APPLICATION OF REAL OPTIONS THEORY TO DoD SOFTWARE ACQUISITIONS

Capt Albert Olagbemiro, USAFR, Johnathan Mun, and Man-Tak Shing

The traditional real options valuation methodology, when enhanced and properly formulated around a proposed or existing software investment employing the spiral development approach, provides a framework for guiding software acquisition decision making by highlighting the strategic importance of managerial flexibility in managing risk and balancing a customer's requirements within cost and schedule constraints. This article discusses and describes how an integrated risk management framework, based on real options theory, could be used as an effective risk management tool to address the issue of requirements uncertainty as it relates to software acquisition and guide the software acquisition decision-making process.

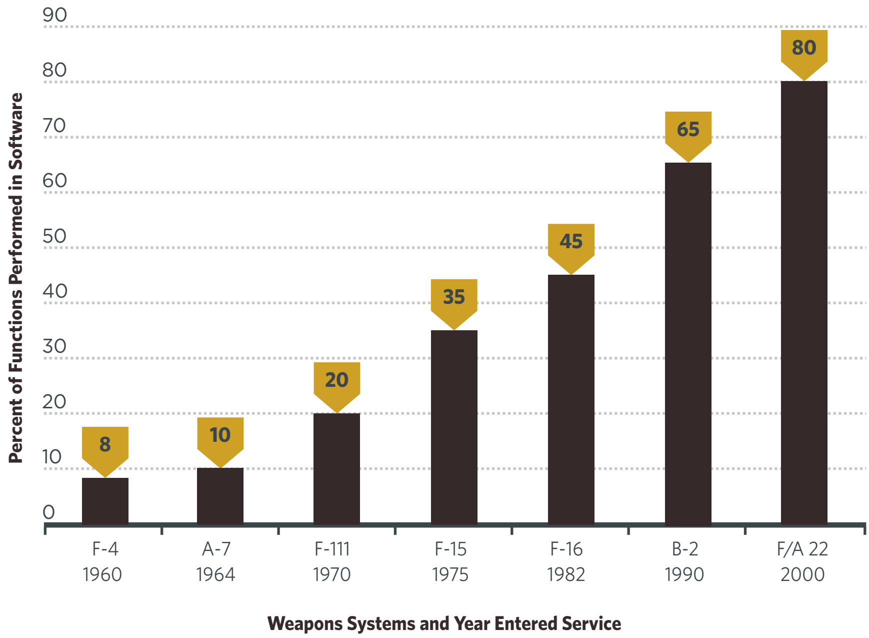
Keywords: *Risk Management, Software Acquisition, Strategic Investment, Evolutionary Acquisition (EA), Real Options Theory*

image designed by Nina Costine »


SHIP TO:
DoD Software Acquisition
WASHINGTON, DC 20301-1400
AX 10U 24H 7D 8838 XTX
THIS SIDE UP

Software is currently the major expense in the acquisition of software-intensive systems (Figure 1), with its role as a technology platform rising from providing a mere 8 percent of weapons systems functionality in 1960 to over 80 percent of functionality in 2000 (Department of Defense [DoD], 2000).

FIGURE 1. SOFTWARE GROWTH IN WEAPONS SYSTEMS



Note. Adapted from *Report of the Defense Science Board Task Force on Defense Software*, November 2000, by Defense Science Board, Department of Defense, pp. 11–12.

Considering the immense presence and ever-increasing role that software plays in weapons systems, software is and should be treated as a capital investment; accordingly, an approach emphasizing a strategic investment methodology in its acquisition is necessary. This approach would emphasize the linking of strategic program management decisions to current and future unknown software requirements within the stipulated parameters of cost, risk, schedule, and functionality. This strategic program management approach is needed to align the software investment under consideration within the context of the overall portfolio of existing/planned software investments to ensure that synergies in efficiencies are leveraged in the delivery of the intended/desired joint capability.

The key to the implementation of a strategic program management framework is a disciplined requirements engineering approach that embodies a risk management-driven model in the acquisition planning process. This framework would link and build on two of the three key processes outlined in the 2009 Joint Capabilities Integration and Development System: requirements; the acquisition process; and the Planning, Programming, Budgeting, and Execution System.

Method

Risk management should be a consideration that is addressed much earlier in the software engineering process—at the acquisition level—during the investment decision-making activities prior to the commitment to acquire and/or develop a software system. The appropriate risk mitigation/reduction strategies or options should be crafted much earlier in the software investment/acquisition process, which leads to the real options approach proposed in this article.

Real Options Valuation

Real options valuation originated from research performed to price financial option contracts in the field of financial derivatives. The underlying premise of its suitability and applicability to software engineering is based on the recognition that strategic flexibility in software acquisition decisions can be valued as a portfolio of options or choices in real "assets"—much akin to options on financial securities that have real economic value under uncertainty (Dixit & Pindyck, 1995). In contrast to financial options, real options valuation centers on real or nonfinancial assets, and is valuable because it enables the option holder (the acquisition executive) to take advantage of potential upside benefits while controlling and hedging risks.

When extended to a real asset such as software, real options could be used as a decision-making tool in a dynamic and uncertain environment.

An option is a contract that confers upon its holder the right, without obligation, to acquire or dispose of a risky asset at a set price within a given period of time. When extended to a real asset such as software, real options could be used as a decision-making tool in a dynamic and uncertain environment. A central and necessary tenet of the real options approach is a requirement for the presence of uncertainties—an inherent characteristic of most software acquisition efforts. Real options are implicit or explicit capabilities created for real assets that provide the option holder with time-deferred and flexible choices (options) regarding future risks or changes of the software, and could explicitly address the issue of software investment choices for future capabilities. Through these capabilities, the option holder may choose to adjust, reduce, increase, or abandon the investment in the future, thereby stabilizing returns from these assets. Prior to its application in any domain, the real-options approach calls for the existence of five pre-conditions. These pre-conditions, as outlined by Mun (2006), follow:

1. A basic financial model must be created to evaluate the costs and benefits of the underlying software asset.
2. Uncertainties must exist during the software acquisition process; otherwise, the real options analysis becomes useless as everything is assumed to be certain and known.
3. The uncertainties surrounding the software acquisition process must introduce risks, which directly impact the decision-making process.
4. Management must have the flexibility or options to make mid-course corrections when actively managing the project.
5. Management must be smart enough to execute the real options when it becomes optimal to do so.

