to many of us who take an engineering view of the world. Let's look at a couple of different ways

of determining priorities, their strengths and weaknesses.

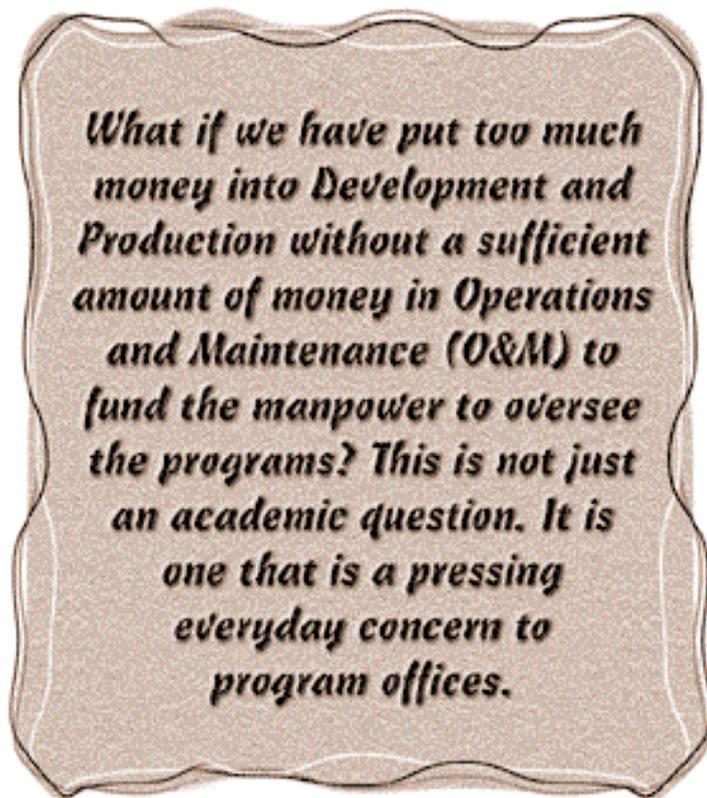
First is the scientific/engineering analysis method. Most processes can be broken down into little pieces, each of which can be measured and quantified, then put back together to make a coherent whole. As a result, we have a number of tools that we use as decision aids. In the engineering and mathematical world, these come under the grandiose heading of “multi-criterion optimization decision support systems.”

The way these tools work (in one way or another) is by taking what we wish to analyze (in this case a group of programs that we wish to prioritize) and determining what the factors are that make up the whole picture we wish to examine. These factors may be such things as user utility, technological maturity, cost, and level of automation. Each of these factors is a certain percentage of the whole puzzle, so each is assigned a weight.

A grading scale is developed for each factor (generally on a zero to one scale). An example might be that for technological maturity, the availability of a commercial solution would be a 1; a non-developmental solution, 0.9; a developmental effort requiring less than two years, 0.7; an effort requiring two to five years, 0.5; five to seven years, 0.3; and any greater time-scale, 0.1. Each program is graded for each factor. We can then multiply the grades by the weights and add up the scores for each program. The highest score is the No. 1 priority, and so on down the line.

Sounds great, right? It's easy, and with seemingly little room for error. Three elements of subjectivity, however, are still apparent that can lead us to the wrong conclusions.

- First is being able to assign the correct weights and grading curves for the factors.
- Second is the assessment of each program's grades for the factors, whether



it is based on reality or merely wishing it were so.

- Third is the simple fact that this scheme attempts to force a single, simplistic “view” of a potentially complex situation.

As we look across the programs, the weights of the factors won't always be constants. One system may have a very high user utility but only if a second system exists; without it, the first system actually has a low utility. The second system by itself has only a mediocre user utility. As we rack and stack the programs (assuming user utility has an effect on our final decisions), the first system ends up with a high priority and the second has a low priority. So in this scenario where the funding lines are drawn later, the first system is funded and the second system is not. The result is that we develop a system on its own that has little utility by itself.

But this isn't the only way we can determine priorities. We can bring the users together and have each one simply rank each program. The programs that are ranked the highest priority by

most users would receive the highest priority. But what if all the users except one represent very small user populations, and the single user alone represents 70 percent of the users? That single user representative may have a very different set of priorities from the those of the others. But being a single voice, that single user's priorities would keep getting bumped down the list.

Well, I've painted a pretty grim picture of prioritization. There seems to be no way to do this properly. But the trick is to adapt. Use some of that gut feel in the process. Use whichever method seems to make the most sense, but understand that each of these systems is going to come up with something that isn't totally correct. Get your priority lists from these methods, and then adjust the list so it does make sense.

