# Managing Virtual Teams:
## Getting the Most From Wikis, Blogs, and Other Collaborative Tools

M. Katherine Brown

Brenda Huettner

Char James-Tanny

# Chapter 5

# Project Planning and Tracking

> *Even if you're on the right track, you'll get run over if you just sit there.*
>
> — Will Rogers, humorist

Whenever two or more people share a task, the total amount of time required to complete that task goes up just a bit because the people involved need to spend additional time coordinating their efforts. Without coordination, they run the risk of duplicating effort or leaving portions of the task undone (or both!). As your team grows, the amount of time needed for this type of coordination effort (or project management) grows right along with it. Virtual teams must not only have a good plan, but also must have a mechanism for tracking the progress against that plan. There are many software packages created just for project planning, and many of the other tools discussed in this book can also serve as project tracking mechanisms.

# Planning

Start every project with a plan. If one person has to accomplish a task that will take one hour, you may not need to write a formal plan document, but you still probably have a plan in mind before you begin work. As the project grows larger, extends for a longer period, and involves more people, it becomes more important that the plan be comprehensive and well-documented.

Lots of reference books and courses are available that teach the basics of project planning and cover a wide variety of specific methodologies. There are also many organizations of professional project managers such as the Project Management Institute (http://www.pmi.org), the International Project Management Association (http://www.impa.ch), and the American Management Association (http://www.amanet.org). Whichever methodology or tools you use, a good project plan must incorporate the following elements:

- Scope of the project
- Assumptions on which the rest of the plan is based
- Requirements for project success
- Tasks included in the plan
- Schedule for completion of each task
- Cost estimate

## Scope of the Project

This first section is the foundation on which the rest of the plan is based. The scope lays out the high-level vision for the project and provides some of the justification for the effort. The description of the project scope must include the following elements:

- **Purpose**. Why are you doing this project? What will be different when you finish the project successfully? Without a clear purpose, you will not be able to tell if the project is successful or not. For example, a project purpose might be "to shorten the customer's learning curve."
- **Goal**. What are you going to do in order to accomplish the purpose? A project with the purpose of shortening the customer learning curve might have a goal of "simplify the user interface" or "improve available training materials." Note that purpose and goals are *not* the same!
- **Major milestones**. How are you going to accomplish the goals, and by when? You do not need to duplicate the task or schedule sections (those come later in the plan); just summarize them at this point. Sticking with our hard-to-learn software example, the milestones might involve conducting a usability study, revising interface programming, and issuing a new release of the software within six months.

## Assumptions

If you are planning a picnic, you make an assumption that the weather will be nice on the scheduled picnic day. If the weather turns stormy, the picnic can be cancelled without anyone blaming the quality or preparation of the food. When planning a project, you make a series of assumptions like "we will have the same personnel for the duration of the project" and "software will be available for testing $x$ days before product release." Documenting these assumptions in the plan accomplishes three things: You get buy-in from management that your assumptions are correct, you increase the likelihood that the assumptions will be borne out as true, and if something does change,

you have justification for changing the other elements of your plan (like schedule and budget) accordingly.

## Requirements

While the assumptions are somewhat general ("equipment will be available for testing"), the requirements section gets more specific. This section includes the details of the things needed to accomplish the stated goals.

■ **People**. Work does not happen by itself; you need people to make your project a success. Include a description of your team in the project plan (see Chapter 2, "Setting Up a Virtual Team"). When describing the people who will work on your project, include the critical non-team members as well. Will you need someone from QA to test your product? Does your company require approval from a legal department before release? Will you be using copy center personnel?

■ **Tools**. Imagine an empty room. What does your team need in that room in order to complete the project? Though some things may seem obvious to you (such as a computer and a license for each of the software tools for each of the team members), document them here anyway. Do you need to upgrade software versions to ensure compatibility across team members? Do you need to provide office supplies (ink, paper, etc.) for remote team members? Do you need to buy special hardware or supplies for testing purposes? For example, if the project is building a multipurpose printer, you probably need a variety of types and sizes of paper stock on which to test. Include reference materials that you want team members to use (books, CDs, online libraries such as LexisNexis, or specific server access), and permissions, logins, and IDs needed to access your servers or to get behind firewalls. Finally, document any special training that team members may need. The cost of a training program is often more than offset by the savings in a team member's time and effort.

As an example, the following table shows some of the tools requirements we had for each writer and for this book project.

