Chapter 6. Project Schedule Management
Learning Objectives

After reading this chapter, you will be able to:

- Illustrate the importance that project schedules and good project schedule management can have in helping to make projects successful
- Discuss the process of planning schedule management
- Define activities as the basis for developing project schedules
- Describe how project managers use network diagrams and dependencies to assist in activity sequencing
- Explain how various tools and techniques help project managers perform activity duration estimates
- Use a Gantt chart for planning and tracking schedule information, find the critical path for a project, and describe how critical chain scheduling and the Program Evaluation and Review Technique (PERT) affect schedule development
- Compare how schedule management is addressed using Agile vs. more predictive project approaches
- Discuss how reality checks and discipline are involved in controlling and managing changes to the project schedule
- Describe how project management software can assist in project schedule management and review words of caution before using this software
- Discuss considerations for agile/adaptive environments

Opening Case

Sue Johnson was the project manager for a consulting company contracted to provide a new online registration system at a local college in nine months or less. This system had to be operational by May 1 so students could use it to register for the fall semester. Her company's contract had a stiff penalty clause if the system was not ready by then, and Sue and her team would get nice bonuses for doing a good job on this project and meeting the schedule. Sue knew that it was her responsibility to meet the schedule and manage scope, cost, and quality expectations. She and her team developed a detailed schedule and network diagram to help organize the project.

Developing the schedule turned out to be the easy part; keeping the project on track was more difficult. Managing personnel issues and resolving schedule conflicts were two of the bigger challenges. Many of the college's employees took unplanned vacations and missed or rescheduled project review meetings. These changes made it difficult for the project team to follow the planned schedule for the system because the team had to have customer sign-off at various stages of the systems development life cycle. One senior programmer on Sue's project team quit, and she knew it would take extra time for a new person to get up to speed, especially since the exiting programmer did a poor job documenting how his code linked to other systems at the college. It was still early in the project, but Sue knew they were falling behind. What could she do to meet the operational date of May 1?

The Importance of Project Schedules

Managers often cite the need to deliver projects on time as one of their biggest challenges and the main cause of conflict. Perhaps part of the reason that schedule problems are so common is that time is easily measured and remembered. You can debate scope and cost overruns and make actual numbers appear closer to estimates, but once a project schedule is set, people remember the projected completion date, and anyone can quickly estimate schedule performance by subtracting the original time estimate from how long it really took to complete the project. People often compare planned and actual project completion times without taking into account the approved changes in the project. Time is the variable that has the least amount of flexibility. Time passes no matter what happens on a project.

Individual work styles and cultural differences may also cause schedule conflicts. For example, you will learn in Chapter 9, Project Resource Management, about the Myers-Briggs Type Indicator. One dimension of this team-building tool deals with attitudes toward structure and deadlines. Some people prefer detailed schedules and emphasize task completion. Others prefer to keep things open and flexible. Different cultures and even entire countries have different attitudes about schedules. For example, in some countries businesses close for several hours every afternoon to have siestas. Countries may have different holidays, which means not much work will be done at certain times of the year. Cultures may also have different perceptions of work ethic—some may value hard work and strict schedules while others may value the ability to remain relaxed and flexible.

Media Snapshot

In contrast to the 2002 Salt Lake City Winter Olympic Games (see **Chapter 4**'s Media Snapshot), planning and scheduling were not well implemented for the 2004 Athens Summer Olympic Games or the 2014 Sochi Winter Olympic Games.

Many articles written before the Athens opening ceremonies predicted that the facilities would not be ready in time. “With just 162 days to go to the opening of the Athens Olympics, the Greek capital is still not ready for the expected onslaught...... By now 22 of the 30 Olympic projects were supposed to be finished. This week the Athens Olympic Committee proudly announced 19 venues would be finished by the end of next month. That’s a long way off target.”* However, many people were pleasantly surprised by the amazing opening ceremonies, beautiful new buildings, and state-of-the-art security and transportation systems in Athens. For example, traffic flow, considered a major pre-Games hurdle, was superb. One spectator at the games commented on the prediction that the facilities would not be ready in time: “Athens proved them all wrong...... It has never looked better.”* The Greeks even made fun of critics by having construction workers pretend that they were still working as the ceremonies began. Unfortunately, the Greek government suffered a huge financial deficit because the games cost more than twice the planned budget.

The 2014 Winter Olympic Games in Sochi, Russia, suffered even greater financial losses. Originally budgeted at US$12 billion, final costs reached over US$51 billion, making it the most expensive games in history. Unlike Greece’s humorous response to challenges at the Athens games, Russian citizens were much more serious. Although in 2006, 86 percent of Sochi residents supported the games, by 2013 only 40 percent supported them. Russian citizens, especially those in Sochi, understood the negative impacts of hosting the games when they heard about the poor planning and huge cost overruns. “. . . the Sochi Olympics will continue to be a burden for the Russian state, with expenses for operation, maintenance and foregone interest and tax revenue in the order of USD 1.2 billion per year. The event also did not manage to improve the image of Russia in the world and among the domestic population support dropped over the seven years of its implementation, most notably among the local population.”*

With all the possibilities for schedule conflicts, it is important for project managers to use good project schedule management. **Project schedule management** , simply defined, involves the processes required to ensure timely completion of a project. Six main processes are involved in project schedule management:

1. *Planning schedule management* involves determining the policies, procedures, and documentation that will be used for planning, executing, and controlling the project schedule.

2. *Defining activities* involves identifying the specific activities that the project team members and stakeholders must perform to produce the project deliverables. An **activity** or **task** is an element of work normally found on the work breakdown structure (WBS)

that has expected duration, cost, and resource requirements.

3. *Sequencing activities* involves identifying and documenting the relationships between project activities. Requirements, a resource breakdown structure, and project documents updates.

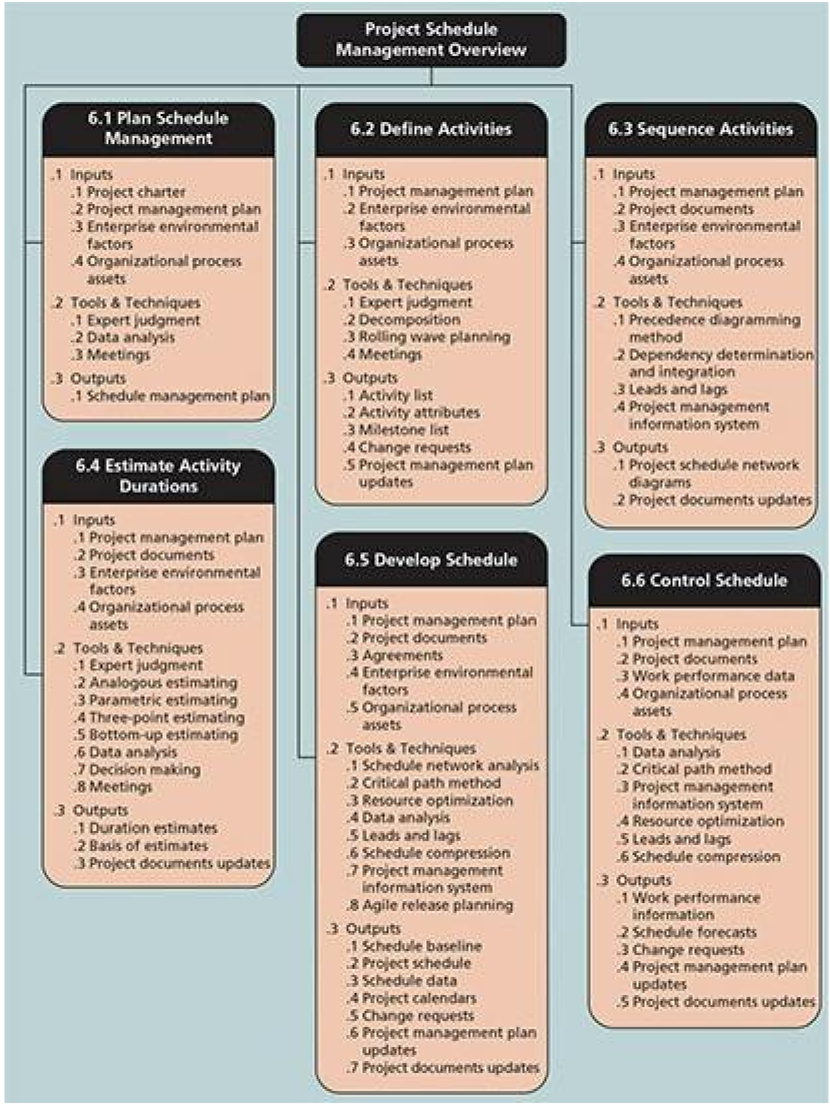
4. *Estimating activity durations* involves estimating the number of work periods that are needed to complete individual activities.

5. *Developing the schedule* involves analyzing activity sequences, resource requirements, and activity duration estimates to create the project schedule.

6. *Controlling the schedule* involves controlling and managing changes to the project schedule.

Figure 6-1 summarizes the inputs, tools and techniques, and outputs of project schedule management.

Figure 6-1. Project schedule management overview



Source: *PMBOK® Guide – Sixth Edition*. Project Management Institute, Inc. (2017). Copyright and all rights reserved. Material from this publication has been reproduced with permission of PMI.

You can improve project schedule management by performing these processes and by using some basic project management tools and techniques. Every manager is familiar with some form of scheduling, but most managers have not used several of the tools and techniques that are unique to project schedule management, such as Gantt charts, network diagrams, and critical path analysis.

Planning Schedule Management

The first step in project schedule management is planning how the schedule will be managed throughout the life of the project. Project schedules grow out of the basic documents that initiate a project. The project charter often mentions planned project start and end dates, which serve as the starting points for a more detailed schedule. After reviewing the project management plan, project charter, enterprise environmental factors, and organizational process assets, the project team uses expert judgment, analytical techniques, and meetings to develop the schedule management plan.

The schedule management plan, like the scope management plan, can be informal and broad or formal and detailed, based on the needs of the project. In general, a schedule management plan includes the following information:

- *Project schedule model development*: Many projects include a schedule model, which contains project activities with estimated durations, dependencies, and other planning information that can be used to produce a project schedule. See Appendix A (available on the Companion website for this text) for information on using Microsoft Project 2016 to create a schedule model.

- *Level of accuracy and units of measure*: This section discusses how accurate schedule estimates should be and determines whether time is measured in hours, days, or another unit.

- *Control thresholds*: Variance thresholds, such as ±10 percent, are established for monitoring schedule performance.

- *Rules of performance measurement*: For example, if team members are expected to track the percentage of work completed, this section specifies how to determine the percentages.

- *Reporting formats*: This section describes the format and frequency of schedule reports required for the project.

- *Process descriptions*: The schedule management plan also describes how all of the schedule management processes will be performed.

Defining Activities

You might think that all project work has been defined in enough detail after planning scope management, but it is often necessary to describe activities in more detail as part of schedule management. Defining activities involves identifying the specific actions that will produce the project deliverables in enough detail to determine resource and schedule estimates. The project team reviews the project management plan, enterprise environmental factors, and organizational process assets to begin defining activities. Outputs of this process include an activity list, activity attributes, a milestone list, change requests, and project management plan updates.

The **activity list** is a tabulation of activities to be included on a project schedule. The list should include the activity name, an activity identifier or number, and a brief description of the activity. The **activity attributes** provide schedule-related information about each activity, such as predecessors, successors, logical relationships, leads and lags, resource requirements, constraints, imposed dates, and assumptions related to the activity. The activity list and activity attributes should agree with the WBS and WBS dictionary. Information is added to the activity attributes as it becomes available; this information includes logical relationships and resource requirements that are determined in later processes. Many project teams use an automated system to keep track of activity-related information.

