Valuation of Learning Options in Software Development Under Private and Market Risk

Erdogmus, Hakan

National Research Council Canada

Conseil national de recherches Canada

Institute for Information Technology

Institut de technologie de l’information

NRC-CNRC

Valuation of Learning Options in Software Development Under Private and Market Risk *

Erdogmus, H.

2002

* published in The Engineering Economist, Vol. 47, No. 3, pp 308-353. NRC 44962.

Canada

VALUATION OF LEARNING OPTIONS IN SOFTWARE DEVELOPMENT UNDER PRIVATE AND MARKET RISK

HAKAN ERDOGMUS
National Research Council, CANADA

ABSTRACT

Commercial software development is an inherently uncertain activity. Private risk is high, schedule and cost overruns are common, and market success is elusive. Such circumstances call for a disciplined project evaluation approach. This paper addresses the of use of market and earned value management data in assessing the economic value of commercial software development projects that are simultaneously subject to schedule, development cost, and market risk. The assessment is based on real options analysis, a financial valuation technique that can tackle dynamic investment decisions under uncertainty. The paper demonstrates the application of real options analysis to a development scenario that consists of two consecutive stages: a mandatory prototyping stage and an optional full-development stage. The full-development stage is undertaken only if the prototype is successful and the market outlook is sufficiently positive at the end of the prototyping stage, thus giving the full-development stage the flavor of an option. The project's staged design increases its value. Real options analysis captures the extra value due to optionality.

INTRODUCTION

According to the Standish Group's CHAOS report [1], more than 20% of software projects fail and only less than 20% of software projects are completed on time and on budget. The direct cost of these failures and overruns are staggering, but even more significant are probably the immeasurable effects of lost opportunity costs. Process improvement initiatives can help improve both predictability and quality, but they cannot eliminate uncertainty in the environment. Uncertainty is inherent in software development, where just about everything – requirements, technology, skills base, business climate, culture – is in a constant state of flux. A disciplined approach to reasoning about value under uncertainty is essential. Such an approach should:

- facilitate the identification of an optimal project structure that mitigates risk and increases value,
- be able to take advantage of current information and relevant historical data,
- support dynamic decisions, and ultimately,
- improve the quality of decision making at both the technical and managerial level.

Huchzermeier and Loch [2] identify five value drivers for R&D projects. Adapted to the software context, these are: product quality (subsumes performance), development budget (cost), development schedule (time), market requirement for quality, and market payoff. Uncertainty in one or a combination of these drivers affects the value of the project to various extents. This paper illustrates how uncertainty in these factors can simultaneously be tackled in software development using an integrated valuation approach. The approach takes project structure into account, uses available data, and supports dynamic decisions.

Budget and schedule are dependent drivers. Since labor is the dominating cost driver in software projects, cost overruns are frequently a consequence of schedule overruns. The treatment considers this dependency by relating budget uncertainty to schedule uncertainty, but can also treat budget uncertainty independently.

METHODOLOGY

Two techniques drive the valuation approach described in this paper: *real options analysis* and *earned value management.*

Real options analysis is a framework for valuing real assets under uncertainty. Earned value management is a technique for project planning and monitoring.

Real options analysis drives the design of the project structure so that the value of the project is maximized. The project is divided into distinct stages where the earlier stages are aimed at partially resolving project uncertainty and later stages *operationalize* the project and ultimately bring it to completion. As such, the earlier stages provide learning opportunities to steer the remainder of the project. When the project structure is viewed as a sequence of learning options, option-pricing techniques can be employed to value the overall project. The main insight provided by the real options approach is this: Staging gives rise to learning options that increase the flexibility of the project. The rational exercise of such options increases the value of the project by limiting the downside risk while retaining the upside potential.

Earned value management concepts are employed to capture the budget and schedule aspects of the project plan and to estimate budget and schedule risk based on historical data. Therefore, uncertainty about project costs and completion times can be factored into the analysis in a sound manner. Earned value management emphasizes the importance of project planning and tracking for the valuation of future projects that are subject to budget and schedule volatility. With rigorous project tracking, future decisions can be linked to past experience.

SUMMARY OF THE GENERAL VALUATION METHODOLOGY

The following steps illustrate the general methodology used:

1. Identify the critical sources of uncertainty.
2. Structure the project as a sequence of learning stages followed by an implementation stage where the learning stages are designed to partially resolve one or more sources of uncertainty. Each stage either represents an active strategy (with a required investment outlay, e.g., a prototyping stage) or a passive strategy (with no associated investment outlay, e.g., a waiting stage).
3. Determine the optimal continuation strategy for each stage conditional on the outcome of the previous stage. Each subsequent stage is viewed as a *real option* on the previous stage.
4. If the outcome of a stage is dependent on private risk – such as product quality, the ability of the product to meet market requirement, cost uncertainty or schedule uncertainty – use data from past projects to generate a probability distribution for the possible outcomes. If no data is available, treat the probability distribution as a sensitivity variable and repeat the analysis for different distributions.
5. Create a budget and schedule plan for each stage. The budget plan can also be conditional on the outcome of preceding stages.
6. Use earned value data from past projects to estimate budget and schedule uncertainty. Determine a set of possible completion times for the project.
7. Use the estimates of budget and schedule uncertainty to develop a conditional distribution of total development cost for each stage. The distribution is conditional on the deviation from the planned schedule

for that stage. It also takes into account the expected deviation from the budget plan independent of the schedule uncertainty.

8. Determine the project cutoff time. The cutoff time is the total project schedule such that the probability of completing the whole project later than this schedule is below a chosen threshold value.
9. Use market data to estimate the market payoff and its volatility. Select an interval size (granularity) and build a discrete, tree-based model of the potential development of the market payoff based on these estimates. The cutoff time determines the time horizon. If the market payoff depends on the project completion time, then construct a separate tree for each possible completion time.
10. Convert the market payoff tree into a decision tree. Introduce branches corresponding to different continuation strategies at the various decision points. Also introduce binary branches for potential delays at those points where the probability of delay is higher than a threshold value.
11. Assign a development cost and a market payoff value for each terminal node in the resulting tree depending on what terminal state the node represents.
12. Fold back the tree using dynamic programming and risk-neutral valuation to calculate the value at the root node. Subtract the initial investment, the cost the very first stage, from this value. The result represents the discounted net project value, and subsumes the option value resulting from the dynamic selection of optimal continuation strategies.

A LEARNING OPTION MODEL WITH MULTIPLE SOURCES OF UNCERTAINTY

The particular problem being addressed by the main section of the paper is the valuation of a learning real option subject to four different kinds of uncertainty. Three of the underlying sources of uncertainty involve private risks. Two of these are interdependent risks. The fourth source involves market risk and is independent of the other three sources.

For a given development project, let $S = \{S_1,..., S_k\}$ be a set of k possible schedules with an associated probability distribution. Let $DC = \{DC_1,..., DC_k\}$ denote the total development cost estimates for the k possible schedules in S and MP be the expected market payoff of the completed project. Both DC and MP are expressed in future value terms.

Let $L(T)$ be a learning option with an associated exercise, or *continuation*, decision at time T such that $T < S_i$, for all $1 \le i \le k$. $L(T)$ is a real option whose underlying asset is *MP* and whose exercise price is one of the DC_k, depending on which one of the possible schedules are realized upon the exercise of $L(T)$, and ultimately upon the completion of the project. If $L(T)$ is not exercised, the continuation decision is to stop, and the project is aborted with zero market payoff.

Suppose that the continuation decision of $L(T)$ is made based on a nonlinear payoff function f. The nonlinear payoff f is a function of the net payoffs distribution at time T and a binomial random variable V representing the likelihood of a favorable learning outcome. The function f is optimal in that it never yields a negative value, and in the case of a favorable learning outcome, extracts the highest value from the net payoff distribution. V constitutes an additional source of uncertainty, and behaves as a guard that destroys the learning option in the case of an unfavorable learning outcome. This scheme is consistent with the rational exercise principle of option pricing.

$L(T)$ is a learning option in that although the realized value of k is not known until the project's completion, the value of the random variable V and some information about the market payoff *MP* (as captured by the net payoff distribution at time T) are revealed just before T. Although for fixed k, the DC_k is certain, since k is not known a priori, the exercise cost is effectively uncertain. The underlying asset MP is also uncertain and follows a standard lognormal diffusion process. Thus $L(T)$ is a real option with an uncertain, lognormal underlying asset and an uncertain exercise price with an arbitrary discrete distribution. We assume that V and k are uncertain because of private factors and *MP* is uncertain because of market factors.

The main question tackled by the paper is this: What is the present value of the learning option $L(T)$ given S, DC, MP, and V?

A real option valuation technique originally suggested by Smith and Nau [3] is at the focal point of the solution. This technique, the *integrated rollback procedure*, is adapted to the problem at hand to handle the multiple sources and types of uncertainty involved. Integrated rollback procedure combines traditional decision-tree analysis with a standard option pricing method known as risk-neutral valuation. This method is based on a binomial, lattice model of the movement of the underlying asset. It assumes that the risk of the underlying asset can reasonably be priced using a market proxy, such that standard market theory and option pricing assumptions apply.

To solve the valuation problem for $L(T)$, first k parallel binomial lattices are constructed to represent the possible development of the market payoff *MP*. Since *MP* is specified in future value terms and it is realized only upon project completion, k models are required, each with a different initial price. The initial price is set to the present value of *MP*'s expectation, discounted back S_k periods using a risk-adjusted rate.

The lattice structure is converted into a decision tree with additional branches corresponding to the k possible schedules and the continuation decision. Each leaf node of the decision tree represents a combined state with two components:

(1) the realized schedule or continuation action, whichever applies, associated with that node and,

(2) the corresponding final state of the binomial lattice associated with the realized schedule.

A net payoff function is determined for each type of leaf node based on this composite state. The net payoffs are then computed for each leaf node according to the payoff function identified for that type of node.

The next step involves folding the decision tree back using a dynamic programming scheme. Three techniques are interleaved during this process. Whenever the continuation decision (an action) is involved, the expected payoff for the parent node is computed from the child nodes based on the nonlinear function f and the probability distribution of the learning outcome V. Whenever a state change is driven by a deviation from the current schedule, the expected payoff for the parent node is computed from the child nodes using the probability distribution of S. In both cases, the expected payoffs are discounted using the risk-free rate because both S and V are privately sourced. Finally, whenever a state change is driven by change in the value of the market payoff expectation, *risk-neutral valuation* is applied. Risk-neutral valuation involves the construction of a *replicating portfolio* composed of a risk-free asset and a market proxy for *MP*. The value of this portfolio is computed for the parent node such that its value at each of the child nodes equals the payoff previously computed for that child node. To eliminate arbitrage opportunities, the expected payoff of the parent node must equal the value of the underlying replicating portfolio at that node. This process is repeated recursively, selecting the proper rollback technique one step at a time depending on the type of the parent node. Ultimately, a value is obtained for the root node of the decision tree. The value of the root node equals the value of $L(T)$.

Smith and Nau prove that this procedure is sound so long as the decision maker is risk averse or risk-neutral. Here we assume that the decision maker is risk-neutral, with a linear utility function.

SCOPE OF THE METHODOLOGY

Some aspects subsumed within the different steps of the general methodology are beyond the scope of the paper.

- Budget and schedule estimation are not treated. Historical project data is used for estimating budget and schedule variability for future projects, but not for developing budget and schedule plans for them. Methods for developing budget and schedule plans are therefore not discussed. The reader interested in this aspect is referred to the general literature on software cost estimation [4, 5].
- Project tracking and measurement is not treated in detail. We assume that historical data from past projects *similar* to the one under evaluation is available in a certain format. In particular, the notion of project similarity is not formalized.
- The estimation of the market payoff is not treated in detail. We assume that relevant projections are provided together with a market proxy. The market proxy is used for estimating the variation around the given projections based on a standard uncertainty model.

