CHAPTER 3

Microsoft Project and the Project Management Domain

In this chapter

Chapter 3 provides the context for the use of Microsoft Project within the domain of project management. It identifies accepted standards and methodologies that are in use within the field and provides some examples of how they are used with Project.

History of Project Management

When Patrick Henry said, "I know of no way of judging the future but by the past," he could have been talking about project management. When faced with projects that have never been done before, all project managers can do is look at what has come before them.

While project management has been practiced for thousands of years, evidenced by the Egyptian and Roman dynasties, modern project management can be traced back to the late nineteenth century and the rise in large-scale government projects and growing technological advancements. Fredrick Taylor, the Father of Scientific Management, applied scientific reasoning to analyzing and improving labor, and Henry Gantt studied management of Navy ship construction during World War I. Gantt's use of charts, task bars, and milestone markers is still practiced today, and they bear his name. One of the major projects that brought detailed project planning, controlling, and coordination to the forefront was the Hoover Dam project, which involved $175 million dollars, six different companies, a major worksite with no existing infrastructure, and approximately 5,200 workers. The project was brought in under budget and ahead of schedule.

After developments in project management during two World Wars and the growing Cold War, major changes to project management were brought about with the launch of Sputnik. Fearful that the United States was falling behind in the race to space, the USA introduced several major programs to focus on science and exploration. Several agencies, including the Advanced Research Project Agency, a high-level research and development program that later became DARPA and NASA were founded. These agencies led the way in the development of project management.

Two other major developments for project management to grow out of this period were the Critical Path Method (CPM) and the Program Evaluation and Review Technique (PERT). CPM was devised by Du Pont and Remington Ran for use with the UNIVAC-1 computer mainframe. PERT was invented by the Program Evaluation Branch of the Special Projects office of the U.S. Navy, for use with the POLARIS missile program, and was also used on the Apollo program for NASA. CPM/PERT gave managers more control over extremely large and complex projects, but could only be calculated within large mainframe computer systems and were used mainly for government sector projects.

With the computer revolution of the 1980s and the move from mainframe computers to personal computers with the ability to multitask, project management software became more accessible to other companies. The Internet and networked systems only made project managers more efficient at controlling and managing the different aspects of their projects. More information on previously completed projects is available today than ever before, making the project manager's job of estimating the future by looking at the past easier than ever.

Exploring Project Management Industry Standards

Anyone can create a file using Microsoft Project. Organizing that file into a logical flow of work, however, requires a solid understanding of how projects should be managed and decomposed into logical units. To understand project management, one must understand the standards and methodology behind it. While Gantt Charts and other similar resources are used in almost all project management schedules, there are several different ways of using those resources. This chapter will discuss prominent industry standards that are often used to set a framework for building schedules. A variety of methodologies, team styles, and life cycles will also be explored. The approach and techniques will vary but the software can still be used to support virtually any approach to scheduling that an individual or organization chooses to use.

Project Management Body Of Knowledge (PMBOK)

The Project Management Institute, or PMI, is an internationally recognized organization that has developed standards for the domain of project management including standards for portfolio management, program management, project management, and work breakdown structures. PMI has several hundred thousand members in over 65 countries. It is widely recognized for its certification programs and continues to grow through a combination of volunteer efforts, certification programs, local chapter events, international seminars, and special interest groups.

The standards created by PMI are authored by a vast network of project management professionals who volunteer their time to create and update these standards on a regular basis. The standards groups are from many different countries across the globe; they research topics and collaborate to bring together the latest thinking and techniques from their collective experience.

The PMI standard that is of primary importance for this chapter of the book is in its third edition and is known as: “A Guide to the Project Management Body of Knowledge,” also known as the PMBOK Guide. It will be discussed in some detail in this chapter to help in understanding all of the components that should be considered when creating a schedule. Because PMI is a standards and certification organization, it does not prescribe methodologies or “how to” approaches, rather it defines specific standards and offers certifications in the field of project management. The PMBOK provides a context for a way to do things, rather than the process that should be followed. Inexperienced project managers often try to make their schedules follow PMBOK as if it were a recipe for success. This can lead them into traps and complexity that is not useful in completion of their projects. Instead, they should look to the PMBOK for support of the methodology and life cycle that they choose to follow.