A **milestone** on a project is a significant event that normally has no duration. It often takes several activities and a lot of work to complete a milestone, but the milestone itself is a marker to help in identifying necessary activities. Milestones are also useful tools for setting schedule goals and monitoring progress. For example, milestones on a project like the one in the chapter's opening case might include completion and customer sign-off documents, such as design documents and test plans; completion of specific products, such as software modules or installation of new hardware; and completion of important process-related work, such as project review meetings and tests. Not every deliverable or output created for a project is really a milestone. Milestones are the most important and visible events. For example, in the context of child development, parents and doctors check for *milestones*, such as a child first rolling over, sitting, crawling, walking, and talking. You will learn more about milestones later in this chapter.

Activity information is a required input to the other schedule management processes. You cannot determine activity sequencing, durations, develop the schedule, or control the schedule until you have a good understanding of project activities.

Recall the triple constraint of project management—balancing scope, time, and cost goals—and note the order of these items. Ideally, the project team and key stakeholders first define the project scope, then the time or schedule for the project, and then the project's cost. The order of these three items reflects the basic order of the processes in project schedule management: defining activities (further defining the scope), sequencing activities (further defining the time), and estimating activity resources and activity durations (further defining the time and cost). These processes are the basis for creating a project schedule.

The goal of defining activities is to ensure that the project team completely understands all the work it must do as part of the project scope so they can start scheduling the work. For example, a WBS item might be "Study report." The project team must understand what the item means before team members can make schedule-related decisions. How long should the report be? Does it require a survey or extensive research to produce? What skill level does the report writer need to have? Further defining the task will help the project team determine how long it will take to do and who should do it.

The WBS is often dissected further as the project team members continue to define the activities required for performing the work. For example, the task "Study report" might be broken down into several subtasks describing smaller deliverables required to produce the report, such as survey development, survey administration, draft report, report edits, and final report. This process of progressive elaboration, one of the project attributes listed in **Chapter 1**, is sometimes called "rolling wave planning."

As stated earlier, activities (or tasks, the term used in most project management software) are elements of work performed during the course of a project; they have expected durations, costs, and resource requirements. Defining activities also results in supporting detail to document important product information as well as assumptions and constraints related to specific activities. The project team should review the activity list and activity attributes with project stakeholders before moving on to the next step in project schedule management. If the team does not review these items with project stakeholders, it could produce an unrealistic schedule and deliver unacceptable results. For example, if a project manager simply estimated that the "time required to complete the Study report" deliverable would be one day and then had an intern or trainee write a 10-page report to complete that task, the result could be a furious customer who expected extensive research, surveys, and a 100-page report. Clearly defining the work is crucial to all projects. If there are misunderstandings about activities, requested changes may be required.

In the opening case of this chapter, Sue Johnson and her project team had a contract and detailed specifications for the college's new online registration system. They also had to focus on meeting the May 1 date for delivering an operational system so the college could start using the new system for the new semester's registration. To develop a project schedule, Sue and her team had to review the contract, detailed specifications, and desired operational date, then create an activity list, activity attributes, and milestone list. After developing more detailed definitions of project activities, Sue and her team would review them with the customers to ensure that the team was on the right track.

What Went Wrong?

At the U.S. Federal Bureau of Investigation (FBI), poor time management was one of the reasons behind the failure of Trilogy, a "disastrous, unbelievably expensive piece of vaporware, which was more than four years in the (un)making. The system was supposed to enable FBI agents to integrate intelligence from isolated information silos within the Bureau."* In May 2006, the Government Accounting Agency said that the Trilogy project failed at its core mission of improving the FBI's investigative abilities and was plagued with missed milestones and escalating costs.

The FBI rushed to develop the new system beginning in 2001, in response to the attacks on September 11. The need for new software was obvious to former FBI agent David J. Williams, who recalls joining a roomful of agents shortly after 9/11 to help with intelligence. Agents were wearing out the casters on their chairs by sliding back and forth among 20 old computer terminals, many of which were attached to various databases that contained different information.

Congressional hearings revealed numerous problems with the project. Its requirements were very loosely defined, there were many leadership changes throughout the project, and several contracts had no scheduled milestones for completion or penalties if work was late. The new system was finally completed in 2006 and cost more than $537 million —more than a year late and $200 million over budget.

The FBI replaced Trilogy with a new system called Sentinel and began training employees to use it in May 2007.* Unfortunately, history repeated itself as troubles still loomed with Sentinel in 2012. According to the FBI's original plan, Sentinel was to be completed by December 2009 at an estimated cost of $425 million. The FBI later increased the estimated cost to $451 million and extended the project completion date twice, so that the project was now two years behind schedule. During a test exercise in 2011, the system experienced two outages, and the FBI determined that its current hardware structure was inadequate. In 2014, several people complained that the system still wasn't working well. "You can't find what you need most of the time, or you get junk you don't want, but other than that, the FBI's long troubled, half-billion-dollar Sentinel computerized file system is coming along just fine."*

Sequencing Activities

After defining project activities, the next step in project schedule management is sequencing them or determining their dependencies. Inputs to the activity sequencing process include the project management plan, project documents (like the activity attributes, activity list, assumption log, and milestone list), enterprise environmental factors, and organizational process assets. The sequencing process involves evaluating the reasons for dependencies and the different types of dependencies.

Dependencies

A **dependency** or **relationship** pertains to the sequencing of project activities or tasks. For example, does a certain activity have to be finished before another can start? Can the project team do several activities in parallel? Can some overlap? Determining these relationships or dependencies among activities is crucial for developing and managing a project schedule.

There are several types of dependencies among project activities:

- **Mandatory dependencies** are inherent in the nature of the work being performed on a project. They are sometimes referred to as hard logic. For example, you cannot test code until after the code is written.

- **Discretionary dependencies** are defined by the project team. For example, a project team might follow good practice and not start the detailed design of a new information system until the users sign off on all of the analysis work. Discretionary dependencies are sometimes referred to as soft logic and should be used with care because they may limit later scheduling options.

- **External dependencies** involve relationships between project and non-project activities. For example, the installation of a new operating system and other software may depend on delivery of new hardware from an external supplier. Even though delivery of the hardware may not be included in the scope of the project, you should add an external dependency to it because late delivery will affect the project schedule.

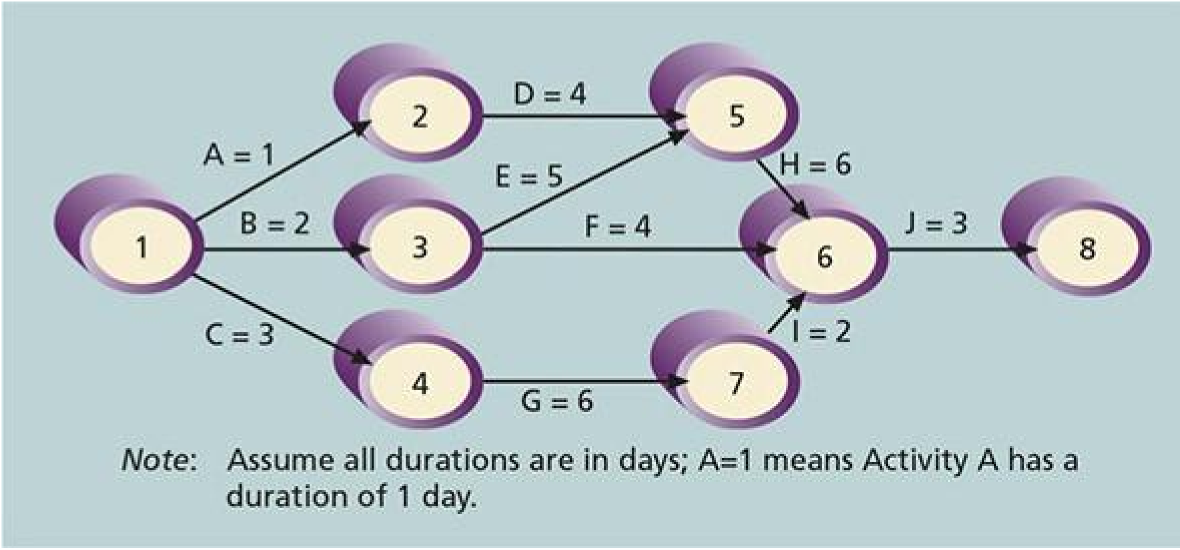
- **Internal dependencies** involve relationships between project activities that are generally inside the project team's control. For example, if software is developed by the team, they can create dependencies such as performing unit testing before system testing.

Note that dependencies can be internal and mandatory, external and discretionary, and so on. As with activity definition, it is important that project stakeholders work together to define the activity dependencies in their project. If you do not define the sequence of activities, you cannot use some of the most powerful scheduling tools available to project managers: network diagrams and critical path analysis.

Network Diagrams

Network diagrams are the preferred technique for showing activity sequencing. A **network diagram** is a schematic display of the logical relationships among project activities and their sequencing. Some people refer to network diagrams as project schedule network diagrams or PERT charts. PERT is described later in this chapter. Figure 6-2 shows a sample network diagram for Project X.

Figure 6-2. Network diagram for project X



Note the main elements on this network diagram. The letters A through J represent activities with dependencies that are required to complete the project. These activities come from the WBS and activity definition process described earlier. The arrows represent the activity sequencing or relationships between tasks. For example, Activity A must be done before Activity D, and Activity D must be done before Activity H.

The format of this network diagram uses the **activity-on-arrow (AOA)** approach or the **arrow diagramming method (ADM)** —a network diagramming technique in which activities are represented by arrows and connected at points called nodes to illustrate the sequence of activities. A **node** is simply the starting and ending point of an activity. The first node signifies the start of a project and the last node represents the end.

Keep in mind that the network diagram represents activities that must be done to complete the project. It is not a race to get from the first node to the last node. *Every* activity on the network diagram must be completed in order to finish the project. Note also that not every item on the WBS needs to be shown on the network diagram; only activities with dependencies need to be shown. However, some people like to have start and end milestones and to list every activity. It is a matter of preference. For large projects with hundreds of activities, it might be simpler to include only activities with dependencies on a network diagram. Sometimes it is enough to put summary tasks on a network diagram or to break down the project into several smaller network diagrams.

Assuming that you have a list of the project activities and their start and finish nodes, follow these steps to create an AOA network diagram:

1. Find all of the activities that start at Node 1. Draw their finish nodes, and draw arrows between Node 1 and each of the finish nodes. Put the activity letter or name on the

associated arrow. If you have a duration estimate, write it next to the activity letter or name, as shown in **Figure 6-2**. For example, A = 1 means that the duration of Activity A is one day, week, or other standard unit of time. Be sure to put arrowheads on all arrows to signify the direction of the relationships.

2. Continue drawing the network diagram, working from left to right. Look for bursts and merges. **Bursts** occur when two or more activities follow a single node. A **merge** occurs when two or more nodes precede a single node. For example, in Figure 6-2, Node 1 is a burst because it goes into Nodes 2, 3, and 4. Node 5 is a merge preceded by Nodes 2 and 3.

3. Continue drawing the AOA network diagram until all activities are included.

4. As a rule of thumb, all arrowheads should face toward the right, and no arrows should cross on an AOA network diagram. You may need to redraw the diagram to make it look presentable.

Even though AOA or ADM network diagrams are generally easy to understand and create, a different method is more commonly used: the precedence diagramming method. The **precedence diagramming method (PDM)** is a network diagramming technique in which boxes represent activities. It is particularly useful for visualizing certain types of time relationships.

Figure 6-3 illustrates the types of dependencies that can occur among project activities based on a Microsoft Project help screen. After you determine the reason for a dependency between activities (mandatory, discretionary, or external), you must determine the type of dependency. Note that the terms *activity* and *task* are used interchangeably, as are *relationship* and *dependency*. The four types of dependencies or relationships between activities include the following:

- **Finish-to-start dependency** : A relationship in which the "from" activity or predecessor must finish before the "to" activity or successor can start. For example, you cannot provide user training until after software or a new system has been installed. Finish-to-start is the most common type of relationship or dependency, and AOA network diagrams use only finish-to-start dependencies.