BACKGROUND AND RELATED WORK

TRADITIONAL VALUATION

Traditional valuation techniques such as discounted cash flow and net present value are adequate for treating investment decisions with static project structures. However, these techniques are not particularly suitable for dealing with dynamic decisions under uncertainty. According to these techniques, the project's value is determined by the sum of its discounted cash flows, where the discount rate captures the opportunity cost of the investment. The discount rate is determined either by the market (based on the Capital Asset Pricing Model, where the expected return on an investment is proportional to the systematic risk borne) or by the minimum required return on investment for projects of similar risk (for example, by the weighted average cost of capital). Although private risk can be accounted for using unbiased estimates – expected period cash flows computed using a set of predetermined outcomes with subjective probabilities – there is no recognition of managerial flexibility once the project is underway.

Thus traditional valuation techniques fall short if the project is structured with milestones where the outcome of each milestone determines the continuation strategy for the rest of the project. Introductory corporate finance texts provide discussions of basic valuation concepts, such as discounted cash flow, net present value, and the Capital Asset Pricing Model. Sufficient coverage of these topics is provided by Ross et al. [6] and Brealey and Myers [7].

DECISION TREE ANALYSIS

The choice of the right discount rate is a contentious issue in discounted cash flow valuation. Decision theoretic approaches, decision tree analysis in particular [8], can handle dynamic aspects of valuation as well as risk preferences of decision makers, but they compound the discount rate problem. These approaches model a dynamic investment scenario as a tree with state nodes that represent possible outcomes in response to an action and action nodes that represent possible actions in response to an outcome. Probabilities are associated with state nodes. Contingent claims on future payoffs can easily be captured with this modeling technique. The decision tree is folded back by calculating expected payoffs using optimal decision rules and discounting these payoffs as one moves towards the root of the tree. However, as the tree branches, the risk structure of the project changes, thus requiring different discount rates to be used at different branches.

OPTION PRICING AND REAL OPTIONS ANALYSIS

Classical option pricing techniques tackle the discount rate problem by a different strategy. Instead of trying to solve the discount rate problem, option pricing relies on replicating the payoffs of a contingent claims scenario by an arbitrage-free trading strategy. The trading strategy involves a dynamically updated portfolio of assets. Therefore the discount rate problem in traditional decision tree analysis is reduced to finding a replicating trading strategy in option pricing. To avoid arbitrage opportunities, the value of the contingent claim must equal the value of the initial portfolio underlying the trading strategy.

Myers was one of the first to realize that financial option pricing techniques could be applied to the evaluation of projects [9]. He recognized that contingent claims on real assets have characteristics of options, and coined the term *real option*. This idea later lead to the development of a class of techniques, collectively referred to as *real options analysis,* for valuing real assets under uncertainty and dynamic decision-making,

In its most general form, an option is a discretionary future action. Options are a form of derivate, in that they their value depends on the value of some underlying asset, whether real or financial.

Viewing contingent claims on real assets, for example the milestones and different stages of a project, as options on an underlying asset gives rise to a powerful reasoning tool. The main insight is that the value of a capital budgeting project lies not only in the static stream of cash flows it is expected to generate, but also in the managerial flexibility embedded in the project. Such flexibility creates the potential to capitalize on the underlying uncertainties. Substantial value can be generated if those uncertainties are managed optimally. Using the real options perspective, option pricing techniques and decision tree analysis can effectively be combined to give rise to a robust valuation framework.

Many excellent, high-level articles discuss the use of option pricing theory in valuing options on real assets [10, 11]. Myers explains how the real options approach links strategy and finance [9]. Copeland and Antikarov [12] and Amram and Kulatalika [13] focus on applications of real options. Trigeorgis provide several applications in various industries [14, 15].

Applications of real options to information technology in general and software development in particular have also been tackled in many articles. Erdogmus and Favaro et al. address investment decisions in software development [16-21]. Sullivan et al. focus on applications to software design [22, 23] and Baldwin and Clark address applications to hardware design [24]. More generalized treatments in the context of information technology investments are also available. For examples, see articles by Taudes, Benaroch, and Dos Santos [25-29].

OPTION PRICING THEORY

Hull [30] provides an undergraduate-level overview of derivative securities, including options, the general techniques for their pricing, and derivative markets. Pindyck and Dixit [31] offer a deeper and more theoretical exposition of option pricing theory together with the econometric foundations of standard option pricing methods.

The seminal paper on option pricing is by Black and Scholes [32], which explains the original derivation of their (and Robert Merton's) Nobel-prize winning model. Cox, Ross, and Rubinstein [33] provide a much simpler derivation of the same model using the binomial model and the risk-neutral approach. The Black-Scholes model has many variations with closed-form solutions. The most relevant ones are discussed by Margrabe and Carr [34, 35]. Margrage derives a formula for the option to exchange two risky assets, of which the Black-Scholes model is a special case [34]. This formula can be used when the exercise cost of a standard call option is uncertain [17]. Carr provides a comprehensive discussion of Margrabe's formula and other, more complex variations, including compound options. Kumar [36] provides a compact discussion of the impact of volatility on option value.

Sundaram [37] gives the best exposition of the binomial model and risk-neutral valuation. Smith and Nau [3] explain the relationship between option pricing and decision tree analysis, and demonstrate how the two models together can account for market and private risk simultaneously.

EARNED VALUE MANAGEMENT

Earned value management is a well-known project planning and tracking technique. It was developed and is advocated by the Project Management Institute [38, 39]. The U.S. Department of Defense adopted and mandated earned value management for effective schedule and cost management of defense contracts. The application of earned value management to software development projects has drawn considerable attention since the early nineties [40-42].

In earned value management, all work is budgeted (planned), scheduled, and tracked in time-phased increments that constitute a baseline for planning and measurement. This baseline is referred to as the *project measurement baseline* (*PMB*). For each unit of work, and for the project as a whole, budgeted (planned) costs, actual work performed relative to budgeted costs, and actual (realized) costs are tracked and accumulated.

As work is performed, it *earns* value on the same basis as it was planned, in dollars or some other quantifiable unit such as person-hours. This is not business value per se, but rather value relative to budgeted work expressed as a percentage of the allocated budget. Thus earned value measures the volume of work accomplished relative to work planned. The difference between earned value and the actual costs measures the deviation from the planned budget. These deviations are tracked, and corrective action is taken when warranted.

Earned value data from past projects can aid in the estimation of cost and schedule uncertainty for future projects. In software development, this information is useful for project planning, valuation, and risk management.

FIGURE 1 illustrates the earned value terminology using the project measurement baseline of a completed project. The *PMB* tracks cumulative costs and progress as a function of time (schedule). (We will deviate slightly from the standard earned value terminology for clarity.)

- *Budgeted cost of work scheduled* (*BCWS*) refers to planned cost. The cumulative *BCWS* curve always terminates at 100% of the total planned schedule. The total planned schedule is referred to as the *budget at completion* (*BAC*). In FIGURE 1, time T represents the *schedule at completion* (*SAC*), which refers to the planned schedule.

- *Budgeted cost of work performed* (*BCWP*) tracks the so-called *earned value*. This is the relative value that a project earns as it progresses. The value is relative to the planned expenditures. The *BCWP* curve extends beyond the SAC if the project is delayed. It terminates before the SAC if the project is ahead of schedule. The project in Figure 1suffered a delay.
- *Actual cost of work performed* (*ACWP*) is the *realized* cost. The cumulative *ACWP* curve tracks actual costs in absolute terms.

As the project progresses, deviation from the planned schedule and budget can be tracked through different metrics. FIGURE 1 illustrates these metrics: *cost variance* (*CV*), *horizontal schedule variance* (*HSV*), and *vertical schedule variance* (*VSV*). The sign of these metrics indicate whether the project is suffering a budget or schedule overrun (indicated by negative values) or it is ahead of planned budget or schedule (indicated by positive values) at any given time.

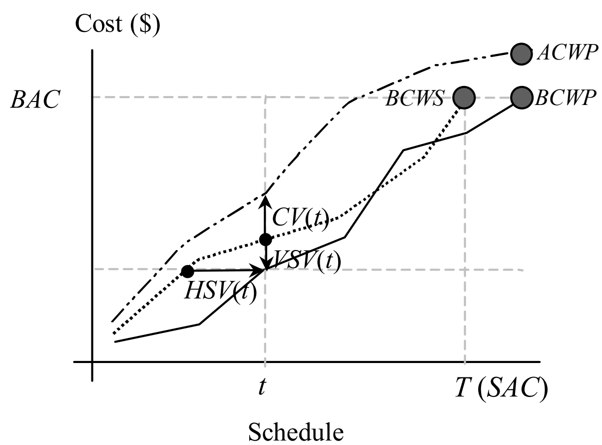

FIGURE 1: Earned value metrics that measure the deviation from cumulative budgeted schedule and cost: horizontal schedule variance (*HSV*), vertical schedule variance (*VSV*), and cost variance (*CV*).

CASE STUDY: A SOFTWARE PROTOTYPING PROJECT

The rest of the paper illustrates the valuation methodology outlined through a case study that involves a fictitious software company contemplating a major extension to its flagship product.

PROBLEM CONTEXT

Polysis, a developer of mathematical software, is considering the extension of its flagship product with a set of innovative user interface and automated analysis features. The new feature set is dubbed MathWizards. If the feature set is successful, the company expects to capture 5% of the total market in mathematical software. Polysis considers MathSoft, a public company, as its main competitor. The company analyst predicts that MathSoft currently holds 50% of the market targeted by Polysis.

Factors to Consider

The mathematical software market looks favorable, but is highly volatile. The company managers are reluctant to take a risk without a reasonable expectation that MathWizards will be well received by the end users.

Data from a similar past project indicates that development schedules and costs are uncertain. Budget risk and schedule risk are considered as private risk because they are unique to the organization.

Questions

How should Polysis proceed? What strategy should it adopt? How should it assess the value of its adopted strategy?

Selected Strategy

The driver for the following strategy is the market requirement risk as reflected by the statement "The company managers are reluctant to take a risk without a reasonable expectation that MathWizards will be well received by the end users." To reduce the risk of MathWizards failing in the market with actual end users, Polysis will undertake the development in two stages. The strategy of Polysis is illustrated in FIGURE 2:

- **Stage 1: Prototype Development and Evaluation.** First, Polysis will develop only a user interface prototype of MathWizards, without implementing the full functionality. If the prototype fails the usability evaluation, Polysis will shelve the idea, and the next version of its flagship product will be released without MathWizards. If the prototype passes the usability evaluation, Polysis will base its decision on the then outlook of the mathematical software market. The prototype will be ready in four months and cost $200,000 in present value terms to develop. The prototype will then undergo the usability evaluation, which will take an additional two months and cost $100,000 dollars. The total cost of the first stage is then $300,000 and the total schedule is six months.

- **Stage 2: Full Development.** At the end of Stage 1, company managers will decide whether to proceed with the full development of MathWizards and its integration with the company's flagship product. If the market outlook at that time is still positive, Stage 2 will proceed; otherwise it will be foregone. If the company decides to undertake Stage 2, it will have to incur the full development and integration cost. At the end of Stage 2, the company will release its flagship product with MathWizards, and only then will it reap the benefits from the project. Stage 2, if undertaken, is expected to be completed in 18 months, but this figure is subject to estimation errors and variability due to technical problems, skill set of the development team, turnover rate, and technological change.

