NPS-CS-09-005

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

Application of Real Options Theory to DoD Software Acquisitions

by

A. Olagbemiro, J. Mun, and M. Shing

20 February 2009

Approved for public release; distribution is unlimited

Prepared for: Acquisition Research Program - OUSD_08
Graduate School of Business and Public Policy
Naval Postgraduate School
Monterey, CA 93943

THIS PAGE INTENTIONALLY LEFT BLANK

NAVAL POSTGRADUATE SCHOOL
Monterey, California 93943-5000

Daniel T. Oliver
President

Leonard A. Ferrari
Executive Vice President and Provost

This report was prepared for the Naval Postgraduate School and funded by the NPS Acquisition Research Program - OUSD_08.

This report was prepared by:

____
Man-Tak Shing
Associate Professor of Computer Science
Naval Postgraduate School

Reviewed by:

____
Peter J. Denning, Chairman
Department of Computer Science

Released by:

____
Karl van Bibber
Dean of Research

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** February 2009	**3. REPORT TYPE AND DATES COVERED** Technical Report (Oct 2007 – Dec 2008)	
4. TITLE AND SUBTITLE: Title (Mix case letters) Application of Real Options Theory to DoD Software Acquisitions			**5. FUNDING NUMBERS** **Project #:F08-023, DWAM80020**
6. AUTHOR(S) A. Olagbermiro, J. Mun, and M. Shing			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER** NPS-CS-09-005
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) Acquisition Research Program - OUSD_08 Graduate School of Business and Public Policy Naval Postgraduate School Monterey, CA 93943			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this report are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			**12b. DISTRIBUTION CODE**
13. ABSTRACT (maximum 200 words) 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.			
14. SUBJECT TERMS Real Options, Strategic Investments, Software Acquisitions, Risk Management, Software Engineering			**15. NUMBER OF PAGES** 43
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

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

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

In the U.S. Department of Defense (DoD), technology acquisitions in the form of software-intensive weapons systems serves as the cornerstone of the transformation strategy currently adopted by the U.S. Military in its efforts to modernize its fleet of weapons systems for future conflicts. However, the benefits of these force "enablers" continue to be plagued by massive cost and schedule overruns. The resulting impact has often led to a reduction in the scope of desired functionality as depicted in Table 1, leaving the war-fighters' needs unfulfilled.

Program	Initial Investment	Initial Quantity	Latest Investment	Latest Quantity	% Unit Cost Increase	% Quantity Decrease
Joint Strike Fighter	$189.8 billion	2,866 aircraft	$206.3 billion	2,459 aircraft	26.7	14.2
Future Combat Systems	$92 billion	18 System	$163.7 billion	14 systems	54.4	77.7
F-22A Raptor	$81.1 billion	648 aircraft	$65.4 billion	181 aircraft	188.7	72.1

Table 1: Program Management Failures of Top Three Major Weapons Systems [1]

The software component plays a critical role in the success of each of these acquisition programs. As it stands today, software is the major expense in the acquisition of software-intensive systems with its role as a *technology platform*, rising from providing a mere 8% of weapons systems functionality in 1960 to over 80% of functionality in 2000 (Fields, 2008) (Figure 1).

[1] Numbers were compiled from various GAO reports and were current as of 2007.

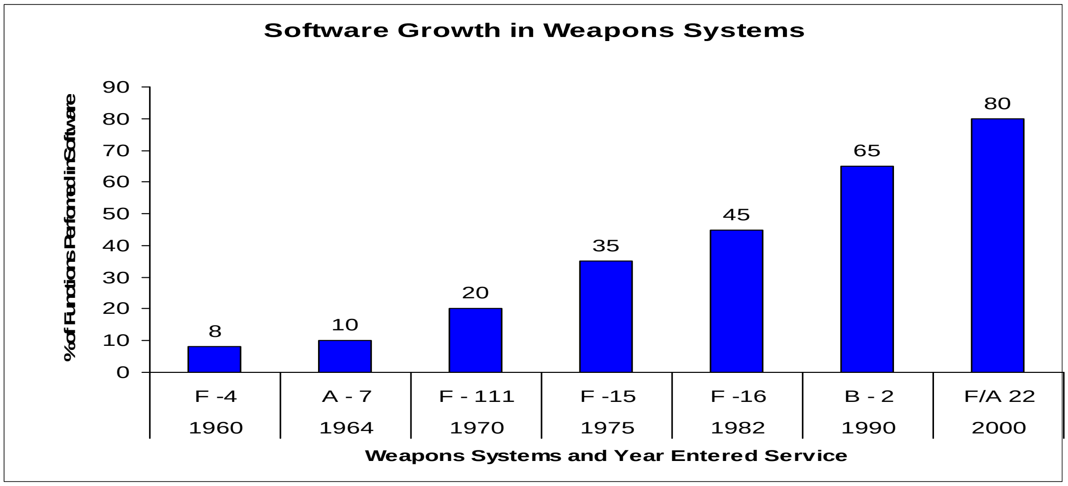

Figure 1: Software Growth in Weapons Systems **(Fields, 2008)**

Considering the immense presence and ever-increasing role which software plays in weapons systems, software is, and should be, treated as a capital investment, and 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 overcome the limitations of the spiral development process currently utilized in the Evolutionary Acquisition (EA) approach as adopted in the DoD 5000 series acquisition directives—it assumes the end-state requirements are known at the inception of the development process (Sylvester & Ferrara, 2003), albeit a misrepresentation of reality in the acquisition of DoD software-intensive weapons systems. The spiral development process is a risk-driven development approach consisting of four main phases namely: determining objectives/alternatives, risk analysis, development and planning. The phases are iteratively followed one after the other building progressively on the first iteration until a complete software product is built. Of the four phases, the risk analysis phase is the most important because the project's success is highly dependent on the ability to identify and resolve risk. Risks are continuously discovered and high-priority risks drive the development process. However, addressing risk at the development phase is a somewhat costly approach.

THIS PAGE INTENTIONALLY LEFT BLANK

II. METHODS

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.