**Table 5-1: Example tools requirements**

|          | Writers | Project |
|----------|---------|---------|
| Hardware | Laptop or desktop computer<br><br>Microphone and speakers for VoIP<br><br>Printer | Server to host wiki |
| Software | Microsoft Word 2002<br><br>Skype 2.0 or higher<br><br>AOL version 9 or AIM<br><br>Email access | Wiki software<br><br>Demo versions of other software discussed in book |
| Other    | High-speed Internet access<br><br>Permission to access wiki<br><br>Style guide from publisher<br><br>Standard office supplies (paper, ink, etc.) | |

Once you have listed everything you need, you can check to see which of these things you already have. In the above example, some of these things we owned already — each of us owns a laptop, so though computers are required for the project, they would not have to go into the budget. Some we had to purchase, like an external PC microphone.

It is often difficult to determine just how detailed a task list needs to be. Certainly, you do not want to spend more time creating the list than it will take to accomplish the tasks! The sample list above goes to three levels (7.3.2). For any project, the lowest level should be small enough for one person to accomplish but big enough to measure in hours.

This might seem like a lot of effort up front, but it will serve you well over the course of the project. In addition to helping you create a more accurate schedule, a detailed task list helps to better track the progress of the project (see "Tracking Your Progress" later in this chapter). A task list is the input to the "Work Breakdown Structure," or WBS, which is the foundation of many project management software packages. If you take the time now to create a detailed task list, it can also become a template for later, similar projects.

## Schedule

Once you have a task list in place, simply assign a time frame to each task to determine your project schedule. Sounds easy, right? The difficult part is knowing how long each task will take. If you are working with or for an organization that has done many similar projects in the past, you may have access to the statistics or metrics about those prior projects. Metrics can give you a good starting place and help to improve your estimation accuracy over time.

But most of the time, you just have to make an educated guess. There are two ways to approach scheduling. If you get to pick your deadline, assign each task a start date and a duration (how long it takes to accomplish that task). The start date for each will usually follow from the end date of the task before it. If you are using project management software, these are called dependencies, or indications of tasks that must be complete before the current task can begin. For example, you cannot begin a review task until the development task is complete. When you enter all this information into project management software, the system calculates the end dates and total time spent for you.

In real life, however, you will often be given a deadline by upper management, or have a deadline based on some external event like a major trade show. If you have an external, unmovable deadline, use a percentage method to determine approximate start and end dates. A project that has a research phase, a development phase, a testing phase, and a production phase may break down to 15 percent research, 60 percent development, 15 percent testing, and 10 percent production. If your deadline is 12 weeks away, you know that the research must be complete and the development must start after 1.8 weeks. Remember that virtual teams may require some extra time for communication issues. Go through your task list and determine how long each task should take, and then compare your task durations to the schedule time frames based on percentage. If you have a conflict, such as a task that takes four weeks to accomplish but is due in just one week, you then have two choices: You can either reduce the scope of the project to decrease the number of tasks you take on or you can add more people to your team.

> **Note:** Be cautious when adding people to alleviate a tight schedule. Some tasks cannot be speeded up by assigning more workers. *"Crash programs fail because they are based on the theory that, with nine women pregnant, you can get a baby a month."* — Wernher von Braun, rocket scientist

Project management software does a pretty good job of assigning work only to workdays (that is, skipping weekends), and most packages include a function that lets you specify major holidays. If you have an international team, make sure you are taking into account the holidays for each country. For example, many people in the United States do not work on Independence Day on July 4, but people in England do not celebrate that holiday. Labor Day is a holiday on May 1 in Hong Kong, Peru, and Finland; on November 23 in Japan; and on the first Monday in September in the United States. For a quick reference of international holidays, see http://www.earthcalendar.net. You can view the lists of holidays by date, by country, or by religion.

In addition to allowing for holidays, try to take into account any other times that people may be unavailable for work. For example, if your project is scheduled to last a year, do not schedule 52 weeks of effort for a person who gets four weeks of vacation per year. There may also be delays if your industry has a large annual conference that most of your team will attend (this can cost you a week or more) or if you have team members in an area with high potential for severe weather. For example, the northeast part of the United States often gets severe snowstorms that can knock out power (and communications) for days in January and February.

No schedule can be 100 percent accurate, but the more you can allow for various delays, the more accurate your schedule will be.

Though project management software does an excellent job of calculating schedules and lets you easily track progress throughout a project, you do not have to use dedicated software. For very small projects with a small team and short time frame, you may be able to use tools like Microsoft Word and Excel to create and distribute your schedule. Whichever method you use, ensure that all team members have the software required to read your output.

