Capt. Norbert Doerry, USN

Enabling CAIV through Design, Modularity, and Program Management

ABSTRACT

Many naval ship acquisition programs have not been able to effectively implement Cost as an Independent Variable (CAIV). With a CAIV approach, operational requirements provided by the customer are given in terms of threshold and objective values. The range between these two values provides the Program Manager with trade-space to match the available funds with the capabilities that can be bought for that amount – the total program cost remains a constant. In practice though, much of the final cost of an acquisition program is fixed early in the design process through basic design decisions on architecture and the allocation of operational requirements and derived requirements to systems. Unfortunately, while much of the cost is determined early in the design process, estimating that cost to any degree of certainty is nearly impossible. As the costs of the program are better understood, the remaining design flexibility to adjust to increasing costs may not be sufficient to enable the Program Manager time to take corrective action when the cost estimates indicate a possible problem. This paper describes techniques, including the use of modularity, cost contingencies, and set-based design to provide the program manager with sufficient flexibility to implement CAIV.

The views expressed in this paper are those of the author and are not necessarily official policy of the U.S. Navy or any other organization. The intent of this paper is to foster dialogue to gain a better understanding of how to develop an acquisition strategy and systems engineering strategy to enable CAIV.

INTRODUCTION

Implementing CAIV has proven very difficult in many naval ship acquisition programs. In CAIV, the operational requirements provided by the customer are given in terms of threshold and objective values. The range between these two values is intended to provide the Program Manager with trade-space to match the available funds with the capabilities that can be bought for that amount.

An inherent assumption in CAIV is that the requirements flexibility afforded by the threshold and objective values can be translated into design and cost flexibility that can be used to keep the program cost a constant. Figure 1 presents a cost model that illustrates the different elements of cost that impact the ability to implement CAIV.

CAIV Target - Dollar Value that the program office is managing to.

Design Cost Estimate - Current estimate for the cost of the design without taking program and technical risks into account. This cost corresponds to a specific design point between the threshold and objective values for each requirement.

Risk Contingency - Estimate of an "insurance premium" to cover program and technical risks.

Cost Estimate - The sum of the Design Cost Estimate and the Risk Contingency representing the mean value of the estimate for the program.

Cost Margin - The difference between the CAIV Target and the Cost Estimate.

Cost Uncertainty Region - The range that the Cost Estimate can take on within a given confidence level. The Cost Uncertainty Region is a function of the Cost Estimate and the inherent uncertainty in the cost estimation methods used. The boundaries of the Cost Uncertainty Region are the Maximum and Minimum Cost Bounds. The Maximum Cost Bound corresponds to the maximum cost that the program is likely to reach for the given confidence level.

Design Flexibility - The maximum reduction in the cost estimate realizable by optimally adjusting design points closer to threshold requirements. As a design progresses, design flexibility generally decreases as design points are locked in, and changing design points would require rework. In later stages of design and in construction, moving closer to a threshold may increase cost due to rework needed to eliminate a capability.

Minimum Required Design Flexibility - The difference between the maximum Cost Bound and the CAIV Target.

CAIV Margin - The difference between the Design Flexibility and the Minimum Required Design Flexibility. This CAIV Margin can account for unknown risks.

Maximum Committed Cost – The difference between the Maximum Cost Bound and the Design Flexibility.

Mean Committed Cost (not shown) - The difference between the Cost Estimate and the Design Flexibility.

For CAIV to work, the CAIV Margin should remain positive over the life of the acquisition. This is difficult because all of the elements of the cost model (with the exception of the CAIV Target) will likely change. Over time, we would expect the Risk Contingency to decrease as risks are either realized or mitigated; the Cost Uncertainty Region is likely to become smaller as uncertainties are resolved, the Design Cost Estimate will mature as more is known about the details of the design and the cost of design, construction, and testing, and the design flexibility can be expected to decline as design decisions are made.