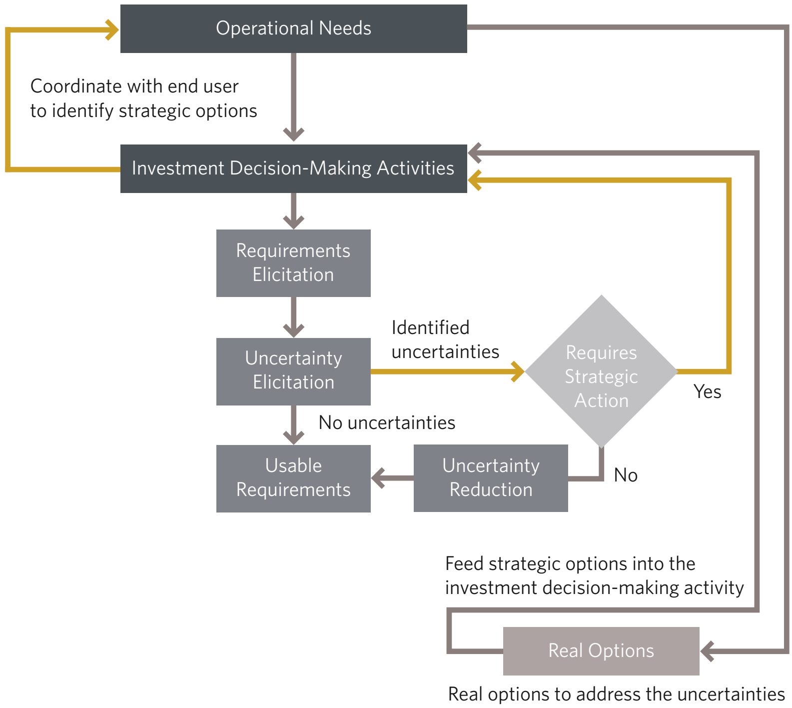
Since software acquisition encapsulates the activities related to procurement decision making, development, implementation, and subsequent maintenance, each of these pre-conditions can be directly correlated to the various activities associated with a software acquisition effort. The uncertainties that surround these activities manifest themselves in the form of risks and could range from changing or incomplete requirements or insufficient knowledge of the problem domain, to decisions related to the future growth, technology maturation, and evolution of the software.

While risks associated with large-scale software acquisition have been effectively managed through the application of stochastic frameworks and project management techniques, a framework based on the real options approach is best suited for the DoD acquisition process because of its capacity to overcome the limitations of classical financial analysis techniques, such as the discounted cash flow (DCF) or net present value (NPV) approach, both of which treat projects/investments as passively managed, rather than actively managed projects/investments, albeit a gross misrepresentation of the norm in software acquisition.

Software Acquisition Uncertainties

To tackle the issue of uncertainties surrounding software acquisition, a formal and distinct uncertainty elicitation phase is proposed as part of the software investment decision-making process (Figure 2) to obtain information on the relevant uncertainties from

FIGURE 2. UNCERTAINTY ELICITATION MODEL

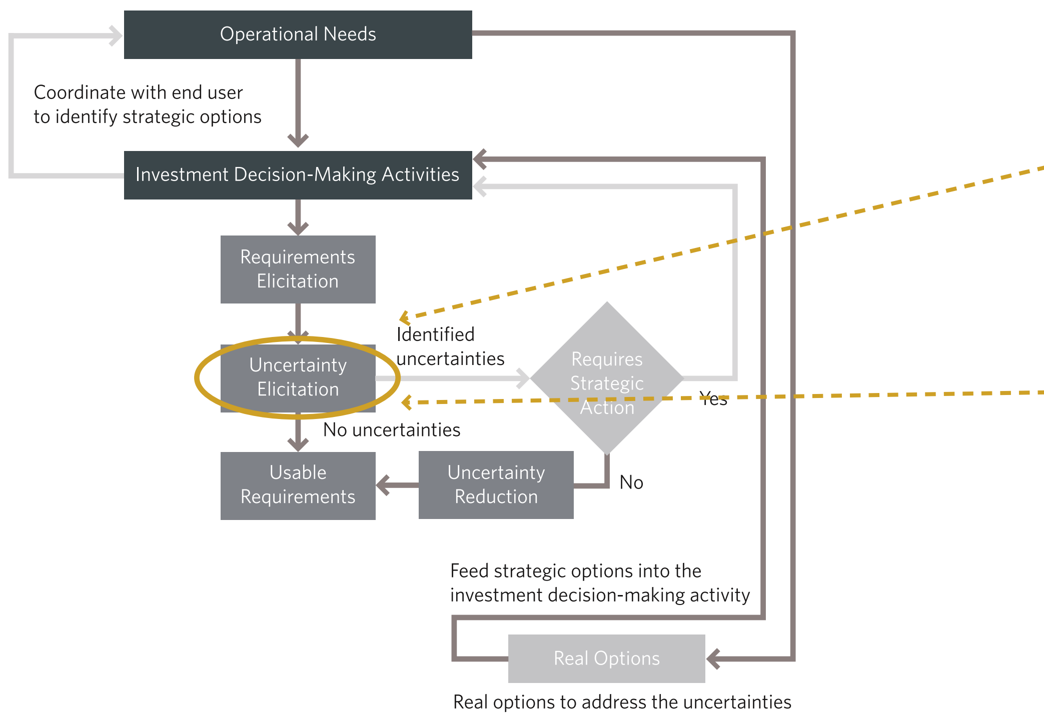


a strategic point of view. Stakeholders in this phase would normally include representatives of the Joint Requirements Oversight Council, in concert with independent requirements subject matter experts, to identify and document uncertainties as they are revealed from an independent point of view.

Implementing an explicit uncertainty elicitation phase would facilitate the identification of uncertainties very early on in the acquisition process, so the necessary steps could be taken to either refine the requirements to address the uncertainties or identify strategic options to mitigate the risks posed by the uncertainties.

During the uncertainty elicitation step in the model, uncertainties are captured from two perspectives—the managerial and technical perspective—as illustrated in Figure 3. Managerial uncertainties of people, time, functionality, budget, and resources contribute to both estimation and schedule uncertainties, which are considered to be pragmatic uncertainties.[1] Technical uncertainties—incomplete, ambitious, ambiguous, changing, or unstable requirements—contribute to software specification uncertainties, which lead to software design and implementation, software validation, and software evolution uncertainties—all of which can be categorized as exhibiting both Heisenberg-type[2] and Gödel-like[3] uncertainties.

FIGURE 3. PRECONDITION 2

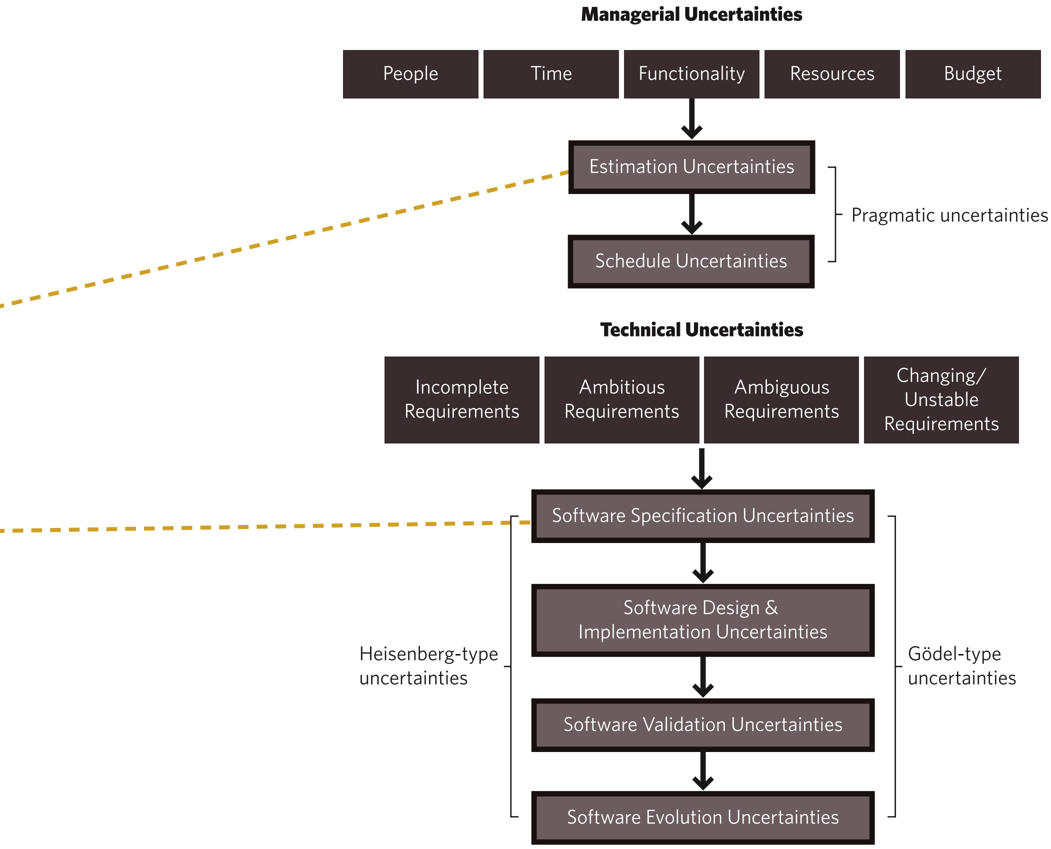


If the uncertainty cannot be resolved, strategic real options could be developed to address the risks posed by the uncertainty, providing management the flexibility to address the risks posed by the uncertainties when they become revealed at a later date during the acquisition effort.