TIP

PMI makes a clear distinction between a *project plan* and a *project schedule*. The *plan* is a formal document that includes narrative on communication approaches, assumptions, deliverables, and execution of the project. The *schedule* is one component of the plan that focuses on the timeline for the activities to be performed. As scheduling tools become more sophisticated, they are gradually including more elements that used to reside only in the plan. Microsoft Project Desktop still focuses on scheduling functions, but the server components have added capabilities to support more of the plan functions.

The PMBOK Guide has established five process groups to define the project management process. These processes are as follows:

- **Initiating Process Group**—Defines and authorizes the project or a project phase
- **Planning Process Group**—Defines and refines objectives, and plans the course of action required to attain the objectives and scope that the project is to address
- **Executing Process Group**—Integrates people and other resources to carry out the project management plan
- **Monitoring and Controlling Process Group**—Regularly measures and monitors progress to identify variances from the project management plan so that corrective action can be taken when necessary to meet project objectives
- **Closing Process Group**—Formalizes acceptance of the product, service, or result and brings the project or a project phase to an orderly end

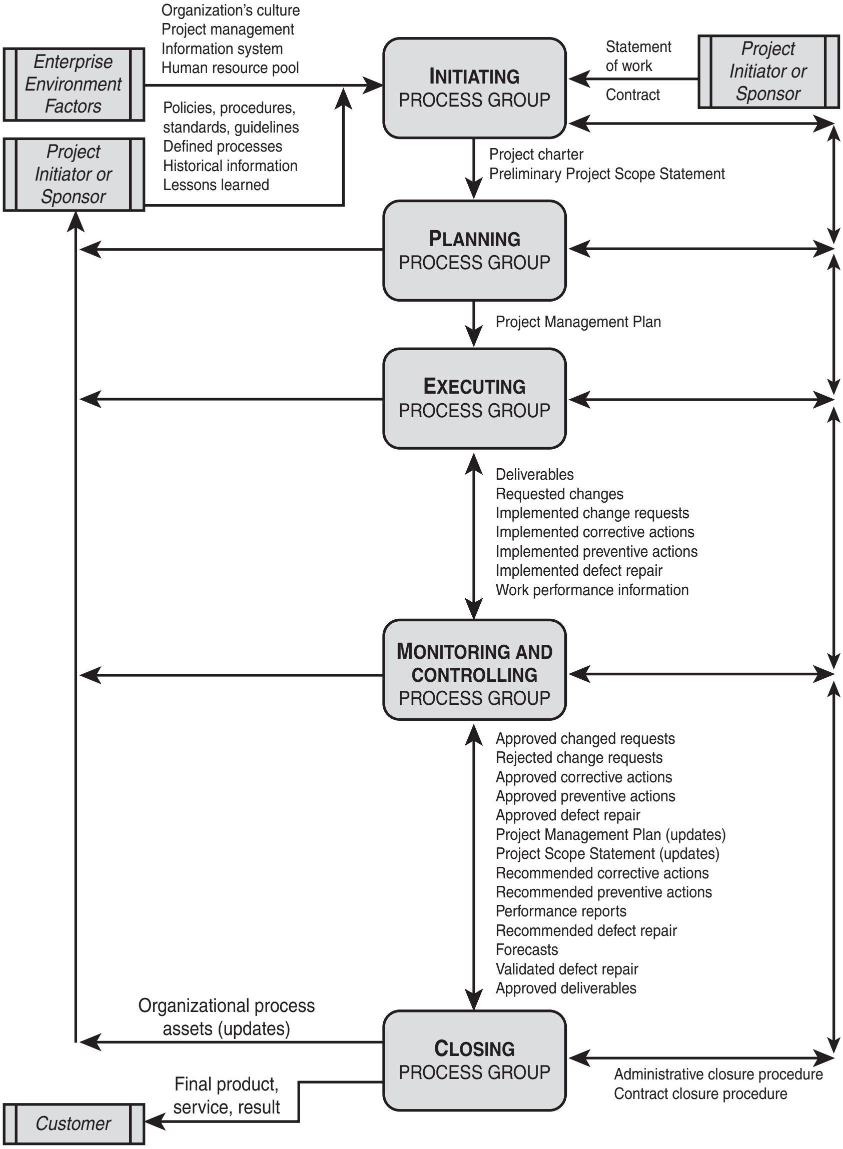
Projects are created and implemented in environments that are larger in scope than the projects themselves. All projects must have a beginning and an end, as shown by the Initiating and Closing Process Groups. In between, a project will be engaged continually with the other three process groups, as shown in Figure 3.1.