Asset Allocation—in Particular Doling out Dollars

In one sense, asset allocation is the ultimate prioritization, but in the final analysis, it isn't. Why? Because where we put the money is our highest priority—and that isn't always necessarily

just asset allocation. But we have to start asset allocation with those prioritization lists. And digressing a little bit, let me say that although a lot is wrong with our budgeting system and its colors of money, the colors of money at least prevent us from a monolithic system of spending. What do I mean by that?

As we examine our *needs* (and I use the word here just a little differently from its previous usage in this article), we have some basic functions. We must pay for the upkeep and operations of the military machine we already have functioning (Operations and Maintenance [O&M] dollars). We must maintain force levels or bring force levels up to determined levels of strength (production and some O&M dollars). We must look to the future and develop improved systems (Research and Development [R&D]). And we must put some attention into the near term, mid term, and long term.

We must be able to assess (also known as *test*) the capabilities that we have or the ones we are building (R&D, production, and O&M dollars, depending on what we are testing). We need to support the people who actually make the system run (O&M and Military Construction [MILCON] dollars). Finally (and sometimes forgotten), as we plan for expanded capability, we need the ability to assess the capabilities we desire for the future (R&D and MILCON dollars).

If it weren't for the different colors of money, the greatest tendency would be to sacrifice the future for the sake of the present—or sometimes (though this is less likely), sacrifice too much now for the sake of the future. The current system of assigning colors of money prevents us from doing either of those too easily. There may be reasons to do one or the other, but we really have to desire the outcome a great deal to put up with the paperwork necessary to make it happen.

So the first trick of allocating the money is to determine in which general cate-

gory the money belongs: R&D, production, O&M, or MILCON. But our issues don't end there.

If we put money into R&D, we must determine how much we allocate toward the particular areas. How much do we devote to near-term development such as systems integration, systems demonstration, and in some cases low rate initial production? And of that amount, how much do we use to extend the capabilities of existing equipment and how much toward developing whole new systems?

Next we must decide how much to spend on long-term development such as concept development or advanced concept demonstration. And finally, we must decide how much to expend on the technical base (research and test development), and how much of that to advocate for more practical research and how much for pure research. Still missing, however, is *test*. Test is actually wrapped up in these areas as we fund particular programs (so much of each program's budget is devoted to test).

Production also has categories, but these are easier in that we fund each of them depending upon the priority lists and the programs that are actually funded. MILCON is almost as easy, the only decision being a basic determination (made during prioritization) of how much to devote to morale and member support and how much to devote to business construction.

O&M is almost as troublesome, if not more so, than R&D. Certainly the scope of O&M is broader. Included in O&M are salaries, everyday operating expenses (and even this covers a very wide range of actions and items), equipment maintenance, and even the procurement of already developed and fully fielded equipment to cover shortages. In fact, based upon what I know just from my own experience working in acquisition, I'd say that O&M is the most consistently underfunded area in all of the DoD. I can even remember times when it was questionable if we would be able to cover everyone's salaries.

Now that we have a basic understanding of the big picture, we have to ask a very basic question regarding the rules by which we shall allocate money: Are we going to fund strictly in order of priority, or shall we attempt to "optimize" the list? And if the latter, do we mean getting the "best" combination of programs or funding as many programs as possible? It is critical to understand that going strictly by the priority listing doesn't necessarily provide the best bang for the buck. This is due, at least in part, to the fact that the needs, requirements, and prioritization process are somewhat disjointed (or as my boss prefers, "are not optimally connected"). But also, please understand that none of these three "systems" is necessarily bad or wrong.

Strict Priority

This is a very straightforward approach, at least to start. We have a prioritized list of programs, and actually we may have several different lists. We have a pot of money. We start with the program listed as the No. 1 priority and fund it. We proceed down the line for as long as we have money to fund programs. Let us say that we have 10 programs, and sufficient funding only for the first six. The question now becomes, what we should do with the money remaining after we have funded those first six programs.

Do we partially fund the priority No. 7 program? Can the program be split, stretched out, or descoped so it can be partially funded? If we do any of the latter, does the program retain the same priority rating? Do we skip No. 7 (and potentially other programs) if there is sufficient money left over to fund one or more of programs 8 through 10?

Optimized Lists (Greatest Number)

Let's take our example of a list of 10 programs, of which we can fund the first six and part of No. 7. What do we do if, for example, we didn't fund the No. 3 priority program (which is very expensive), and the money saved by not funding the No. 3 program could pay for all of the remaining programs? That

is, we funded No. 1, No. 2, and priorities No. 4 through No. 10. Is this situation changed if the program we needed to *not* fund was the No.1 program instead of the No. 3 program?