A. 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 acquisitions decisions can be valued as a portfolio of options or choices in real "assets", much akin to options on financial securities which have real economic value under uncertainty (Dixit & Pindyck, 1995). In contrast to financial options, real options valuation centers on real or non-financial assets, and is valuable because it enables the option holder (software program manager) 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. Real options are implicit or explicit capabilities created for real assets that provide the software manager 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 software manager may choose to adjust, reduce, increase, or abandon the investment in the future, thereby stabilizing returns from these assets.

A necessary and key tenet of the real options approach is a requirement for the presence of uncertainties, an inherent characteristic of software acquisitions decision-making. Software acquisitions encapsulate the activities related to software procurement, development, implementation, and subsequent maintenance. The uncertainties which surround these activities are compounded by increasingly complex requirements demanded by the warfighter and present themselves in various forms ranging from changing or incomplete requirements, insufficient knowledge of the problem domain, decisions related to the future growth, technology maturation and evolution of the software.

To tackle the issue, 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 a strategic point of view. While this phase would not include members of a typical requirements team, they would work in tandem with the requirements team, 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 that 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.

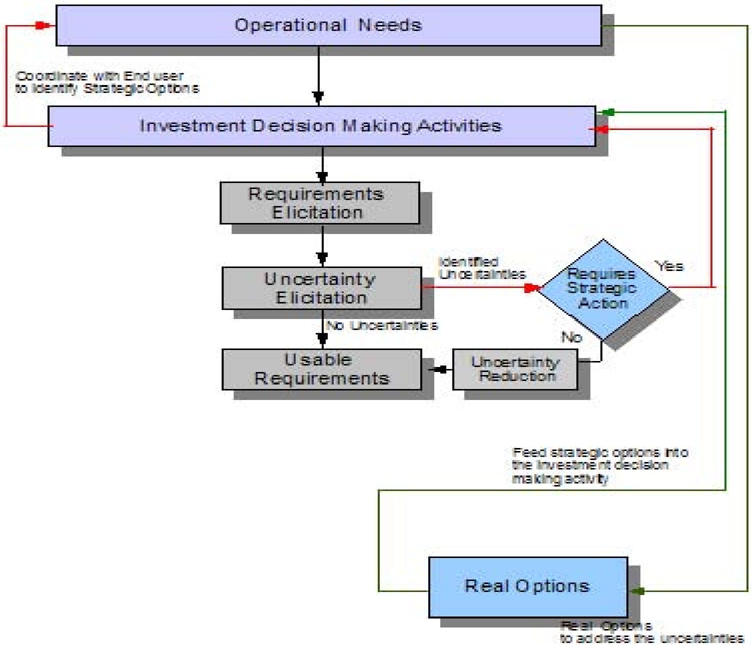

Figure 2: Uncertainty Elicitation Model

During the uncertainty elicitation step in the model, uncertainties are captured from two perspectives (the managerial and technical perspective) using what we call the "2 **T**" approach 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[2]. Technical uncertainties of incomplete requirements, 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[3] and Gödel-like[4] uncertainties.

---

2 Pragmatic uncertainties are problems in actually performing the development activities.

3 Heisenberg-type uncertainties occur as the system is being developed and grows during use and exhibit themselves in the form of changing requirements either due to unsatisfactory behavior post implementation.

4 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.

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.

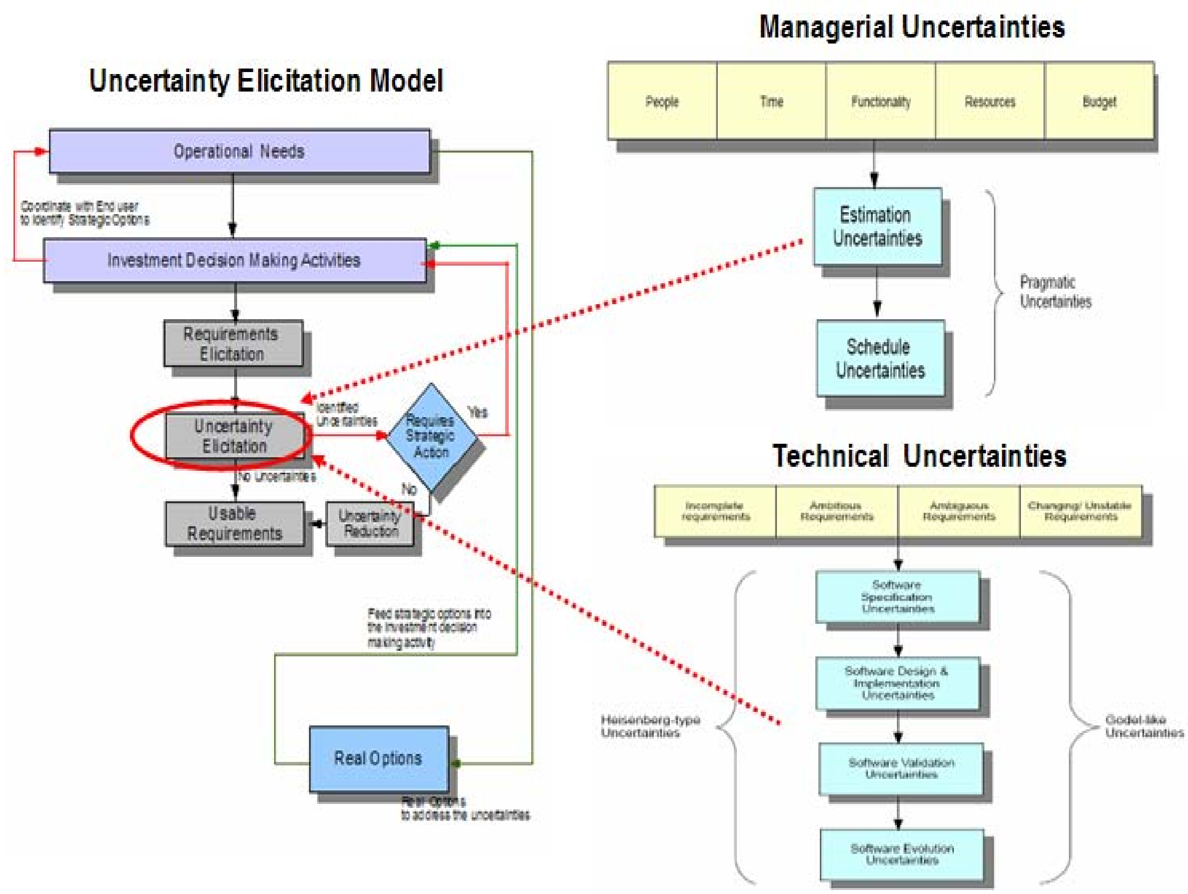

Figure 3: Expanded View of Uncertainty Elicitation Model

B. 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 develop a generalized Real

Options framework (Figure 4) in line with the 5 preconditions outlined in (Mun, 2006). This proposed framework consists of the following six phases each of which explicitly addresses and establishes compliance with the preconditions.