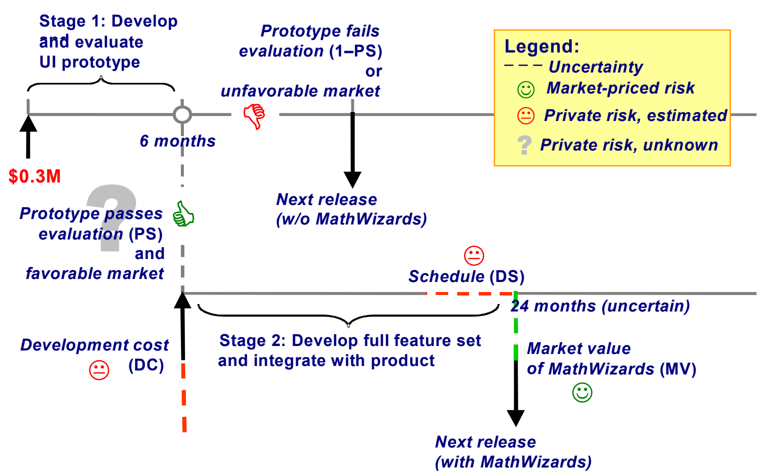

FIGURE 2. The development strategy of Polysis and the underlying sources of uncertainty.

ELABORATION OF THE SOURCES OF UNCERTAINTY

We assume that the prototyping cost and schedule estimates are fairly accurate. With this assumption, the strategy is subject to four remaining sources of uncertainty, two of which are dependent. The first three of these sources constitute private risk. The last one constitutes market risk, and will be priced using market data.

- **Success of Prototype (*PS*).** The probability of the MathWizards prototype passing the usability evaluation at the end of Stage 1. This probability is unknown a priori, and therefore *PS* will be treated as a sensitivity variable. Thus, (*PS*, 1 – *PS*) is the probability distribution of the binary learning outcome (the variable *V* in the general model).
- **Total Stage-2 Schedule (*DS*).** Total time (in calendar months) required to complete Stage 2. The planned schedule is 18 months for Stage 2, but this projection is uncertain. The uncertainty of this variable will be estimated using the *PMB* of a reference project from the company's database of past projects.
- **Total Stage-2 Cost (*DC*).** Total cost of Stage 2. The total expected cost for Stage 2 is one million dollars. The actual cost will be determined by the actual Stage-2 schedule. In addition to a possible deviation from the planned schedule, actual costs on a per-period (e.g., monthly) basis may turn out to be different from the estimated period costs. The reference project will be used to adjust the cost estimates.
- **Market Payoff (*MP*).** The future value of total revenues from the sales of MathWizards. This variable represents the market payoff of the completed project. The uncertainty of *MP* is determined by the market demand for mathematical software products. *MP* will be realized only if the MathWizards prototype is successful and Stage 2 is undertaken. Both *MP* and its uncertainty, σ(*MP*), will be estimated using market data.

PROBLEM FRAMING

Since the full development stage (Stage 2) is conditional on the success of the prototyping stage (Stage 1) and the market outlook at the end of Stage 1, the overall strategy can be viewed as a learning option. Stage 1 does not yield any immediate benefits. Its purpose is to resolve uncertainty and create the *option* to continue with the full development stage. The cost of full development is the exercise cost of the option. The underlying asset, the asset being targeted, is the uncertain market payoff MP.

DATA SETUP AND ESTIMATION

SCHEDULE AND BUDGET PLAN

The prototype will be ready in four months and cost $200,000 to develop. The prototype will then undergo a usability evaluation, which will take an additional two months and cost $100,000. Without further elaboration of Stage 1, we skip to the budget plan for Stage 2, the full development stage.

The plan for Stage 2 is shown in FIGURE 3. The vertical axis represents cumulative cost. The total planned budget, or *BAC*, for Stage 2 is thus one million dollars. Resources are allocated on a per-month basis, from the start of Stage 2. The flatter region in the middle represents an expected decrease in resources due to another project scheduled during that region.

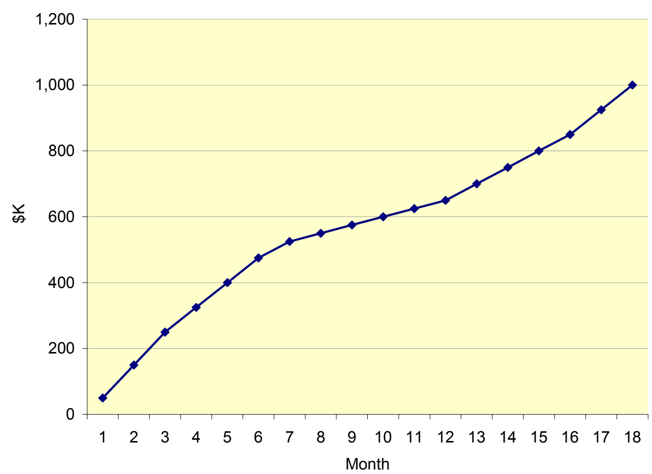

FIGURE 3. Estimated budget and schedule for the full development stage (Stage 2) of MathWizards.

ESTIMATION OF PROJECT UNCERTAINTY USING A REFERENCE PROJECT

Identification of Benchmark Data: The Reference Project

The company's database contains a data point from a similar past project. It is expected that Stage 2 will operate under a similar resource commitment scheme, will be subject to similar technical risks, and will be undertaken by a team with a similar mix of skills. This past project constitutes the *reference project*. MathWizards' full development stage will be subject to a budget and schedule variability comparable to that of the reference project.

FIGURE 4 shows the *PMB* of the reference project. The *PMB* consists of the three different cumulative plots illustrated in FIGURE 1. The horizontal axis represents the percentage budgeted schedule. Some characteristics of the reference project are:

- The *BCWS* curve indicates that the project was planned to incur a total cost of $875K at completion. This is the budget at completion (*BAC*).
- The *BCWP* curve, or the *earned value* curve, extends beyond the 100% schedule mark (*SAC*), terminating at 125% of the total budgeted schedule. The $875K-worth of total planned work was completed with a 25% schedule overrun.
- The *ACWP* curve terminates at a total value of $975K. Thus the project completed with a cost overrun of 975 – 875 = $100K, or 23% of *BAC*.

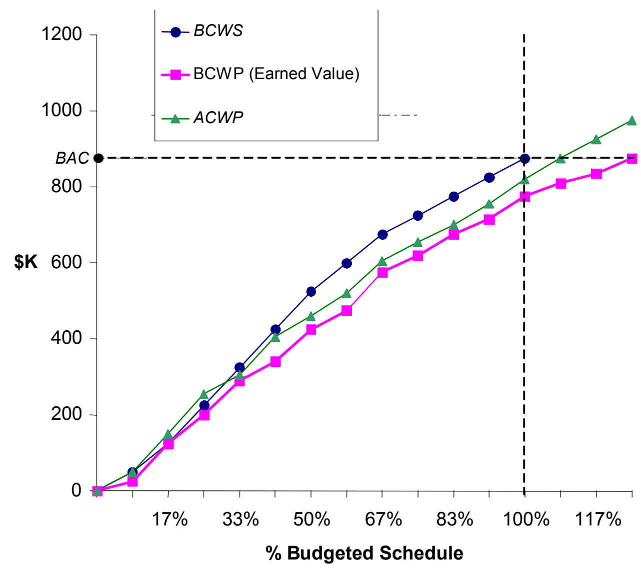

FIGURE 4. Performance measurement baseline (*PMB*) for the reference project.

Analysis of the Reference Project

A single data point is not sufficient to make an overall prediction with a reasonable degree of statistical significance. In this case, a cost overrun of 23% and a schedule overrun of 25% cannot be credibly predicted for the MathWizards project based on the total delay and cost overrun suffered by the reference project. Instead, we will rely on period-to-period observations, and inspect the deviation from the planned schedule and cost throughout the progression of the reference project. An analysis of the *PMB* of the reference project using period observations yields the required estimates regarding the cost and schedule uncertainty of the project under evaluation.

The first step is to estimate a set of most likely schedules corresponding to different project completion times, together with a probability distribution for these completion times. The *HSV* metric is used for this purpose.

The next step is to estimate the completion cost based on the budget plan of FIGURE 3, under a given deviation from the planned schedule. Note that the budget plan is developed independent of any reference project. To predict the deviation from the budgeted period costs under a specified deviation from the planned schedule, we take advantage of the metrics *CV* and *VSV*. *VSV* is used to estimate the *BCWP* of the project under evaluation from its original plan – that is from the project's *BCWS* – based on a hypothesized schedule slip. *CV* can then be used to estimate the project's *ACWP*, which leads to the cost estimate corresponding to that schedule slip.

The details of these calculations can be found in the APPENDIX.

FIGURE 5 plots the percentage horizontal schedule variance, %*HSV*, for the reference project. %*HSV* has a mean value of –13% and a standard deviation of 6%. Before the %*HSV* data can be used for estimating schedule uncertainty, it needs to be transformed into a more usable form to be able to generate a probability distribution for the deviation from the planned schedule.

FIGURE 6 plots this transformed metric, the *estimated relative completion time* (ERCT), as a function of relative budgeted schedule. Over the lifetime of the project, the ERCT has a mean value of 116%, corresponding to a relative mean delay of 16%, and a standard deviation of 8.1%.

By fitting an appropriate probability distribution to the ERCT data, completion and delay probabilities can be estimated for a given schedule. For the data in FIGURE 6, a lognormal distribution is chosen. FIGURE 7 plots the resulting completion probabilities, again against the percent deviation from the budgeted schedule. According to these estimates, there is virtually no chance that the project will be completed ahead of planned schedule. The probability of completing the project as planned, at 100% of the budgeted schedule, is less than 3%. A delay of more than 17%, corresponding roughly to one quarter, is likely with a probability of about 45%. A delay of more than 33%, or roughly two quarters, is likely with a probability of less than 2%. A delay of more than two quarters is highly unlikely.

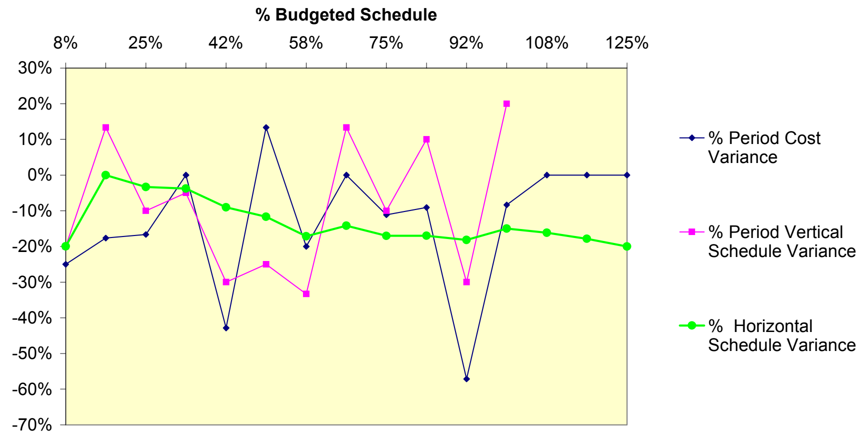

FIGURE 5. Percentage deviation of actual cost and schedule from budgeted cost and schedule, respectively, for the reference project.