- **Start-to-start dependency** : A relationship in which the "from" activity cannot start until the "to" activity or successor is started. For example, on IT projects, a group of activities might start simultaneously, such as the many tasks that occur when a new system goes live.

- **Finish-to-finish dependency** : A relationship in which the "from" activity must be finished before the "to" activity can be finished. One task cannot finish before another finishes. For example, quality control efforts cannot finish before production finishes, although the two activities can be performed at the same time.

- **Start-to-finish dependency** : A relationship in which the "from" activity must start before the "to" activity can be finished. This type of relationship is rarely used, but it is appropriate in some cases. For example, an organization might strive to stock raw materials just in time for the manufacturing process to begin. A delay in starting the manufacturing process should delay completion of stocking the raw materials. Another example would be a babysitter who wants to finish watching a young child but is dependent on the parent's arrival. The parent must show up or "start" before the babysitter can finish the task.

Figure 6-3. Task Dependency Types

Task dependencies

The nature of the relationship between two linked tasks. You link tasks by defining a dependency between their finish and start dates. For example, the "Contact caterers" task must finish before the start of the "Determine menus" task. There are four kinds of task dependencies in Microsoft Project.

Task dependency	Example	Description
Finish-to-start (FS)	A B	Task (B) cannot start until task (A) finishes.
Start-to-start (SS)	A B	Task (B) cannot start until task (A) starts.
Finish-to-finish (FF)	A B	Task (B) cannot finish until task (A) finishes.
Start-to-finish (SF)	A B	Task (B) cannot finish until task (A) starts.

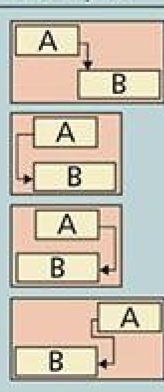
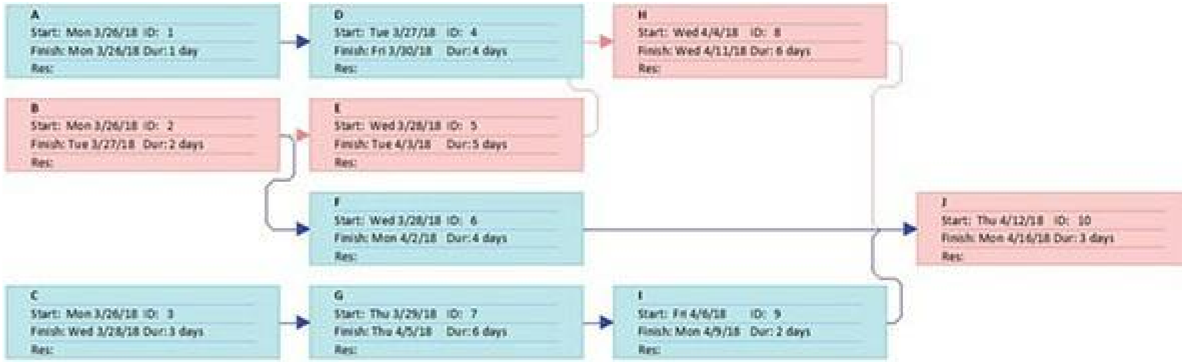


Figure 6-4 illustrates Project X using the precedence diagramming method. Notice that the activities are placed inside boxes, which represent the nodes on this diagram. Arrows show the relationships between activities. This figure was created using Microsoft Project, which automatically places additional information inside each node. Each task box includes the start and finish dates, which are labeled Start and Finish; the task ID number, labeled ID; the task's duration, labeled Dur; and the names of resources, if any, that are assigned to the task. These resources are labeled Res. The border of boxes for tasks on the critical path appears automatically in red in the Microsoft Project network diagram view.

Figure 6-4. Precedence diagramming methods (PDM) network diagram for project X



The precedence diagramming method is used more often than AOA network diagrams and offers a number of advantages over the AOA technique. First, most project management software uses the precedence diagramming method. Second, using this method avoids the need to use dummy activities. **Dummy activities** have no duration and no resources, but are occasionally needed on AOA network diagrams to show logical relationships between activities. These activities are represented with dashed arrow lines and have zeros for their duration estimates. Third, the precedence diagramming method shows different dependencies among tasks, whereas AOA network diagrams use only finish-to-start dependencies. You will learn more about activity sequencing using Project 2016 in Appendix A (available on the Companion website for this text).

Estimating Activity Durations

After working with key stakeholders to define activities and determine their dependencies, the next process in project schedule management is to estimate the duration of activities. It is important to note that **duration** includes the actual amount of time worked on an activity *plus* elapsed time. For example, even though it might take one workweek or five workdays to do the actual work, the duration estimate might be two weeks to allow extra time needed to obtain outside information. The people or resources assigned to a task will also affect the task duration estimate. As another example, if someone asked for an estimate of when you plan to finish reading a particular book, you might give an answer of two months. Two months would be the duration estimate, even if you only plan to spend 20 hours actually reading the book.

Do not confuse duration with **effort** , which is the number of workdays or work hours required to complete a task. (Duration is normally entered in the Duration column in software such as Microsoft Project 2016, while effort is entered in a Work column. See Appendix A—available on the Companion website for this text—for more information.) A duration estimate of one day could be based on eight hours of work or 80 hours of work, assuming that multiple people are working on a task that day. Duration relates to the time estimate on a calendar, not the effort estimate. In the previous example, you might plan to spend 20 hours reading a book—the effort estimate—and spread that time out over two months—the duration.

Of course, duration and effort are related, so project team members must document their assumptions when creating duration estimates. The people who will actually do the work, in particular, should have a lot of say in duration estimates because their performances will be evaluated based on their ability to meet the estimates. It is also helpful to review similar projects and seek the advice of experts in estimating activity durations.

Project team members must also update the estimates as the project progresses. If scope changes occur on the project, the duration estimates should be updated to reflect those changes.

There are several inputs to activity duration estimates, including the project management plan, project documents, enterprise environmental factors, and organizational process assets. In addition to reviewing past project information, the team should review the accuracy of the duration estimates thus far on the project. For example, if team members find that all of their estimates have been much too long or short, they should update the estimates to reflect what they have learned.

One of the most important considerations in making activity duration estimates is the availability of resources, especially human resources. What specific skills do people need to do the work? What are the skill levels of the people assigned to the project? How many people are expected to be available to work on the project at any one time? Estimating activity resources is described in Chapter 9, Project Resource Management.

The outputs of activity duration estimates include the estimates themselves, the basis of estimates, and project documents updates. Duration estimates are often provided as a discrete number, such as four weeks; as a range, such as three to five weeks; or as a three-point estimate. A **three-point estimate** includes an optimistic, a most likely, and a pessimistic estimate, such as three weeks for the optimistic scenario, four weeks for the most likely scenario, and five weeks for the pessimistic scenario. The optimistic estimate is based on a best-case scenario, while the pessimistic estimate is based on a worst-case scenario. The most likely estimate, as you might expect, is based on a most likely or expected scenario. A three-point estimate is required for performing PERT estimates, as described later in this chapter, and for performing Monte Carlo simulations, as described in Chapter 11, Project Risk Management. Other duration estimating techniques include analogous and parametric estimating and reserve analysis, as described in Chapter 7, Project Cost Management. Expert judgment is also an important tool for developing good activity duration estimates.

Advice for Young Professionals

Some people find estimating to be challenging, especially for their own work. However, it is very important to develop this skill. You don't want to be stressing yourself out when you gave an overly optimistic estimate and need to work around the clock to get something done, nor do you want to lose out on an opportunity because you made a very high estimate that was unacceptable to a potential client or sponsor. Practice estimating how long it takes you to do different activities, and then take actual measurements. Defining the activity in detail, as described earlier in this chapter, will help you to make better estimates about each part of the activity. Also, don't be afraid to ask for help in making estimations. We often think that things take less time than they actually do, and people with experience can help us be more realistic with estimates. You could start with activities like writing reports, arranging meetings, or creating a presentation. You might think you can develop a presentation in four hours, for example, but you could find that you actually take twice or three times that much because you have to think it through, revise, get opinions, format graphics, or other similar parts of the activity. If you realize that an activity estimate might not be a good one, let your team know as soon as possible so that adjustments can be made early in the project.

Developing the Schedule

Schedule development uses the results of all the preceding project schedule management processes to determine the start and end dates of the project and its activities. Project schedule management processes often go through several iterations before a project schedule is finalized. The ultimate goal of developing a realistic project schedule is to provide a basis for monitoring project progress for the time dimension of the project. The main outputs of this process are a schedule baseline, project schedule, schedule data, project calendars, change requests, project management plan updates, and project documents updates. Some project teams create a computerized model to create a network diagram, enter resource requirements and availability by time period, and adjust other information to quickly generate alternative schedules. See Appendix A (available on the Companion website for this text) for information on using Project 2016 to assist in schedule development.

A few of the tools and techniques for schedule development include the following:

- A Gantt chart is a common tool for displaying project schedule information.
- Critical path analysis is a very important tool for developing and controlling project schedules.

- Critical chain scheduling is a technique that focuses on limited resources when creating a project schedule.
- PERT analysis is a means for considering schedule risk on projects.

The following sections provide samples of each of these tools and techniques and discuss their advantages and disadvantages.

Gantt Charts

Gantt charts provide a standard format for displaying project schedule information by listing project activities and their corresponding start and finish dates in calendar form. Gantt charts are sometimes referred to as bar charts because the activities' start and end dates are shown as horizontal bars. Figure 6-5 shows a simple Gantt chart for Project X created with Microsoft Project. Figure 6-6 shows a Gantt chart that is more sophisticated and is based on a software launch project from a template provided by Microsoft. The activities on the Gantt chart are driven by the deliverables on the WBS, and should coincide in turn with the activity list and milestone list. Notice that the Gantt chart for the software launch project contains milestones, summary tasks, individual task durations, and arrows showing task dependencies.