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

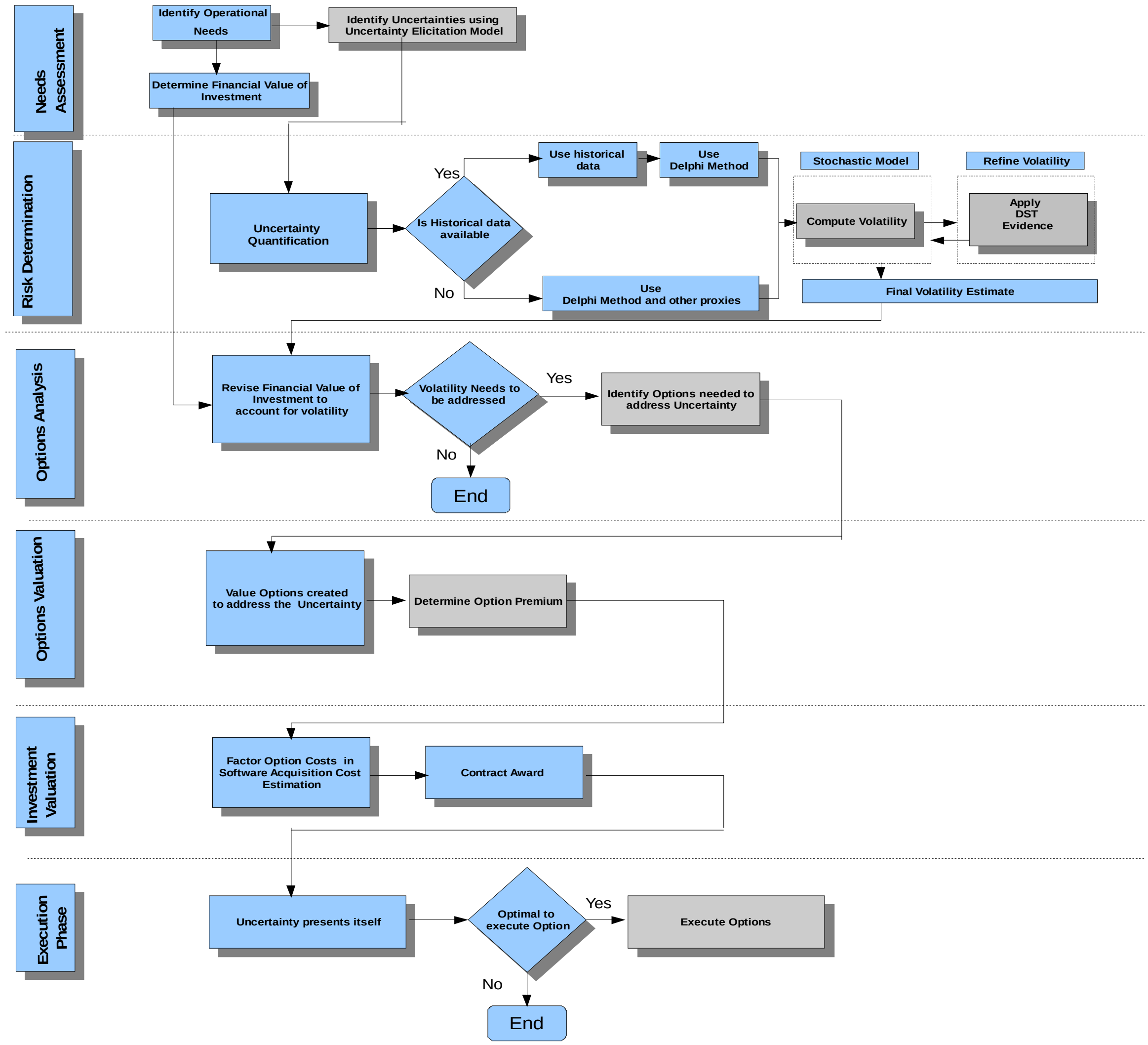

Figure 4: Real Options Framework

III. SAMPLE APPLICATION OF THE FRAMEWORK

This section provides a sample application of the framework using a software component, the Future Combat Systems Network (FCSN), of the U.S. Army Future Combat Systems program (Congressional Budget Office, 2006).

A. PHASE I: NEEDS ASSESSMENT

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

$$NPV = \sum \frac{(C_t - M_t)}{(1+r)^t} - I$$

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*).

A NPV of $6.4 trillion was computed for the FCSN using estimated values based on key assumptions in (Olagbemiro, 2008)[5].

(b) 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 *technology maturation* issues (GAO Report 08-467sp, 2008) plagued the FCSN program and introduced several other corresponding uncertainties. Thus the following uncertainties were determined to have been retroactively predictable.

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

---

[5] NPV of $6.4 trillion is computed based on a 1) Value of the FCSN program, (future value less operating costs. i.e. sum of (C – M) was $10 trillion), 2) Initial development cost I was $163.7 billion, 3) r is 3%, 4) Time t to develop the FCSN is 13 years.

B. PHASE II: RISK DETERMINATION

The risk determination phase consists of two steps: (a) *uncertainty* quantification and (b) *volatility* determination.