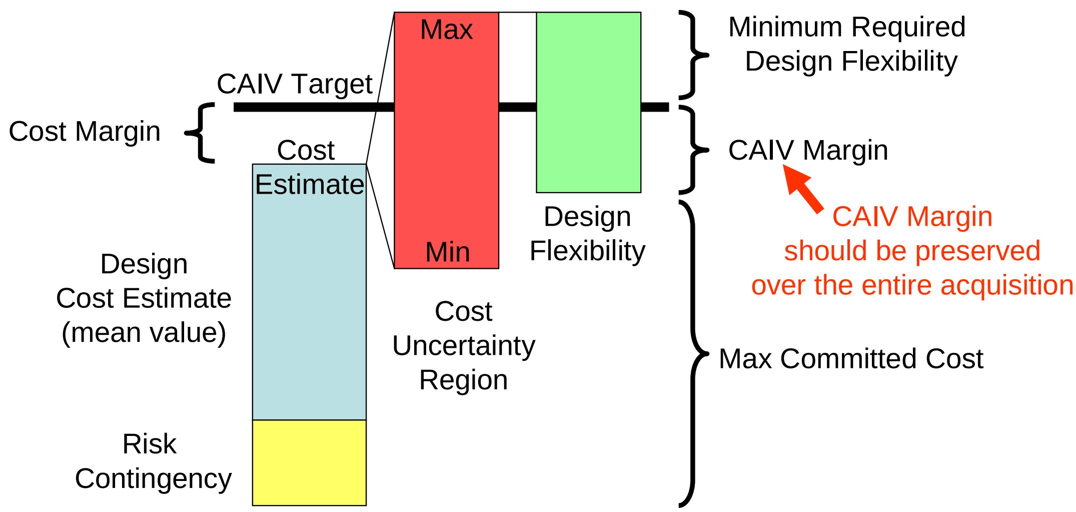

Figure 1: CAIV Cost Model

A close examination of Figure 2 can easily identify difficulties with implementing CAIV. Much of the final cost of an acquisition program is normally fixed early in the design process through basic design decisions on architecture and the allocation of operational requirements and derived requirements to systems. Unfortunately, developing accurate estimates is generally not possible. As a program matures and the costs better understood, the remaining design flexibility to adjust for cost fluctuation may not be sufficient for the Program Manager to take sufficient corrective action. There are however, several techniques that can provide the Program Manager with sufficient flexibility to implement CAIV. Instead of Figure 2, the goal would be a systems engineering process and acquisition strategy that would enable the Program Manager to still have flexibility when cost problems are identified as shown in Figure 3.