Figure 6-5. Gantt chart for project X

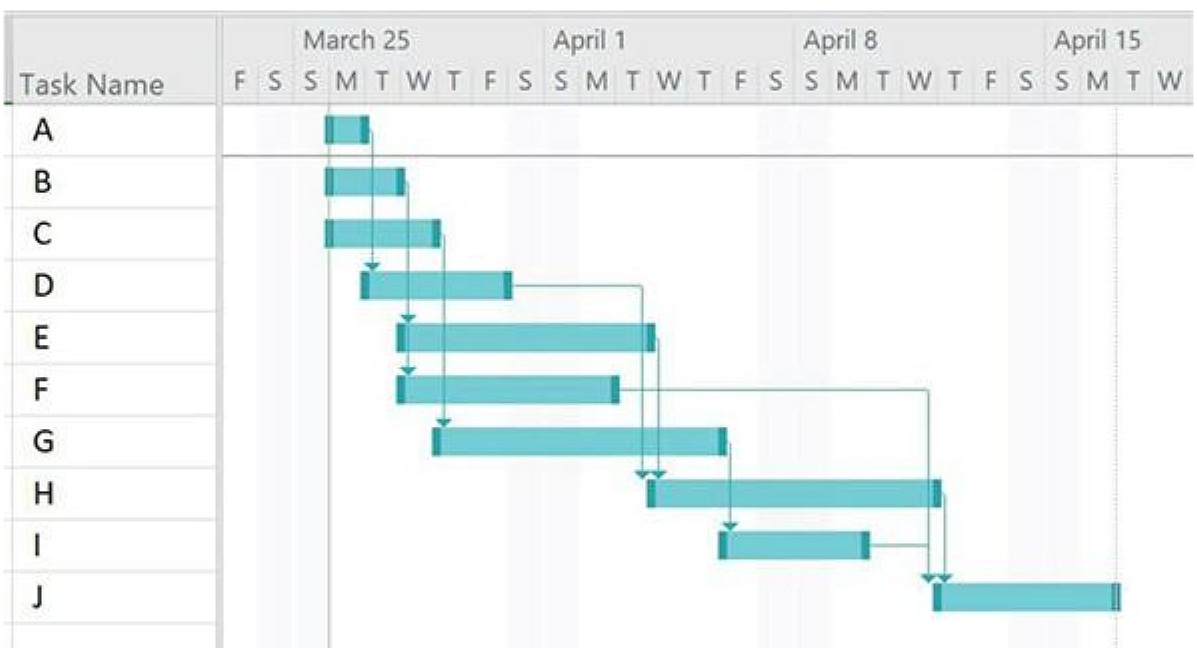
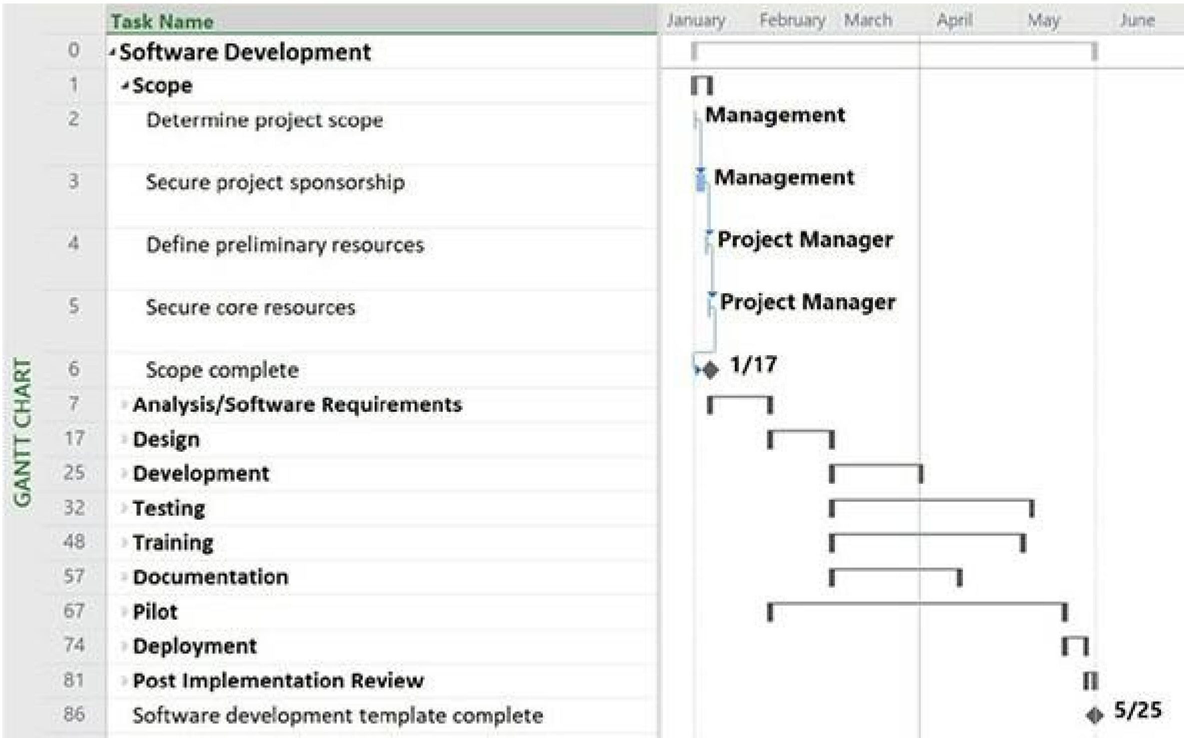


Figure 6-6. Gantt chart for software launch project



Notice the different symbols on the Gantt chart for the software launch project (see **Figure 6-6**):

- The black diamond represents a milestone. In **Figure 6-6**, Task 6, "Scope complete," is a milestone that occurs on January 17. For very large projects, top managers might want to see only milestones on a Gantt chart. Microsoft Project allows you to filter information displayed on a Gantt chart so you can easily show specific tasks, such as milestones.
- The thick black bars with rectangles at the beginning and end represent summary tasks. For example, the activities listed in rows 2-5 are all subtasks of the summary task called "Scope," Task 1. WBS activities are referred to as tasks and subtasks in most project management software.
- The light blue horizontal bars for Tasks 2, 3, 4, and 5 represent the duration of each individual task.
- Arrows connecting these symbols show relationships or dependencies between tasks. Gantt charts often do not show dependencies, which is their major disadvantage. If dependencies have been established in Microsoft Project, they are automatically displayed on the Gantt chart.

Adding Milestones to Gantt Charts

Milestones can be a particularly important part of schedules, especially for large projects. Many people like to focus on meeting milestones, so you can create them to emphasize important events or accomplishments on projects. Normally, you create milestones by entering tasks with zero duration. In Microsoft Project, you can mark any task as a milestone by checking the appropriate box in the Advanced tab of the Task Information dialog box. The duration of the task will not change to zero, but the Gantt chart will show the milestone symbol to represent that task based on its start date. See Appendix A (available on the Companion website for this text) for more information.

To make milestones meaningful, some people use the SMART criteria to help define them. The **SMART criteria** are guidelines suggesting that milestones should be:

- Specific
- Measurable
- Assignable
- Realistic
- Time-framed

For example, distributing a marketing plan is specific, measurable, and assignable if everyone knows what should be in the marketing plan, how it should be distributed, how many copies should be distributed and to whom, and who is responsible for the actual delivery. Distributing the marketing plan is realistic and able to be time-framed if it is an achievable event and scheduled at an appropriate time.

Best Practice

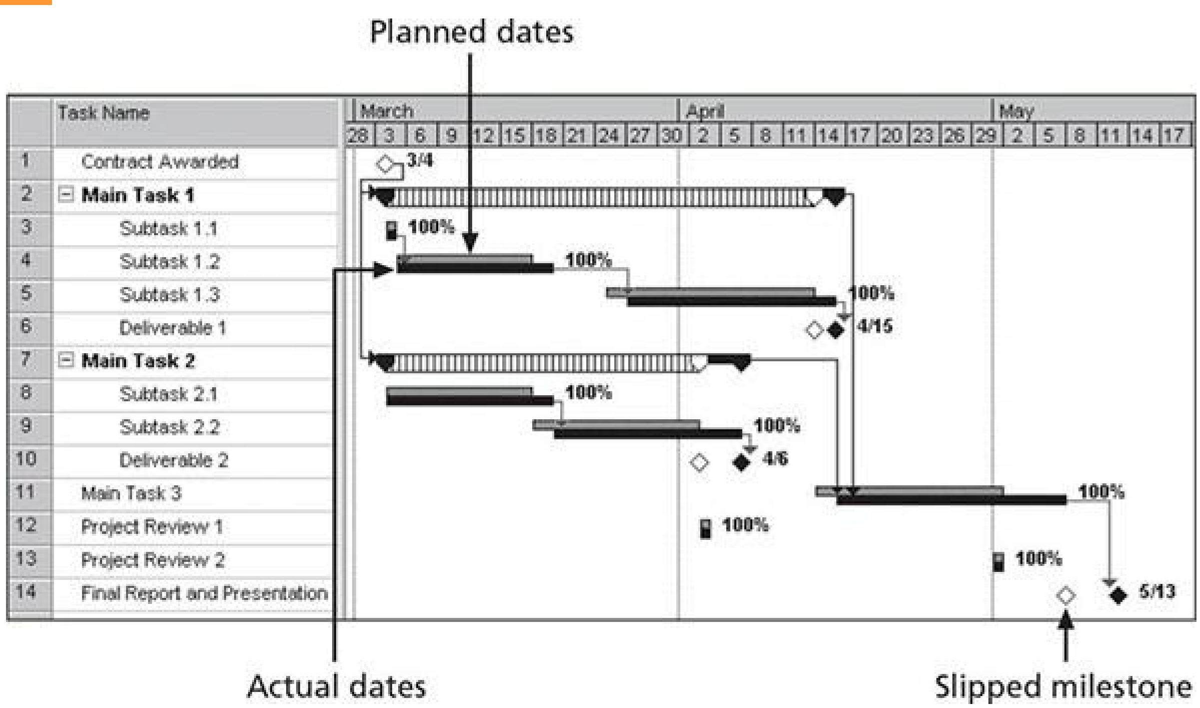
In his book, *The Happiness Advantage*, Shawn Achor shares principles of positive psychology that can fuel success at work. One problem many people have at work is feeling overwhelmed. They complain that they are working all the time, but they cannot get their work done. Achor suggests that the 20-second rule can help people improve their focus by minimizing barriers to change.

The 20-second rule capitalizes on people's preference for taking the path of least resistance. For example, you can more easily resist having an extra scoop of ice cream if you have to wait 20 seconds in line for it rather than have it served to you. Many people have trouble focusing on work because they are easily distracted by checking their e-mail, stocks, news, social media, etc. Achor recommends making it more difficult for yourself to be distracted. For example, keep e-mail closed while you are working. Don't leave your favorite non-work-related websites open or have your passwords saved in them. You can actually "save time by adding time"—but only to the distracting behaviors at work.*

Using Tracking Gantt Charts to Compare Planned and Actual Dates

You can use a special form of a Gantt chart to evaluate progress on a project by showing actual schedule information. **Figure 6-7** shows a **Tracking Gantt chart** —a Gantt chart that compares planned and actual project schedule information. The planned schedule dates for activities are called the **baseline dates** , and the entire approved planned schedule is called the **schedule baseline** . The Tracking Gantt chart includes columns labeled "Start" and "Finish" to represent actual start and finish dates for each task, as well as columns labeled "Baseline Start" and "Baseline Finish" to represent planned start and finish dates for each task. (These columns are hidden in Figure 6-7.) In this example, the project is completed, but several tasks missed their planned start and finish dates.

Figure 6-7. Sample tracking Gantt chart



To serve as a progress evaluation tool, a Tracking Gantt chart uses a few additional symbols:

- Notice that the Gantt chart in **Figure 6-7** often shows two horizontal bars for tasks. The top horizontal bar represents the planned or baseline duration for each task. The bar below it represents the actual duration. Subtasks 1.2 and 1.3 illustrate this type of display. If these two bars are the same length, meaning they start and end on the same dates, then the actual schedule was the same as the planned schedule for that task. This scheduling occurred for Subtask 1.1, in which the task started and ended as planned on March 4. If the bars do not start and end on the same dates, then the actual schedule differed from the planned or baseline schedule. If the top horizontal bar is shorter than the bottom one, the task took longer than planned, as you can see for Subtask 1.2. If the top horizontal bar is longer than the bottom one, the task took less time than planned. A striped horizontal bar, as illustrated by Main Tasks 1 and 2, represents the planned duration for summary tasks. The black bar adjoining it shows progress for summary tasks. For example, Main Task 2 clearly shows that the actual duration took longer than planned.

- A white diamond on the Tracking Gantt chart represents a **slipped milestone** . A slipped milestone means the milestone activity was completed later than originally planned. The last task provides an example of a slipped milestone because the final report and presentation were completed later than planned.

- Percentages to the right of the horizontal bars display the percentage of work completed for each task. For example, 100 percent means the task is finished, and 50 percent means the task is still in progress but is 50 percent completed.

A Tracking Gantt chart is based on the percentage of work completed for project tasks or the actual start and finish dates. It allows the project manager to monitor schedule progress on individual tasks and the whole project. For example, **Figure 6-7** shows that the project is completed. It started on time, but it finished a little late, on May 13 (5/13) instead of May 8.