(a) Uncertainty Quantification: Risk implies uncertainty, and 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 is utilized. The objective of the evidence gathering activity is to equate the software engineering uncertainties of the current investment effort to a quantifiable property (risk factor) based on historical evidence in order 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, data obtained from the Joint Strike Fighter program was fitted and utilized as a source of historical information for comparative purposes. The risk of requirements changes in the FCSN program was estimated to be 12% (as oppose to 0.28% for the JSF program which is one fifth the size of the FCSN program) using the Capers Jones formula shown below (Kulk & Verhoef, 2008) [6].

$$r = \left(\sqrt[t]{\frac{SizeAtEnd}{SizeAtStart}} - 1 \right) \cdot 100.$$

[6] The requirements volatility of 12% was computed based on start and ending SLOC for the FCSN program. SLOC is used for demonstration purposes only. A more suitable metric such as function points is recommended.

(b) Volatility Determination: Volatility is used to quantify the effect of the risk in the form of variations in the returns associated with the investment. Volatility accuracy is a key factor in real options valuation because it drives the value of a real 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 because greater volatility 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, 2008). A Monte Carlo simulation of the risk model (Figure 5) was run using the Risk Simulator software, taking into account interdependencies between the risk variables to emulate all potential combinations and permutations of outcomes. The analysis indicated that requirements volatility introduced an overall volatility of 0.0866% in the FCSN program. The volatility of 0.0866% resulted in a reduction in the NPV of the FCSN program from $6.4 Trillion to $6.1 Trillion. This reduction in NPV is as a result of the potential of increased costs in light of the risks facing the FCSN program, which ultimately reduces the value of the investment effort from a financial point of view.

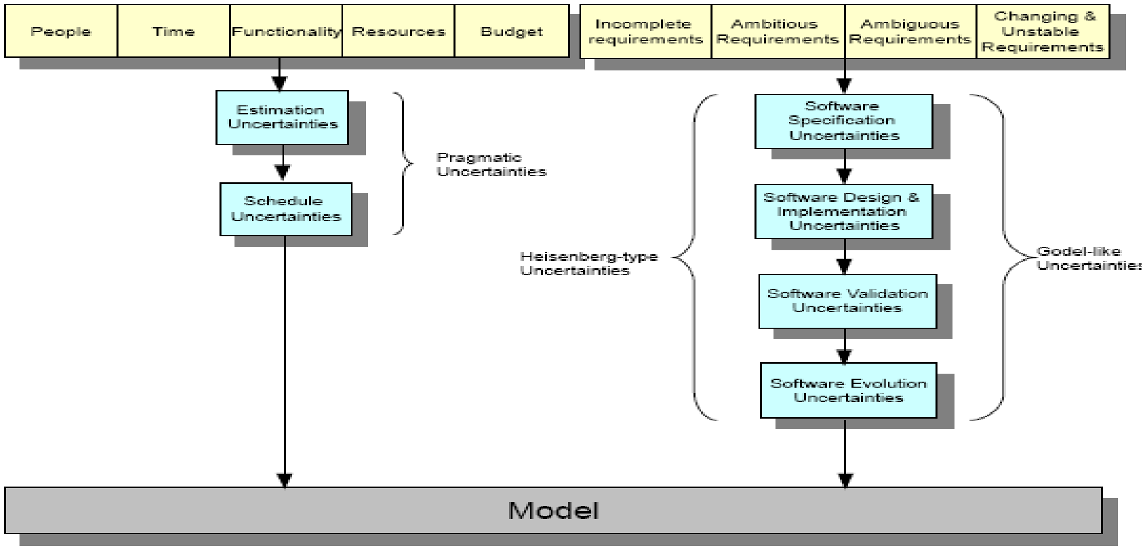

Figure 5: Modeling Software Engineering Uncertainties[7]

To improve the accuracy of the volatility estimates, we chose to refine the volatility using the Dempster Shaffer Theory of Evidence (DST) (Arnborg**,** Kungliga & Hogskolan, 2006) which aims to provide increased belief, partial belief, ignorance or conflict with our initial estimates. This is accomplished by establishing "belief functions" that reflect the "degrees of belief" between our NPV estimates 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 Analysis (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, and in situations where

[7] Both the Managerial and Technical uncertainties are fed into a risk model and epistemic and aleotoric uncertainties characterized from the inputs.

the combined belief functions reflected conflict with our estimates, our estimates were revised accordingly, to reflect the estimates computed using the DST approach and we run the Monte Carlo simulation of the model with the revised risk estimates again. Based on the risk of requirements uncertainty[8] presented in the FCSN, a resulting “refined” volatility of 0.0947% was obtained. The derived volatility which reflects an increase from the initial volatility estimate of 0.0866% results in a further reduction of NPV of the FCSN program from $6.1 Trillion to $5.7 Trillion. Details of the computation can be found in (Olagbemiro 2008).

C. PHASE III: OPTIONS ANALYSIS

This phase involves the identification of options. Once the volatility of the software investment 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 acquisitions.

1) Expand/Growth options

2) Wait/Deferment options

3) Contract/Switch/Abandon options.

[8] While there are several risk factor based on the technical and managerial uncertainties, we focus on requirements risk due to its overwhelming impact on the FCSN.