The Real Options Valuation Framework

To develop the appropriate options to hedge against the risks due to the uncertainties surrounding a software acquisition effort, we formulated a generalized real options framework (Figure 4) in line with the five preconditions outlined by Mun (2006). This proposed framework consists of the following six phases, each of which explicitly addresses and establishes compliance with the preconditions.

1. Needs Assessment Phase
2. Risk Determination Phase
3. Options Analysis Phase
4. Options Valuation Phase
5. Investment Valuation Phase
6. Execution Phase



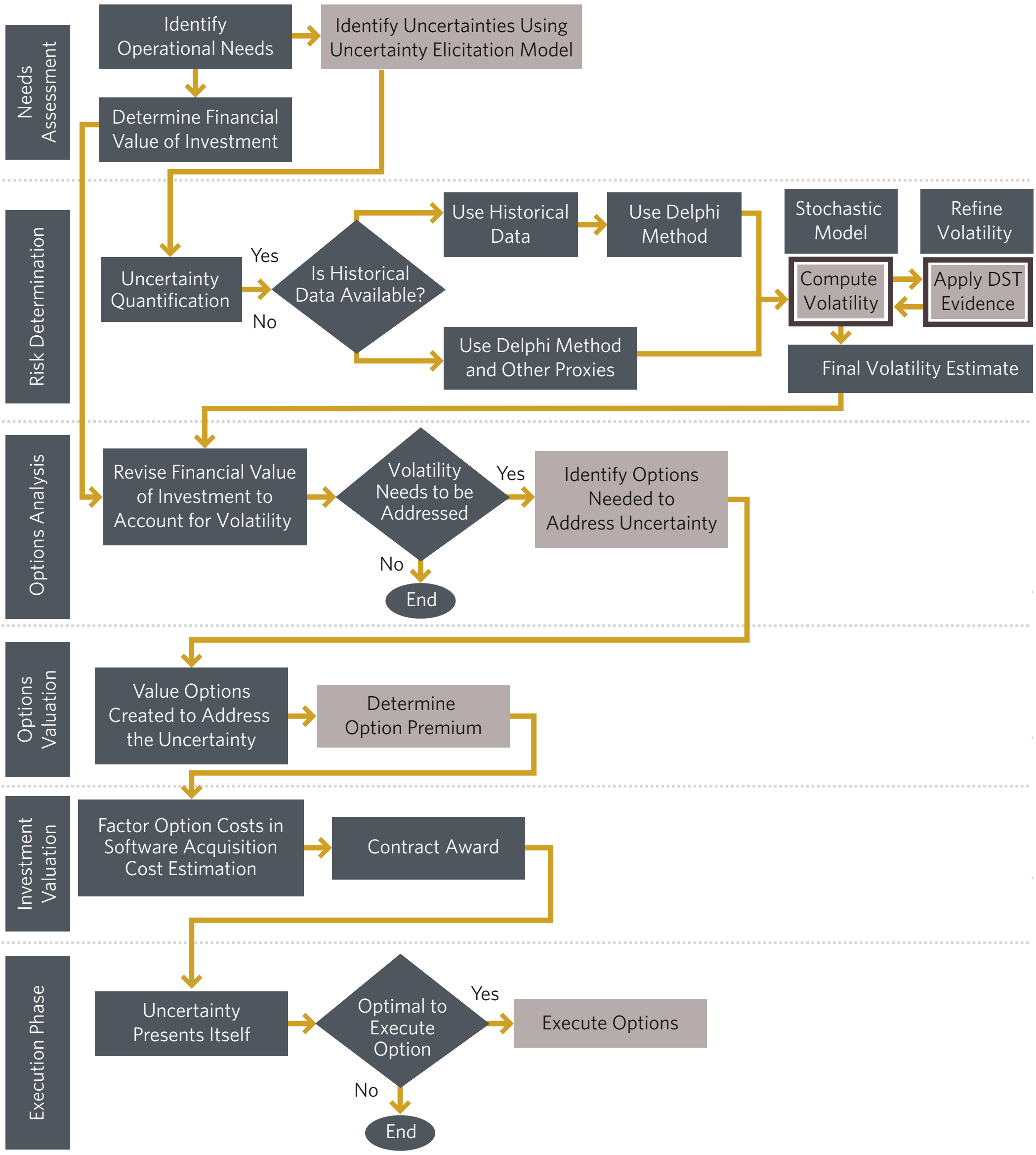
We further validated the framework and illustrated its viability, as an example, by applying it to the Future Combat Systems Network (FCSN), the software component of the U.S. Army Future Combat Systems program (Congressional Budget Office, 2006, pp. 2–21)

Results

Phase I: Needs Assessment

Business Case. The needs assessment phase culminates with the establishment of a business case along with the associated financial model. The financial model is used to evaluate the costs and benefits of the underlying software asset being considered for acquisition in compliance with the first precondition of the real options approach. The traditional discounted cash flow model with a net present value[4] (NPV) is employed to satisfy this requirement, and NPV is computed in terms of five high-level determinants (Erdogmus & Vandergraaf, 2004):

FIGURE 4. REAL OPTIONS FRAMEWORK



Note. DST = Dempster–Shafer Theory. A mathematical theory of evidence/generalization of probability theory where probabilities are assigned to sets as opposed to mutually exclusive and exhaustive propositions termed “singletons.” Information from multiple sources can be combined in the form of belief assignments, which serves to aggregate the information with respect to its constituent parts.

$$NPV = \Sigma \frac{(C_t - M_t)}{(1+r)} - I \quad (1)$$

I is the (initial) *development cost of the FCSN*
t is the (initial) *development time* or time to deploy the FCSN.
C is the *asset value of the FCSN* over time t
M is the *operation cost of the FCSN* over time t
r is the rate at which all future cash flows are to be discounted (the *discount rate*) where the standard assumption in [1] is $(C - M)$ is always positive.

An NPV of $6.4 trillion was computed for the FCSN using estimated values of $163.7 billion, 13 years, and 3.0 percent for variables I, T, and r respectively based on key assumptions in Olagbemiro (2008, pp. 121–148).[5] Furthermore, a value of $C - M$ = $10 trillion was estimated along the lines of the assumptions by Olagbemiro (2008, pp. 121–148).