## Cost Estimate

At this point, you know how long your project will take, how many people will work on it, and which tools you will need to purchase. You just need a few more items to create the cost estimate for the project.

Time is money, and for most projects, the cost of your team's time will be the largest percentage of your expenses. When calculating the cost of a person's time, salary is just the starting point. You will need to add in other costs associated with that person (called overhead or "burden"). Expressed as a percentage, it includes all the other expenses that are associated with each person, such as benefits, insurance, taxes, office space, etc. Some companies use a single percentage for all employees, while others distinguish between types of workers to determine more accurate numbers. For example, an engineer who makes $60 per hour in salary might need a 25 percent burden, resulting in a $75 per hour cost to the project. Your company's financial department will be able to tell you what percentages to use.

If your project will be using sources outside of your company, include the estimates from those sources in your costs.

# Tracking Your Progress

All team members need to know where their efforts fit into the larger project, how they are doing compared to the other team members, and how the team is faring compared to initial project estimates. Ensure that you have processes in place to allow team members to perform the following basic tasks:

- Update progress on individual tasks. This category may also include bug tracking or similar types of error management software.

- View overall project status.

- Adjust the plan as needed.

- Maintain an archive for documenting actions and decisions.

- Calculate and store metrics that may help improve accuracy of estimates for future projects.

All of these can be accomplished by a combination of individual applications like word processing software, calendar software, spreadsheets, and file sharing systems, or can be quickly and easily processed by dedicated project management tools. There are many different packages available, with a wide variety of capabilities and a wide range of prices.

Before choosing a package, determine which features you need, some of which are described below:

- **Output types.** Most packages let you enter tasks and milestones, and then generate a variety of charts and reports. Two common charts are Gantt charts and PERT charts, which are graphical representations of tasks and schedules. Higher-end project management systems increase the options for output and may include capabilities for custom reports and graphics.
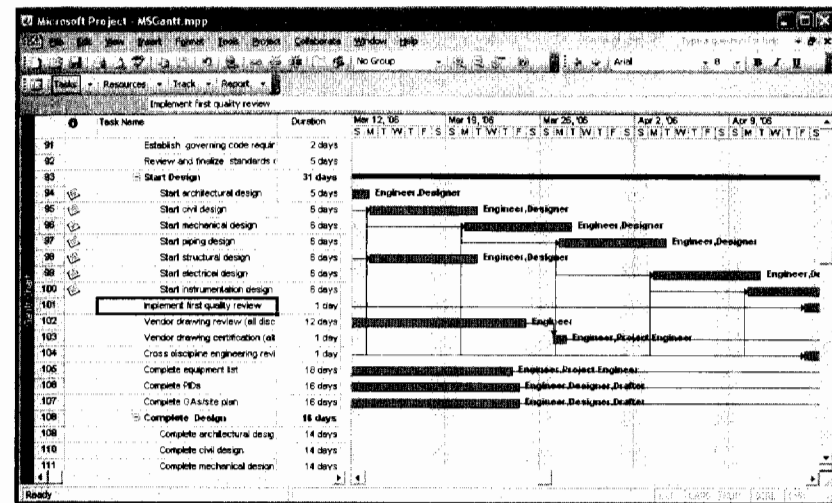


Figure 5-1: Gantt chart for an engineering design project

- **Compatibility.** Consider the existing software within your team, and also the project management needs of other teams with which you might need to share information. Look at the compatibility of output files with other software packages. Can a person without the project management software view the output? Can the output be easily pulled into other applications, like word processing or spreadsheets? If you are already using a database product like QuickBase, you may want to stick with that tool for the project management tasks rather than invest in purchasing and learning new software.

- **Security.** Some project management packages now have file management functions like the ability to check files in and out, which can be useful when many people are accessing the same task and scheduling files. Particularly with the newer web-based project management systems, consider the level of security, passwords, and IDs.

- **Other tasks**. How detailed do you want to get in the tracking of your project? Do you need to be able to track individual workloads as well as team progress? Do you want to track specific issues? Is there a need for tracking financial information?

- **Desktop vs. web-based**. With the proliferation of distributed teams, many products are now being hosted by the vendors so that users can access their project management information from any web-enabled system. Unlike the traditional desktop packages like Microsoft Word or Kidasa Milestones, the web-based systems require that you set up user names and passwords. Generally, you pay a monthly service fee rather than purchase the software outright. Web-based systems make virtual collaboration a bit easier than desktop systems do, but they might not have all the security functionality that some companies require.