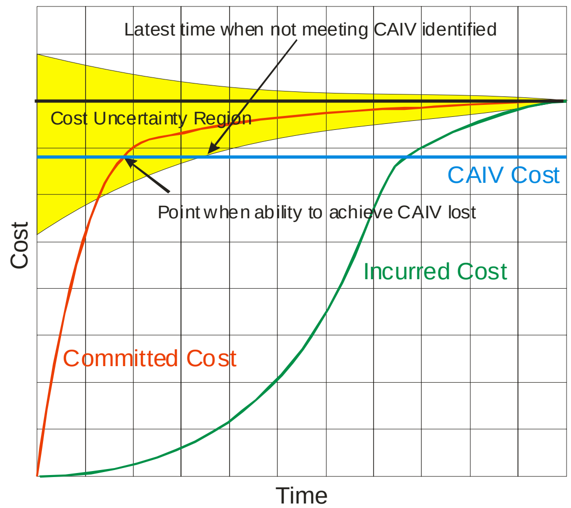

Figure 2: Traditional Cost Commitment vs. Cost Incurred curves

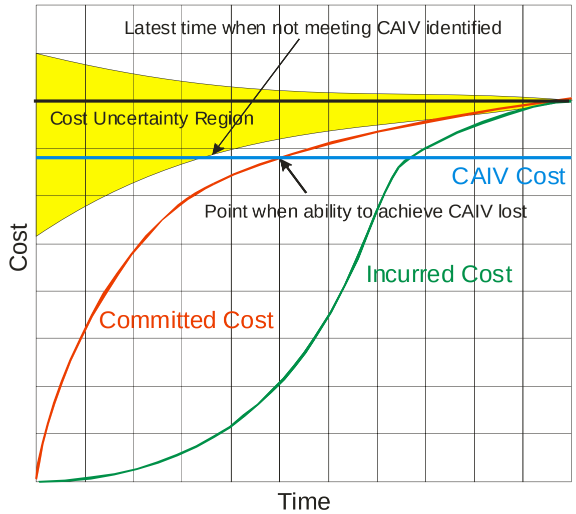

Figure 3: Improved Cost Commitment vs. Cost Incurred Curves

DIFFICULTIES IMPLEMENTING CAIV

The best way to control costs is to have sufficient funds available to get the job done and manage those funds wisely. Unfortunately, for a variety of reasons a Program Manager will discover that the program does not have sufficient funds to execute the current program plan. When funding becomes "tight," the usual response includes:

- Spreading cuts along all cost accounts – increasing the risk that the work can not be done correctly and on schedule with the amount of available funds. Rework will result in increased costs.

- Reviewing every task to cut any perceived margin – increasing the risk of a cost overrun. Tasks that are perceived underfunded are rarely plussed up. By cutting only the "overfunded" tasks without increasing the "underfunded" tasks, on average the program will be underfunded.

- Deferring work to post-delivery – often at an increase in overall cost because work is done onboard where labor efficiency is much lower than work performed in a shop environment during the construction process.

- Cut engineering, analysis, documentation, testing, and Government engineering oversight – increasing risk that technical issues will be discovered late when corrective action is very expensive. Keane, Fireman and Billingsley (2005) provide evidence that "the most important factor in ensuring that programs are delivered on time and on budget is increased funding in the early stages of development." Yet many programs reduce this early stage work and rush into production in a generally unsuccessful attempt to control costs.

- Descoping capability – If not preplanned, then the cost to eliminate a capability from a design will require significant engineering (and potentially production) rework. If not descoped early enough, removing capability may increase costs. In any case, if not preplanned, the cost to restore a descoped capability can be much larger than the amount of funds recovered from the descoping effort.

While each of these responses can in the short-term appear to cut costs, the resultant increase in risk over the life of the program will likely result in increases in cost and schedule slippage as individual risks are realized. As shown in Figure 4, a RAND study for the U.K. MoD found that 69% of schedule slippage was due to change orders, late product definition, and lack of technical information (Arena et. al. 2005). These results are consistent with the normal program management response to predicted cost over-runs.

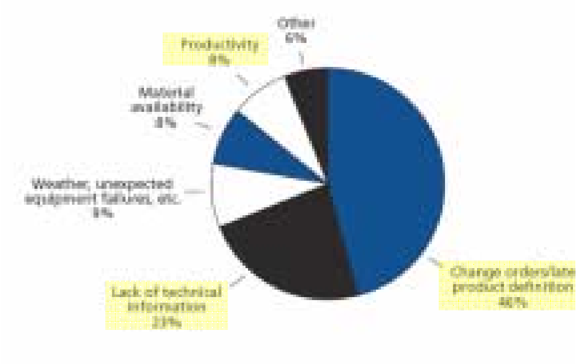

Figure 4: Causes of Schedule Slips Reported by Shipbuilders (percentage) (Arena et. al. 2005)

In practice, the traditional responses to predicted cost over-runs often increase program risk, but current methods of establishing risk contingencies are not sensitive to individual risk items. The net impact is that the acceptance of risk results in an increase of the cost uncertainty, such that the region of uncertainty includes the CAIV target. In this way, the Program Manager can be convinced that achieving the CAIV target is possible, when in reality, the likelihood of success is even lower. The impact of these typical responses is shown in Figure 5. Because the Committed cost is already above the CAIV Cost target when the cost problem is identified, the "Corrective Action" merely appears to solve the cost problem by increasing the size of the Cost Uncertainty Region to encompass the CAIV Cost target.

CAIV is intended to provide the program manager with the flexibility to trade performance for cost. For this to successfully happen, the Program Manager must have the ability to identify a potential cost problem and take corrective action before the Committed Cost curve crossed the CAIV Cost target as shown in Figure 6.