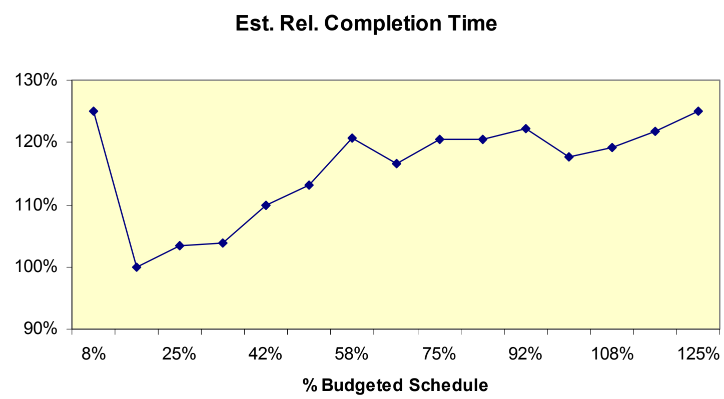

FIGURE 6. Estimated relative completion time for the reference project.

Given a deviation granularity of one quarter (three months) and a cutoff probability of 5%, the most likely schedules are represented by delays of 17% and 33%. The likelihood of the project to sustain a delay of more than 17% is

45%. A delay of more than 33% is below the cutoff probability, and therefore is not considered.

FIGURE 8 plots the estimated future value of total development cost for Stage 2 of the MathWizards project as a function of the realized schedule. The cost estimates are obtained using the *CV* and *VSV* metrics.

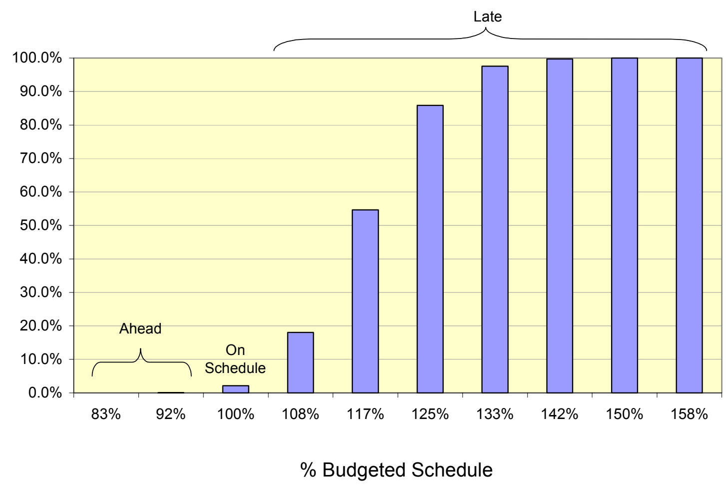

FIGURE 7. Completion probability estimates for Stage 2.

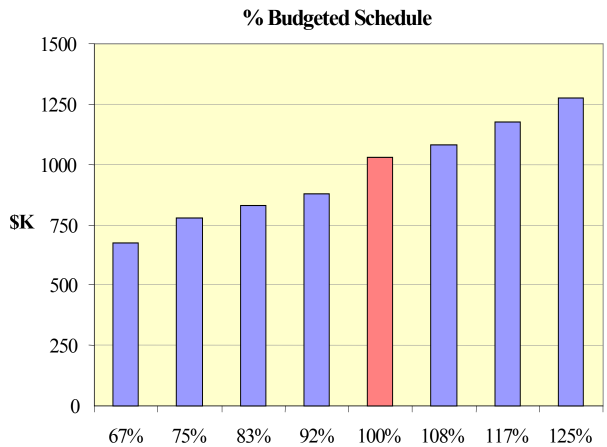

FIGURE 8. Future value of total development cost for Stage 2.

ESTIMATION OF THE MARKET PAYOFF

If MathWizards successfully passes the user evaluation study following the prototyping stage, Polysis expects to capture 5% of the total market in mathematical software. Polysis considers MathSoft, a public company, as its main competitor. The company analyst predicts that MathSoft has currently 50% of the market share targeted by Polysis. At the time of the project valuation, MathSoft had a price-to-sales ratio of 2.36 and a market capitalization of 64.9 million dollars. Based on this information, we perform a simplistic assessment of the market payoff for the whole project.

Total current revenues for MathSoft thus amount to 64.9/2.36 = 27.50 million dollars. The total size of the market targeted by Polysis is twice of this figure, given that MathSoft's market share is estimated to be 50%. Polysis hopes to capture 5% of this market with the launch of MathWizards, again conditional on the feature set passing the usability evaluation. The future value of the market payoff expected by Polysis conditional on the completion of the full development stage is (27.5)(2)(5%) = $2.75 million. This figure is taken to be the expected future market payoff, or MP.

The present value of *MP* depends on the completion time. Since MP is subject to market risk, it should be discounted back to the present from using a risk-adjusted rate *r*. Polysis uses a continuously compounded annual hurdle rate of 20% for new initiatives to account for market risk. Thus we set r = 20%. TABLE 1 gives the present value of *MP* under the three possible completion times that were identified in the previous subsection. This scheme is consistent with the Capital Asset Pricing Model.

ESTIMATION OF THE MARKET PAYOFF UNCERTAINTY

The MP estimates constitute a good starting point, but the calculated present values are still not certain. Since the market outlook at the end of the prototyping stage will affect the full development decision, the uncertainty of these figures will affect the project value. We use the stock performance of the competitor MathSoft as a proxy for the outlook of the mathematical software market, thereby substituting the stock's volatility for the volatility of the parameter MP. The two companies are assumed to operate in the same target market and are subject to the same market risk.

FIGURE 9 plots the weekly performance of the MathSoft stock over the year preceding the time of the project's valuation (the data is split-adjusted). The choice of weekly observations over a one-year range is somewhat arbitrary. We assume that while the growth rates may vary rapidly and erratically, growth rate volatility is fairly stable in the software industry over overlapping ranges of one year. At the time of the analysis, empirical data appeared to support this hypothesis [16]. Daily observations are too fine-grained to be of much value and monthly observations yield too few data points for the chosen range. Weekly observations seem like a good compromise.

The uncertainty of the market payoff, $\sigma(\text{MP})$, is captured by the *volatility* of the market proxy. Volatility is formally defined as the standard deviation of percentage change in the value of the proxy over the given range, annualized appropriately. Several references explain how to calculate volatility; see [13, 16]. The MathSoft stock exhibited a very high volatility of 115% over the selected range. The standard deviation of the weekly percentage changes in the stock's price was 16%. When annualized, this yields an approximate volatility of $\sqrt{52} \cdot 16\% = 115\%$ (under the assumption that the development of the proxy follows a standard lognormal diffusion process). Provided that Polysis continue to be subject to comparable market uncertainty over the next year and a half, we can take $\sigma(\text{MP}) = 115\%$.

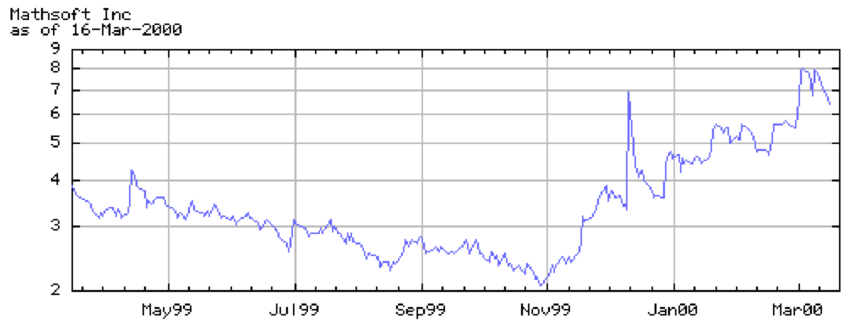

FIGURE 9. MathSoft's stock performance (source: yahoo.finance.com).

PROJECT UNCERTAINTY SUMMARY

In TABLE 1, the cost estimates and the expected present value of the market payoff for the identified completion times are shown for Stage 2 of the MathWizards project. The completion times represent delays of one and two quarters beyond the planned schedule. *D* denotes percent deviation from the planned, or budgeted, schedule; *PV* denotes present value; and *FV* denotes future value.

The estimated completion probabilities associated with each completion scenario are also shown. Although the likelihood of completing the project on schedule is marginal, we still consider this possibility because the cutoff probabilities are used only to determine the cutoff time, but not the potential delay points.

TABLE 1. Project estimates for Stage 2.

Delay Suffered	On time (No delay)	1st Delay (Delay 1)	2nd Delay (Delay 2)
k (quarters)	0	1	2
D (%)	0	17%	33%
Development Cost			
$FV_k(DC)$, \$M	1.1	1.2	1.4
Market Payoff			
$PV_k(MP)$, \$M	1.8	1.75	1.67
Annual volatility	115%	115%	115%
Probability			
Completed	2%	55%	98%
Further delay	98%	45%	2%

VALUATION

We proceed with the valuation of the two-stage strategy of Polysis regarding the development of MathWizards. Factors to be considered are:

- The first stage is for prototype development and usability evaluation. This is a learning stage. The cost and schedule of this stage are assumed to be certain. The first stage does not yield any immediate benefits, but is a prerequisite for the subsequent stage.
- The second stage, the full development of MathWizards, is an option on the first stage. The decision to continue with the full development stage is undertaken at the end of the first stage only if the prototype is successful and the market outlook at that time is favorable. The cost and schedule of the second stage are uncertain. Also uncertain is the ultimate expected market payoff of the overall project.

In all calculations to follow, the risk-free rate r_f is fixed at 7% per annum.

STATIC NET PRESENT VALUE

First we calculate the standard, static net present value (NPV) of the overall project to highlight a flaw in traditional discounted cash flow valuation. The static NPV treats the second stage as a mandatory stage by disregarding the full-development decision at the end of the first stage, but it still takes the cost and schedule risk of the second stage into account.

The static NPV is given by:

$$\begin{aligned}\text{Static NPV} = &-(\text{Total Cost of First Stage}) + \\ &E[PV(MP) - PV(\text{Expected Completion Cost of Second Stage})]\end{aligned}$$

Here PV denotes present value and $E[.]$ denotes the expectation operator. Since the total cost of the first stage was already expressed in present value terms, no discounting is necessary for this term. The second term is the unbiased net value for the second stage. The expectation is calculated using the probability distribution of the project schedule provided in TABLE 1. The development costs are discounted back to the present time using the risk-free rate. In millions of dollars, the static NPV equals:

$$\text{Static NPV} = -0.3 + 0.63 = 0.33$$

This calculation, however, is seriously flawed. The MP estimate is conditioned on the final product matching the market requirement for new features in mathematical software. If this requirement is not matched, MP will never be realized. The prototyping stage is designed particularly to resolve the market requirement uncertainty. If the continuation strategy is independent of the outcome of the learning stage, the learning stage is useless. In addition, the learning stage will, as a side benefit, provide an opportunity to reevaluate the market outlook. Even when the prototype is successful, if the market evolves into an unfavorable state relative to the continuation cost at the end of the learning stage, the second stage should be forgone. We need to resort to a more sophisticated valuation technique that can account for this multi-criterion, dynamic nature of the continuation decision.

DEALING WITH UNCERTAINTY AND OPTIONALITY

Modeling Market Payoff Uncertainty Using a Binomial Lattice

To be able to reason about the market outlook, a model of the dynamics of the market payoff MP is needed. We use the volatility estimate $\sigma(MP)$ to develop such a model.

The *binomial model* [37] is frequently used in option pricing to model the development of an uncertain asset for solving valuation problems with structures that cannot be handled by closed-form option pricing formulae.

In the binomial model, the underlying asset of an option is modeled using a multiplicative binary random-walk process. Starting from an initial expected or observed value, the asset moves either up or down in a fixed interval. The same process is repeated for successive intervals such that two consecutive opposite moves always takes the asset to its previous value, generating a *binomial lattice*. The resulting structure represents the possible evolution of the asset in discrete time, starting with an initial value. The structure is a binary tree with merging upward and downward branches.

