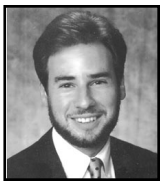

Steve McConnell

Feasibility Studies

EDITOR: Steve McConnell • Construx Software Builders • stevemcc@construx.com

Is a cancelled project a bad project? After surveying about 8,000 IT projects, the Standish Group reported that about 30 percent of all projects were cancelled ("Charting the Seas of Information Technology," 1994). Capers Jones reports that the average cancelled project in the US is about a year behind schedule and has consumed 200 percent of its expected budget by the time it's cancelled (*Assessment and Control of Software Risks*, Yourdon Press, 1994). Jones estimates that work on cancelled projects comprises about 15 percent of total US software efforts, amounting to as much as $14 billion per year in 1993 dollars.

In spite of these grim statistics, cancelling a project is, in itself, neither good nor bad. Cancelling a project *later than necessary* is bad. The trick is to perform the minimum amount of work necessary to determine that the project should be cancelled.

Press Cancel to Exit or OK to Continue

How do you cancel a project with the least work? One of the most effective ways is to conduct a feasibility study to determine whether the full-scale project is workable. This study culminates in a feasibility review, at which the project team, customer, or upper management make a go/no go decision about the rest of the project. The review usually involves a meeting, but sometimes the project team simply distributes study materials for individual examination.

Feasibility studies are a time-tested practice, but they aren't used very much. A KPMG survey found that 84 percent of companies that had had runaway projects proposed to use feasibility studies as one means of preventing future problems ("Runaway Projects—Cause and Effect," *Software World*, Vol. 26, No. 3, 1995). This suggests that these companies hadn't performed feasibility analyses of their runaway projects. (If they had, why would they expect performing them in the future to make any difference?)

One reason feasibility studies aren't used very often might be the term "feasability study" itself. The term conjures up questions of technical feasibility, and for those of us working in mainstream languages on mainstream computers, questions of technical feasibility rarely enter our minds. If we know our project is technically feasible, the thinking goes, why would we need to conduct a feasibility study?

For a few projects, technical feasibility is a significant concern: Is it technically feasible to build a Star Wars missile defense system? Is it technically feasible to build a natural language English-to-French translator? For most projects, however, feasibility depends on nontechnical issues: Are the project's cost and schedule assumptions realistic? Does the project have an effective executive sponsor? Does the company have a business case for the software when the real cost—rather than the initial blue-sky, wishful-thinking cost—is considered?

What Work is Involved?

During the feasibility study, the project team should create or obtain the following materials:

- clear commitment from the project's key decision maker or executive sponsor
- vision statement for the project
- business case for the software
- original effort and schedule targets
- current effort and schedule estimates
- list of the top risks to the project and plans to manage each

Continued on page 119

Continued from page 120

- detailed user interface prototype, if the system has a significant user interface element
- requirements specification
- software quality assurance plan
- detailed software development plan

Creating each of these materials addresses one or more significant common project hazards. As the Standish Group survey noted, poor requirements, lack of effective executive sponsorship, and inadequate planning are all major causes of project failure.

If the team can't prepare these materials for the feasibility study, don't hold the review meeting, because you won't have enough information to determine the project's viability. If the project team tries to create these materials and repeatedly fails to do so, assume that the project is somehow being prevented from preparing for success and faces a great risk of cancellation downstream.

The amount of calendar time required to create these materials depends mostly on how much work is needed to identify the software's requirements. If end users know exactly what software they want built, this period might take only 10 percent of the software's total development schedule. More typically it takes up to 20 percent of the total development schedule. On some projects, the hardest part of development is helping users figure out what they want built, so occasionally this part of the project can take 25 percent or more of the total development schedule. The initial funding request and plans for the feasibility review should account for this variability.

The Feasibility Review

Focus the feasibility review on the following questions:

- Is the product concept viable?
- Will it be possible to develop a product that matches the project's vision statement?
- What are the current estimated cost and schedule for the project?
- How big is the gap between the original cost and schedule targets and current estimates?
- Is the business case for the software justified when the current cost and schedule estimates are considered?
- Have the major risks to the project been identified, and can they be surmounted?
- Is the requirements specification complete and stable enough to support remaining development work?
- Have users and developers been able to agree on a detailed user interface prototype? If not, are the requirements really stable?
- Is the software development plan complete and adequate to support further development work?

The work done during the first 10 to 20 percent of the project should sufficiently answer these questions and give the client or top management enough information to decide whether to fund the rest of the project.

Why Bother?

Breaking a software project into a feasibility study phase and a main development phase helps software organizations in at least three ways.

First, some people view any cancelled project as a failure, but a project cancelled at the 10- to 20-percent complete point should be considered a clear success. Cancelling one project that ultimately goes nowhere after it is 10 to 20 percent instead of 80 to 90 percent complete (or, as Jones points out, 200 percent complete) can pay for the exploratory phases of a lot of other projects.

Second, a feasibility study sets up a project manager to make more accurate funding requests than average. The project manager first requests funding for the feasibility study phase, during which the first 10 to 20 percent of the project is completed. After the feasibility study has been completed and the people holding the project's purse strings have made a "go" decision, the manager requests funding for the remainder of the project. At this point there will still be a large potential variation in project cost, but the exploratory work will reduce the cost variation from a factor of 4 either way to about 50 percent (Barry Boehm et al., "Cost Models for Future Software Life Cycle Processes: COCOMO 2.0," *Annals of Software Engineering, Special Volume on Software Process and Product Measurement*, J.D. Arthur and S.M. Henry, eds., Baltzer Science Publishers, Amsterdam, 1995).

Finally, requiring the project team to complete 10 to 20 percent of a project before requesting funding for the rest of it forces a focus on upstream activities that are critical to a project's success. Otherwise, these activities are often abbreviated or ignored, and the damaging consequences of such neglect won't become apparent until late in the project. ❖