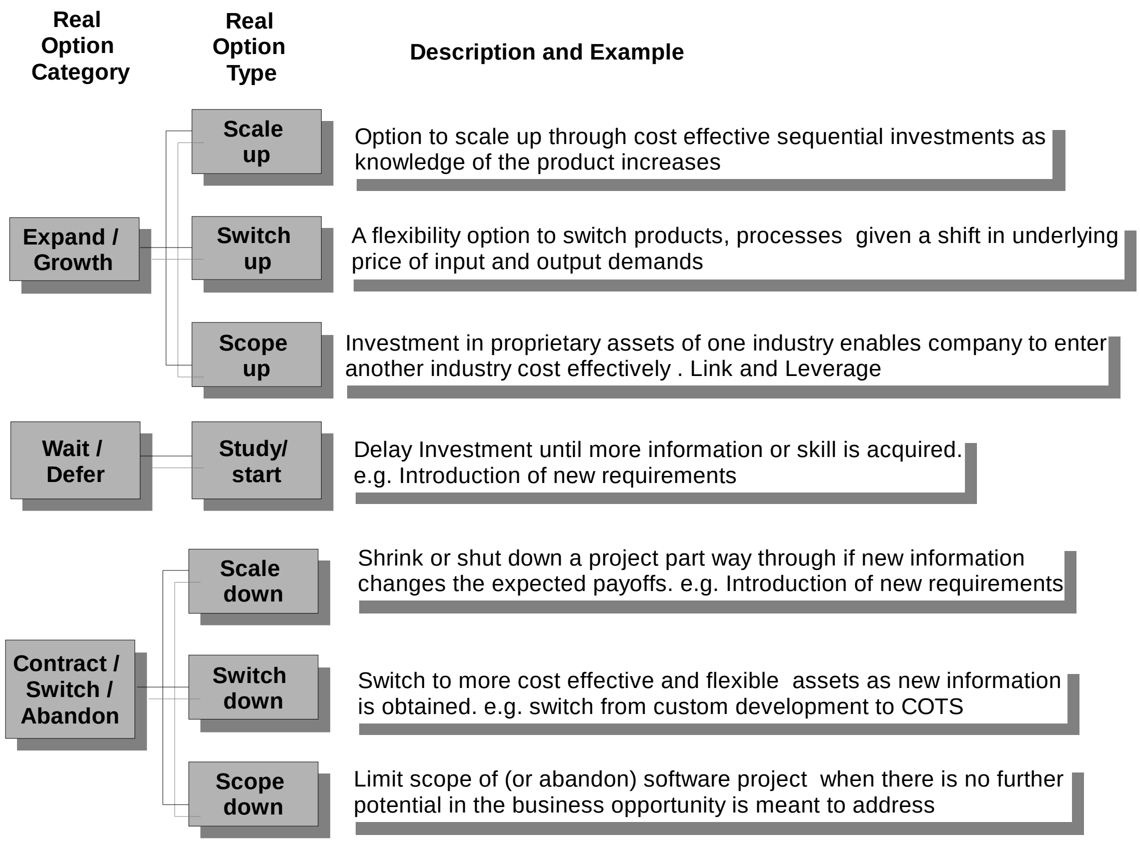

Figure 6: Sample Options to Address Software Investments **(Mun, 2006)**

To take advantage of the options identified, the issue of software design is revisited. From a real options perspective, the decomposition of the software into components, modules or subsystems serves to introduce flexibility which the software executive or program manager could exploit and benefit from. As 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, 2007)

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. Given that the FCS software effort has been decomposed into the following six components: Combat Identification, Battle Command and Mission Execution, Network Management System, Small Unmanned Ground Vehicle, Training Common Component, and Systems of Systems Common Operating Environment (GAO Report 08-409, 2008), the FCS software development effort could be recast as a series of Deferment/Learning Options and Investment/Growth Options during which the option to Start, Stop, Scale Down staff, and Reallocate Resources, and Resume Development when uncertainty is resolved or Defer Development in the face of requirements uncertainty is utilized. This whole strategy is based on the correct partitioning/decomposition of the FCS Network into the appropriate systems or subsystems.

To highlight this strategy, we present a scenario.

Scenario 1: At least one out of the 6 software components is not facing requirements uncertainty

In this scenario, we assume that of the six component systems, one is not facing 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 1) Compound Option 2) Deferment Option

Compound Option

In the event that at least one of the software components is not facing requirements uncertainty, with all the others 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 which the staff engineers are being reallocated to work on is not already behind schedule and hence does not violate *Brooks Law*[9]). If the development effort which the staff are being assigned to work on is late (behind schedule), the number of staff, experience level and role which the added staff would play in the software development effort must be taken into consideration. We therefore frame the real options in this case as: an *Option to Contract and Scale Down* from an uncertain system, *Option to Switch* resources to another system, *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.

[9] Brooks law states that adding people to a late project makes it later.

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* to include the 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*.