The difficulties experienced in keeping committed costs low for a prolonged period of time include:

- Often, the point where a design will fall between the threshold and objective values of a requirement will be fixed early in the design process through the selection of equipment and systems. Once equipment decisions are made and the design evolves to incorporate the equipment, the flexibility offered by the threshold and objective values is largely eliminated – the design point becomes a de facto fixed requirement[1]. For CAIV to work, the system architecture must be scalable such that the design point between the threshold and objective can be affordably adjusted to respond to cost perturbations over as much of the life of the acquisition program as possible.

- To minimize costs, some Program Managers (or customers) immediately direct that the program only fund for threshold performance. The view is "If the minimum wasn't good enough, it wouldn't be the minimum." The budget is then established at the current cost estimate for meeting only the threshold requirements. As normal variances in the projected cost become apparent over time, the typical responses listed above are implemented. The net result is that the program is not executed in a CAIV environment, but rather on a fixed set of requirements, with the normal increase in costs due to the typical response to reduce the apparent cost resulting in actual cost increases.

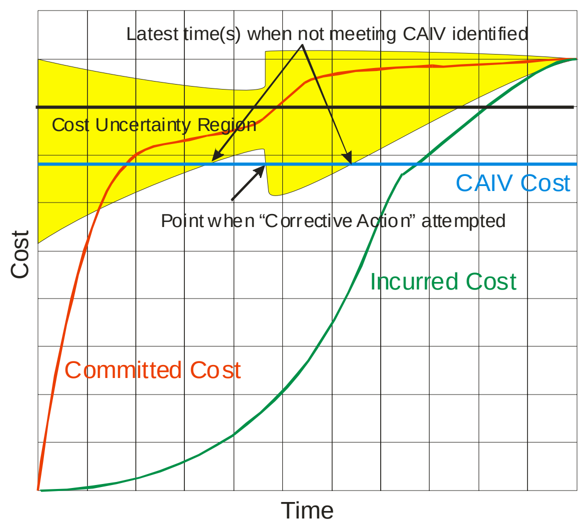

Figure 5: Impact of Late Corrective Action on achieving CAIV Cost Target

[1] More precisely, the threshold to objective range is still useful to account for uncertainty in system performance.

Approved for Public Release

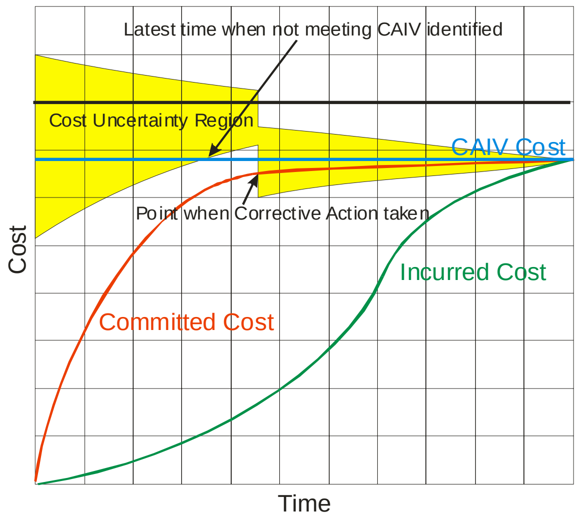

Figure 6: Successful Corrective Action to achieve CAIV Cost Target

SOURCES OF COST UNCERTAINTY

Figures 1 through 6 clearly show that minimizing cost uncertainty early could also provide the program manager with sufficient time to take corrective action before the Committed Cost line crosses the CAIV Cost target. Unfortunately, the cost for the development and construction of a complex system is difficult to predict with precision. Often, the acquisition schedule will span ten to twenty years, and many of the assumptions used to develop a cost estimate will prove to be incorrect. Sources of uncertainty include:

- Changes in labor rates
- Changes in material rates
- Uncertainty in the amount of man-hours needed (especially true for new technology)
- Contractor expertise (competition for workforce with other industries)
- Cash Flow impacts (Generally a result of program funding instability)
- Poorly specified, misunderstood, or emergent safety requirements requiring rework.
- Realized Risks – Problems
- Unpredicted Problems
- Waste

Furthermore, existing financial management policies discourage program managers from maintaining a contingency fund for addressing much of the cost uncertainty. Funds allocated for change orders can only be used to address poorly specified, misunderstood or emergent safety requirements requiring rework. Management Reserve is used to address realized risks and unpredicted problems. Funds are not typically allocated to cover other sources of uncertainty.