The main advantage of using Gantt charts is that they provide a standard format for displaying planned and actual project schedule information. In addition, they are easy to create and understand. The main disadvantage of Gantt charts is that they do not *usually* show relationships or dependencies between tasks. If Gantt charts are created using project management software and tasks are linked, then the dependencies *will* be displayed, but differently than they would be displayed on a network diagram. Whether you view dependencies on a Gantt chart or network diagram is a matter of personal preference.

Critical Path Method

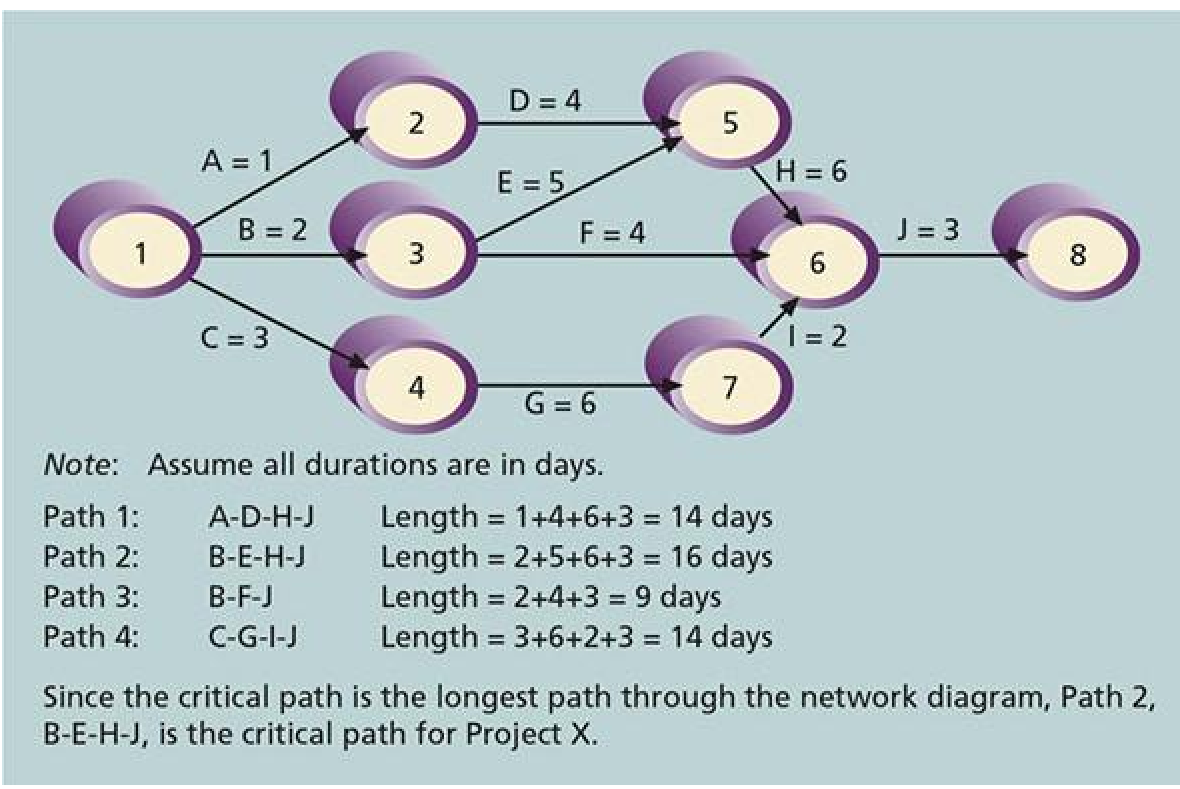
Many projects fail to meet schedule expectations. **Critical path method (CPM)** —also called **critical path analysis** —is a network diagramming technique used to predict total project duration. This important tool helps you combat project schedule overruns. A **critical path** for a project is the series of activities that determine the *earliest* time by which the project can be completed. It is the *longest* path through the network diagram and has the least amount of slack or float. **Slack** or **float** is the amount of time an activity may be delayed without delaying a succeeding activity or the project finish date. Normally, several tasks are done in parallel on projects, and most projects have multiple paths through a network diagram. The longest path or the path that contains the critical tasks is what drives the completion date for the project. You are not finished with the project until you have finished *all* the tasks.

Calculating the Critical Path

To find the critical path for a project, you must first develop a good network diagram, which in turn requires a good activity list based on the WBS. Once you create a network diagram, you must also estimate the duration of each activity to determine the critical path. Calculating the critical path involves adding the durations for all activities on each path through the network diagram. The longest path is the critical path.

Figure 6-8 shows the AOA network diagram for Project X again. Note that you can use either the AOA or precedence diagramming method to determine the critical path on projects. Figure 6-8 shows all of the paths—a total of four—through the network diagram. Note that each path starts at the first node (1) and ends at the last node (8) on the AOA network diagram. This figure also shows the length or total duration of each path through the network diagram. These lengths are computed by adding the durations of each activity on the path. Because path B-E-H-J has the longest duration at 16 days, it is the critical path for the project.

Figure 6-8. Determining the critical path for project X



What does the critical path really mean? Even though the critical path is the *longest* path, it represents the *shortest* time required to complete a project. If one or more activities on the critical path take longer than planned, the whole project schedule will slip *unless* the project manager takes corrective action.

Project teams can be creative in managing the critical path. For example, Joan Knutson, a well-known author and speaker in the project management field, often describes how a gorilla helped Apple Inc. complete a project on time. Team members worked in an area with cubicles, and whoever was in charge of the current task on the critical path had a stuffed gorilla on top of his or her cubicle. Everyone knew that person was under the most time pressure and did not need distractions. When a critical task was completed, the person in charge of the next critical task received the gorilla.

Growing Grass Can Be on the Critical Path

People are often confused about what a project's critical path really means. Some people think the critical path includes the most critical activities, but it is concerned only with the time dimension of a project. The fact that its name includes the word *critical* does not mean that it includes all critical activities. For example, Frank Addeman, Executive Project Director at Walt Disney Imagineering, explained in a keynote address at a PMI-ISSIG Professional Development Seminar that growing grass was on the critical path for building Disney's Animal Kingdom theme park! The 500-acre park required special grass for its animal inhabitants, and some of the grass took years to grow. Another misconception is that the critical path is the shortest path through the network diagram. In some areas, such as transportation modeling, identifying the shortest path in network diagrams is the goal. For a project, however, each task or activity on the critical path, as well as other paths, must be done in order to complete the project. It is not a matter of choosing the shortest path.

Other aspects of critical path analysis may cause confusion. Can there be more than one critical path on a project? Does the critical path ever change? In the Project X example, suppose that Activity A has a duration estimate of three days instead of one day. This new duration estimate would make the length of Path 1 equal to 16 days. Now the project has two longest paths of equal duration, so there are two critical paths. Therefore, a project *can* have more than one critical path. Project managers should closely monitor performance of activities on the critical path to avoid late project completion. If there is more than one critical path, project managers must keep their eyes on all of them.

The critical path on a project can change as the project progresses. For example, suppose everything is going as planned at the beginning of the project. Suppose that Activities A, B, C, D, E, F, and G all start and finish as planned. Then suppose that Activity I runs into problems. If Activity I takes more than four days, it will cause path C-G-I-J to be longer than the other paths, assuming they progress as planned. This change would cause path C-G-I-J to become the new critical path. Therefore, the critical path can change on a project.

Using Critical Path Analysis to Make Schedule Trade-Offs

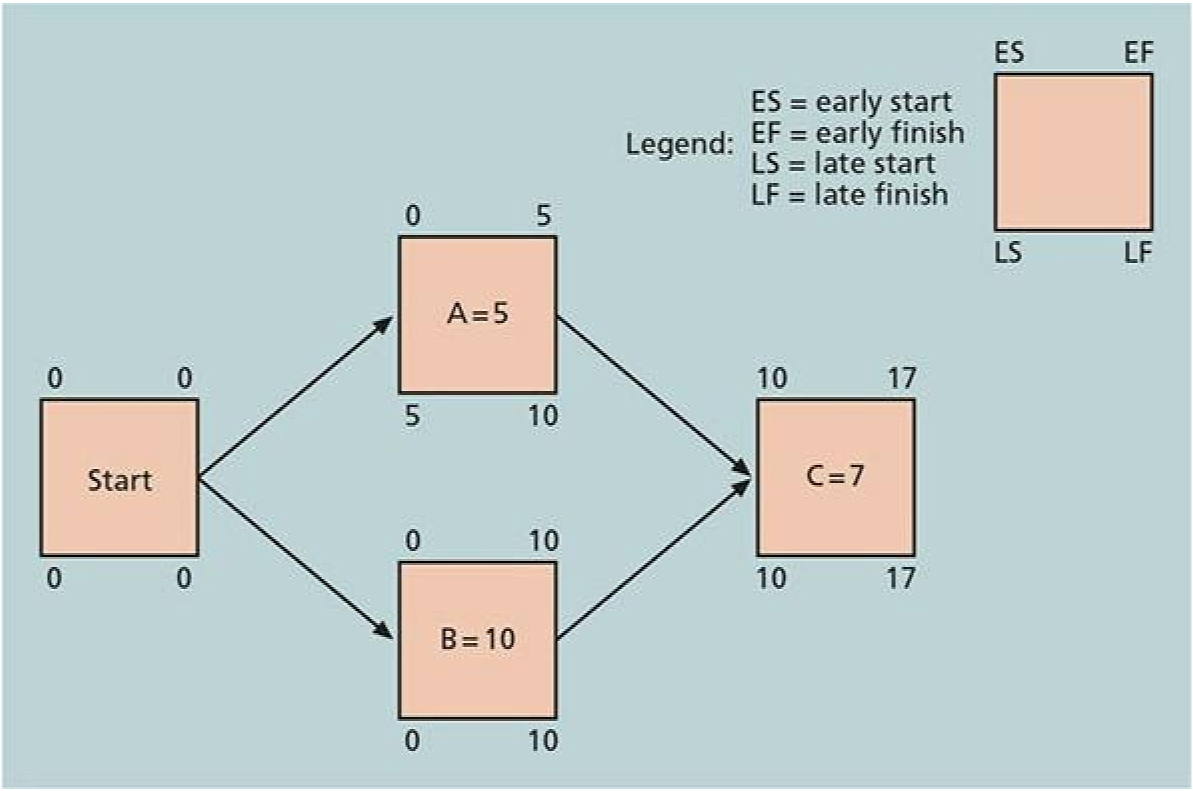
It is important to know the critical path throughout the life of a project so the project manager *can* make trade-offs. If a task on the critical path is behind schedule, the project manager must be aware of the problem and decide what to do about it. Should the schedule be renegotiated with stakeholders? Should more resources be allocated to other items on the critical path to make up for that time? Is it acceptable for the project to finish behind schedule? By keeping track of the critical path, the project manager and the team take a proactive role in managing the project schedule.

A technique that can help project managers make schedule trade-offs is determining the free slack and total slack for each project activity. **Free slack** or **free float** is the amount of time an activity can be delayed without delaying the early start date of any immediately following activities. The **early start date** is the earliest possible time an activity can start based on the project network logic. **Total slack** or **total float** is the amount of time an activity can be delayed from its early start without delaying the planned project finish date.

Project managers calculate free slack and total slack by doing a forward and backward pass through a network diagram. A **forward pass** determines the early start and early finish dates for each activity. The **early finish date** is the earliest possible time an activity can finish based on the project network logic. The project start date is equal to the early start date for the first network diagram activity. The early start plus the duration of the first activity is equal to the early finish date of the first activity. It is also equal to the early start date of each subsequent activity unless an activity has multiple predecessors. When an activity has multiple predecessors, its early start date is the latest of the early finish dates of those predecessors. For example, Tasks D and E immediately precede Task H in Figure 6-8. The early start date for Task H, therefore, is the early finish date of Task E, because it occurs later than the early finish date of Task D. A **backward pass** through the network diagram determines the late start and late finish dates for each activity in a similar fashion. The **late start date** is the latest possible time an activity might begin without delaying the project finish date. The **late finish date** is the latest possible time an activity can be completed without delaying the project finish date.