In the current example, the underlying uncertain asset is represented by the market payoff of the end product. This asset does not exist independently: It is created only if the second stage of the project is undertaken. Moreover its value cannot be observed. *W* already estimated the conditional value of this asset, *MP*, in present value terms for three different completion times. The value of the overall scenario depends on the behavior of this asset. In FIGURE 10, three parallel binomial lattices are shown for this asset, one for each completion time. The root node is represented by the values 1.67, 1.75, and 1.84, which are the estimated present values of the variable *MP* for the "Delay 2", "Delay 1" and "No Delay" scenarios, respectively. The maximum duration of the project, including the two most likely schedule slips, is 30 months, but the decision to undertake the full development stage (Stage 2) will be made in six months. The 30-month horizon is divided into ten intervals of three months, or a quarter, long. A coarser (e.g., biannual) interval size generates too few states at the decision point corresponding to the end of the first stage. A finer (e.g., monthly) granularity results in a large tree without any significant impact on the valuation results.

Month	0	3	6	9	12	15	18	21	24	27	30
Delay 2	1.67	2.96	5.27	9.36	16.64	29.6	52.5	93.4	165.9	294.9	524.1
Delay 1	1.75	3.12	5.54	9.84	17.49	31.08	55.24	98.16	174.44	310.01	
No Delay	1.84	3.28	5.82	10.35	18.39	32.67	58.07	103.19	183.39		
Delay 2		0.94	1.67	2.96	5.27	9.36	16.64	29.57	52.54	93.37	165.94
Delay 1		0.99	1.75	3.12	5.54	9.84	17.49	31.08	55.24	98.16	
No Delay		1.04	1.84	3.28	5.82	10.35	18.4	32.7	58.1		
Delay 2			0.53	0.94	1.67	2.96	5.27	9.36	16.64	29.57	52.54
Delay 1			0.56	0.99	1.75	3.12	5.54	9.84	17.49	31.08	
No Delay			0.58	1.04	1.84	3.28	5.82	10.35	18.4		
Delay 2				0.30	0.53	0.94	1.67	2.96	5.27	9.36	16.64
Delay 1				0.31	0.56	0.99	1.75	3.12	5.54	9.84	
No Delay				0.33	0.58	1.04	1.84	3.28	5.82		
Delay 2					0.17	0.30	0.53	0.94	1.67	2.96	5.27
Delay 1					0.18	0.31	0.56	0.99	1.75	3.12	
No Delay					0.18	0.33	0.58	1.04	1.84		
Delay 2						0.09	0.17	0.30	0.53	0.94	1.67
Delay 1						0.10	0.18	0.31	0.56	0.99	
No Delay						0.10	0.18	0.33	0.58		
Delay 2							0.05	0.09	0.17	0.30	0.53
Delay 1							0.06	0.10	0.18	0.31	
No Delay							0.06	0.10	0.18		
Delay 2								0.03	0.05	0.09	0.17
Delay 1								0.03	0.06	0.10	
No Delay								0.03	0.06		
Delay 2									0.02	0.03	0.05
Delay 1									0.02	0.03	
No Delay									0.02		
Delay 2										0.01	0.02
Delay 1										0.01	
Delay 2											0.01

FIGURE 10. A binomial lattice for MathWizards' market payoff.

The values associated with the subsequent nodes of the binomial lattice are determined using the volatility estimate of 115% per year. From the volatility estimate, first we calculate an upward factor u that is greater than unity and a downward factor d that is smaller than unity. Over each interval, the value of the asset either increases by a factor of u or decreases by a factor of d. The upward and downward factors are chosen to be consistent with the volatility estimate. Recall that volatility is defined as the standard deviation of the rate of percentage change in an asset's value. If the volatility is σ, then u and d can be chosen as follows [33]:

$$u = \exp(\sigma\sqrt{\tau}) \text{ and } d = 1/u,$$

where τ is the chosen interval size expressed in the same unit as σ and exp denotes the exponential function. An annual volatility of 115% and an interval size of three months (.25 years) yield an upward factor $u = 1.78$ and a downward factor $d = 0.56$. Before proceeding, we to need verify that the chosen risk-free rate of 7% is consistent with these factors. An annual, continuously compounded rate of 7% is equivalent to a rate of 2% compounded quarterly, yielding a quarterly multiplier of $\exp(0.07 \times 0.25) = 1.02$. We check that this multiplier is greater than the downward factor d and smaller than upward factor u, a condition that must be satisfied to be able to apply risk value valuation to the scenario.

The lattice for each completion time is rolled out beginning with the initial market payoff for that completion time and multiplying it with the upward and downward factors repeatedly until the completion time. This process yields 11, 10 and 9 terminal nodes for the "Delay 2 ", "Delay 1", and "No Delay" scenarios, respectively. Each figure represents a possible future market payoff for MathWizards given an annual volatility of 115% and the present value of the market payoff for a specific completion time. Any of these values may be realized if the second stage of the project is undertaken.

Transformation of the Binomial Lattice into a Decision Tree

The next step is to map the binomial lattice structure onto a corresponding decision tree. This is where new branches are introduced to capture the effect of the private risks and the continuation decision for the second stage. The resulting decision tree is shown in FIGURE 11. The values in the decision tree have been computed for a fixed success probability $PS = 0.5$ using the integrated rollback procedure.

The structure of the decision tree mirrors the structure of the underlying binomial lattices. The lattice structure is augmented with different kinds of branches. Three kinds of branches are introduced at three points along the time horizon of ten quarters.

1. **The full development decision**. The decision to undertake the full development stage (Stage 2) will be made at the end of the second quarter, in six months. This decision is represented in FIGURE 11 by the Pass-Fail branches. The decision depends both on the success of the prototype (represented by the probability PS) and the value of the remainder of the project at that time. Once the market requirement uncertainty is resolved through prototyping, the remainder of the

project is subject to the cost and schedule uncertainty of the full development stage and the uncertainty of the targeted market.

2. **1st Delay**. Based on an analysis of the reference project, the estimated minimum time to complete the full development stage (Stage 2) is six quarters, which corresponds to eight quarters in total elapsed time from the beginning of Stage 1. The possibility of an initial schedule slip is represented in FIGURE 11 by the “Delay-Done” branches at the end of the eighth quarter. Each of the “Done” branches leads to a terminal node in the decision tree, whereas each of the “Delay” branches is expanded further into a subtree.
3. **2nd Delay**. Again, based on the analysis of the reference project, following an initial delay (“Delay 1”), the estimated minimum time to compete Stage 2 is seven quarters, or nine quarters in total elapsed time from the beginning of Stage 1. The possibility of a second schedule slip is represented in FIGURE 11 by the “Delay-Done” branches at the end of the ninth quarter. As with the “Delay 1” scenario, each of the “Done” branches leads to a terminal node, whereas each of the “Delay” branches is expanded into a subtree.

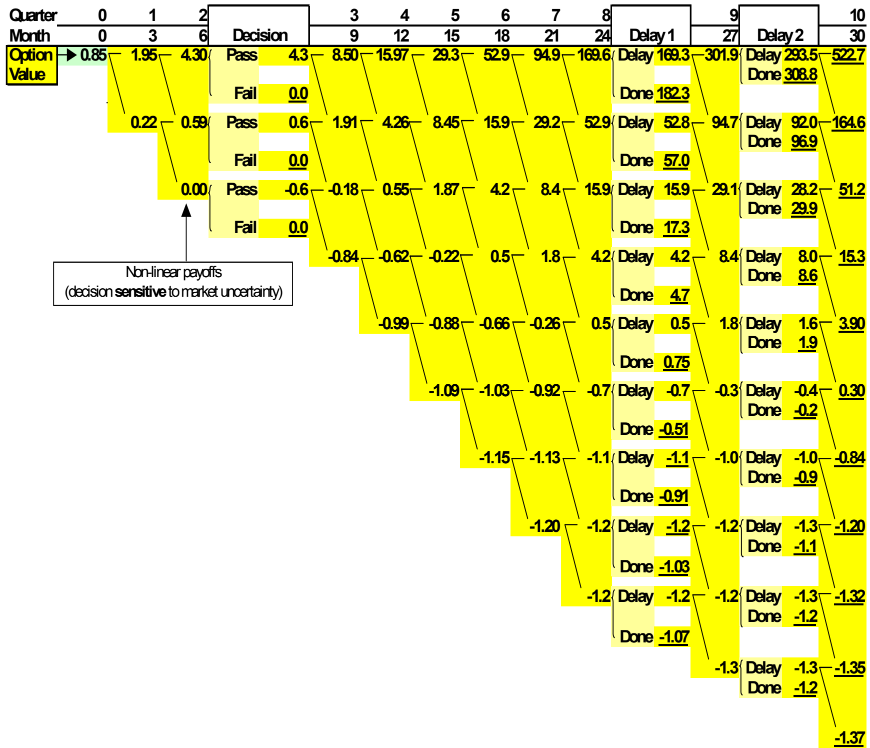

FIGURE 11. MathWizards decision tree for a success probability of 0.5.

Following a second delay, the project will continue another quarter before it is completed. Since the probability that the full development will extend beyond this point is very small according the reference project data, the tree is cut off at the tenth quarter.

Calculating the Terminal Net Payoffs

Once the structure of the decision tree is determined, the net payoff is calculated for each terminal node. First we identify a payoff function for each type of terminal node. There are four types of such nodes overall. Starting with the farthest nodes, these are:

A. The full development stage has been undertaken and the project has suffered two consecutive delays. In FIGURE 11, these are the nodes under the tenth quarter.

B. The full development stage has been undertaken and the project has suffered a single delay. In FIGURE 11, these are the nodes to the right of the "Done" branches under the column labeled "Delay 2".

C. The full development stage has been undertaken and the project was completed on time. In FIGURE 11, these are the nodes to the right of the "Done" branches under the column labeled "Delay 1".
D. The full development stage has been forgone. In FIGURE 11, these are the nodes to the right of the "Fail" branches under the column labeled "Decision".

The net payoff for a terminal node *n* of type A, B, and C is:

$$MP_k(n) - FV_k(DC),$$

where $MP_k(n)$ is the future market payoff at the corresponding final state in the original binomial lattice associated with *k* delays, and $FV_k(DC)$ is the expected future value of the total development cost after *k* delays, where *k* equals 2 ("Delay 2") for a type-C node, 1 ("Delay 1") for a type-B node, and 0 ("No Delay") for a type-A node.

For a type-D node, the net payoff is simply zero because the full development stage is not undertaken.

Discounting the Net Payoffs

The net payoffs are discounted back to the present time by folding back the decision tree in a recursive manner. The process involves dynamic programming: an optimal decision is computed whenever a decision node is reached. The process, referred to as *integrated rollback*, has been suggested by Smith and Nau [3]. It yields the present value for the overall scenario at the root of the tree.

During the rollback process, payoffs that are subject market risk and private risk are handled differently. In addition, payoffs subject to an immediate decision require special treatment since their payoff function deviates from the usual linear form seen in classical decision tree analysis. The next subsection explains in detail how the integrated rollback process works.

Treating Market Risk

To handle market risk, the same two principles that underlie traditional option pricing models are invoked: *replicating portfolio* and the *law of one price*. We develop on the fly the resulting general valuation technique that is known as *risk-neutral valuation*.

Consider the top two terminal nodes of the decision tree in FIGURE 11 with the corresponding net payoffs of 522.7 and 164.6, respectively. What is the expected discounted payoff at the beginning of the preceding interval?

One approach is to attach probabilities to the upward and downward branches in the binomial lattice, calculate the expected payoff using these probabilities, and then discount the result back one interval using a proper discount rate. This approach would work, except that neither the probabilities nor the proper discount rate are known in this case. (We could have used the company hurdle rate of 20%, but this should not apply to costs that are subject purely to private risk to be consistent with standard market theory. Moreover, with each new development in the market outlook, theoretically, the discount rate should also change to reflect the new expected return).