The PMBOK identifies nine knowledge areas that a project manager should consider throughout the entire life cycle of a project. Knowledge areas focus on a specific aspect of the overall domain and identify the elements that need to be considered to properly manage a project.

- **Project Integration Management**—This knowledge area looks at the processes and activities that are needed to identify, define, combine, unify, and coordinate the different actions within a Project Management Process Group.
- **Project Scope Management**—This knowledge area handles scope planning, scope definition, creating a WBS (decomposition of the scope into smaller components), scope verification, and scope control.
- **Project Time Management**—This knowledge area concerns five different steps: activity definition, activity sequencing, activity resource estimating, activity duration estimating, and schedule development.
- **Project Cost Management**—This knowledge area involves planning, estimating, budgeting, and controlling costs so a project can be finished within budget.
- **Project Quality Management**—This knowledge area determines policies, objectives, and responsibilities to meet a project's quality standards.

- **Project Human Resource Management**—This knowledge area helps organize and manage a project's team, the people necessary for the completion of the project.
- **Project Communications Management**—This knowledge area involves the processes that ensure timely generation, collection, distribution, storage, retrieval, and disposition of information.
- **Project Risk Management**—This knowledge area envelopes risk management planning, identification, analysis, responses, monitoring, and controlling of a project.
- **Project Procurement Management**—This knowledge area involves the processes necessary to purchase products, services, or results from outside the project team.

Figure 3.1
The relationship among the PMBOK process groups (taken from Figure 3-4, *PMBOK Guide, Third Edition*).



Enterprise Environment Factors
Organization's culture
Project management
Information system
Human resource pool
Project Initiator or Sponsor
Policies, procedures, standards, guidelines
Defined processes
Historical information
Lessons learned
INITIATING PROCESS GROUP
Statement of work
Contract
Project Initiator or Sponsor
Project charter
Preliminary Project Scope Statement
PLANNING PROCESS GROUP
Project Management Plan
EXECUTING PROCESS GROUP
Deliverables
Requested changes
Implemented change requests
Implemented corrective actions
Implemented preventive actions
Implemented defect repair
Work performance information
MONITORING AND CONTROLLING PROCESS GROUP
Approved changed requests
Rejected change requests
Approved corrective actions
Approved preventive actions
Approved defect repair
Project Management Plan (updates)
Project Scope Statement (updates)
Recommended corrective actions
Recommended preventive actions
Performance reports
Recommended defect repair
Forecasts
Validated defect repair
Approved deliverables
Organizational process assets (updates)
CLOSING PROCESS GROUP
Final product, service, result
Customer
Administrative closure procedure
Contract closure procedure

NOTE

Do not confuse the PMBOK process groups with life-cycle phases of projects (more on phases later in this chapter). This is a common tendency when a project manager tries to decompose a project into logical components. Process groups pertain to all projects; life cycles vary by the type of project, the domain of the work, the complexity and time frame, and many other factors.

The nine knowledge areas are specifically designed to work with the five process groups to identify possible areas for management within the scope of the project. They are referred to here as the “5×9 checklist.” When the two components are combined, they provide guidance for what elements should be considered at what time in a project. In the context of MSP desktop, the key knowledge areas are scope, time, and cost. These components will help you build the initial project schedule framework. All of the other knowledge areas will play a part in building the working baseline schedule, and it is recommended that each be considered in the context of the framework. For example, the resources to be applied to a project may involve both the Project Human Resource Management and the Project Procurement Management knowledge areas. Until you have a general idea of the scope, expected time-frames, and budget for that work, however, you will not be able to identify which area is most appropriate. The emphasis for each knowledge areas varies by phase of project; some will be more important in one phase than another, but all of the nine are used throughout the project.

PRINCE2

PRINCE2, which stands for *Projects in Controlled Environments*, is a project management methodology developed by the United Kingdom government. It is in its second release and was originally known as the PRINCE technique. The first release was established in 1989 by the Central Computer and Telecommunications Agency (CCTA) as a standard for information technology project management. Because of its success in IT, the methodology was republished about seven years later in a version that could be applied across many other disciplines. PRINCE2 was again updated in 2005 by the Office of Government Commerce (OGC) and has become the standard for project management in the UK, and is now used in 50 other countries. You can become certified in the use of PRINCE2 at either one of two levels: Foundation and Practitioner.