Project managers can determine the early and late start and finish dates of each activity by hand. For example, **Figure 6-9** shows a simple network diagram with three tasks, A, B, and C. Tasks A and B both precede Task C. Assume that all duration estimates are in days. Task A has an estimated duration of 5 days, Task B has an estimated duration of 10 days, and Task C has an estimated duration of 7 days. There are only two paths through this small network diagram: Path A-C has a duration of 12 days (5 + 7), and path B-C has a duration of 17 days (10 + 7). Because path B-C is longer, it is the critical path. There is no float or slack on this path, so the early and late start and finish dates are the same. However, Task A has 5 days of float or slack. Its early start date is day 0, and its late start date is day 5. Its early finish date is day 5, and its late finish date is day 10. Both the free and total float amounts for Task A are 5 days.

Figure 6-9. Calculating early and late start and finish dates



Using project management software is a much faster and easier way to determine early and late start and finish dates and free and total slack amounts for activities. **Table 6-1** shows the free and total slack for all activities on the network diagram for Project X using the data from Figure 6-8 and assuming that Tasks A, B, and C started on March 26, 2018. (The network diagram is shown in Figure 6-4, which was created with Microsoft Project.) The data in this table was created by selecting the Schedule Table view in Microsoft Project.

Table 6-1. Free and total float or slack for project X

	Start	Finish	Late Start	Late Finish	Free Slack	Total Slack
A	3/26/2018	3/26/2018	3/28/2018	3/29/2018	0d	2d
B	3/26/2018	3/27/2018	3/26/2018	3/28/2018	0d	0d
C	3/26/2018	3/28/2018	3/28/2018	4/2/2018	0d	2d
D	3/27/2018	3/30/2018	3/29/2018	4/4/2018	2d	2d
E	3/28/2018	4/3/2018	3/28/2018	4/4/2018	0d	0d

F	3/28/2018	4/2/2018	4/6/2018	4/12/2018	7d	7d
G	3/29/2018	4/5/2018	4/2/2018	4/10/2018	0d	2d
H	4/4/2018	4/11/2018	4/4/2018	4/12/2018	0d	0d
I	4/6/2018	4/9/2018	4/10/2018	4/12/2018	2d	2d
J	4/12/2018	4/16/2018	4/12/2018	4/16/2018	0d	0d

Knowing the amount of float or slack allows project managers to know whether the schedule is flexible and how flexible it might be. For example, at 7 days (7d), Task F has the most free and total slack. The most slack on any other activity is only 2 days (2d). Understanding how to create and use slack information provides a basis for negotiating project schedules. See the Help information in Microsoft Project or research other resources for more detailed information on calculating slack.

Using the Critical Path to Shorten a Project Schedule

It is common for stakeholders to want to shorten a project schedule estimate. A project team may have done its best to develop a project schedule by defining activities, determining sequencing, and estimating resources and durations for each activity. The results of this work may have shown that the project team needs 10 months to complete the project, but the sponsor might ask if the project can be done in eight or nine months. (Rarely do people ask the project team to take longer than suggested.) By knowing the critical path, the project manager and the team can use several duration compression techniques to shorten the project schedule. One technique is to reduce the duration of activities on the critical path. The project manager can shorten the duration of critical-path activities by allocating more resources to those activities or by changing their scope. Several experts suggest that it is extremely difficult to shorten a reasonable project schedule by more than 25 percent.

Recall that Sue Johnson in the opening case was having schedule problems with the online registration project because several users missed important project review meetings and one of the senior programmers quit. If Sue and her team created a realistic project schedule, produced accurate duration estimates, and established dependencies between tasks, they could analyze their status in terms of meeting the May 1 deadline. If some activities on the critical path had already slipped and they did not build in extra time at the end of the project, then they would have to take corrective actions to finish the project on time. Sue could request that her company or the college provide more people to work on the project in an effort to make up time. She could also request that the scope of activities be reduced to complete the project on time. Sue could also use project schedule management techniques, such as crashing or fast tracking, to shorten the project schedule.

Crashing is a technique for making cost and schedule trade-offs to obtain the greatest amount of schedule compression for the least incremental cost. For example, suppose that one of the items on the critical path for the online registration project was entering course data for the new semester into the new system. If this task has not yet been done and was originally estimated to take two weeks based on the college providing a part-time data entry clerk, Sue could suggest that the college have the clerk work full time to finish the task in one week instead of two. This change would not cost Sue's company more money, and it could shorten the project end date by one week. If the college could not meet this request, Sue could consider hiring a temporary data entry person for one week to help get the task done faster. By focusing on tasks on the critical path that could be done more quickly for no extra cost or a small cost, the project schedule can be shortened.

The main advantage of crashing is shortening the time needed to finish a project. The main disadvantage of crashing is that it often increases total project costs. You will learn more about project costs in Chapter 7, Project Cost Management.

Another technique for shortening a project schedule is fast tracking. **Fast tracking** involves doing activities in parallel that you would normally do in sequence. For example, Sue Johnson's project team may have planned not to start any of the coding for the online registration system until *all* of the analysis was done. Instead, they could consider starting some coding activity before the analysis is completed. The main advantage of fast tracking, like crashing, is that it can shorten the time needed to finish a project. The main disadvantage is that it can increase cost and actually lengthen the project schedule because starting some tasks too soon often increases project risk and results in rework.

Importance of Updating Critical Path Data

In addition to finding the critical path at the beginning of a project, it is important to update the schedule with actual data. After the project team completes activities, the project manager should document the actual durations of those activities. The project manager should also document revised estimates for activities in progress or yet to be started. These revisions often cause a project's critical path to change, resulting in a new estimated completion date for the project. Again, proactive project managers and their teams stay on top of changes so they can make informed decisions and keep stakeholders involved in major project decisions.

Critical Chain Scheduling

Another technique that addresses the challenge of meeting or beating project finish dates is an application of the Theory of Constraints called critical chain scheduling. The **Theory of Constraints (TOC)** is a management philosophy developed by Eliyahu M. Goldratt and discussed in his books *The Goal* and *Critical Chain*. The Theory of Constraints is based on the metaphor of a chain and its weakest link: Any complex system at any point in time often has only one aspect or constraint that limits the ability to achieve more of the system's goal. For the system to attain any significant improvements, that constraint must be identified, and the whole system must be managed with it in mind. **Critical chain scheduling** is a method that considers limited resources when creating a project schedule and includes buffers to protect the project completion date.

An important concept in critical chain scheduling is the availability of scarce resources. Some projects cannot be done unless a particular resource is available to work on one or several tasks. For example, if a television network wants to produce a show centered around a particular celebrity, it must first check the availability of that celebrity. As another example, if a particular piece of equipment is needed full time to complete each of two tasks that were originally planned to occur simultaneously, critical chain scheduling acknowledges that you must either delay one of those tasks until the equipment is available or find another piece of equipment in order to meet the schedule. Other important concepts related to critical chain scheduling include multitasking and time buffers.

Although many people are proud to say they are good at multitasking, multitasking is not a good thing to do if you want to finish a project in a timely manner. **Multitasking** occurs when a resource works on more than one task at a time. This situation occurs frequently on projects. People are assigned to multiple tasks within the same project or different tasks on multiple projects. For example, suppose someone is working on three different tasks—Task 1, Task 2, and Task 3—for three different projects, and each task takes 10 days to complete. If the person did not multitask, and instead completed each task sequentially starting with Task 1, then Task 1 would be completed after day 10, Task 2 would be completed after day 20, and Task 3 would be completed after day 30, as shown in Figure 6-10a. However, because many people in this situation try to please all three parties who need their tasks completed, they often work on the first task for some time, then the second, then the third, then go back to the first task, and so on, as shown in Figure 6-10b. In this example, the tasks were all half-done one at a time, then completed one at a time. Task 1 is now completed at the end of day 20 instead of day 10, Task 2 is completed at the end of day 25 instead of day 20, and Task 3 is still completed on day 30. This example illustrates how multitasking can delay task completion. Multitasking also often involves wasted setup time, which increases total duration.

Figure 6-10a. Three tasks without multitasking

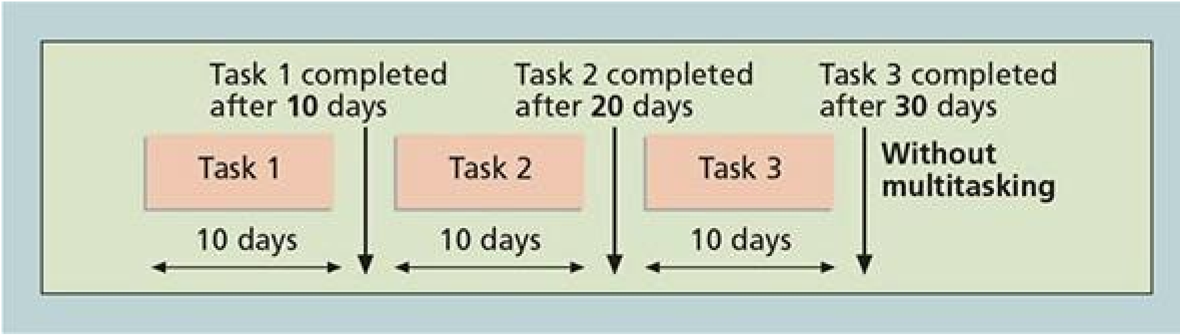
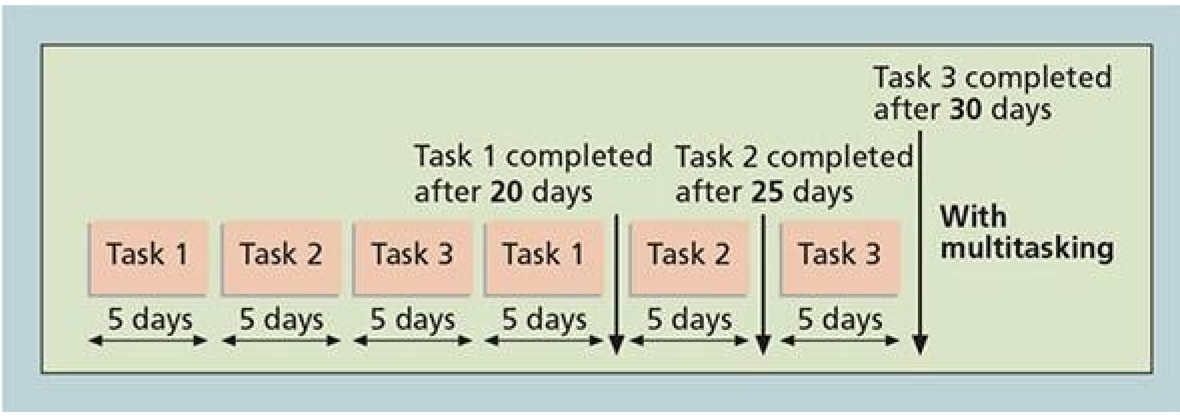