Instead, we appeal to the concept of a *replicating portfolio*. The payoffs of 522.7 and 164.6 at the terminal nodes can also be realized artificially by forming a portfolio composed of a *twin security* (a market proxy) and a fixed-interest loan. The movement of the twin security parallels that of the market payoff. Since we based the estimate of the market payoff volatility on the competitor's historical stock performance, the twin security in this example is simply the stock of the competitor. The absolute value of the twin security is not important as long as its movement is positively correlated with the underlying asset, or the market payoff MP in this case. When the market payoff moves up or down, the twin security also moves up or down by the same factor. Assume that the value of the twin security at the beginning of a quarter-long interval is M.

The replicating portfolio is formed at the beginning of the interval as follows:

- Buy n units of the twin security. This represents the position of the replicating portfolio in the underlying asset.
- Take out a loan in the amount of B at the risk-free rate of interest to partly finance this purchase. This represents the position of the replicating portfolio in the risk-free asset.

The worth of the replicating portfolio at the beginning of the interval then equals $nM - B$. If we can determine the value of n and B, we can calculate the exact value of the replicating portfolio (as we will see, we don't need to know the value of M).

This is the right point to apply the *law of one price*: the expected value of the payoff at the beginning of the interval must equal the value of the replicating portfolio for otherwise an arbitrage opportunity would exist.

Now consider the possible values of the portfolio at the end of the interval. After one interval, the loan must be paid back with interest to receive the payoff. Regardless of what happens to the price of the twin security, the amount of the loan will be $B(1 + \Delta r_f)$, including the principle and the interest accrued. Here Δr_f is the interest rate on the loan over a single interval (in this case, approximately 2%).

If on the one hand, the price of the twin security moves up to uM, the portfolio will then be worth $uMn - B(1 + \Delta r_f)$. For the portfolio to replicate the payoff, this amount should equal 522.7, the payoff after the upward movement. On the other hand, if the price of the twin security falls to dM, the portfolio will be worth $dMn - B(1 + \Delta r_f)$, which must equal 164.6, the payoff after the downward movement. Thus the law of one price gives rise to two equations:

If the price moves up:

$$\begin{aligned} 522.7 = \text{(Terminal payoff)} &= \text{(Terminal value of replicating portfolio)} \\ &= uMn - B(1 + \Delta r_f) \end{aligned}$$

If the price moves down:

$$\begin{aligned} 164.6 = \text{(Terminal payoff)} &= \text{(Terminal value of replicating portfolio)} \\ &= dMn - B(1 + \Delta r_f) \end{aligned}$$

Since Δr_f, u and d are all known, we can solve these two equation for B and n as a function of M, and then calculate the portfolio value at the beginning of the interval by plugging the solution in the expression $nM - B$. Fortunately, the unknown M is eliminated during this process, and the value of the portfolio at the beginning of the interval is calculated as 293.5. If the same procedure is repeated for the remaining pairs of nodes in the column for the tenth quarter, we obtain the portfolio values shown in the column "Delay 2" corresponding to the "Delay" branches. Each figure represents the present value of the net payoff at the corresponding state of the underlying binomial lattice, immediately following the second schedule slip.

Applying Risk-Neutral Valuation

The procedure described in the previous subsection may seem somewhat cumbersome. Fortunately, there is an easier way. Solving a system of simultaneous equations to obtain the portfolio value at the beginning of an interval is equivalent to computing the expected value of the payoffs at the end of the interval using an artificial probability measure, and then discounting back this expected value at the risk-free rate by one interval. FIGURE 12 illustrates this simple technique.

Future payoffs following upward and downward movements

$$C = \frac{p \cdot C^{+} + (1 - p) \cdot C^{-}}{(1 + r_f)}, \text{ where } p = \frac{1 + r_f - d}{u - d}$$

Risk-free rate ***Risk-adjusted probability***

Option value is the ***expected value*** *of future payoffs* ***under the risk-adjusted probabilities*** *discounted* ***at the risk-free rate***

FIGURE 12. Risk-neutral valuation in the binomial model.

The formula in FIGURE 12 applies so long as the interval being considered is only subject to market risk, and does not involve a decision point or a change of state due to private risk. In FIGURE 11, the nonterminal values that can be computed using the risk-neutral valuation formula are those corresponding to the quarters 0, 1, 3 to 7, 10, and the nonterminal values under the columns labeled "Decision", "Delay 1" and "Delay 2".

As an example, consider the computation of value of the root node of the decision tree:

$$0.43 = \frac{0.97 \times p + 0.11 \times (1 - p)}{1 + r_f},$$

$$0.97 = \frac{2.15 \times p + 0.30 \times (1 - p)}{1 + r_f} \text{ and } 0.11 = \frac{0.30 \times p + 0 \times (1 - p)}{1 + r_f},$$

where:

$$p = \frac{1 + r_f - d}{u - d} = 0.37\,,\; 1 - p = 0.63,\; \text{and}\; r_f = 0.02.$$

The quantities 1 and 1 – p in the above equations and in FIGURE 12 are referred to as *risk-neutral* or *risk-adjusted* probabilities. They don't represent the actual probabilities of the upward and downward movements of the underlying asset, yet they are used to compute an expected value (in FIGURE 12, the numerator in the equation on the left). The expected value is simply discounted back at the risk-free rate r_f. The artificial probabilities p and 1 – p depend on the spread between u and d, the upward and downward movement factors of the twin security. In a way then, p and 1 – p capture the variation, or the risk, of the underlying asset relative to the risk-free asset.

The general method of computing the present value of an asset based on the principles of replication and no-arbitrage is known as *risk-neutral valuation*. Risk-neutral valuation is commonly used in valuing asymmetric contingent claims on risky assets, options in particular.

A number of features are remarkable about this technique. First, the value calculated does not require the actual probability distribution underlying the asset's movement. Second, it does not require a discount rate provided that the initial value of the underlying asset is given. Third, the procedure is independent of how the future payoffs are calculated. Since the rule used to calculate the payoffs doesn't matter, the process is the same for any payoff function.

Treating Private Risk

The integrated rollback procedure deviates from risk-neutral valuation where private risk is involved. Schedule slips constitute private risk. To treat schedule uncertainty, we need the delay probabilities computed in TABLE 1 from the reference project's earned value data. These probabilities are used to calculate the values under the eighth and ninth quarters. The probabilities are used to collapse the payoffs associated with the nodes following a delay node into an *unbiased* estimate:

Unbiased Payoff Immediately Before a Delay
= Payoff[Delay]·P[Delay] + Payoff[Done]·P[Done],

where P[Delay] is the estimated (conditional) probability of a delay at that delay point and P[Done]= 1 – P[Delay] is the estimated (conditional) probability of the project completing without a further delay. The probability is conditional if a previous delay has already been sustained.

Given an estimated 45% probability of a second delay (conditional on a previous delay), the payoff for the top row of column nine is calculated as:

$$301.9 = 293.5 \times 0.45 + 308.8 \times 0.55$$

No further adjustment for risk is required since the underlying risk is purely private, and therefore technically diversifiable. The central assumption here is that the decision maker is risk-neutral, with a linear utility function. If the risk preferences of the decision makers are to be taken into account, a convex utility function can be used to transform the unbiased estimate, allowing a further adjustment for risk averseness.

The remaining values under the ninth column are calculated in a similar manner. For the eight quarter, the procedure is similar except that the delay probability equals 98%.

Treating the Decision Point

The decision nodes at the second quarter are treated in a similar way since the risk of the prototype successfully passing the user evaluation is also private. However, these nodes in addition involve a decision that is similar to the exercise decision of an option, resulting in a nonlinear payoff. Since the project will be abandoned if either the prototype fails or the revised estimate of the final payoff is negative, we have:

$$\begin{aligned}&\text{Unbiased Payoff Immediately Before Prototype Evaluation}\\&= \text{Max}\{\text{Payoff[Pass]}, 0\}\ PS + 0 \times (1 - PS)\\&= \text{Max}\{\text{Payoff[Pass]}, 0\} \times PS\end{aligned}$$

where PS is the probability of the prototype passing the evaluation and 1 – PS is the probability of the prototype failing the evaluation. Note that if the prototype succeeds, but the subsequent discounted payoff is still negative, the project will still be abandoned, thereby avoiding a likely loss. Hence if Payoff[Pass] is negative, the unbiased estimate yields a value of zero. For example, the value of the bottom node of column for the second quarter in FIGURE 11 is calculated as:

$$0 = \text{Max}\{-0.6, 0\} \times PS$$

The remaining two nodes under the second quarter in FIGURE 11 have been calculated using a *PS* value of 0.5 for illustration purposes.

The nonlinear payoff function prevents a negative value to be propagated past the decision point. Hence the payoff at the root of the decision tree is always positive.

Remarkably, the decision tree has nodes with negative values beyond the second quarter. While the underlying binomial lattices for the market payoff may not contain negative-valued nodes, the decision tree may contain negative-valued nodes beyond the points where optimal decisions are made based on a nonlinear net payoff function. In the current example, the continuation decision is made at quarter two, and beyond this action point, all values are determined by some future state of nature based on the payoff at that state net of the development cost associated with that state.

VALUTION RESULTS – THE NET PRESENT VALUE OF THE OVERALL PROJECT

The root value of the decision tree represents the option value of the second stage investment (full development) conditional on undertaking the first stage (prototyping). However, it does not take into account the $300K (in present value terms) cost associated with the prototyping stage. This cost must be deducted from the option value. The expanded net present value, or the NPV viewing the second stage as an option on the first stage, equals:

$$\text{Expanded NPV} = (\text{Option Value of Second Stage}) - (\text{Total Cost of First Stage})$$

With a success probability $PS = 0.5$, the option value equals 0.43, and expanded NPV, in millions of dollars, equals:

$$\text{Expanded NPV} = 0.43 - 0.3 = 0.13$$

The positive NPV indicates that the project is expected to be profitable at a *PS* level of 0.5.

However, if we disregard the effect of the market payoff on the continuation decision, the payoff function at the decision node becomes linear for a given probability of success. The negative branches following a “Pass” verdict are no longer pruned, the continuation decision is no longer optimal, and the calculation reduces to a standard expected value calculation. The expanded

NPV, as a result, decreases. FIGURE 13 depicts the reduced decision tree where the full development decision is a function of only the prototype's success. If the market payoff uncertainty is disregarded, at a success probability *PS* = 0.5, the option value of the second stage and the expanded NPV fall respectively to 0.32 and 0.02.

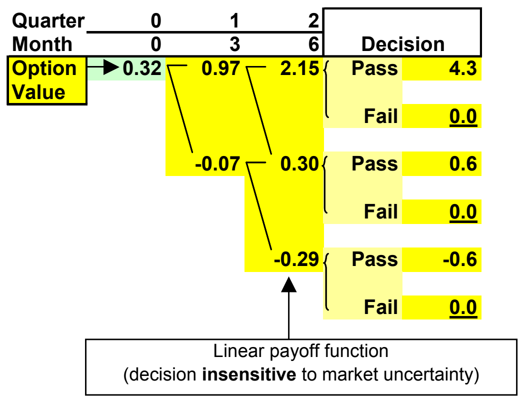

FIGURE 13. Reduced decision tree for the MathWizards project. The full development decision is insensitive to market uncertainty**.**

As the success probability *PS* increases, the expanded NPV increases. For *PS* > 35%, if the continuation decision considers the market outlook at the time of the decision, the expanded NPV is positive. But if the continuation decision disregards the market outlook at the time of the decision, the expanded NPV will remain negative until about *PS* = 50%.