PRINCE2 uses a simple four-step process to explain what each project will need, as shown in Figure 3.2. This process is explained in more detail using the following eight different processes, sometimes known as the Validation, Quality, Verification, and Approval steps:

- **Start-up**—This is when a project manager is chosen. Then the need for the project is defined and outlined as to how it will be executed.
- **Direction**—The project manager, who reports to the Project Board, is responsible for managing the details. The Project Board is responsible for the overall success of the current project and defines the direction in which the project will be heading.
- **Initiation**—The Project Initiation Document is prepared and submitted to the Project Board for approval and possible revision.

- **Stage Control**—During this stage, the project is broken down into several different manageable stages. The number of stages will depend on the size and risk level of the project, and each stage must also plan for the succeeding stage. Before any new stage can begin, the current stage must be fully finished.
- **Stage Boundary Management**—At this stage, the Project Board must review the current stage and then develop the process for the next stage. It is only after the approval for the execution of the current stage and the planning of the next stage that the project can continue.
- **Planning**—This stage is used for deciding what products will be produced and what is required for their production. Then estimates are made for cost, time, and any other resources, plus any risk analysis, activity scheduling, and process streamlining that is necessary for the project.
- **Product Delivery Management**—This is the production stage, where the project manager confirms that the right goods are being produced correctly and on schedule.
- **Closing**—Once everything is finished, the project manager must perform a Post Project Review, which evaluates the outcome of the project. When this review is approved by the Project Board, the project is complete.

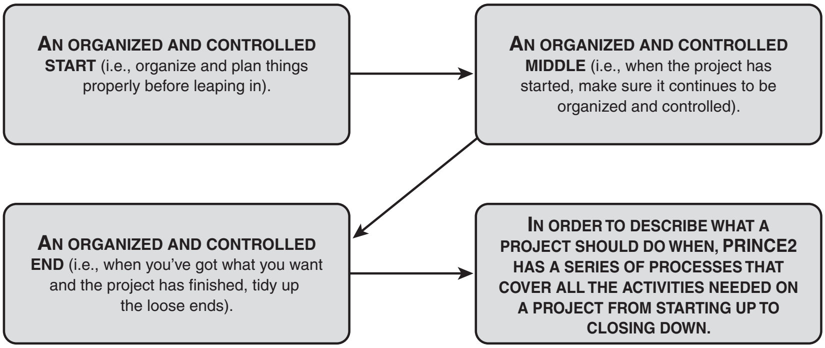

AN ORGANIZED AND CONTROLLED START (i.e., organize and plan things properly before leaping in).
AN ORGANIZED AND CONTROLLED MIDDLE (i.e., when the project has started, make sure it continues to be organized and controlled).
AN ORGANIZED AND CONTROLLED END (i.e., when you've got what you want and the project has finished, tidy up the loose ends).
IN ORDER TO DESCRIBE WHAT A PROJECT SHOULD DO WHEN, PRINCE2 HAS A SERIES OF PROCESSES THAT COVER ALL THE ACTIVITIES NEEDED ON A PROJECT FROM STARTING UP TO CLOSING DOWN.

Figure 3.2
PRINCE2 four-step process.

NOTE

> There is no conflict between PMBOK and PRINCE2. They can be used together if a project manager chooses to use both. PRINCE2 is a methodology and focuses more on deliverables, whereas PMBOK is a standard and focuses on the process and knowledge areas. PRINCE2 establishes a Validation of Process (through a very specific focus on deliverables and the activities around them), whereas PMBOK focuses on the processes used to manage the deliverables.

In addition to consideration of these standards and methods, project managers need to understand the environment in which they will be working before they create a schedule. They need to be aware of the various methodologies and approaches that can be used to help them (or confuse them, if they do not understand how and when the methodologies and approaches should be applied). The following section provides an introduction to this information.