Figure 6-10b. Three tasks with multitasking

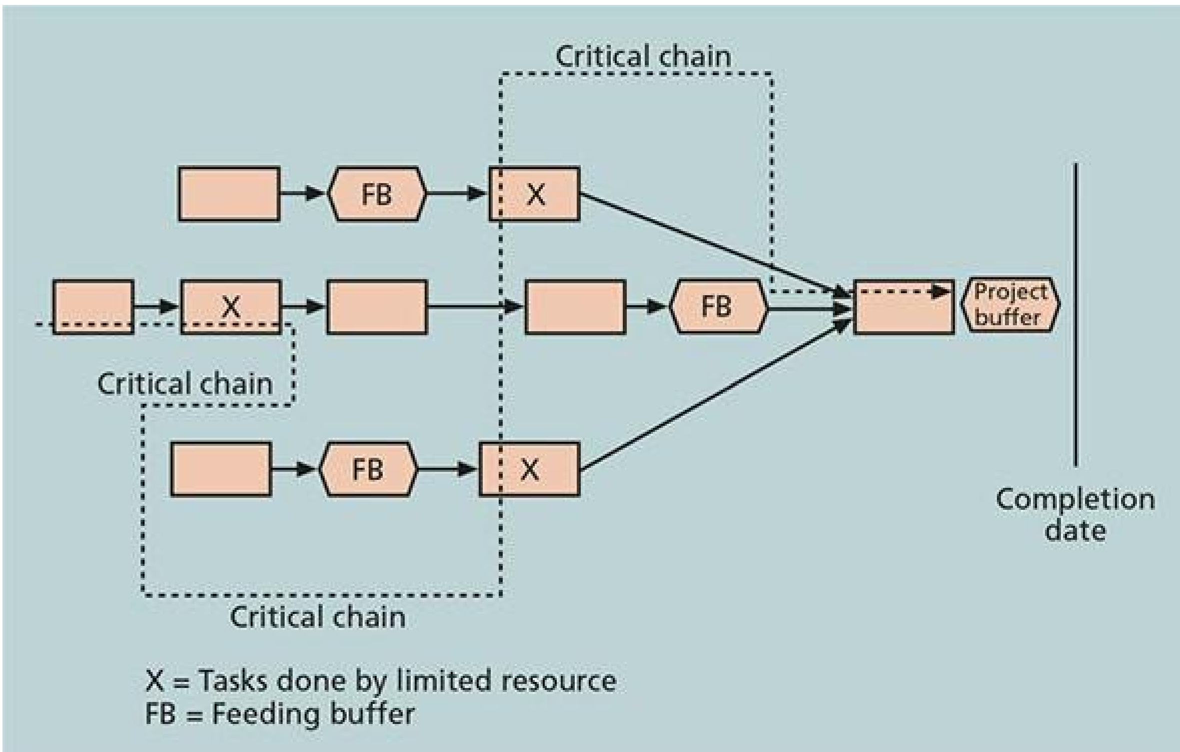


Critical chain scheduling assumes that resources do not multitask or at least minimize multitasking. Someone should not be assigned to two tasks simultaneously on the same project when critical chain scheduling is in effect. Likewise, critical chain theory suggests that projects be prioritized so that people who are working on more than one project at a time know which tasks take priority. Preventing multitasking avoids resource conflicts and wasted setup time caused by shifting between multiple tasks over time.

An essential concept to improving project finish dates with critical chain scheduling is to change the way people make task estimates. Many people add a safety or **buffer** —additional time to complete a task—to an estimate to account for various factors. These factors include the negative effects of multitasking, distractions, and interruptions, fear that estimates will be reduced, and Murphy's Law. **Murphy's Law** states that if something can go wrong, it will. Critical chain scheduling removes buffers from individual tasks and instead creates a **project buffer** , which is time added before the project's due date. Critical chain scheduling also protects tasks on the critical chain from being delayed by using **feeding buffers** , which consist of time added before tasks on the critical chain if they are preceded by other tasks that are not on the critical path.

Figure 6-11 provides an example of a network diagram constructed using critical chain scheduling. Note that the critical chain accounts for a limited resource, X, and the schedule includes use of feeding buffers and a project buffer in the network diagram. The tasks marked with an X are part of the critical chain, which can be interpreted as being the critical path using this technique. The task estimates in critical chain scheduling should be shorter than traditional estimates because they do not include their own buffers. Not having task buffers should mean fewer occurrences of **Parkinson's Law** , which states that work expands to fill the time allowed. The feeding and project buffers protect the date that really needs to be met —the project completion date.

Figure 6-11. Example of critical chain scheduling*



* Eliyahu Goldratt, *Critical Chain* (Great Barrington, MA: The North River Press, 1997), p. 218.

Several organizations have reported successes with critical chain scheduling. For example, the What Went Right? feature shows how the healthcare industry is learning to think differently by using the Theory of Constraints.

What Went Right?

Experts in healthcare quality management have thought carefully about how scheduling works at a doctor's office or healthcare clinic and have considered how it might be improved:

> Consider a relatively simple system of a physician's office or clinic. The steps in the process could be patients checking in, filling out forms, having vital signs taken by a nurse, seeing the physician, seeing the nurse for a prescribed procedure such as vaccination, and so forth. These steps could take place in a simple linear sequence or chain…… Each link in this chain has the ability to perform its tasks at different average rates. In this example, the first resource can process 13 patients, charts, or blood samples per hour; the second can process 17, and so forth. One may think that this process can produce 13 per hour, the average of all resources. In fact, this process or chain only can produce an average of eight per hour. The chain is only as strong as its weakest link and the rate of the slowest resource in this example, the weakest link, is eight. This is true regardless of how fast each of the other resources can process individually, how much work is stuffed into the pipeline, or how complex the process or set of interconnected processes is to complete. Moreover, improving the performance of any link besides the constraint does nothing to improve the system as a whole.*

In fact, the Avraham Y. Goldratt Institute (www.goldratt.com) applied this thinking at a client's site and showed just how well critical chain scheduling can work. The National University Hospital in Singapore decreased patient admission times by more than 50 percent. Improved scheduling lowered average wait times, which went from six to eight hours to less than three hours. By applying critical chain techniques, the hospital admitted 63 percent of patients in less than 1.5 hours. Although these times still seem long, they show significant improvement.*

As you can see, critical chain scheduling is a fairly complicated yet powerful tool that involves critical path analysis, resource constraints, and changes in how buffers are used as part of task estimates. Some people consider critical chain scheduling one of the most important new concepts in the field of project management.

Program Evaluation and Review Technique (PERT)

When there is a high degree of uncertainty about the individual activity duration estimates, the network analysis **Program Evaluation and Review Technique (PERT)** can be used to estimate project duration. PERT applies the critical path method (CPM) to a weighted average duration estimate. This approach was developed at about the same time as CPM, in the late 1950s, and it also uses network diagrams, which are still sometimes referred to as PERT charts.

PERT uses **probabilistic time estimates** —duration estimates based on using optimistic, most likely, and pessimistic estimates of activity durations—instead of one specific or discrete duration estimate, as CPM does. To use PERT, you calculate a weighted average for the duration estimate of each project activity using the following formula:

$$\text{PERT weight average} = \frac{\text{optimistic time} + 4 * \text{most likely time} + \text{pessimistic time}}{6}$$

By using the **PERT weighted average** for each activity duration estimate, the total project duration estimate takes into account the risk or uncertainty in the individual activity estimates. Note that the above formula assumes a normal distribution or bell curve. You can also use other distributions. Consult other sources for more details.

Suppose that Sue Johnson's project team in the opening case used PERT to determine the schedule for the online registration system project. The team would have to collect numbers for the optimistic, most likely, and pessimistic duration estimates for each project activity. Suppose that one of the activities was to design an input screen for the system. Someone might estimate that it would take about two weeks or 10 workdays to do this activity. Without using PERT, the duration estimate for that activity would be 10 workdays. Using PERT, the project team would also need to estimate the pessimistic and optimistic times for completing this activity. Suppose an optimistic estimate is that the input screen can be designed in eight workdays, and a pessimistic time estimate is 24 workdays. Applying the PERT formula, you get the following:

$$\begin{aligned}\text{PERT weight average} &= \frac{\text{optimistic time} + 4 * \text{most likely time} + \text{pessimistic time}}{6} \\ &= 12\ \text{workdays}\end{aligned}$$

Instead of using the most likely duration estimate of 10 workdays, the project team would use 12 workdays when doing critical path analysis. These additional two days could help the project team get the work completed on time.

The main advantage of PERT is that it attempts to address the risk associated with duration estimates. Because many projects exceed schedule estimates, PERT may help in developing schedules that are more realistic. PERT's main disadvantages are that it involves more work than CPM because it requires several duration estimates, and there are better probabilistic methods for assessing schedule risk. (See the information on Monte Carlo simulations in Chapter 11, Project Risk Management.)

Agile and Schedule Management

Recall from **Chapter 2** that two of the core values of the Manifesto for Agile Software Development are "customer collaboration over contract negotiation" and "responding to change over following a plan." These values seem to contradict some of the structured project schedule management processes and tools described in this chapter. Agile methods like Scrum were designed to address collaboration and flexibility, especially on projects with a complex scope of work. For example, the product owner defines and prioritizes the work to be done within a sprint, so customer collaboration is designed into the process. The short period of each sprint (normally two to four weeks) and daily Scrum meetings provide an environment where team members collaborate to focus on completing specific tasks.

An adaptive or agile approach can be used on large projects, even after they are started using a more prescriptive approach. For example, the FBI says that switching to an agile approach is what helped them complete the Sentinel project described in the What Went Wrong? feature earlier in the chapter. The system went live "thanks to the fact that it was rescued by following an agile software development methodology…… The primary lesson? Agile actually works. It's not perfect, and many people in government and in the contractor community still struggle with it, but it's succeeding where it counts—enabling the rollout of large-scale IT projects that are on time, on budget and actually do what stakeholders want them to do. Imagine that."*

How was project schedule management different after moving to an agile approach? After changing to a Scrum method in 2010, the work for the Sentinel project was organized into user stories based on the original systems requirements specifications. Each user story was allocated a number of story points based on difficulty. At the beginning of each two-week sprint, the development team identified which user stories to do during each sprint. At the end of each sprint the team demonstrated the system, regardless of whether all of the work was completed, but only the user stories that passed tests (approved by the customers) were reported as complete. If user stories were not complete, they were moved to the product backlog. This approach helped the team focus on delivering a working system that met customer needs in a specified timeframe. In other words, the emphasis was on completing some useful work for the customer in short time increments versus trying to define all the work required first and then scheduling when it could be done.*

Controlling the Schedule

The final process in project schedule management is controlling the schedule. Like scope control, schedule control is a portion of the integrated change control process in project integration management. The goal of schedule control is to know the status of the schedule, influence the factors that cause schedule changes, determine that the schedule has changed, and manage changes when they occur.

The main inputs to schedule control are the project management plan, project documents (like the lessons-learned register, project calendars, project schedule, resource calendars, and schedule data), work performance data, and organizational process assets. Some of the tools and techniques include the following:

- Data analysis tools, including
    - Earned value analysis, described in detail in Chapter 7, Project Cost Management
    - Iteration burndown charts, as described in **Chapter 3**
    - Performance reviews
    - Trend analysis
    - Variance analysis
    - What-if-scenario analysis
- Critical path method, described earlier in this chapter
- Project management information systems
- Resource optimization, such as resource leveling, described in Chapter 9, Project Resource Management
- Leads and lags, described earlier
- Schedule compression, such as crashing and fast tracking

The main outputs of schedule control include work performance information, schedule forecasts, change requests, project management plan updates, and project documents updates.

Many issues are involved in controlling changes to project schedules. It is important first to ensure that the project schedule is realistic. Many projects, especially in IT, have very unrealistic schedule expectations. It is also important to use discipline and leadership to emphasize the importance of following and meeting project schedules. Although the various tools and techniques assist in developing and managing project schedules, project managers must handle several personnel-related issues to keep projects on track. “Most projects fail because of people issues, not from failure to draw a good PERT chart.”* Project managers can perform a number of reality checks that will help them manage changes to project schedules. Several soft skills can help project managers to control schedule changes.

