

# Tool Support for Project Tracking

Prospecting for programmer's gold.

YOU TRACK A PROJECT TO MAKE SURE that it's following your plan—that it's meeting its schedule, cost, and quality targets. This helps you detect problems early, when there's still time to do something about them. If you don't track your project, you can't control it. And, if a project isn't being controlled, it's out of control. You have no way to monitor risks or know whether your plans are being carried out.

Effective project tracking is hard. Capers Jones reports that "software progress monitoring is so poor that several well-known software disasters were not anticipated until the very day of expected deployment" ("Patterns of Large Software Systems: Failure and Success," *IEEE Software*, March 1995). Similarly, David H. Kitson and Stephen Masters found that 75 percent of the 59 sites they assessed between 1987 and 1991 needed to improve project tracking and oversight ("An Analysis of SEI Software Process Assessment Results, 1987-1991," *Proc. 15th Int'l Conf. Software Eng.*, IEEE Computer Soc. Press, 1993).

Still, the situation doesn't have to be so grim. The capabilities of modern software tools are coalescing in exciting ways that make effective project tracking easier than ever.

**KEEPING YOUR STORIES STRAIGHT.** Version control isn't just for source code anymore. In addition to providing for archiving and retrieval of multiple versions of code, most version control systems will store any electronic file: documents, project plans, spreadsheets, design diagrams, test cases, and so on. Once these work products have been placed under networked version control, anyone who has permission to access the system can examine project plans, requirements, designs, coding standards, user interface prototypes, or other work products. Getting a current copy of a document does not depend on finding the person who "owns" that document. Although this might seem to be a minor benefit, in many projects it can be difficult to get a current copy of the requirements, for example, simply because the person in charge of that document is on vacation! Few software developers who have worked on projects using version control ever want to work again on projects that don't.

**HOME SWEET HOME PAGES.** The core indicators of a project's cost, quality, and schedule status should be made visible to the entire project team. One way to do this is to create a project intranet home page with links to general project information, including project planning and tracking information, technical work products, and project deliverables. Figure 1 on page 119 lists the contents that such a

**If you don't track your project, you can't control it. And if a project isn't being controlled, it's out of control.**

home page could have. You might also offer a similar Web page to clients, though you would typically not want to give them access to all the technical details and potentially sensitive information on your team's home page.

When you use both version control software and a project intranet home page, current status information is only a mouse click away. You can embed code in the project home page to retrieve the latest version of each plan, graph, document, or other work product automatically. Alternatively, some version control systems can be configured to post the most recent version of each file to a specified network directory, including the directory where the home page files reside.

If the Web site offers these materials, project stakeholders—possibly including upper managers and customers—can easily access current project information without knowing how to use the version control system and without needing to attend a formal status review. Having materials available on demand eliminates a significant source of friction in some projects—the friction caused by sponsors not being able to obtain timely information about their project's progress.

*Continued on page 119*

## Editor:

**Steve McConnell**  
 Construx Software Builders  
 PO Box 6922  
 Bellevue, WA 98008  
 stevemcc@construx.com

Continued from page 120

**ARE YOUR RISKS DEFECTIVE?** An effective way to manage risk is to create a list of the top 10 risks to the project, then review and update that list twice a month or so. You can simplify this task by entering the risks into the project's defect tracking system, separately from the project's defects.

Defect tracking systems typically let you mark risks open or closed. You can prioritize them, and assign them to specific team members for resolution. You can print lists of risks by priority, by the amount of time they've been open, or by person responsible. You can also track the steps taken to resolve a risk, who took the step, and so on. Used in this way, a defect tracking system can take some of the tedium out of maintaining the risks list.

**WHO WAS THAT MASKED MAN?** Reporting good news projectwide and up the management chain seldom creates problems for the reporter, but reporting bad news can. There should be an anonymous channel for problem reporting and feedback that project members can use to report status and risk information up the management chain. If developers are turning their code over to testing later than scheduled, a concerned tester can report that. If testers are releasing builds to documentation that have not been well tested, a concerned technical writer can report that. If a project manager is exaggerating the project's progress in reports to upper management, a concerned developer can report that.

### Project tracking

- Percent of schedule used (actual)
- Percent of resources used (actual and planned)
- Percent of defects found (actual and planned)
- Graphs of actual vs. planned resources and defects
- Current task list
- Current defect list
- Top 10 risks list
- Anonymous feedback bulletin board

### Project planning

- Software development plan
- Quality assurance plan
- Detailed software construction plans

### Technical work products

- Requirements specification
- Top-level design document
- Detailed design documents
- Project coding standard

### Project deliverables

- Current version of the software
- Deployment document or cut-over handbook
- Release checklist

Figure 1. Project home page contents.

One way to support an anonymous reporting channel is to set up an electronic project bulletin board to which people can post comments anonymously. This bulletin board should be visible to other project team members and upper management—preferably, it should be directly accessible from the project's home page.

## The manager shouldn't become the bottleneck through which all status information must pass.

**MANY HANDS MAKE LIGHT WORK.** Automated tools make it much easier to track the status of each task. Tasks can be listed in an electronic planning tool such as project management software or a spreadsheet, stored in the project's version control system, and made accessible via the project's intranet home page. When a developer or tester completes a task, he or she can check out the task list from version control, update it to show that the task is complete, then check it back in. When this approach is used, the project manager isn't burdened with maintaining a detailed task list. More important, the manager doesn't become the bottleneck through which all status information must pass before it can be used by the rest of the project team and upper management.

**MANUAL LABOR.** Although current tools make collecting and distributing tracking information easier than ever, some small amount of manual work remains. About once a week, the project manager or the manager's assistant should review the project's progress. This review should include

- ◆ collecting summary data from the project planning and defect tracking tools,
- ◆ comparing actual tasks completed to the plan,
- ◆ comparing actual defects reported to predicted defects,
- ◆ comparing actual effort to planned effort,
- ◆ reviewing and updating the Top 10 Risks list,
- ◆ reviewing and acting upon anonymous feedback, and
- ◆ updating information on the project home page as needed.

Weekly data collection and analysis also lay the foundation for maintaining a software project log during the project and for creating a software project history at the end of the project.

**PUTTING THE PIECES TOGETHER.** Different project stakeholders are sensitive to different issues. When you expose task status, current risks, current defects, and anonymous problem reports on a project home page that all project stakeholders can use, you go a long way toward ensuring that project managers and upper management will have the information they need to address problems promptly and effectively. ◆