Uncertainty Identification. Uncertainty identification is the next crucial step performed during the needs assessment phase. In this step, the uncertainty elicitation model is used as a mechanism to identify uncertainties. When applied to the FCSN, it was determined that requirements uncertainty fostered by technological maturation issues (Government Accountability Office [GAO], 2008a, pp. 89–90) plagued the FCSN program from the onset and introduced several other corresponding uncertainties. Thus, the following uncertainties were determined to have been retroactively predictable within the context of the proposed real-options framework.

Technical Uncertainties

1. Requirements uncertainties
2. Integration uncertainties
3. Performance uncertainties

Managerial Uncertainties

1. Estimation uncertainties (size and cost of the software)
2. Scheduling uncertainties

Phase II: Risk Determination

The risk determination phase consists of two steps: *uncertainty* quantification and *volatility* determination.

Uncertainty Quantification. Uncertainty implies risk; consequently, uncertainty must be duly quantified as a risk factor with the goal being to assign an appropriate numerical value to the uncertainty. This is accomplished by gathering evidence using historical data from previous acquisition efforts that faced similar risks. In the absence of historical data, the Delphi method[6] is suggested. The objective of the evidence-gathering activity is to equate/approximate the software engineering uncertainties of the current software acquisition effort to a quantifiable property (risk factor) based on historical evidence depicted by previous software acquisition efforts. Such evidence-gathering activity is necessary to gauge the magnitude/impact of the risk on the underlying asset. In our study, while a suitable proxy for the FCSN program was not readily available (from a size perspective, FSCN represented the largest software investment/development effort to date), data obtained from the Joint Strike Fighter[7] (JSF) program (JSF software component was one-fifth the size of the FCSN program) were extrapolated and fitted accordingly to mirror the size of the FCS. These data were then utilized as a source of historical information for comparative purposes. The risk of requirements changes in the FCSN program was estimated to be 12 percent (as opposed to 1.44 percent for the JSF program, which is one-fifth the size of the FCSN program) using the Capers Jones formula shown below (Kulk & Verhoef, 2008).[8]

$$r = \left(\sqrt{\frac{SizeAtEnd}{SizeAtStart}} - 1 \right) .100 \qquad (2)$$

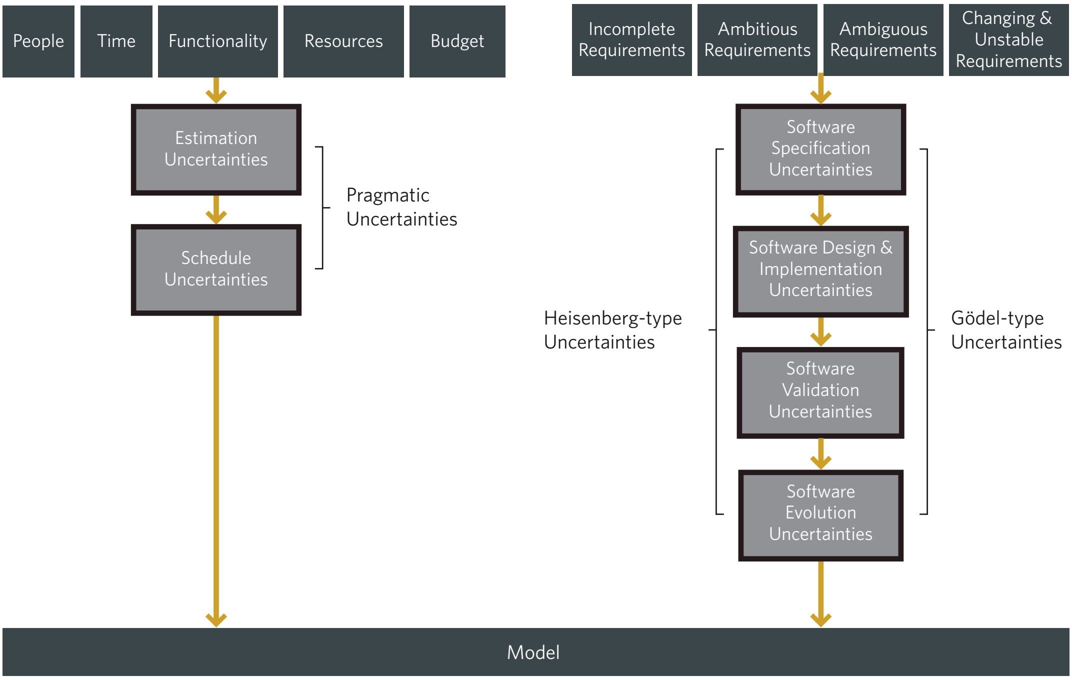
where t is the time period in years during which the estimates were observed.

The Capers Jones approach, which is a transposition from the financial industry, assumes requirements are compounded within a project and asserts that the method of average percentage of change of the overall requirements volume lacks information because it does not give any information on the time in which the change occurred. Determining time is an important, key factor in software engineering since requirements changes become more expensive to implement the further we are into the software development process.

Volatility Determination. Volatility is used to quantify the effect of the risk in the form of variations in the returns associated with the software investment, and the accuracy of its estimation is a key factor in real options valuation because it drives the value of an option and is positively related to value. While high volatility signifies high risk and implies a higher discount rate and lower value in traditional NPV valuation, a high volatility in real options analysis is linked to high-option value. This link results from greater volatility, which creates a wider range of possible future values of the opportunity as the option would only be exercised if the value of the opportunity exceeds the exercise price (Hevert, 2007).

Figure 5 depicts identified uncertainties, which were fed into a Monte Carlo model—Risk Simulator[9] software—taking into account interdependencies between both the technical and managerial uncertainties associated with the software acquisition effort. The software emulated all potential combinations and permutations of outcomes (i.e., to determine the effects of requirements volatility of 12 percent on integration, performance, scheduling, estimation, and its overall impact on the software acquisition effort). The analysis indicated that requirements volatility introduced an overall volatility of 0.0866 percent in the FCSN program. The volatility of 0.0866 percent resulted in a reduction in the NPV of the FCSN program from $6.4 trillion to $6.1 trillion. This reduction in NPV is a result of the potential of increased costs in light of the risks facing the FCSN

FIGURE 5. MODELING SOFTWARE ENGINEERING UNCERTAINTIES



Note. Adapted from *Real Options Analysis: Tools and Techniques for Valuing Strategic Investment and Decisions,* by J. Mun, 2006. The Risk Simulator software was developed by Mun.

program, which ultimately reduces the value of the investment effort from a financial point of view.

To improve/refine the accuracy of the volatility estimates, the Dempster-Shafer Theory of Evidence (DST)[10] is employed to provide increased belief, partial belief, ignorance, or conflict with the initial estimates (Arnborg & Högskolan, 2006). This is accomplished by establishing "belief functions" that reflect the "degrees of belief" between the revised NPV estimate, computed at $6.1 trillion in light of the risks posed by requirements uncertainty and the FCSN cost estimates provided by two independent sources—the Cost Analysis Improvement Group (CAIG) and the Institute of Defense Analyses (IDA) (Congressional Budget Office, 2006).

The independent belief functions based on the CAIG and IDA, which inferred basic probability assignments associated with each of the FCSN risk factors (requirements, integration, estimation risk, etc.), were combined using an orthogonal matrix to determine the most probable beliefs for the set of risk factors. Where the combined functions reflected "belief" in our estimates, our estimates were considered to be valid and were left untouched. In situations where the combined belief functions reflected conflict with our estimates, our estimates were revised accordingly to reflect the estimates computed using the DST approach. Further, we ran the Monte Carlo simulation with the revised risk estimates again, thus resulting in a "refined" volatility of 0.0947 percent. The derived volatility, which reflects an increase from the initial volatility estimate of 0.0866 percent, results in a further reduction of NPV in the FCSN program from $6.1 trillion to $5.7 trillion. This reduction implies a $7 billion shortfall ($6.4 trillion–$5.7 trillion) between the original and the refined NPV as a result of the volatility of the software investment. Details of the volatility computation can be found in Olagbemiro (2008, pp. 121–148).