WBS, Phases and Control Points, Methodologies, and Life Cycles

Many of the preceding terms are used by project managers to describe the approach that is used to define and execute a project. Each of these has been explained in many other texts and references. Since the focus of this book is using Microsoft Project, the topics are brought up here to provide context only; there is no attempt to provide the definitive use of any of the terms. Rather, the hope is that the reader can apply the concepts and techniques as appropriate when building a schedule. The work that needs to happen during a project's life does not automatically conform to a particular methodology.

Before building any schedule, the project manager must consider two key components: work decomposition (what work needs to be done, the Work Breakdown Structure or WBS) and managerial control (stages, phases, and life cycle requirements). The discipline used for either will depend on the environment in which the project is executed, so the formality will vary but both components must be considered. The tasks or activities and milestones (how the work will be accomplished) should not be defined in a project schedule until the WBS and control framework are determined. WBS will help the project manager set parameters around the scope of work to be done; the life cycle sets the controls in place for decisions during project execution. If these two components are kept in control, the project will have a much higher opportunity for success.

Work Breakdown Structure (WBS)

Step one in building a schedule is to begin with a Work Breakdown Structure (WBS) that allows decomposition of the scope of the project from major components to the smallest set of deliverables, called *work packages*. As a best practice, this process is completed before a true schedule is built. It can be done using Microsoft Project as long as ongoing "use" rules are defined and followed to keep the WBS components intact after the project is approved and baselined.

As mentioned earlier in this chapter, PMI has developed a standard for the WBS. It is a primary component of good project management practices because it forces the discipline of scope definition and control.

→ For detailed information of how to build a WBS, **see** Chapter 4, "Getting Started After the Business Initiative Is Approved," **p. 87**.

If the scope of your project is managed through a WBS, all of the tasks and milestones will be created in support of specific work packages and can be rolled up through the structure for tracking progress using Earned Value Management techniques. This practice eliminates some of the common failure points in project management, such as scope creep and fuzzy requirements. All work is clearly linked to the production of a deliverable, and progress against that deliverable can be monitored.

Managerial Control

So many terms are used in the context of managerial control that a few definitions are in order. Hundreds of resources are available to provide detailed explanations; the purpose here is context only. The hope is that these simple descriptions will help the user's understanding when building a project schedule, as discussed in the following sections.

Phases and Gates

Many organizations have established processes for deciding what projects will be approved and for overseeing the projects once they have been launched. In some organizations, the processes are rigorous and robust; in others, the processes may be simple guidelines that have been put in place to help project managers. In either case, a defined set of standard phases and control points (often called *gates*) simplify the decisions that need to be made when running a project. In most cases, templates can be created that standardize the phases and the required control points for different types of projects.

Phases and gates can allow more management control of the project, as they break the project down into smaller components. This helps to keep executive and team focus aligned on the same set of activities. A change between phases is usually defined by some kind of transfer. In many cases, the transfer requires a formal review before the project is allowed to move into the next phase. It is not unusual, however, for phases to begin before the completion of the previous phase, especially when the risks are judged as acceptable. Each organization will make its own determination of the level of control required.

Building the phases and control points into templates is an excellent way to minimize the amount of work that needs to be done when building a new schedule. Many examples are already available in Microsoft Project, and the organization can build additional ones as needed.

➔ For additional information on building and using templates, **see** Chapter 18, "Managing Project Files," on **p. 581**.

Methodologies

As organizations mature in the project management discipline, they very often adopt more formal management control systems. These systems are typically described as methodologies that include processes, rules, standards, and methods for how work will be done. In this section, we will identify a few of the ones used in specific industry segments. Each industry has its own set of methodologies, and this chapter will not attempt to identify all of them. The purpose here is to show how managing projects using Microsoft Project can be included in the methodology to assist in the enforcement and usability of the tools.

Life Cycles

Like methodologies, project life cycles are unique to the industries and disciplines in which they are used. Although all projects have a beginning and an end, they will vary greatly in how the work is accomplished. It is nearly impossible to define an ideal life cycle. Some companies and organizations use a single, standardized life cycle for every project, whereas others permit the project manager to choose the best life cycle for the project. In others, a variety of life cycles exists to accommodate different levels of complexity and different styles or types of work.

Regardless of the organization's choices regarding methodologies and life cycles, all choices can use a scheduling tool to help with project execution. The key to success in every case is that the schedule must be focused on the deliverables to be produced rather than the process. The process must be set up to assist with producing deliverables. The next section of this chapter provides several examples of methodologies and life cycles in the field of software development to illustrate how Microsoft Project can be used to enable management of a wide variety of projects.