PROVIDING THE PROGRAM MANAGER WITH COST FLEXIBILITY

Key to implementing CAIV is keeping the Committed Cost outside of the Cost Uncertainty region. Unfortunately, determining either the Committed Cost or the Cost Uncertainty region with any precision is currently not possible. Still, there are a number of practices that provide the Program Manager with cost flexibility:

- Modularity
- Requirements Stability
- Trade Space
- Cost Contingencies
- Set Based Design
- Eliminate Sources of Cost Risk

Improving Design Flexibility through Modularity

Modularity implemented in a scalable architecture enables the development of subsystems independent of the overall platform development. To work in a CAIV environment, providing scalable performance at scalable cost is critical. Furthermore, the architecture should enable the decision for how much performance to provide to be delayed as long as possible without impacting the cost-performance relationship.

The systems architect should use modularity strategically to control costs. Areas to apply modularity include:

- Material solutions to address operational requirements with a threshold and objective value. The modularity should enable a scalable solution to cover most or the entire threshold to objective range.

- Material solutions for technologies that are anticipated to become obsolete and not logistically supportable during the design service life of the system.

- Material solutions for operational requirements likely to change over the life of the system.

In each of these cases, the modularity must enable a cost effective change in system capability.

Examples of modularity that preserve flexibility for the Program Manager in adjusting system performance to meet cost targets include:

- Sizing modular radar arrays to achieve the objective value, but only partially populating the radar array.

- For distributed systems such as electrical power and chill water, design the system for full service life allowances, but only populate generation and distribution system “modules” to meet the delivery condition. The system design must incorporate the ability to easily add the modules to achieve full service life requirements.

- Sizing network equipment racks to hold the full number of blade-servers to meet objective requirements, but only partially populating racks with blade-servers.

- Designing a scalable software architecture that is capable of achieving objective requirements, but only developing, testing, and installing software modules to achieve a lesser level of performance.

- For ships with an Integrated Power System, design the power generation and propulsion motors to achieve a sustained speed greater than threshold speed. Use some portion of the power generation installed above threshold speed as a design and construction margin and/or service life margin.

Reinertsen (1997) describes three underlying principles for developing a product architecture:

Make decisions with regard to how modular to make the product

Partition the design to control the impact of variability

Manage the internal interfaces of the design

With respect to modularity, he states that the secret art of product architecture is that the benefits will only come when the system is portioned properly and the interfaces are properly defined and stable. Stable interfaces require an adequate margin to prevent changes during the design and the resultant rework.

Reinertsen emphasizes that a broad benefit of modularity is that it permits reuse of modules from other designs. A carefully designed reuse plan can save enormous amounts of design time and expenses. Within the CAIV environment, each increment of performance corresponds to a different systems design and corresponding cost. The re-use in this context is the re-use of design work for different levels of performance.

Estimating system costs of modular systems is not easy. At the interface level, costs usually increase because we add parts and potentially complexity. At the module level, costs can either rise or fall because the module is designed to meet the needs of many system designs instead of just one. The cost impact of modularity depends on both cross-program economics and the need to accommodate many “designs” to implement CAIV and cannot be assessed on the basis of a single design. For a CAIV program, the greater the number of times that requirements are adjusted to maintain the cost target, the required non-recurring engineering to implement the change in requirements will likely be increasingly less for modular systems than for non-modular systems.