Phase III: Options Analysis

This phase involves the identification of options. Once the volatility of the software acquisition effort has been determined, possible options could be identified to manage the risks associated with the software investment effort (Figure 6). In this study, three broad categories of options are explored relative to software acquisition.

1. Expand/Growth options
2. Wait/Deferment options
3. Contract/Switch/Abandon options

To take advantage of the options identified, the issue of software design is revisited. From a software architectural perspective,

FIGURE 6. SAMPLE OPTIONS TO ADDRESS SOFTWARE INVESTMENTS

Real Option Category	Real Option Type	Description and Example
Expand/ Growth	Scale up	Option to scale up through cost-effective, sequential investments as knowledge of the product increases
	Switch up	A flexibility option to switch products, processes, given a shift in underlying price of input and output demands
	Scope up	Investment in proprietary assets of one industry enables company to enter another industry cost effectively – Link and Leverage
Wait/ Defer	Study/Start	Delay investment until more information or skill is acquired, e.g., introduction of new requirements
Contract/ Switch/ Abandon	Scale down	Shrink or shut down a project partway through if new information changes the expected payoffs, e.g., introduction of new requirements
	Switch down	Switch to more cost-effective and flexible assets as new information is obtained, e.g., switch from custom development to Commercial Off-the-Shelf
	Scope down	Limit scope of (or abandon) software project when there is no further potential in the business opportunity the software is meant to address

Note. Adapted from *Real Options Analysis: Tools and Techniques for Valuing Strategic Investment and Decisions,* by J. Mun, 2006.

the decomposition of the software into components, modules, or subsystems serves to introduce flexibility from which the program manager could exploit and benefit. Since the software design is a key activity aimed at conceiving how a software solution would solve a particular problem, factoring modular decomposition into the design would support the following two propositions (Damodaran, 2002, pp. 796–815):

1. Some projects that look attractive on a full-investment basis may become even more attractive if the project is partitioned or decomposed into components because we are able to reduce downside risk at the lowest possible level.
2. Some projects that are unattractive on a full-investment basis may be value-creating if the firm can invest in stages.

A successful modular decomposition would introduce flexibility into the acquisition process by recasting the software effort as a series of options to start, stop, expand, or defer the development of a module or subsystem when requirements uncertainty is encountered. Note that the FCSN software effort has been decomposed into six components: Combat Identification, Battle Command and Mission Execution, Network Management System, Small Unmanned

Ground Vehicle, Training Common Component, and System of Systems Common Operating Environment (GAO, 2008b, pp. 2–31). The FCSN software development effort could be recast as a series of deferment/learning options and investment/growth options. Such options may include start, stop, scale down (e.g., staff), reallocate resources, or resume development when uncertainty is resolved; or defer development in the face of requirements uncertainty. This whole strategy is based on the correct partitioning/decomposition of the FCSN into the appropriate systems or subsystems.

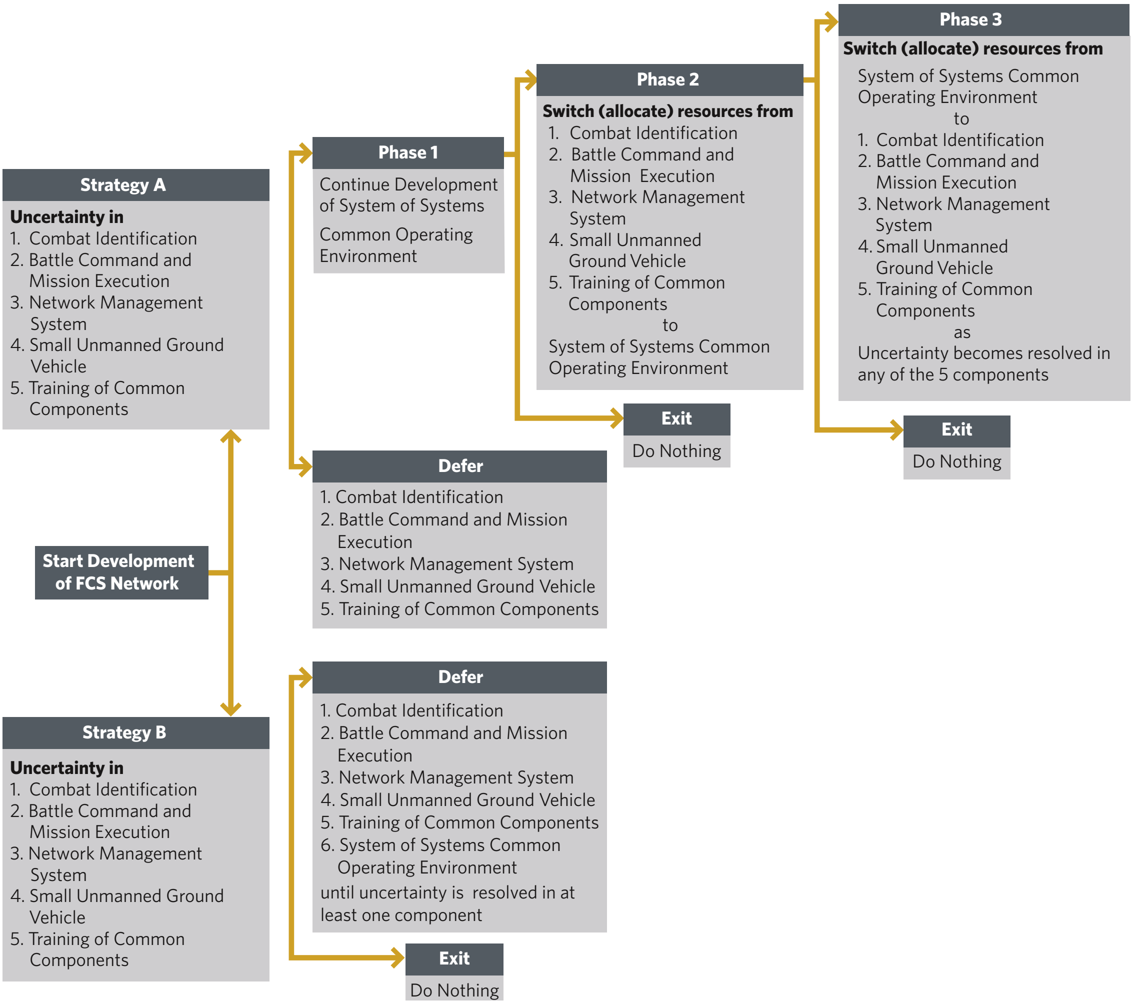
To highlight this strategy, we present a scenario.

Scenario: At least one out of the six software components is not facing requirements uncertainty. In this scenario, we assume that of the six component systems, one is not facing any form of uncertainty, while five of the software components are facing uncertainty. We proceed to develop different options to address this scenario. For our study, we examine two possible options: compound option and deferment option.