FIGURE 14 shows how the expanded NPV varies as a function of the success probability *PS*. "[Market + Private]" implies both private and market risk are taken into account in the continuation decision. "[Private]" implies only private risk is taken into account in the continuation decision. The value of the project increases as more information is used in making the continuation decision.

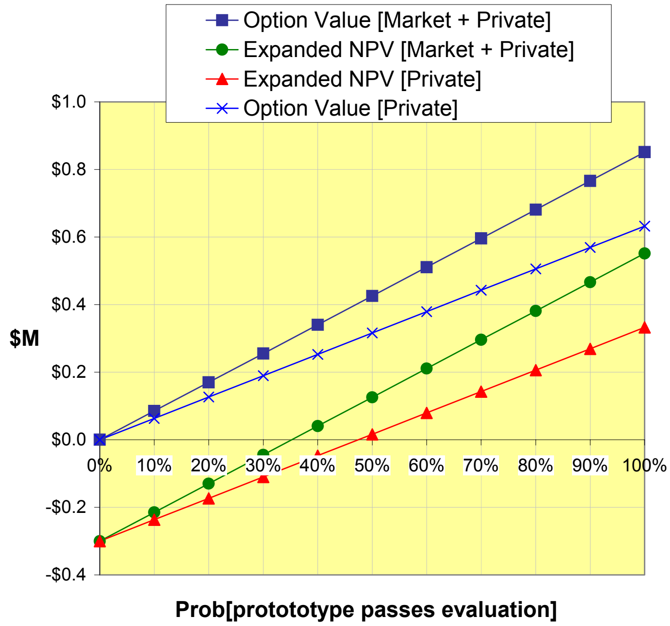

FIGURE 14. Sensitivity analysis for the MathWizards project.

SUMMARY

A valuation methodology must be able to handle both private and market risk to be applicable to commercial software development projects. Such projects are subject to multiple sources of uncertainty. Sources of uncertainty include not only market payoff, product quality, and the likelihood of the product to meet the market requirement, but also schedule and budget. Software projects are notorious for significant schedule and budget overruns, and have a high probability of failure in the market. Thus it is imperative that software projects are structured to manage the underlying risks. Optimal management of risks in turn increases the value of a project.

The paper presented a valuation methodology that can handle dynamic risk management strategies under multiple sources of risk. The methodology combines real options analysis with earned-value based estimation. Projects are constructed as sequences of stages. A continuation strategy is associated with each stage. The continuation strategy determines the subsequent stage and the conditions under which the subsequent stage will be undertaken. Earlier stages tend to resolve uncertainty, whether technical or market-related, and later stages focus on implementation. Each stage is viewed as a real option on the previous stage.

A case study illustrated the application of the methodology. A two-stage development project was considered. The first stage of the project involved prototyping. This was a learning stage designed to resolve the market requirement uncertainty. The subsequent stage was the full development stage, during which the target product was implemented. The full development was undertaken only if the prototype passed the usability study (and thus met the market requirement concerning usability) and the market payoff expectation was still favorable. The cost and schedule of the full development stage were uncertain.

The project was valued as a learning option using a combination of techniques. Uncertainty in the market payoff was estimated using a market proxy. Schedule uncertainty and budget uncertainty were estimated using earned value concepts and earned value data from a reference project. These estimates were then converted into a set of likely outcomes. Uncertainty concerning the usability of the product was treated as a sensitivity variable.

The market proxy was used to develop a tree-based model of the possible development of the market payoff projection. This model was subsequently extended into a decision tree with optionality based on the project's staged structure and the likely budget and schedule outcomes that had previously been identified. The technique integrates decision tree analysis with standard risk-neutral valuation to simultaneously handle market and private risk. The valuation demonstrated that the more information the continuation decision relied on, the more valuable the project became. Managing uncertainty increases value.

LIMITATIONS

AVAILABILITY AND RELEVANCE OF HISTORICAL DATA

The methodology relies on historical project data to estimate schedule and budget uncertainty. In particular, it assumes the existence of a reference project that shares similarities with the project under evaluation. The information gleaned from this reference project is transferred to the project under evaluation. This assumption is rather a strong one to make. Many technology projects are unique. Even if the organization is meticulous in data collection, a reasonable match may be difficult to obtain. In addition, the fitness of a candidate project to serve as the reference project depends on many factors such as scale, scope, risk characteristics, technical difficulty, team skills, and available technological solutions. Therefore, the identification of the reference project is a nontrivial task.

Several improvements are possible depending on the richness of the available data. To improve the statistical relevance of the estimates, historical data from several candidate projects can be combined. Using multiple projects allows the uncertainty estimates to be computed along two orthogonal dimensions: intra-project and inter-project. Estimates obtained in one dimension can be used to smooth the estimates obtained in the other dimension. If uncertainty tends to vary within projects, then projects can be decomposed into distinct phases. Separate estimates are computed and these estimates are later applied to the corresponding phases of the project under evaluation.

To take advantage of the probability distribution of the ERCT metric for a single reference project to estimate the completion and delay probabilities for a future project, we must assume that the distribution of schedule deviation is stable both across and within projects. By picking the reference project carefully, we make sure that the information gleaned from the reference project can be ported to the project under evaluation. The stability of ERCT within a single project is more problematic. This condition does not necessarily hold in the traditional waterfall model of software development where the distinct phases of the project (requirements, design, coding, testing, deployment) may be subject to varying degrees of schedule risk. In iterative and agile software development [43], the distinctions among the phases are blurred, and consequently, it would be safer to assume that the distribution of ERCT throughout the project is reasonably stable.

DATA RELIABILITY

As in any quantitative analysis, in valuation, the quality of the outputs depends on the quality of the inputs. Option valuation in particular is sensitive to the volatility estimate of the underlying asset. Where the quality of the input data is in doubt, sensitivity analyses should be preferred over point estimates. However, with several input variables, sensitivity analysis is difficult.

LIMITATIONS OF OPTION PRICING

The suitability of the financial options analogy and of the techniques developed to price financial options to the valuation of real options scenarios is frequently debated. The main differences between financial and real options are summarized in TABLE 2.

In particular, two points are important to keep in mind while applying option-based models and techniques such as the binomial model and risk-neutral valuation.

First, the existence of trading markets and assumptions regarding their efficiency, completeness, and liquidity are critical for financial option pricing techniques. These techniques are designed to treat risks that can be priced in such markets. The suitability of a specific option pricing technique to value a real options scenario may thus depend on the nature of the uncertainty being treated and the *closeness-to-the-market* of the underlying assets. In software development projects, where private risk is an important factor, the financial options analogy may be weak.

In cases where the financial option-real option analogy is weak, the option values yielded should thus be treated as idealized values computed under the assumption of an the existence of a market equivalent that reasonably closely tracks the risk being tackled. Fortunately, sometimes risks that seem purely private at first can be market-priced, thanks to an expanding and vibrant securities market in the technology sector. An example is provided in [16]. For further discussion of the analogy between financial and real options and the limitations of this analogy, the reader is referred to the sidebar in [10].

TABLE 2: Comparison of financial and real options.

Financial Options	Real Options
Complete markets. A replicating portfolio can emulate any payoff structure.	*Incomplete markets.* A replicating portfolio that emulates a particular payoff structure may not exist.
Traded asset. The underlying asset is traded in the financial markets.	*Twin security.* The underlying asset is not traded; instead, the existence of a proxy, or twin security whose value is correlated with the underlying asset must be assumed.
Observed current price. The current price of the underlying asset is observed.	*Hypothetical asset.* The current price of the underlying asset is not observed. It must be estimated in present value terms.
No discount rate. A discount rate is not needed to value the option because of the existence of an observed price and the replication and no-arbitrage (law of one price) assumptions.	*Discount rate needed.* A discount rate is often needed to calculate the present value of future payoffs.
No interaction. Financial options are self-contained, fixed-structure contracts. They don't interact.	*Extensive interaction.* Real options within a project or across different projects often have complex interactions. The behavior of one option may affect the value of the other.
Limited sources of uncertainty. Financial options involve one or two uncertain underlying assets.	*Multiple sources of uncertainty.* Real options frequently involve multiple underlying assets or assets with multiple sources of uncertainty.
Single ownership. Financial options have defined ownership.	*Shared ownership.* Real options are often shared among competitors. A company's exercise of a real option may kill or significantly undermine a similar real option for a competitor, and vice versa.
Value leakage. The holder of a financial option may be subject to loss of benefits, such as dividends, that are available to the holders of the underlying asset, but not to the holders of an option. This behavior can be modeled using historical data or industry conventions.	*Value leakage or amplification.* The holder of a real option may experience a reduction or an increase in benefits as a result of random actions by other market players. However such actions often do not necessarily follow defined patterns, and as such, are difficult to model.

REFERENCES

1. THE STANDISH GROUP, "CHAOS Report," 1998.
2. A. HUCHZERMEIER and C.H. LOCH, "Project management under risk: using the real options approach to evaluate flexibility in R&D," *Management Science*, Vol. 47, 2001, pp. 85-101.

3. J.E. SMITH and R.F. NAU, "Valuing risky projects: option pricing theory and decision analysis," *Management Science*, Vol. 41, 1995.
4. B. BOEHM, *Software Engineering Economics*: Prentice Hall, 1981.
5. B. BOEHM, C.W. HOROWITZ, R. MADACHY, and R. SELBY, "Cost models for future software lifecycle processes: COCOMO 2.0," *Annals of Software Engineering*, Vol. 1, 1995, pp. 45-60.
6. S.A. ROSS, R.W. WESTERFIELD, B.D. JORDAN, and G. ROBERTS, *Fundamentals of Corporate Finance*: Times Mirror Professional Publishing, 1996.
7. R. BREALEY and S.C. MYERS, *Principles of Corporate Finance*. New York, N.Y.: McGraw-Hill, 1987.
8. S.W. HESS, "Swinging on the branch of a tree: project selection applications,," *Interfaces*, Vol. 23, 1993, pp. 5-12.
9. S.C. MYERS, "Finance theory and financial strategy," in *Interfaces*, Vol. 14, 1984, pp. 126-137.
10. M. AMRAM and N. KULATILAKA, "Disciplined decisions: aligning strategy with the financial markets," *Harvard Business Review*, January-February 1999, pp. 95-104.
11. T.A. LUEHRMAN, "Investment opportunities as a portfolio of real options," *Harvard Business Review*, July-August 1998, pp. 51-67.
12. COPELAND and V. ANTIKAROV, *Real Options: A Practitioner's Guide*. New York, N.Y.: Texere, 2001.
13. M. AMRAM and N. KULATILAKA, *Real Options: Managing Strategic Investment in an Uncertain World*: Harvard Business School Press, 1999.
14. L. TRIGEORGIS, *Real Options: Managerial flexibility and Strategy in Resource Allocation*: MIT Press, 1994.
15. L. TRIGEORGIS, *Real Options and Business Strategy: Applications to Decision Making*. London, U.K.: Risk Books, 1999.
16. H. ERDOGMUS, "Value of commercial software development under technology risk," *The Financier*, Vol. 7, Nos. 1-4, 2000.
17. H. ERDOGMUS, "Management of license cost uncertainty in software development: a real options approach," in *Proceedings of the Fifth Annual Conference on Real Options: Theory Meets Practice*, UCLA, Los Angeles, 2001.
18. H. ERDOGMUS and J. VANDERGRAAF, "Quantitative approaches for assessing the value of COTS-centric development," in *Proceedings of the Sixth International Software Metrics Symposium*, Boca Raton, Florida: IEEE Computer Society, 1999, pp. 279-291.