Using Microsoft Project with Methodologies and Life Cycles

Almost all organizations have at least a small number of technology projects underway, so software development is an excellent example of the wide variety of project-scheduling approaches available to organizations. The types of projects range from simple to complex, short duration to multiyear, and projects that have clear goals to those that are open-ended research. The following examples discuss the associated software development life cycle (SDLC) and how Microsoft Project can be set up to support the life cycle. As you review the examples, you should also keep in mind that these projects should be planned and executed using the principles described in the previous sections on project standards (the PMBOK Guide and PRINCE2). While strict adherence to the standards is not required or necessary on every project, it is useful to remember that there are five major process groups to be managed on each project and that there are nine knowledge areas that should be considered throughout the project's life cycle.

Waterfall Development Process

Traditional software development is often described as a waterfall model because it is sequential in nature. The assumption with this model is that phases can be completed in order with little or no need to repeat the previous activities. Development is described as a waterfall, steadily falling down through traditional phases such as definition, preliminary design, detailed design, coding, testing, implementation, and transition to operations.

This method of development is used in many organizations today, especially those involved in multiyear programs. The phases can be lengthy and the work can be very exacting. Although the name suggests that all work from one phase is completed before moving into the next phase, these types of projects are often set up with overlapping phases so that design can begin on certain deliverables as soon as the definition of the work for those deliverables is completed. In addition, there is typically some level of iterative development involved in almost all projects but the term "waterfall" is still in common use today.

In this type of project, the tendency is to set up the project schedule in the same order as the major phase names. Instead, the project can be set up so that it is broken up into logical work packages that can be monitored and measured separately.

Iterative Development

Iterative development provides a strong framework for planning purposes and also flexibility for successive iterations of software development. The Rational Unified Process (RUP) and the Dynamic Systems Development Methods are two frameworks for this type of project life cycle. RUP is not only a methodology for software engineering project management, it also has a set of software tools for using the specific methodology that is the Rational Unified Process. Figure 3.3 depicts the RUP workflow.

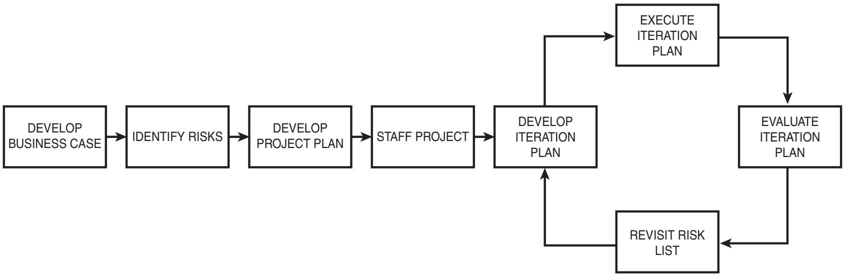


Figure 3.3
RUP workflow.

The goal for this type of software development life cycle (SDLC) is to allow the developers to learn through incremental development and the use of prototypes instead of trying to complete detailed requirements before the development work begins. Agile and XP can also be considered to be iterative methods.

Agile Development Process

Agile is a philosophy of project management that moves away from the classic project management methods and focuses less on planning and more on execution. In Agile, crucial decisions are made during the project execution phase, instead of the planning phase. As business and project environments become more fluid and dynamic, the amount of time for effective planning becomes less and less. This does not mean that planning will be ignored; rather the focus will shift to supporting decisions during project execution instead of finalizing all decisions during the planning stage.

Agile is not an all-or-nothing methodology either; it is possible to combine Agile with more classic project management ideas. While classic project management is comprehensive and will work in diverse situations, Agile can add various ideas for facing new and unique situations that can be found in creative, knowledge-based industries.