If not done properly, modularity can affect performance. Interfaces can act as bottlenecks as compared to a tightly coupled non-modular system. As a result, Reinertsen differentiates between low-expense architectures, low-cost architectures, high-performance architectures, and fast-development architectures. He particularly emphasizes that architecture should be an economic decision, not a technical one. Technical people are still likely to play a dominate role in selecting the architecture; however, they cannot do the job alone. Acquisition professionals, ship design engineers and cost engineers must collaborate from the earliest stages of design.

Reducing Risk Contingencies through Requirements Stability

Requirements Stability is extremely important to CAIV. Requirements instability can quickly result in unplanned design (and production) rework. This rework usually results in additional costs that must be offset by reductions elsewhere. In general, making design changes late in the design process or during construction should be avoided to the greatest extent practical. Unless unavoidable, requirements should not be altered following the Preliminary Design Review, and configurations should not be altered following the Critical Design Review. If a specific requirement can not be fixed or there is risk that it may change late in design or construction, then the systems architecture should be modular and scalable as indicated in the previous section. This implies that a program should continuously evaluate the risk of a requirement changing over the design and construction period, as well as during the service life of the system. The choice of how to implement modularity must also account for when the risk is likely to be realized (during design, construction, or in-service).

Requirements Stability is not limited to growth in requirements. Late reductions in requirements, as in descoping efforts to reduce program costs, are also sources of additional work that often consume much of the cost that is intended to be saved.

Improving Design Flexibility by maintaining a Trade-space

For CAIV to work, the Program Manager must have flexibility to trade performance for cost. If the Program Manager only budgets to achieve the threshold requirements then the Program Manager has lost all flexibility to address unforeseen cost increases. Early in the design of a system, the budget should be set to achieve close to the objective values (about 65%-85% of the threshold to objective range). The difference in cost for the capability between the threshold capability and the budgeted capability becomes a margin that can be consumed during the design and construction of the system. This can only work if the system design is such that the management flexibility is preserved (through modularity for example) to enable the consumption of this margin.

Better Risk Contingencies – Budgeting for Risk

For many programs, cost estimates for a system do not directly account for technical risks. If technical risks are accounted for at all, their impact is assessed as a gross fraction of the total ship cost. Most alarming, risk-reduction activity is considered “non-value added” because this activity does not impact the material properties of the end product. By not properly accounting for risk in cost estimation, a program manager will be tempted to cut risk reduction activity because the cost estimation methodology only includes the cost of the risk reduction activity and not the reduction in the cost of the risk contingency due to the resulting reduction in risk. Within these cost models, risk-reduction activity only adds costs; hence they suggest that risk reduction activity should perversely be eliminated.

Ideally a cost contingency should be incorporated for each risk in the program risk register. The cost contingency should be considered an “insurance” payment to account for the impact on the ship program should the risk be realized. Because the likelihood of a risk outcome is not 100%, (if so, then it would be a problem and not a risk) the cost contingency reserved for a risk should typically be a fraction of the cost to recover from the risk outcome. This fraction will depend on the aggregation of all program risks, and the program’s risk tolerance. Within a CAIV environment, the sum of the cost margin plus the cost contingency should have a high probability (say 90%) of being sufficient to fund the aggregation of realized risks as well as risk mitigation efforts designed to reduce cost contingency requirements.

Implementing a good cost contingency method requires careful definition of risk outcomes as well as allocating cost contingencies only when risks are realized or for cost effective risk mitigation. Risk outcomes should be defined in terms of precisely what adverse event will occur and what required efforts would be needed to recover from the adverse event. A program should conduct a risk mitigation activity if the cost of the risk mitigation activity is less than the reduction in cost contingency for that risk that is realized by the risk mitigation effort.

If cost contingencies are allocated to non-risk mitigation activities before risks are realized, the funds will likely be spent without mitigating or recovering from the realized risk. The effect of "Money allocated is Money Spent" becomes evident. Careful management of the cost contingency funds is needed to ensure work is conducted in a controlled-risk manner to avoid unforeseen problems while undertaking cost-effective risk mitigation efforts. For further reading on cost contingencies see NRC (2005), Kujawski (2002) (2004) and Kujawski et. al. (2002).