**Table 5-2: Examples of project tracking software**

| Tracking Software | Description |
|---|---|
| @Task (http://www.attask.com) | Web-based; files can be checked in and out |
| eProject (http://www.eproject.com/products/index.htm) | Web-based; files can be checked in and out, has forum/message board capability |
| Kidasa Milestones (http://www.kidasa.com) | Desktop; limited document and resource management functionality |
| Microsoft Project (http://www.microsoft.com/) | Desktop; many outputs, customizable and compatible with most other tools, does not have check-in/check-out capability |

# Incorporating Progress into Your Plan

The plan you create does not go away just because it has been approved. Once you begin implementation, you will need to measure your progress against your initial estimates. No plan is ever 100 percent accurate, but by measuring your estimates against the actual results of your project, you can create a more accurate plan the next time around.

When you begin tracking a project, ensure that you are keeping track of the information that will be most useful to you later. Much of this may be determined by your upper management — are they more concerned with hours spent or dollars spent? Do they want to track individual productivity, or are they more concerned with process improvement? Are they more interested in evaluating the process with a view to improving future projects, or are they focused on measuring the specific output values (quality, cost, on-time) of the current project?

For most projects, the raw data can be divided into three categories:

- **Input**. The cost of the project and the amount of effort expended, usually tracked as number of hours devoted to each task or expenses accrued, such as software purchases, outsourced services, or travel.

- **Process**. A measurement of things that happen during the course of the project — the number of review cycles, for example, or the number of mistakes made or bugs fixed. These measures can include variance measurements — that is, how far over or under budget you are, or how far ahead or behind the estimated schedule.

- **Output**. A measurement of the finished project — the number of widgets produced, the number of lines written, the pages of text printed.

If you are using a standard project management package, many of these numbers will be automatically stored for you as long as you input the raw data. Otherwise, you will need to track them manually, as did the users of the first Gantt charts back in 1910 (see Figure 5-2).
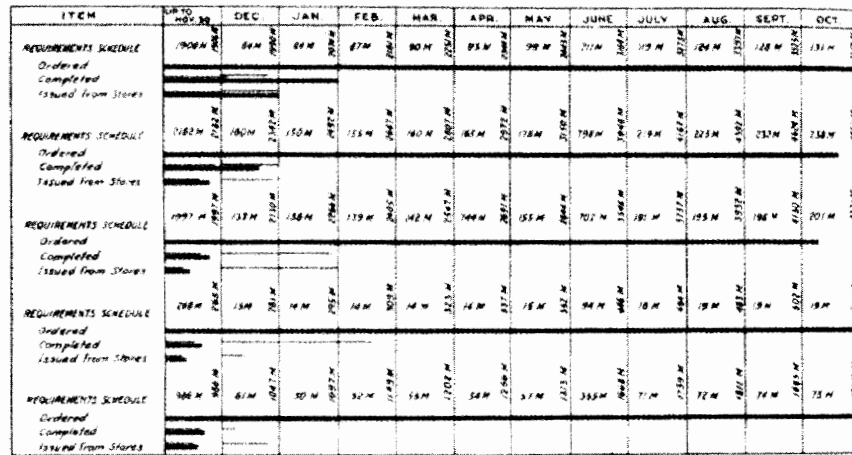


Figure 5-2: Gantt chart from *Industrial Management* magazine, 1918

The key here is to be diligent in keeping up to date with the status of your plan. Record the actual hours expended, the actual number of lines written, or whatever measure is appropriate for your project.

In the example of this book project, we began by counting chapters, and then moved on to counting pages. Though we did not measure the number of hours expended, we did have a scheduled due date.

For most projects, particularly ones of longer duration, update your actual numbers fairly frequently but at least once a week. Note that at this point you are simply storing information, not yet analyzing it in any way. When you begin the analysis of the numbers, by looking at percentages or by comparing values in different categories, you are actually creating the metrics by which you can evaluate the success of your project (see Chapter 9, "Evaluating Project Success").

## Summary

Every project needs a project plan. Ideally, project plans need to include the scope, assumptions, requirements, tasks, schedule, and costs for the project. The more detailed your plan, the better you will be able to track your progress as time goes by.

## Related Resources

Garton, Colleen, and Erika McCulloch. *Fundamentals of Technology Project Management*. MC Press Online, 2004.

Project Management Institute. *A Guide to the Project Management Body of Knowledge*, (PMBOK Guide). 3rd ed. PMI Publications, 2004.