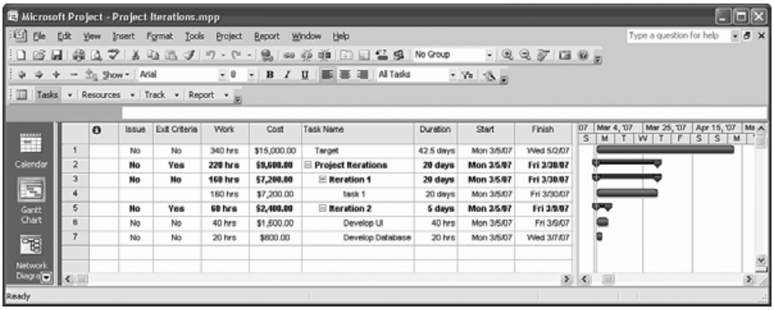
Here are some of the attributes of an Agile SDLC:

- Short development cycles are used to produce working software in weeks rather than months.
- Communication between the business users and the developers occurs daily.
- Documentation of working functionality is captured after the software is completed; there is limited documentation of the requirements or design.
- Timeboxing is used to force tough decisions early in the project.

- Changes to requirements are expected; they are a result of early working prototypes and are a goal of the process.
- The project manager for an Agile team is focused on ensuring excellent communication as the primary mechanism to maintain progress.

Agile development can be difficult for large organizations to embrace because it does not require a focus on formal planning of an entire project. The use of Agile should not be used as an excuse to avoid planning or managing a budget. The approach is meant to provide a lighter and faster method to reach a goal, but the goal is still required. The major difference is that the primary measurement of progress is frequent delivery of small amounts of working software. With a focus on feature delivery, it can sometimes be difficult to understand the overall picture, so strong project management must provide this clarity.

In this type of environment, a project team can still use Microsoft Project to support its goals. In an Agile environment, the tool is not used to develop a robust schedule with a beginning-to-end flow of tasks and resources. Its use in this case supports communication to management and ensures that changes are captured and the backlog of work is moved through each successive iteration of the project schedule. In the following example, the project manager has established a budget summary task to provide rollup of budget for management purposes. Successive sets of work are defined in small iterations while the overall timeframe and budget are obvious for all (see Figure 3.4). This approach allows the team to perform iterative planning while still meeting the business requirements of not exceeding a specific timeframe and budget. By establishing a project schedule with an overall goal, the needs of the team and their management can be met. Refer to Figure 3.4 for an example of a short project that is expected to complete within a target effort of 340 hours. The work is not fully defined at the beginning so that the team has the flexibility to decide what work will happen in what order. Management is still able to see overall metrics of planned work, actual work, and the current estimate of work remaining.



		Issue	Exit Criteria	Work	Cost	Task Name	Duration	Start	Finish
1		No	No	340 hrs	$15,000.00	Target	42.5 days	Mon 3/5/07	Wed 5/2/07
2		**No**	**Yes**	**220 hrs**	**$9,600.00**	**⊟ Project Iterations**	**20 days**	**Mon 3/5/07**	**Fri 3/30/07**
3		**No**	**No**	**160 hrs**	**$7,200.00**	**⊟ Iteration 1**	**20 days**	**Mon 3/5/07**	**Fri 3/30/07**
4				160 hrs	$7,200.00	task 1	20 days	Mon 3/5/07	Fri 3/30/07
5		**No**	**Yes**	**60 hrs**	**$2,400.00**	**⊟ Iteration 2**	**5 days**	**Mon 3/5/07**	**Fri 3/9/07**
6		No	No	40 hrs	$1,600.00	Develop UI	40 hrs	Mon 3/5/07	Fri 3/9/07
7		No	No	20 hrs	$800.00	Develop Database	20 hrs	Mon 3/5/07	Wed 3/7/07

Figure 3.4
An Agile project showing overall budget, work, and timeframe with iterative development.

Agile is an extremely successful method of software development that is very well suited to an environment with self-motivated teams, open communications, and leadership that is comfortable with a prototyping approach to work. It does not fit all projects, but when it works, it works very well. The schedule created in Microsoft Project for this type of approach becomes a tool for communication, overall budget and time goals, and historical tracking purposes.

NOTE

For organizations that use the Project Server tool, this method will allow them use an Agile approach and yet have oversight of the entire project portfolio. Agile projects coexist with standard iterative projects in their Project Server environment; the projects have planned timeframes, resources, and budgets but are not required to have all of the work scoped out at the beginning of the project.