Improving Design Flexibility through Set-Based Design

Set-Based Design as described by Bernstein (1998) preserves design flexibility through three basic tenets:

"*Understand the design space*
- Define feasible regions
- Explore tradeoffs by designing multiple alternatives
- Communicate sets of possibilities

Integrate by intersection
- Look for intersection of feasible sets
- Impose minimum (maximum) constraint
- Seek conceptual robustness

Establish feasibility before commitment
- Narrow sets gradually while increasing detail
- Stay within set once committed
- Control by managing uncertainty at process gates"

As an example of how set-based design has been applied in commercial industry, Ward et. al. (1995) describe Toyota's successful implementation of set-based design to produce competitive automotive designs faster and cheaper than traditional design methods.

In a set-based design process, engineers of different systems (i.e. electrical systems, combat systems, hull design, etc.) communicate ranges of solutions with associated derived requirements on other systems and levels of performance. As shown in Figure 7, regions of feasibility are determined by the intersections of the different ranges of solutions offered by the different engineering disciplines. Initially, the ranges of discipline solutions may need to grow to enable a sufficiently large region of feasibility at the intersection of independent solutions. The range of solutions for each engineering discipline is then reduced at the process gates to eliminate sub-system solutions that are not likely to contribute to a total system solution. Following the reduction in design space, engineers produce additional levels of details of the subsystems to refine the solution, improve cost estimates, and reduce risk. Within a CAIV environment, the size of the feasible design space must remain large enough to encompass the cost uncertainty. The design space is only reduced at a process gate if the design is sufficiently detailed to enable an accurate enough cost estimate to eliminate regions of the design space.

A marine engineering example of set based design would be the interaction of hull shape, propeller selection, and propulsion motor selection. For a range of required displacements and deck area, the hull designer would provide the range of speed – Effective Horsepower (EHP) curves and propeller size limitations. For this range, the propeller designer would provide the marine engineer with achievable propeller efficiencies, associated shaft speed – shaft power – ship speed curves along with maximum shaft speeds to preclude cavitation. The propulsion engineer would look at the range of powers and shaft speed required, and identify a motor architecture that could cover that region. The cost engineer would identify the cost and cost uncertainty that would apply to the different design spaces.

Initially, intersections of the different solutions would be identified. Areas of the design space that are Pareto – dominated, that is, there are solutions which perform better at lower cost, are eliminated from consideration. Likewise, regions of the design space for which the estimated cost minus cost uncertainty exceed the CAIV target are also eliminated because there is a small probability that the CAIV target will be achieved in that portion of the design space. In this manner a design solution is arrived at by eliminating potential solutions rather than by trying to make a point design "work."

Because a portion of the cost uncertainty will not be realized until after the design is completed, set-based design is not sufficient by itself to ensure CAIV. Other techniques that can be implemented after design is complete, such as modularity, can be combined with set-based design to implement an overall CAIV acquisition strategy.

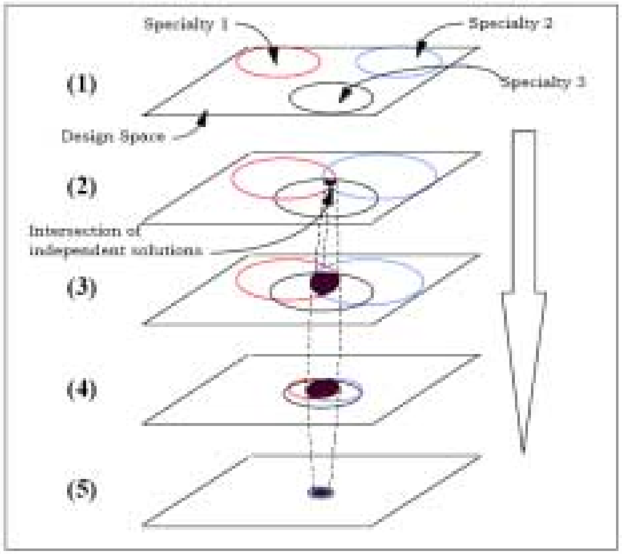

Figure 7: Design Convergence using Set-Based Design (Bernstein 1998)