19. H. ERDOGMUS and J. FAVARO, "Keep your options open: Extreme programming and the economics of flexibility," in *Extreme Programming Perspectives*, L. Williams, D. Wells, M. Marchesi, and G. Succi, Eds.: Addison-Wesley, 2002.
20. J.M. FAVARO and K.R. FAVARO, "Strategic analysis of application framework investments," in *Building Application Frameworks: Object Oriented Foundations of Framework Design*, M. Fayad and R. Johnson, Eds.: John Wiley and Sons, 1999.
21. J.M. FAVARO, K.R. FAVARO, and P.F. FAVARO, "Value based software reuse investment," *Annals of Software Engineering*, Vol. 5, 1998, pp. 5-52.
22. K.J. SULLIVAN, P. CHALASANI, S. JHA, and V. SAZAWAL, "Software design as an investment activity: a real options perspective," in *Real Options and Business Strategy: Applications to Decision Making*, L. Trigeorgis, Ed.: Risk Books, 1999.
23. K.J. SULLIVAN, W.G. GRISWOLD, Y. CAI, and B. HALLEN, "The structure and value of modularity in software design," in *Proceedings of the ACM SIGSOFT Symposium on the Foundations of Software Engineering (FSE)*, Vienna, 2001.
24. C. Y. BALDWIN and K. B. CLARK, *Design Rules: The Power of Modularity*. Cambridge, Mass.: MIT Press, 1999.
25. A. TAUDES, "Software growth options," *Journal of Management Information Systems*, Vol. 15, 1998, pp. 165-185.
26. A. TAUDES, M. FEURSTEIN, and A. MILD, "Options analysis of software platform decisions: a case study," *MIS Quarterly*, Vol. 24, 2000, pp. 227-243.
27. B.L. DOS SANTOS, "Justifying investments in new information technologies," *Journal of Management Information Systems*, Vol. 7, 1991, pp. 71-90.
28. M. BENAROCH and R.J. KAUFFMAN, "A case for using option pricing analysis to evaluate information technology project investments," *Information Systems Research*, Vol. 10, 1999, pp. 70-86.
29. M. BENAROCH and R.J. KAUFFMAN, "Justifying electronic banking network expansion using real options analysis," *MIS Quarterly*, Vol. 24, 2000, pp. 197-225.
30. J.C. HULL, *Options, Futures, and Derivatives*. Englewood Cliffs, N.J.: Prentice-Hall, 1997.
31. R.S. PINDYCK and A. DIXIT, *Investment Under Uncertainty*: Princeton University Press, 1992.

32. F. BLACK and M. SCHOLES, "The pricing of options and corporate liabilities," *Journal of Political Economy*, Vol. 81, 1973, pp. 637-659.
33. J. COX, S. ROSS, and M. RUBINSTEIN, "Option pricing: a simplified approach," *Journal of Financial Economics*, Vol. 7, 1979, pp. 229-263.
34. W. MARGRABE, "The value of an option to exchange one asset for another," *Journal of Finance*, Vol. 33, 1978, pp. 177-186.
35. P. CARR, "The Valuation of Sequential Exchange Opportunities," *Journal of Finance*, vol. 18, pp. 1235-1256, 1988.
36. L.R. KUMAR, "A note on project risk and option values of investments in information technologies," *Journal of Management Information Systems*, Vol. 13, 1996, pp. 187-193.
37. R.K. SUNDARAM, "Equivalent martingale measures and risk-neutral pricing: an expository note," *The Journal of Derivatives*, Fall 1997, pp. 85-98.
38. Q.W. FLEMING and J.M. KOPPELMAN, *Earned Value Project Management*, 2nd ed: Project Management Institute Publications, 2000.
39. D.S. CHRISTENSEN, "The costs and benefits of the earned value management process," *Acquisition Review Quarterly*, Fall 1998, pp. 373-386.
40. D.S. CHRISTENSEN, "Cost overrun optimism: fact or fiction?," *Acquisition Review Quarterly*, Winter 1994, pp. 25-37.
41. Q.W. FLEMING and J.M. KOPPELMAN, "Earned value project management: a powerful tool for software projects," *Crosstalk*, July 1998, pp. 19-23.
42. D.S. CHRISTENSEN and D.V. FERENS, "Using earned value management for performance measurement on software development projects," *Acquisition Review Quarterly*, Spring 1995, pp. 155-171.
43. J. HIGHSMITH, *Adaptive Software Development*: Dorset House, 1999.

BIOGRAPHICAL SKETCH

HAKAN ERDOGMUS is a senior research officer with the Institute for Information Technology, National Research Council of Canada. He holds a Master’s degree in Computer Science from McGill University, Montreal, and a Ph.D. in Telecommunications from Université du Québec. His current research is in software economics, focusing on project design and evaluation of software development processes and practices. He delivered numerous lectures and published several articles on these topics in finance and software engineering venues.

APPENDIX: ESTIMATION OF SCHEDULE AND COST UNCERTAINTY USING EARNED VALUE DATA

ESTIMATION OF SCHEDULE UNCERTAINTY

Horizontal Schedule Variance (HSV)

The horizontal schedule variance at a given period t, $HSV(t)$, is the schedule shortfall (excess) resulting from the difference between t and the elapsed time in the budgeted schedule (*BCWS*) at which the cumulative budgeted work (*BCWP*) corresponding to t should have been completed. Then *percentage horizontal schedule Variance* at period t, $\%HSV(t)$, equals $HSV(t)/t$. A negative *HSV* indicates a schedule shortfall or lag. A positive *HSV* indicates a schedule excess or lead.

Vertical Schedule Variance (VSV)

Vertical schedule variance is another way to capture the deviation from the planned schedule. It is given by the difference between cumulative *BCWP* and cumulative *BCWS*. For a given period t, (cumulative) *VSV* equals:

$$VSV(t) = BCWP(t) - BCWS(t)$$

The *percentage period vertical schedule variance* (*%pVSV*) is calculated from the *VSV* as follows:

$$\%pVSV(t) = \frac{[BCWP(t) - BCWP(t-1)] - [BCWS(t) - BCWS(t-1)}{BCWS(t) - BSCWS(t-1)}$$

where $BCWP(t) - BCWP(t–1)$ and $BCWS(t) - BCWS(t–1)$ denote incremental or period costs (as opposed to cumulative costs).

Cost Variance (CV)

Cost variance is the deviation from the planned cost. It is given by the difference between *ACWP* and *BCWP*. For a given period t, (cumulative) *CV* is equals:

$$CV(t) = BCWP(t) - ACWP(t).$$

The *percentage period cost variance* (%*pCV*) is calculated from *CV* as follows:

$$\%pCV(t) = \frac{[BCWP(t) - BCWP(t-1)] - [ACWP(t) - ACWP(t-1)}{BCWP(t) - BSCWP(t-1)}$$

where $BCWP(t)$ – $BCWP(t–1)$ and $ACWP(t)$ – $ACWP(t–1)$ again denote incremental or period costs (as opposed to cumulative costs).

IDENTIFICATION OF THE MOST LIKELY SCHEDULES

Estimated Relative Completion Time (ERCT)

Using only the information at period t – namely, the percentage *HSV* at that period – it is possible to estimate a relative completion time for the overall project by a linear extrapolation of the period lead or lag. If $\%HSV(t) = x$, then the project has been performing at $100 \times (1 + x)$ percent on average up to that period. If the same lag or lead is sustained throughout, the project will be completed at $100/(1 + x)$ of the budgeted schedule. For example, if $\% HSV(t) = –20\%$, the estimated relative completion time for period t equals 125%, indicating a delay of 25% at completion given the information at period t. *ERTC* is computed as:

$$ERCT(t) = 1/(1 + \%HSV(t)).$$

Probabilities of Completion and Delay

Let $pdf_{\mathrm{ERCT}}(x)$ denote the cumulative distributive function of *ERCT*. Assume that this distribution does not change over time as the project progresses. Then the probability that the project is competed within y% (where $y > 0$) of the planned schedule equals the probability that *ERTC* is smaller than or equal to y. Therefore:

$$P[\text{projects completes within } y\% \text{ of budgeted schedule}] = pdf_{\mathrm{ERCT}}(y/100)$$
$$P[\text{project is delayed beyond } y\% \text{ of budgeted schedule}] = 1 - pdf_{\mathrm{ERCT}}(y/100)$$

DEVELOPMENT COST ESTIMATION FOR THE MOST LIKELY SCHEDULES

Estimation of BCWP for a Given Deviation from the Planned Schedule

The mean value of %*pVSV* can be used to estimate the *BCWP* for a future project by adjusting the period *BCWS* values in the original project plan. If the project does not sustain a net schedule slip or is completed ahead of planned schedule, the period *BCWP* estimate will equal the period *BCWS* value. If the project sustains a deviation of *D*% from the planned schedule, then each period *BCWS* value can be adjusted as follows to predict the corresponding period *BCWP*:

$$pBCWP(t) = pBCWS(t) \times (1 + (\mu(\%pVSV) \times D)/(\mu(ERCT) - 1)),\ \ 1 < \ t \ \leq T$$

where *T* is the planned schedule or *SAC*, *pBCWP* and *pBCWS* denote period *BCWP* and period *BCWS*, respectively, $\mu(\%pVSP)$ is the mean value of the period *VSP* of the reference project and $\mu(ERCT)$ is the mean value of the estimated relative completion time of the reference project. The multiplier $D/(\mu(ERCT) - 1)$ scales the period deviation relative to the total predicted deviation *D*. If *D* is negative, then the deviation corresponds to a delay (or slip), and one or more period values for *BCWP* beyond the original schedule should be estimated. These post-schedule values can be estimated by allocating the remaining budget at completion evenly among the extra periods:

$$pBCWP(t) = (BAC - BCWP(T))/(T \times D),\ \ T < \ t \ \leq T \times (1 + D).$$

ESTIMATION OF ACWP

The mean value of %*pCV* is in turn used to estimate the period *ACWP* values by adjusting the estimated period *BCWP* values under a given deviation from the planned schedule. If the project does not sustain a net schedule slip or is completed ahead of planned schedule, the *ACWP* estimate will equal the *BCSW* estimate. If the project sustains a deviation of *D*% from the planned schedule, then each estimated period *BCWP* value can be adjusted as follows to predict the corresponding period *ACWP*:

$$pACWP(t) = pBCWP(t)\ (1 - (\mu(\%pCV) \times D)/(\mu(ERCT) - 1)),$$
$$1 < \ t \ \leq T \times (1 + D),$$

where *pACWP* denotes the *ACWP* for one period only.

Future Value of Cost Estimate Under a Given Schedule Deviation

The total development cost conditional upon a given schedule deviation of *D*% of the planned schedule *T* is simply the estimated cumulative *ACWP* at project completion subject to that deviation. Thus:

$$\text{Total Development Cost} = ACWP(T \times (D + 1)) \ .$$

However, this expression is simply the sum of period costs and does not factor in time value of money. To be able to compare the total development cost to the future market payoff, we need to express the costs in future value terms at the projected completion date because the market payoff will be realized only then. At a constant, continuously-compounded annual risk-free rate of interest r_f, the FV of cumulative cost at period t is given by:

$$FV(ACWP(t)) = pACWP(t) \times \exp((D + 1) \times (T - t) \times r_f/12) + FV(ACWP(t-1)).$$

According to standard market theory, the risk-free rate should be used to carry the period values to the future because development costs are assumed to be subject to only private risk. Since the estimated future cost is already conditional upon a given schedule deviation, that risk has already been factored into the estimate.

Finally, the future value of the total development cost subject to a deviation of D% from a planned schedule of T is given by:

$$FV(\text{Total Development Cost}) = FV(ACWP((D + 1) \times T)).$$