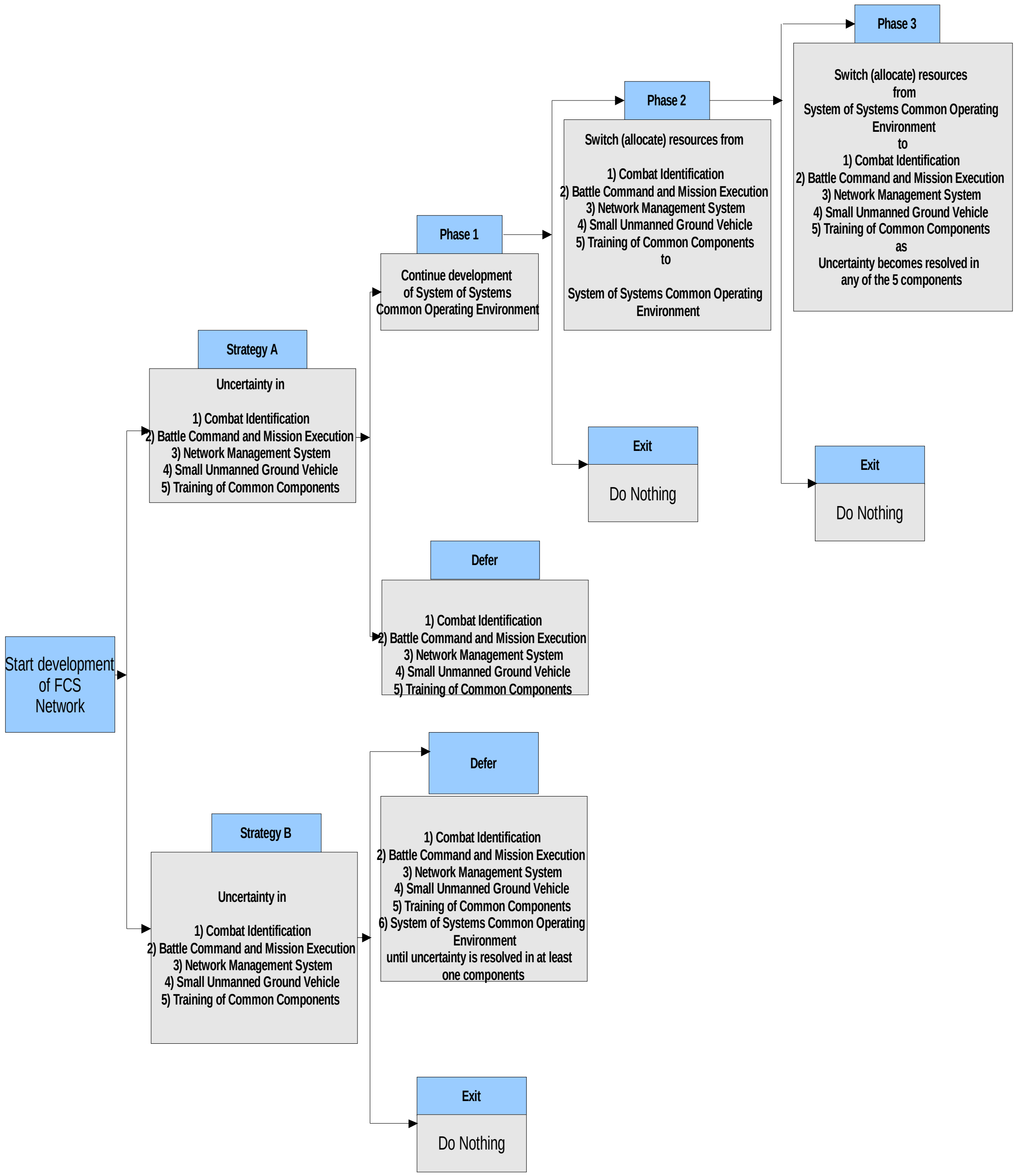

Figure 7: FCS Strategy Tree depicting Strategy A and B for given Scenario

D. 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. There are several methods for computing and valuing real options such as employing the use of closed-form models, partial differential equations, lattices, and so forth. For our study, we utilize the binomial approach and apply risk-neutral probabilities as this method elicits great appeal due to its simplicity, ease to 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 Table 2.

Symbol	Real Option on Software Acquisitions 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, 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 US 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 DST of Evidence)

Table 2: Real Options SLS Inputs

Strategy A

The Real Options SLS software was populated 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%
4. Volatility of our project is 0.0947
5. Duration of software development is 13 years
6. Lattice steps was set to 300

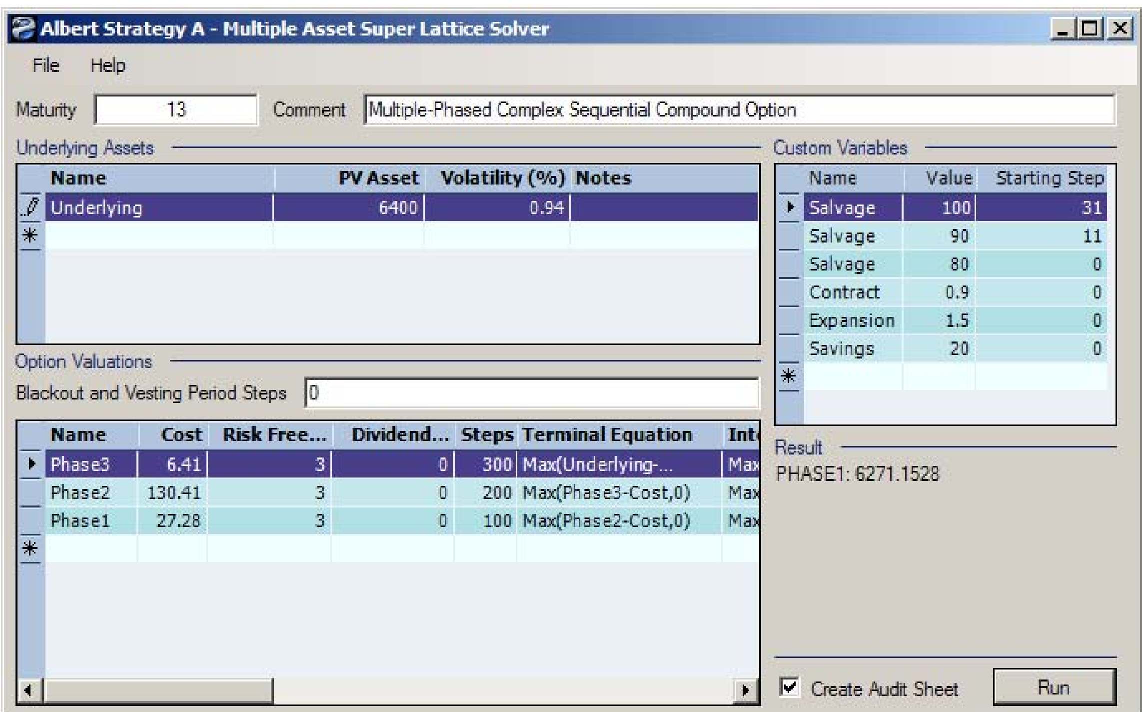

Figure 8: Screen Shot of our Model in the Real Options SLS software

The model was executed and the lattice of the underlying asset (FCSN) (Figure 9) as well as the Option Valuation lattice for (Figure 10) Strategy A 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.

FCSN Underlying Asset Value Lattice

										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 9: Lattice of Underlying Asset (FCS Network)

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

Figure 10: Phase 1 Option Valuation Lattice

The value of the underlying asset was computed as $6.4 trillion (Figure 9). The option analysis which represents the value of the option under Strategy A returned a value of $6.27 trillion (Figure 10). The option valuation lattice of each phase under strategy A was created and values computed using backward induction working from backwards from Phase 3 to Phase 1 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.