Compound Option. In the event that at least one of the software components is not facing requirements uncertainty, while all the others are facing requirements uncertainty, an option could be developed to *scale down* the resources/staff allocated to the software components facing requirements uncertainty. The staff could then be *switched* to work on the software component that is not facing requirements uncertainty, while the uncertainties in the other components are addressed using our uncertainty elicitation model. (Note: The assumption with this approach is the software component development effort upon which the staff engineers are being reallocated to work is not already behind schedule and hence does not violate *Brooks Law.*)[11] If the development effort upon which the staff are being assigned to work is late (behind schedule), the number of staff, experience level, and role that the added staff would play in the software development effort must be taken into consideration. We therefore framed the real options in this case as: *an option to contract and scale down* from an uncertain system, an *option to switch* resources to another system, and *options to expand and scale up* staff assigned to the development of a system not facing uncertainty (shown as Strategy A in Figure 7). This is essentially a *compound option*—an option whose "exercise" is contingent on the execution of the preceding option.

Deferment Option. In the event that five out of the six software components are facing requirements uncertainty, then an option could be developed to *stop and defer all development*, including the

FIGURE 7. FCS STRATEGY TREE DEPICTING STRATEGY A AND B FOR GIVEN SCENARIO



development of the software component that is not facing requirements uncertainty for a specified period until uncertainty is resolved (shown as Strategy B in Figure 7). This is an option to *wait and defer*.

Phase IV: Options Valuation

Valuation plays a central part in any acquisition analysis. Options are usually valued based on the likelihood of the execution of the options. Several methods are available for computing and valuing real options, such as employing the use of closed-form models, partial differential equations, or lattices. For our study, we utilize the binomial approach and apply risk-neutral probabilities as this method elicits great appeal due to its simplicity, ease of use, and the ability to solve all forms of customized real-life options.

We utilize the Real Options Super Lattice Solver (SLS) 3.0 software developed by Real Options Valuation, Inc., for the task. The basic inputs are presented in the Table.

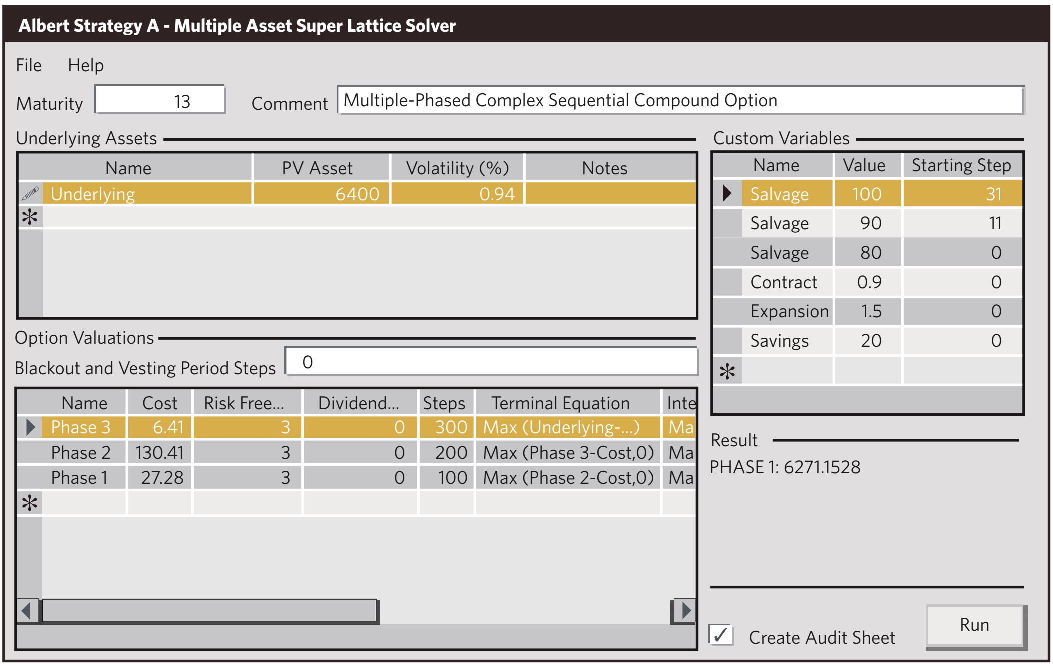
TABLE. REAL OPTIONS SUPER LATTICE SOLVER (SLS) 3.0 INPUTS

Symbol	Real Option on Software Acquisition Project	Description
S	Value of Underlying Asset (Asset Price)	Current value of expected cash flows (expected benefits realized from investing in the software effort [NPV])
K	Exercise Price/Strike Price	Price at which the created option would be realized (investment cost, or cost of investing in options, which is an estimation of the likely costs of accommodating changes)
T	Time-to-Expiration	The useful life of the option (time until the opportunity disappears/maturity date of the option contract)
r	Risk-Free Interest Rate	Risk-free interest rate relative to budget and schedule (interest rate on U.S. Treasury bonds)
cv	Volatility	Uncertainty of the project value and fluctuations in the value of the requirements over a specified period of time (volatility in requirements, cost estimation, and schedule estimation based on Dempster-Shafer Theory of Evidence)

Strategy A. The Real Options SLS software was populated (Figure 8) based on the following underlying values:

1. Development/Implementation cost of FCSN is $163.7 billion.
2. Value of underlying asset is $6.4 trillion.
3. The risk-free rate is 3.0 percent.
4. Volatility of our project is 0.0947.
5. Duration of software development is 13 years.
6. Lattice steps were set to 300.

FIGURE 8. SCREEN CAPTURE OF AUTHORS' MODEL IN THE REAL OPTIONS SLS SOFTWARE



The model was executed, and the lattice of the underlying asset (FCSN) (Figure 9) as well as the options valuation lattice for Strategy A (Figure 10), was created. The terminal values in our lattices (apex of lattice) are the computed values that occur at maturity, while the intermediate values in the lattices are the computations that occur at all periods leading up to maturity. All these values are computed using backward induction.

The option analysis that represents the value of the option under Strategy A returned a value of $6.27 trillion (Figure 10). The options valuation lattice of each phase under Strategy A was created and values computed using backward induction, working backward from Phase III to Phase I to arrive at the results depicted in Figure 10.

Strategy B. In Strategy B, which calls for a "defer and wait" approach, an assumption is made that the duration for deferment option would be 3 years. We set up our model (Figure 11) using the same assumptions used in Strategy A, but set the duration of the deferment option to 3 years.

The model is executed and similar to Strategy A; the lattice of the underlying asset (Figure 12) is generated. In contrast, the option analysis returned a value of $6.25 trillion (Figure 13).

FIGURE 9. LATTICE OF UNDERLYING ASSET (FCS NETWORK)

										6526.47
									6513.71	
								6500.97		6500.97
							6488.27		6488.27	
						6475.58		6475.58		6475.58
					6462.92		6462.92		6462.92	
				6450.29		6450.29		6450.29		6450.29
			6437.68		6437.68		6437.68		6437.68	
		6425.10		6425.10		6425.10		6425.10		6425.10
	6412.54		6412.54		6412.54		6412.54		6412.54	
6400.00		6400.00		6400.00		6400.00		6400.00		6400.00
	6387.49		6387.49		6387.49		6387.49		6387.49	
		6375.00		6375.00		6375.00		6375.00		6375.00
			6362.54		6362.54		6362.54		6362.54	
				6350.10		6350.10		6350.10		6350.10
					6337.69		6337.69		6337.69	
						6325.30		6325.30		6325.30
							6312.93		6312.93	
								6300.59		6300.59
									6288.28	
										6275.98