Reduce Risk Contingencies by Eliminating Sources of Program Cost Risk

Removing sources of cost risk from a program is an effective way of reducing the amount of Risk Contingency needed. Some elements of cost uncertainty are outside the control of a program manager. Inflation for example, is very difficult to predict but can have a major impact on the cost of materials. Forcing a Program Manager to account for inflation within a CAIV environment may in itself consume all cost flexibility. Instead, that portion of the cost of a product allocated to materials can be adjusted according to a standard industry index. The Bureau of Labor Statistics publishes a number of indices that could be used. In previous years, ship acquisition program used this method in the form of Escalation Payments.

Conclusions

For a Program Manager to effectively employ a CAIV acquisition strategy, the design, engineering, and cost estimating methods must be aligned to ensure that costs are not committed so early as to eliminate the flexibility necessary to react to unpredictable cost variances. Techniques that assist the Program Manager and lead design engineer include:

- Implement modularity to provide flexibility
- Stabilize Requirements – use modularity to address requirements risks.
- Provide a trade-space – don't fix a design point too early between the threshold and objective values.
- Establish program budget wisely – include a budget for risk.
- Use cost contingencies wisely – be aware of the effects of "Money allocated is Money Spent"
- Employ Set-Based Design to improve design flexibility
- Eliminate Sources of Cost Risk to reduce the funds allocated to risk contingencies.

By employing these techniques, final design decisions can be delayed without impacting the overall acquisition schedule. By prolonging decisions, flexibility is preserved and cost better controlled.

References

Arena, M. V., J. Birkler, J. F. Schank, J. Riposo, C. A. Grammich, "Monitoring the Progress of Shipbuilding Programmes, How Can the Defence Procurement Agency More Accurately Monitor Progress?," RAND MG-235, 2005.

Bernstein, Joshua, "Design Methods in the Aerospace Industry: Looking for Evidence of Set-Based Practices," Thesis for the degree of Master of Science at the Massachusetts Institute of Technology, Technology and Policy Program, May, 1998.

Committee for Oversight and Assessment of U.S. Department of Energy Project Management, National Research Council, "The Owner's Role in Project Risk Management," ISBN: 0-309-54754-7, 2005. (NRC 2005)

Keane, R. G., H. Fireman, D. W. Billingsley, "Leading a Sea Change in Naval Ship Design: Toward Collaborative Product Development," SNAME Journal of Ship Production, Volume 23, Number 2, May 2007 , pp. 53-71.

Kujawski, Edouard, "Why Projects Often Fail, Even with High Cost-Contingencies," Systems Engineering, Vol. 5, No. 2, 2002.

Kujawski, Edouard, "Incorporating Psychological Influences in Probabilistic Cost Analysis," Systems Engineering, Vol. 7, No 3, 2004.

Kujawski, E., M. L. Alvaro, and W. R. Edwards, "Selection of Technical Risk Responses for Efficient Contingencies," Systems Engineering, Vol. 5, No. 3, 2002.

Reinertsen, Donald G., "*Managing The Design Factory: A Product Developer's Toolkit,"* The Free Press, New York, New York, 1997

Ward, A., J. K. Liker, J. J. Cristiano, and D. K. Sobek II, "The Second Toyota Pardox: How Delaying Decisions Can Make Better Cars Faster," Sloan Management Review, Spring 1995.

Capt. Norbert Doerry (Ph.D. Naval Electrical Power Systems – MIT '91, SMEECS, NE – MIT '89, BSEE USNA '83) is an Engineering Duty Officer currently assigned as the Technical Director for Future Concepts and Surface Ship Design Group (SEA 05D) in the Naval Sea Systems Command. Previous tours at NAVSEA include Technical Director for IPS and Ship Design Manager for JCC(X). He additionally served as an Assistant Project Officer for Aircraft Carrier repair and new construction at SUPSHIP Newport News and as the Assistant Acquisition Manager for LHD 8 within PMS 377. Prior to becoming an Engineering Duty Officer, he served as gunnery and fire control officer on USS *Deyo*.