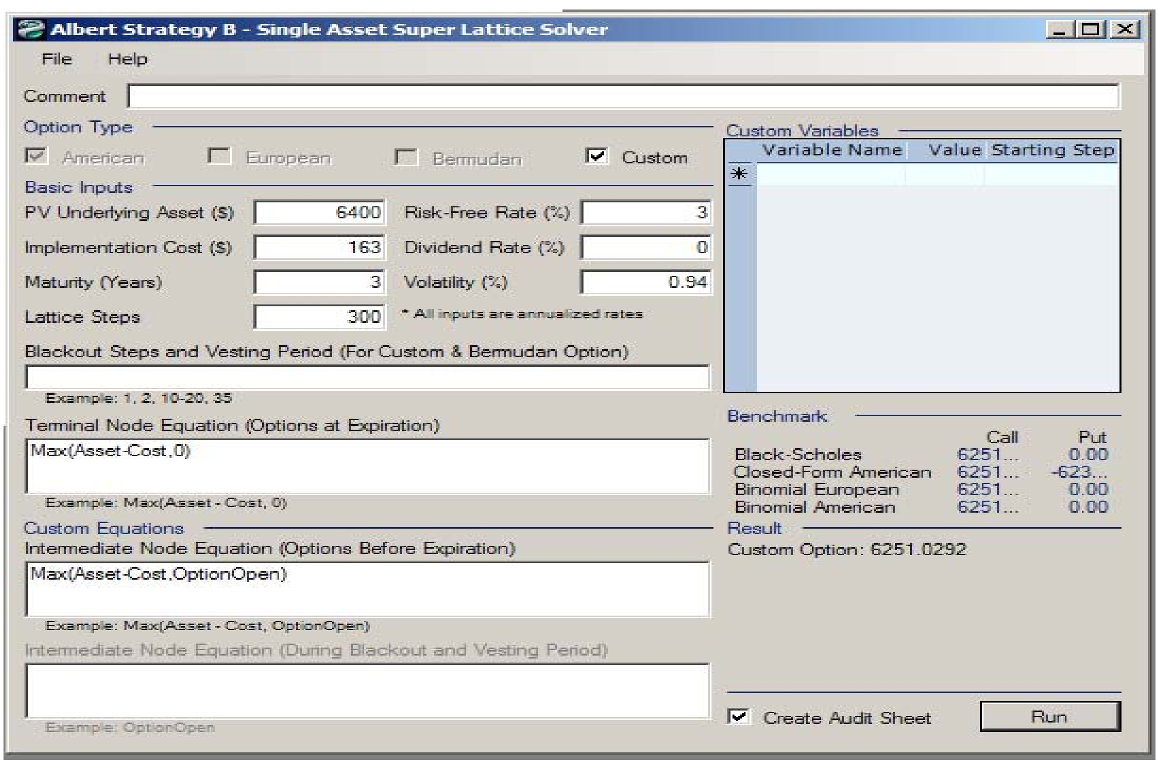

Figure 11: Real Options Super Lattice Solver Deferment Model

The model is executed and similar to strategy A, the value of the underlying asset was computed as \$6.4 trillion (Figure 12). In contrast, the option analysis returned a value of \$6.25 trillion (Figure 13).

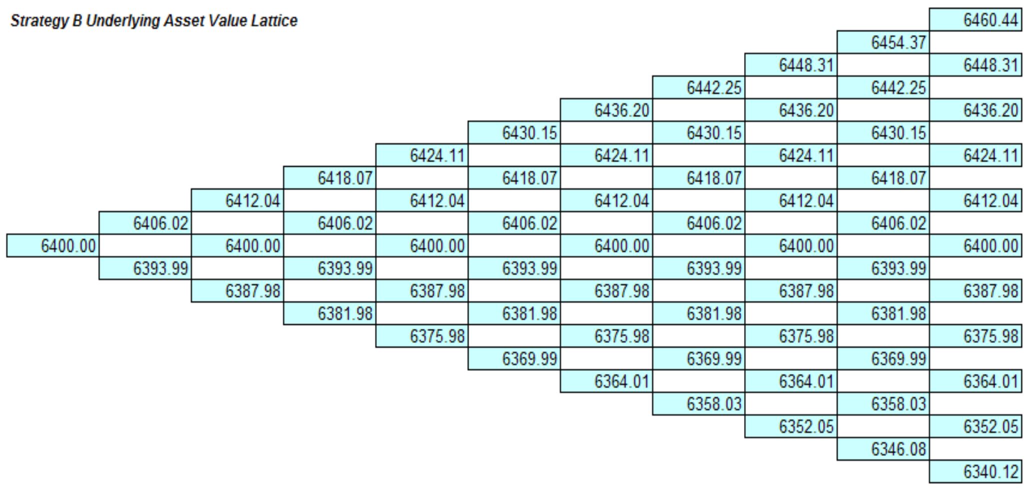

Figure 12: Lattice of Underlying Asset (FCS Network)