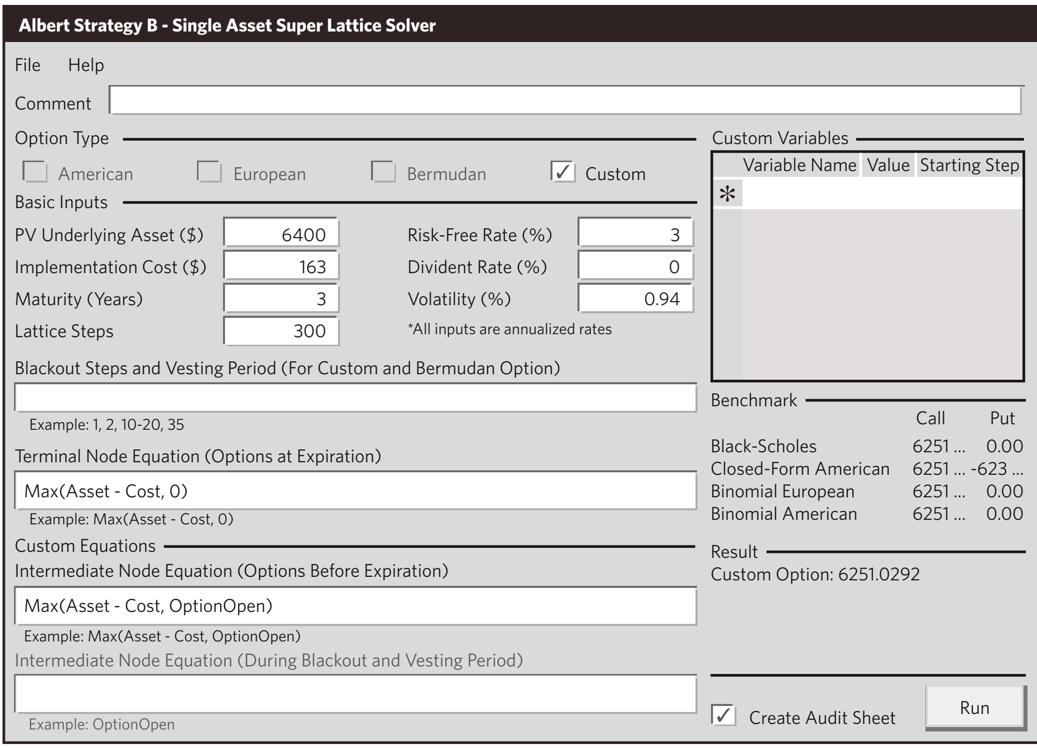
FIGURE 10. PHASE 1 (STRATEGY A) OPTION VALUATION LATTICE

										6395.93
									6383.34	
								6370.78		6370.44
							6358.24		6357.90	
						6345.73		6345.39		6345.05
					6333.24		6332.90		6332.56	
				6320.77		6320.43		6320.10		6319.76
			6308.33		6307.99		6307.66		6307.32	
		6295.91		6295.58		6295.24		6294.90		6294.56
	6283.52		6283.18		6282.85		6282.51		6282.17	
6271.15		6270.82		6270.48		6270.14		6269.81		6269.47
	6258.47		6258.14		6257.80		6257.46		6257.13	
		6245.82		6245.48		6245.15		6244.81		6244.47
			6233.19		6232.85		6232.52		6232.18	
				6220.58		6220.25		6219.91		6219.57
					6208.00		6207.66		6207.33	
						6195.44		6195.11		6194.77
							6182.91		6182.57	
								6170.40		6170.06
									6157.91	
										6145.45

Phase V: Investment Valuation

Given the option value of $6.27 trillion under Strategy A, the intrinsic value of the compound option is determined to be $6.27 trillion – $5.7 trillion = $570 billion. Under Strategy B, the intrinsic value of the deferment option is determined to be $6.25 trillion – $5.7 trillion = $550 billion. This implies that under both Strategies A and B, the program manager should be *willing* to pay no more than (and hopefully less than) the option premium of $570 billion and $550 billion respectively. This amount, in addition to the initial investment cost of $163.7 billion, should increase the chances of receiving the initially projected NPV of $6.4 trillion for the FCSN as

FIGURE 11. OPTIONS SUPER LATTICE SOLVER DEFERMENT MODEL



opposed to the current $5.7 trillion in light of the risks caused by the uncertainties in five of the six software components. This premium would also include the administrative costs associated with exercising an option from an integrated logistics support point of view, i.e., costs associated with contractual agreements, software development retooling costs, costs associated with infrastructure setup of the infrastructure, etc.

In analyzing both strategies, Strategy A is more attractive than Strategy B. Instead of waiting for another 3 years at an additional cost of up to $550 billion (after which uncertainty would hopefully have been resolved) and then proceeding to spend $163.7 billion at once to develop all six software components, the staged-phase approach in Strategy A calls for budgeting up to $570 billion for the option up front. The staged-phase approach also calls for spending some $163.7 billion for the System of Systems Common Operating Environment component, and then investing more over time as the requirements are firmed up for the other five components. Therefore, under these conditions, Strategy A, which employs the compound sequential options, is the optimal approach.

FIGURE 12. STRATEGY B UNDERLYING ASSET VALUE LATTICE (FCS NETWORK)

										6460.44
									6454.37	
								6448.31		6448.31
							6442.25		6442.25	
						6436.20		6436.20		6436.20
					6430.15		6430.15		6430.15	
				6424.11		6424.11		6424.11		6424.11
			6418.07		6418.07		6418.07		6418.07	
		6412.04		6412.04		6412.04		6412.04		6412.04
	6406.02		6406.02		6406.02		6406.02		6406.02	
6400.00		6400.00		6400.00		6400.00		6400.00		6400.00
	6393.99		6393.99		6393.99		6393.99		6393.99	
		6387.98		6387.98		6387.98		6387.98		6387.98
			6381.98		6381.98		6381.98		6381.98	
				6375.98		6375.98		6375.98		6375.98
					6369.99		6369.99		6369.99	
						6364.01		6364.01		6364.01
							6358.03		6358.03	
								6352.05		6352.05
									6346.08	
										6340.12

FIGURE 13. STRATEGY B OPTIONS VALUATION LATTICE UNDER DEFERMENT

										6311.03
									6305.00	
								6298.98		6298.89
							6292.97		6292.88	
						6286.86		6286.87		6286.78
					6280.96		6280.87		6280.78	
				6274.96		6274.87		6274.78		6274.69
			6268.97		6268.88		6268.79		6268.70	
		6262.98		6262.89		6262.80		6262.71		6262.62
	6257.00		6256.91		6256.82		6256.73		6256.65	
6251.03		6250.94		6250.85		6250.76		6250.67		6250.58
	6244.97		6244.88		6244.79		6244.70		6244.61	
		6238.92		6238.83		6238.74		6238.65		6238.56
			6232.87		6232.78		6232.69		6232.60	
				6226.83		6226.74		6226.65		6226.56
					6220.80		6220.71		6220.62	
						6214.77		6214.68		6214.59
							6208.74		6208.65	
								6202.72		6202.63
									6196.71	
										6190.70

Phase VI: Execution

The execution phase deals with the last precondition of real options valuation theory, which asserts that decision makers must be smart enough to execute the real options when it becomes optimal to do so. The options premium has two main components: intrinsic value and time value, both of which contribute to the valuation of the underlying software investment. For example, assuming that the contract for the FCSN includes an option for Strategy A, program managers must then be willing to exercise the compound sequential option when they observe that five of the six software components are at risk due to uncertainties.