Optimized Lists (Biggest Bang for the Buck)

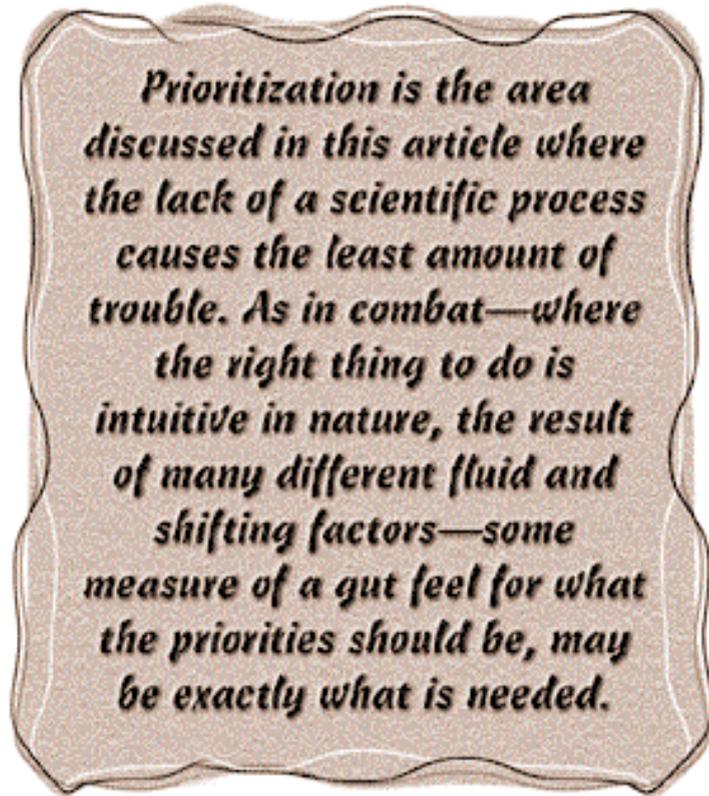
This is only possible if we have some sort of quantitative measurement of the relative importance of each of the programs to each other (such as we discussed in the section on prioritization). Here we could try funding various combinations of programs until we fund the combination with the highest aggregate weighting.

In the section on prioritization, we also discussed the case of two linked programs, where the higher priority program relies upon one of the lower ranked programs. Do we take that into account as we determine asset allocation?

Decisions and Consequences

One more very important factor must be considered as we look at both asset allocation and prioritization. The decisions we make in any particular year have impacts, some of them for decades after those initial decisions. Programs don't just appear and then disappear. They take time to accomplish. And then if successful, they incur costs for a long time following. A successful development program should result in production funding. Once we produce something, we have to maintain it. And even when the useful life is over, we have to dispose of the items we have produced. This is not a trivial set of issues.

Of course, priorities are continually shifting. Last year's No. 1 program is only No.5 this year, and who knows what priority it will be next year. But once committed (and I am not using the word in its governmental fiscal meaning), we cannot change course that quickly. Oh, they try! And that continual shifting of priorities, along with the associated funding of these programs, is the root of many of the inefficiencies in the acquisition system that people just love to complain about.



I believe that in order for acquisition transformation to work, we need to take a much closer look at stabilizing the processes discussed in this article. This is all the more true as we move to using evolutionary acquisition strategies—the latest paradigm on the block.

An Optimistic Look at the Future

A lot needs to happen for acquisition to be reasonably efficient. But the acquisition community can do a lot to help itself by developing efficient processes. By themselves, these processes will not be enough. Stable requirements, stable program authorizations, and stable appropriations are necessary for any real, meaningful transformation of the system.

My goal is to implement visible processes where operational needs are identified and the whole community—users, developers, and testers—develops concurrently requirements documents and acquisition strategies that work together.

I want to see programs prioritized in a manner that addresses the needs of the warfighters in a holistic fashion—real-

izing that certain programs feed off and require certain other programs, and all of these programs require an infrastructure that encourages and supports acquisition of the right weapons systems.

I want a process where programs receive funding with an eye to the long term, where the synergies involved and the consequence of the levels of funding and the schedules involved are taken into account, understood, and not taken lightly.

Once we understand this and start working together as a team—from the Congress and their staffs to the personnel in the Pentagon, from the Program Offices and the test teams (both developmental and operational) to the soldiers, sailors, airmen, and Marines deployed around the world—only then will we have fully earned our reputation as developing and acquiring the world's best weapons systems for our nation's warfighters.

Editor's Note: The author welcomes questions or comments on this article. Contact him at alex.slate@brooks.af.mil.