Communicating Project Drift
Through Cost/Benefit Scenarios

David McComb
First Principles, Inc.

Jill Smith Slater
University of Denver

Software risk-management practices acknowledge the vital, albeit difficult imperative of communicating the status of project risk to stakeholders while attempting to mitigate and/or control risk manifestations. The problem is that risks are typically dynamic, unpredictable, and may be outside the purview and control of the project manager. This article presents a communication mechanism to explain the phenomenon of “project drift” through a series of abstract cost/benefit scenarios. The scenarios may be used either separately or in various combinations to continually reassess risk both at project inception and in light of project history to date. Three important aspects of the cost/benefit scenarios are that they are (1) conceptually simple, (2) useful in assessing and validating some decisions that might not be apparent in the absence of this type of evaluative model, and (3) politically neutral in that they may be used to explain project drift without affixing blame.

Change is perceived as difference, and difference is sometimes unnoticeable when it is continuous and fluid.
— Gilbreath (1986), Winning at Project Management

Like the growth of one’s own children, change in the status of a software project may occur so slowly as to go undetected on a day-to-day basis. Yet it is clear that something is astir in most major software undertakings. Late schedules, excessive costs, unacceptable performance, and user dissatisfaction are often cited ailments. Difficulty of specifying initial requirements, commercial pressures with accompanying tight deadlines, and/or the expansion of requirements after project initiation often drive change requests (Joch & Sharp, 1995).

Distributed computing intensifies problems. For example, distributed computing may lead to (1) duplication of resources, (2) decrease in standardization, (3) difficulty in meshing user needs with those of the overall organization, and (4) minimal transfer of learning from one project to another (Jenkins, 1994). Because distributed computing projects are more widespread, emergent risk problems become exposed to increased managerial scrutiny.

Software process improvement (SPI) efforts in general, and a subset of SPI, risk management, in particular, attempt to improve the software development process in the face of these obstacles. This paper addresses risk management issues and offers a series of conceptual cost/benefit scenarios useful in understanding risk dynamics and in conversing with external management about the cause and consequence of unfolding risks.

Risk Management and Communication.

The Risk Management Program sponsored by Carnegie Mellon’s Software Engineering Institute (SEI) affirms three objectives: risk prevention, risk mitigation and correction, and ensuring safe system failure if the project aborts. Recognizing the difficulty of predicting all risks in advance, the overriding objective of SEI is risk mitigation and correction (Higuera & Haimes, 1996):

The goal of SEI Risk Program is to enable engineers, managers, and other decision makers to identify, suffi-
ciently early, the risks associated with software acquisition, development, integration, and deployment so that appropriate management and mitigation strategies can be developed on a timely basis. Time is critical and the goal is to act early before a source of risk evolves into a major crisis. *In other words, being mainly reactive in risk mitigation and control rather than proactive in risk prevention and control is at the heart of good risk management* [italics in the original] (p. 2).

The uncertainties associated with software development require effective communication. This means speaking openly and sharing concerns on the part of everyone involved both within the domain of the project and with external stakeholders including user management and end users (if different). The SEI Risk Program recognizes the pervasiveness of communication and importance of internal and external communication of risk management by placing communication as an integral component of every risk activity rather than as a supplemental action (Higuera & Haimes, 1996):

… In order to be analyzed and managed correctly, risks must be communicated to and between the appropriate organizational levels. This includes levels within the development project and organization, within the customer organization, and most especially, across that threshold between the developer, the customer, and where different, the user (p. 20).

However, speaking openly with stakeholders outside the project team is often a significant problem because of disparate goals. Understandably, these stakeholders (particularly line managers) are subject to pressures of a different kind than quality software processes. Practitioner literature stresses the difficulty of convincing managers outside the project that time spent improving the development process (including acting promptly on risk mitigation) is important (Strehlo 1996, August 5, 19, 26); Jenkins, 1994; Joch & Sharp, 1995). Used to basing corporate direction on financial arguments, some managers disregard software process improvement efforts because it is difficult to forecast costs (Computer Finance, 1995).

More effective communication structures are needed to meet objectives of risk management and control. One structure is introduced here. The structure offers a generic set of abstract systems diagrams based on cost/benefit analysis diagramming conventions.

Communication by Abstraction

High-level abstraction permits a project manager to convey general risk direction based on archetypes, or generalized stories. Peter Senge modeled the use of archetype diagrams in the *Fifth Discipline* to portray organizational events in terms of broader system implications (Senge, 1990). Archetype diagrams serve as a template to describe events specific to an organization.

The motivation for high-level abstraction is that one may present arguments with less chance of political repercussions. An example from another field is a recent practice in organizational development called the Search Conference. This is a forum for all organizational stakeholders (or a deep, representative slice) to create a common vision and action plans based on that vision. Facilitators suggest that if a diverse group can address issues at a high level, removed from political agendas, they are more likely to achieve agreement on principles (Emery, 1995; Emery & Purser, 1996). The cost/benefit diagrams adhere to this reasoning.

Cost/benefit Diagrams

Cost/benefit analysis is common business language. The diagrams remove the quantification of costs and benefits to provide a conceptual method to explain and manage project drift. Project drift is the tendency of a project to move across its predetermined cost/benefit zone due to changes occurring during the process of project management. Three important aspects of the cost/benefit scenarios are that they are (1) conceptually simple, (2) useful in assessing and validating some decisions that might not be apparent in the absence of this type of evaluative model, and (3) politically neutral in that they are may be used to explain project drift without affixing blame.

The following diagrams provide a lens through which project drift may be explained and managed. The fundamental principle is to reduce the entire project funding decision (or continuation of funding) to a simple cost/benefit equation. Simply defined, a successful project is one that is worth more than it costs.

A very real issue is that many systems being developed or contemplated currently rely extensively on intangible benefits or tangible benefits that are difficult to quantify. Nevertheless, management has the job of determining the overall value and benefit of systems projects and weighing the benefits against their total expected cost. Thus the diagrams assume that any strategic or non-quantified reasons for proceeding with the project are included in the benefit equation. Total expected costs include all planning, design, implementation, conversion, and training costs. Figure 1 is a very simple abstract representation of the context against which to mark changes in a project’s economics. The figure illustrates the total expected benefit of a project, total expected cost, and a diagonal line representing possible break-even points.

Each project has a position within this decision space. Figure 2 positions a typical project. In this case Project A has some moderate cost and some benefit in excess of its cost, giving it a favorable cost/benefit ratio. In the typical justification section of a project plan, the cost/benefit analysis would eventually boil down to being the vertical distance that this project is above the break-even line. In other words, to what