Discussion

Our proposed approach addresses the risks associated with software-related capital investments by taking a proactive approach towards risk management by emphasizing the planning for, and paying for risk up front. This is not to say that risk management strategies are not being adopted today, but rather highlights a failure of management to take a strategic approach towards risk management. The status quo emphasizes the employment of what is deemed to be a "tactical" approach in the form of the spiral development process, which results in the elimination/reduction of much needed functionality from the scope of the software investment effort—usually when the acquisition effort is already in the development phase. Therefore, the proposed methodology in this article would help address some of the limitations of the spiral development process by serving as a mechanism through which the much desired and needed planning associated with the spiral development process is provided.

Conclusions

Uncertainties associated with software-related capital investments lead to unnecessary and sometimes preventable risks. As DoD often sets optimistic requirements for weapons programs that require new and unproven technologies, the application of the real options valuation methodology would be beneficial as it would enable the DoD to incorporate the appropriate *strategic options* into acquisition contracts. The options would serve as a contract between the software executive and the contractor (in the case of a government acquisition) to buy or sell a specific capability known as the options on the underlying project. The real options valuation approach is able to overcome the limitations of traditional valuation techniques by utilizing the best features of traditional approaches and extending their capabilities under the auspices of managerial flexibility. Barring the use of an explicit uncertainty elicitation phase as proposed in our research, and the development of options to hedge against the risk—and ultimately execute the options as they appear—we believe the current acquisition process would continue to be plagued by the risks of cost and schedule overruns.

The cost-reduction strategy of reducing testing resources proposed by DoD on the JSF program, while risky in itself, still did not address the root causes of cost-related increases as identified in GAO Report No. 08-569T (GAO, 2008c, pp. 2–17), further underscoring the importance of a preemptive and strategic approach of identifying uncertainties early on in an acquisition effort and paying for risk up front. By employing our proposed approach, the DoD would be able to optimize the value of their strategic investment decisions by evaluating several decision paths under certain conditions to lead to the optimal investment strategy.

Authors' Note

This work was supported in part by the Naval Postgraduate School Acquisition Research Program, OUSD_08 (Project #:F08-023, JON: RGB58).

Author Biographies

Capt Albert Olagbemiro, U.S. Air Force Reserve, is the Operations Officer, 69th Aerial Port Squadron, 459th Air Refueling Wing, Andrews AFB, Maryland. He is a career logistics readiness officer and is a graduate of the Advanced Logistics Readiness Officer Course. He received his PhD in Software Engineering from the Naval Postgraduate School. He also holds both an MS in Computer Science and an MBA from the Johns Hopkins University.

(E-mail address: albert.olagbemiro@afncr.af.mil)

Dr. Johnathan Mun is a research professor at the Naval Postgraduate School and Founder/CEO of Real Options Valuation, Inc., a software training and consulting firm in Silicon Valley, focusing on advanced risk analytics, risk simulation, stochastic forecasting, portfolio optimization, strategic real options analysis, and general business modeling analytics. He received his PhD and MBA from Lehigh University and holds other financial charters and certifications in risk and financial analysis.

(E-mail address: jcmun@realoptionsvaluation.com)

Dr. Man-Tak Shing is an associate professor at the Naval Postgraduate School. His research interests include the engineering of software-intensive systems. He is on the program committees of several software engineering conferences and is a member of the Steering Committee of the Institute of Electrical and Electronics Engineers (IEEE) International Rapid System Prototyping Symposium. He received his PhD in Computer Science from the University of California–San Diego, and is a senior member of IEEE.

(E-mail address: shing@nps.edu)

REFERENCES

Arnborg, S., & Högskolan, K. T. (2006, July). Robust Bayesianism: Relation to evidence theory. *Journal of Advances in Information Fusion, 1*(1), 1–16.

Congressional Budget Office. (2006). Statement of J. Michael Gilmore, Assistant Director, The Army's Future Combat Systems Program, before the Subcommittee on Tactical Air and Land Forces Committee on Armed Services, U.S. House of Representatives. Washington, DC: U.S. Government Printing Office.

Damodaran, A. (2002). *Tools and techniques for determining the value of an asset* (2nd ed.). Hoboken, NJ: John Wiley & Sons.

Department of Defense. (2000, November). *Report of the Defense Science Board task force on defense software.* Washington, DC: Office of the Under Secretary of Defense for Acquisition and Technology.

Dixit, A. K., & Pindyck, R. S. (1995, May-June). The options approach to capital investment. *Harvard Business Review, 73*(3), 105–115.

Erdogmus, H., & Vandergraaf, J. (2004). *Quantitative approaches for assessing the value of COTS-centric development.* Ottawa, Canada: Institute for Information Technology, Software Engineering Group National Research Council of Canada.

Hevert, K. (2007, March). Real options: Valuing flexibility in strategic investments. *Babson Insight.* Wellesley, MA: Babson College.

Kulk, V., & Verhoef, C. (2008). Quantifying requirements volatility effects. *Science of Computer Programming, 72*(3), 136–175.

Mun, J. (2006). *Real options analysis: Tools and techniques for valuing strategic investment and decisions* (2nd ed.). Hoboken, NJ: John Wiley & Sons.

Olagbemiro, A. (2008). *Application of real options theory to software engineering for strategic decision making in software-related capital investments.* (Doctoral dissertation). Naval Postgraduate School, Monterey, CA.

U.S. Government Accountability Office. (2008a). *Defense acquisitions. Assessments of selected weapon programs* (GAO Report No. 08-467SP). Washington, DC: U.S. Government Printing Office.

U.S. Government Accountability Office. (2008b). *Defense acquisitions. Report to congressional committees: Significant challenges ahead in developing and demonstrating Future Combat System's Network and software* (GAO Report No. 08-409). Washington, DC: U.S. Government Printing Office.

U.S. Government Accountability Office. (2008c). *Joint Strike Fighter: Impact of recent decisions on program risks* (GAO Report No. 08-569T). Washington, DC.: U.S. Government Printing Office.

ENDNOTES

1. Pragmatic uncertainties are problems in actually performing the development activities.
2. Heisenberg-type uncertainties occur as the system is being developed, grow during use, and exhibit themselves in the form of changing requirements due to unsatisfactory behavior post-implementation.
3. Gödel-like uncertainties occur when the properties of a program cannot be known from the representation because the software systems and their specifications are abstract models of the real world.
4. The NPV valuation approach is still utilized because the real options approach "builds" on traditional methods such as the NPV by incorporating strategic flexibility in the form of "options."
5. NPV of $6.4 trillion is computed based on: (a) value of the FCSN program (future value less operating costs, i.e., sum of [C – M] was $10 trillion; (b) initial development cost I was $163.7 billion; (c) r is 3 percent; and (d) time t to develop the FCSN is 13 years.
6. The Delphi method is a subjective estimation methodology based on the elicitation of the opinion of an expert or groups of experts to guide decision making by predicting future events.
7. At the time of this study, the JSF software acquisition effort represented the largest development effort after the FCSN.
8. The requirements volatility of 12 percent was computed based on start and ending SLOC (Source Lines of Code) for the FCSN program. SLOC is used for demonstration purposes only. A more suitable metric such as function points is recommended.
9. The Risk Simulator software was developed by Johnathan Mun.
10. DST is a mathematical theory of evidence/generalization of probability theory where probabilities are assigned to sets as opposed to mutually exclusive and exhaustive propositions termed "singletons." Information from multiple sources can be combined in the form of belief assignments, which serves to aggregate the information with respect to its constituent parts.
11. Brooks Law states that adding people to a late project makes it later.