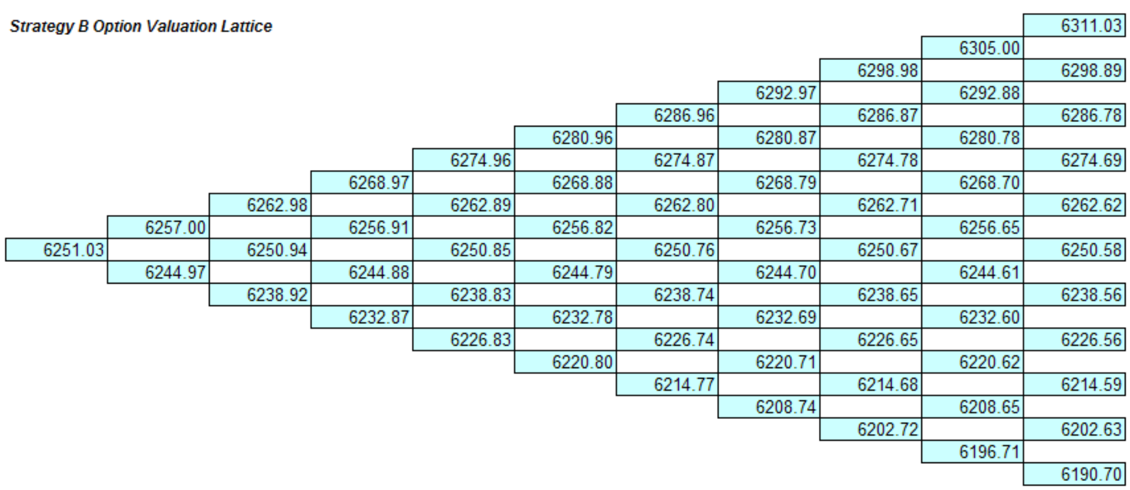

Figure 13: Options Valuation Lattice under Deferment

E. 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 $6.4 trillion – $6.27 trillion = $130 billion. Under strategy B, the intrinsic value of the deferment option is determined to be $6.4 trillion – $6.25 trillion = $150 billion. This implies is that under both strategies A and B, the software executive should be *willing* to pay no more than (and hopefully less than) the option premium of $130 billion and $150 billion respectively in addition to the initial investment cost of $163.7 billion to increase the chances of receiving the initially projected NPV of $6.4 trillion for the FCSN as 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 $150 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 spending up to $130 billion for the option up front plus some of the $163.7 billion for the Systems 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.

VI. 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, assume that the contract for the FCSN includes an option for strategy A, then the software executive must be willing to exercise the compound sequential option when s/he observes that five of the six software components are at risk due to uncertainties.

III. CONCLUSION

The current risk management strategy of reducing risk by employing the spiral development process is not sufficient because it assumes the end-state of requirements are known and takes a reactive approach in dealing with the arising risks. 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 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 report 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.

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 the 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 resource currently proposed by DoD on the Joint Strike Fighter program, while risky in itself, still does not address the root causes of cost related increases as identified in [GAO Report 08-569T, 2008], further underscoring the importance of a preemptive and strategic approach of identifying uncertainties early on in a acquisition effort and paying for risk upfront. 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.

As part of the future work in connection with this research, we would like to formalize and create an automated software acquisition decision-making tool explicitly aimed at managing the risks associated with software-related capital investments using out Real Options approach. Specifically, we would like to gather historical information on previously completed software acquisition programs depicting the number of requirements planned at the onset of the acquisition effort and the number of requirements delivered at the end of the software acquisition effort, as well as the associated cost and schedule information for each of the acquisition programs. We would use all of this data to create a repository of historical programs which would serve as a basis of comparison with current/future acquisition programs to help provide some insight into the issue of requirements volatility and its associated impact on cost and schedule overruns. By gathering historical information into once centralized repository, we hope to alleviate the assumptions we made in our study due to data gathering problems we encountered in this study. We would incorporate the DST volatility refinement technique into our software tool and link our automated software acquisition decision making tool to the repository containing historical data of previously completed software acquisition programs to provide a one “stop-shop” modeling toolkit to better facilitate the acquisition decision making process.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Arnborg, S., and Hogskolan, K. (2006, July). Robust Bayesianism: Relation to Evidence Theory. *Journal of Advances in Information Fusion Vol. 1, No. 1.*

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

Damodaran, A. (2007). *Investment Valuation: The Options To Expand And Abandon: Valuation Implications.* 2nd Edition, Chapter 29. John Wiley & Sons, Inc.

Dixit, A.K. and Pindyck, R.S. (1995, May-June). The options approach to capital investment. *Harvard Business Review,*

Erdogmus, H. (1999, May 17). Valuation of Complex Options in Software Development, *ICSE'99 Workshop on Economics Driven Software Engineering Research* (EDSER1), Los Angeles, CA

Erdogmus, H. and Vandergraaf, J. (2004, October). Quantitative Approaches for Assessing the Value of COTS-centric Development. *Institute for Information Technology, Software Engineering Group National Research Council of Canada*

Fields, C. (2000, November). Task Force On Defense Software. *Defense Science Board*.

Hevert, K. (2007, March). Real Options: Valuing Flexibility in Strategic Investments, *Babson Insight*

Kulk, V. (2008). Quantifying requirements volatility effects. *Science of Computer Programming.*

Mun, J. (2006) *Real Options Analysis: Tools and Techniques for Valuing Strategic Investment and Decisions*. 2nd Edition (Wiley Finance)

Olagbemiro, A. (2008). Application of Real Options Theory to Software Engineering for Strategic Decision Making in Software Related Capital Investments, PhD Dissertation, Naval Postgraduate School, Monterey, California.

Options Industry Council. (2006) What is an Option? *Investor Observer.com*

Sylvester, R., and Ferrara, J. (2003, Winter). Conflict And Ambiguity Implementing Evolutionary Acquisition", *Acquisition quarterly review,* U.S. General Accounting Office. (2008). *Defense Acquisitions, Assessments Selected Weapon Programs*. (U.S. GAO Report GAO 08-467sp). Washington, DC. U.S. Government Printing Office.

U.S. General Accounting Office. (2008). *Defense Acquisitions, Report to Congressional committees: Significant Challenges Ahead in Developing and Demonstrating Future Combat System's Network and Software*. (U.S. GAO Report GAO Report 08-409). Washington, DC. U.S. Government Printing Office.

U.S. General Accounting Office. (2008). *Joint Strike Fighter: Impact of Recent Decisions on Program Risks*. (U.S. GAO Report GAO Report 08-569T). Washington, DC. U.S. Government Printing Office

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. RADM James B. Greene, USN (Ret)
   Graduate School of Business and Public Policy
   Naval Postgraduate School
   Monterey, California

4. Dr. Keith Snider
   Graduate School of Business and Public Policy
   Naval Postgraduate School
   Monterey, California

5. Ms. Karey Shaffer
   Graduate School of Business and Public Policy
   Naval Postgraduate School
   Monterey, California

6. Dr. Peter Denning
   Department of Computer Science
   Naval Postgraduate School
   Monterey, California

7. Dr. Man-Tak Shing
   Department of Computer Science
   Naval Postgraduate School
   Monterey, California

8. Dr. Johnathan Mun
   Department of Information Sciences
   Naval Postgraduate School
   Monterey, California

9. Capt. Albert Olagbemiro
   22 Laurence Brooke Road
   Catonsville, Maryland