EXTREME PROGRAMMING

Extreme Programming, or XP, is another method within the Agile family that has become a simple and flexible way for developing software through the writing of tests. It is designed to be used by a group of two to ten programmers who are able to execute tests in a fraction of a day. It uses short cycles of phases, with early and continuing feedback during each cycle. This flexibility allows it to respond to changing business demands through the implementation of functionality. XP's use of automated tests, written by the programmers to scrutinize development, helps early detection of defects and also allows the cycle of phases to evolve as the project continues. These automated tests depend both on the short-term instincts of programmers and also on the long-term interests of the project. It also relies heavily on a system of oral communication, tests, and source code to help communicate the system structure and intent. These processes allow for the day-to-day programming of a feature, and then moving on to testing, implementation, design, and integration, all packed into each cycle.

The scheduling methods used in the preceding Agile example can again be adapted for XP.

SPIRAL DEVELOPMENT PROJECT

Spiral development was defined by Barry Boehm in 1985 and is often used in fairly large projects that take months to two years or more to complete. The initial focus may be on core functionality, and then the "bells and whistles" such as graphical user interfaces and reporting are added at a later time. This is sometimes considered to be another form of iterative development, but the structure of the plans and schedule focus on a robust core design in the early stages.

RESEARCH PROJECT

A research project may be the most difficult type of project to tackle when it comes to identifying a project schedule. Often, there is no clear goal in mind and there may not even be an expectation of a specific end date or budget. On the other side, however, even research projects must be funded by someone and they must have a working staff, so there is typically some expectation of a result. In most cases, there is also an expectation that the funding is used responsibly, so there must be a process in place to track how the money has been spent.

Microsoft Project can once again be used to support this type of project as a tracking mechanism and a place to bring together the set of work that will be performed. The schedule will not require all of the advanced features of critical path analysis, resource-leveling, and predecessor/successor relationships, but it can be used as an easy method of historical support and a loose prediction of the work that is to be accomplished.

Accommodating Teaming Styles

High-performance teams, self-managed teams, and other nontraditional structures began to emerge over 50 years ago in Great Britain and gained acceptance across the globe as several large corporations began to adopt the concepts. The general idea behind these teaming styles was to loosen managerial constraints in an effort to increase worker performance and make quantum leaps in accomplishment of organizational goals.

When framed correctly, the teams need little direction and excel in accomplishing the goals of their projects. If the dynamics are not understood, however, little is accomplished. From a project management perspective, Agile or XP projects can be a bit intimidating because the team dynamics can overwhelm the designated leader. In reality, successful self-managed teams are not leaderless. They have simply figured out a mechanism to allow many people within the team to play a leadership role.

Even in a team where a project management role has not been defined, someone must take on the job of setting a direction to accomplish a goal. The goal may only be one week in the future, but the team must coalesce around that goals, and the person who makes that happen is a leader. If the project manager understands the dynamics of the team, he or she can use these dynamics to improve the team's focus and increase its performance. The PM must be comfortable with sharing decision making and will need to focus heavily on communication of information within the team and with the stakeholders of the project. Things change quickly in this environment, so communication of status becomes a driving force for the project.

Microsoft Project is an excellent tool to aid the PM in communication. Two components need to be established to make this successful. The overall goal of the project needs to be clear to the team, and the boundaries of the project (overall timeframe, scope, resources, and budget) must be understood. If these components are established within the tool as a baseline, the remainder of the schedule can be flexible or rigid, as dictated by the project structure and the teaming style.

Consultants' Tips

Determine the Approach To Use in Managing Your Project

Microsoft Project 2007 has a rich set of features that enable the project manager and team to track projects at a very detailed level. It also has enough flexibility to allow high-level tracking without a demand for the detail. It can be and has been used to support all industries and all domains within those industries.

Because the software has so many capabilities, it must be well-understood to be used correctly. The scheduling engine will anticipate your needs and move the dates or adjust the amount of work that is to be accomplished based on the parameters that you set. Because it will do this, project managers must have a clear understanding of the approach that they wish to use on their projects before they begin entering tasks.

Use WBS as a First Step in Project Definition

Always start with a WBS to help you be clear on the goals of the project. Wait to add the task-level details until you are sure that you have decomposed the WBS to the work package level that is right for the type of project you are leading. Do not confuse the listing of activities with the completion of deliverables.

Use the 5×9 Checklist for Planning

Remember the 5×9 checklist and consider it when planning and executing each project. As you move through the phases of your project's life cycle, spend a moment to consider which of the five process groups is most dominant at the moment and which of the nine knowledge areas plays the most important part in the project's evolution.