Reality Checks on Scheduling and the Need for Discipline

It is important for projects to have realistic schedule goals and for project managers to use discipline to help meet those goals. One of a project manager's first reality checks is to review the draft schedule that is usually included in the project charter. Although this draft schedule might include only a project start and end date, the project charter sets some initial schedule expectations for the project. Next, the project manager and project team should prepare a more detailed schedule and get stakeholders' approval. To establish the schedule, it is critical to get involvement and commitment from all project team members, top management, the customer, and other key stakeholders.

Another type of reality check comes from progress meetings with stakeholders. The project manager is responsible for keeping the project on track, and key stakeholders like to stay informed, often through high-level periodic reviews. Project managers often illustrate progress with a Tracking Gantt chart that shows key deliverables and activities. The project manager needs to understand the schedule, including why activities are or are not on track, and take a proactive approach to meeting stakeholder expectations. It is also important to verify schedule progress, as Sue Johnson from the opening case discovered (see the **Case Wrap-Up** later in this chapter). Just because a team member says a task was completed on time does not always mean that it was. Project managers must review the actual work and develop a good relationship with team members to ensure that work is completed as planned or changes are reported as needed.

Top management hates surprises, so the project manager must be clear and honest in communicating project status. By no means should project managers create the illusion that the project is going fine when it is actually having serious problems. When serious conflicts arise that could affect the project schedule, the project manager must alert top management and work with them to resolve the conflicts.

Project managers must also use discipline to control project schedules. Several IT project managers have discovered that setting firm dates for key project milestones helps minimize schedule changes. It is very easy for scope creep to raise its ugly head on IT projects. Insisting that important schedule dates be met and that proper planning and analysis be completed up front helps everyone focus on doing what is most important for the project. This discipline results in meeting project schedules.

Using Software to Assist in Project Schedule Management

Several types of software are available that assist with project schedule management. Software for facilitating communications helps project managers exchange schedule-related information with project stakeholders. Decision support models can help project managers analyze various trade-offs that can be made to address schedule issues.

However, project management software was designed specifically for performing project management tasks. You can use project management software to draw network diagrams, determine the critical path for a project, create Gantt charts, and report, view, and filter specific project schedule management information. For example, **Figures 6-4**, 6-5, 6-6, 6-7, and Table 6-1 were all created using Microsoft Project; Appendix A (available on the Companion website for this text) shows many more examples of using this software to assist in project schedule management.

Many projects involve hundreds of tasks with complicated dependencies. After you enter the necessary information, project management software automatically generates a network diagram and calculates the critical path(s) for the project. It also highlights the critical path in red on the network diagram.

Project management software also calculates the free and total float or slack for all activities. Using project management software eliminates the need to perform cumbersome calculations manually and allows for "what if" analysis as activity duration estimates or dependencies change. Recall that knowing which activities have the most slack allows the project manager to reallocate resources or make other changes to compress the schedule or help keep it on track.

Project 2016 easily creates Gantt charts and Tracking Gantt charts, which make it easier for project teams to track actual schedule performance versus the planned or baseline schedule. However, for a project to benefit from using Tracking Gantt charts, actual schedule information must be entered in a timely manner. Some organizations use e-mail or other communications software to send updated task and schedule information to the person responsible for updating the schedule. He or she can then quickly authorize these updates and enter them directly into the project management software. This process provides an accurate and up-to-date project schedule in Gantt chart form.

Project 2016 also includes many built-in reports, views, and filters to assist in project schedule management. For example, a project manager can quickly run a report to list all tasks that are scheduled to start soon, and then send a reminder to the people responsible for these tasks. A project manager who was presenting project schedule information to top management could create a Gantt chart showing only summary tasks or milestones. You can also create custom reports, views, tables, and filters. See Appendix A (available on the Companion website for this text) to learn how to use the project schedule management features of Project 2016.

Global Issues

Most software companies use case studies or customer testimonials to showcase how their software helps customers solve business problems. Microsoft tells the customer story of Mexico's Secretary of Economy, who wanted to ensure that IT initiatives aligned with business goals and improved project management efficiency. Internal IT project requests often took several weeks to fulfill because the organization lacked a centralized method of capturing project information and used several diagramming tools.

The Director General of Information Technology decided to establish an Office of Project Management and search for a solution that combined professional diagramming capabilities with project management and collaboration tools. Assistant Director General of the Office of Information Technology, Carlos Benítez Gonzales, worked with other agencies to find the best solution. After deploying several of Microsoft's tools (Visio, Project, and SharePoint), the organization was able to handle more internal IT projects without adding staff, while fulfilling requests 60 percent faster.

"By documenting and incorporating governance procedures and project lifecycle stages into a standardized workflow process, the Secretary's IT team can finish projects in less time. Benítez estimates that it previously took an average of five business days to fulfill requests. The team can now complete the same work in two business days on average. . . . 'We now have a single platform for managing process information that used to be scattered throughout the organization and inaccessible because it was stored on local computers,' says Benítez. The IT team can now handle four times the number of concurrent projects without the need to hire additional staff."*

Words of Caution on Using Project Management Software

Many people misuse project management software because they do not understand the concepts behind creating a network diagram, determining the critical path, or setting a schedule baseline. They might also rely too heavily on sample files or templates in developing their own project schedules. Understanding the underlying concepts (or even being able to work with the tools manually) is critical to successful use of project management software, as is understanding the specific needs of your project.

Many top managers, including software professionals, have made blatant errors using various versions of Microsoft Project and similar tools. For example, one top manager did not know how to establish dependencies among project activities and entered every single start and end date for hundreds of activities. When asked what would happen if the project started a week or two late, she responded that she would have to reenter all of the dates.

This manager did not understand the importance of establishing relationships among the tasks, which allows the software to update formulas automatically when the inputs change. If the project start date slips by one week, the project management software will update all the other dates automatically, as long as they are not manually entered into the software. If one activity cannot start before another ends, and the first activity's actual start date is two days late, the start date for the succeeding activity will automatically be moved back two days. To achieve this type of functionality, tasks that have relationships must be linked in project management software.

Another top manager on a large IT project did not know how to set a baseline in Microsoft Project. He spent almost one day every week copying and pasting information from Microsoft Project into a spreadsheet and using complicated "IF" statements to figure out what activities were behind schedule. He had never received training on Microsoft Project and did not know about many of its capabilities. To use any software effectively, users must have adequate training in the software and an understanding of its underlying concepts.

Many project management software programs also come with templates or sample files. It is very easy to use these files without considering unique project needs. For example, a project manager for a software development project can use the Microsoft Project Software Development template file, the files from similar projects done in the past, or sample files purchased from other companies. All of these files include suggested tasks, durations, and relationships. There are benefits to using templates or sample files, such as less setup time and a reality check if the project manager has never led a certain type of project before. However, there are drawbacks to this approach. Many assumptions made in these template files might not apply to the project, such as a design phase that takes three months to complete or the performance of certain types of testing. Project managers and their teams should be careful not to rely too much on templates or sample files and ignore the unique concerns of their particular projects.

Considerations for Agile/Adaptive Environments

The *PMBOK® Guide – Sixth Edition* provides the following information for project schedule management:

> Adaptive approaches use short cycles to undertake work, review the results, and adapt as necessary. These cycles provide rapid feedback on the approaches and suitability of deliverables, and generally manifest as iterative schedule and on-demand, pull-based scheduling, as discussed in the section on Key Trends and Emerging Practices in Project Schedule Management.

In large organizations, there may be a mixture of small projects and large initiatives requiring long-term roadmaps to manage the development of these programs using scaling factors (e.g., team size, geographical distribution, regulatory compliance, organizational complexity, and technical complexity). To address the full delivery life cycle for larger, enterprise-wide systems, a rage of techniques utilizing a predictive approach, adaptive approach, or a hybrid of both may need to be adopted. The organization may need to combine practices from several core methods, or adopt a method that has already done so, and adopt a few principles and practices of more traditional techniques.

The role of the project manager does not change based on managing projects using a predictive development life cycle of managing projects in adaptive environments. However, to be successful in using adaptive approaches, the project manager will need to be familiar with the tools and techniques to understand how to apply them effectively.*

Schedule management is radically different using Agile and Scrum in particular. Projects that rely heavily on the critical path method consider meeting the project's estimated completion date as a crucial component of success. Agile projects, on the other hand, may not even need to estimate activity durations or project schedules at all. Why? Because overall project completion time is not important. The customer/product owners are empowered to demand what they want in each and every short iteration (in the product backlog), and they get to see and touch a working product, and approve of it, by the end of each iteration. Customer satisfaction and project progress are extremely transparent at each iteration.

Project schedule management is challenging for all types of projects. This chapter highlights important concepts, processes, and tools and techniques in this knowledge area.

Case Wrap-Up

It was now March 15, just a month and a half before the new online registration system was supposed to go live. The project was in total chaos. Sue Johnson thought she could handle all of the conflicts that kept occurring on the project, and she was too proud and scared to admit to her top management or the college president that things were not going well. She spent a lot of time preparing a detailed schedule for the project, and she thought she was using the project management software well enough to keep up with project status. However, the five main programmers on the project all figured out a way to generate automatic updates for their tasks every week, saying that everything was completed as planned. They paid very little attention to the actual plan and hated filling out status information. Sue did not check most of their work to verify that it was actually completed. In addition, the head of the Registrar's Office was uninterested in the project and delegated sign-off responsibility to one of his clerks, who did not completely understand the registration process. When Sue and her team started testing the new system, she learned they were using last year's course data, which caused additional problems because the college was moving from quarters to semesters in the new term. How could they have missed that requirement? Sue hung her head in shame and anguish as she walked into a meeting with her manager to ask for help. She learned the hard way how difficult it was to keep a project on track. She wished she had spent more time talking face-to-face with key project stakeholders, especially her programmers and the Registrar's Office representatives, to verify that the project was on schedule and that the schedule was updated accurately.

Chapter Summary

Project schedule management is often cited as the main source of conflict on projects. Most IT projects exceed time estimates. The main processes involved in project schedule management include planning schedule management, defining activities, sequencing activities, estimating activity durations, developing the schedule, and controlling the schedule.

Planning schedule management involves determining the policies, procedures, and documentation that will be used for planning, executing, and controlling the project schedule. The main output is a schedule management plan.

Defining activities involves identifying the specific activities that must be completed to produce the project deliverables. It usually results in a more detailed WBS.

Sequencing activities determines the relationships or dependencies between activities. Three reasons for creating relationships are that they are mandatory based on the nature of the work, discretionary based on the project team's experience, or external based on non-project activities. Activity sequencing must be done in order to use critical path analysis.

Network diagrams are the preferred technique for showing activity sequencing. The two methods used to create these diagrams are the arrow diagramming method and the precedence diagramming method. There are four types of relationships between tasks: finish-to-start, finish-to-finish, start-to-start, and start-to-finish.

Estimating activity durations creates estimates for the amount of time it will take to complete each activity. These time estimates include the actual amount of time worked plus elapsed time.

Developing the schedule uses results from all of the other project schedule management processes to determine the start and end dates for the project. Project managers often use Gantt charts to display the project schedule. Tracking Gantt charts show planned and actual schedule information.

The critical path method predicts total project duration. The critical path for a project is the series of activities that determines the earliest completion date for the project. It is the longest path through a network diagram. If any activity on the critical path slips, the whole project will slip unless the project manager takes corrective action.

Crashing and fast tracking are two techniques for shortening project schedules. Project managers and their team members must be careful about accepting unreasonable schedules, especially for IT projects. There is a practical limit to shortening schedules of about 25 percent, and it almost always increases project cost and risk.

Critical chain scheduling is an application of the Theory of Constraints (TOC) that uses critical path analysis, resource constraints, and buffers to help meet project completion dates.

The Program Evaluation and Review Technique (PERT) is a network analysis technique used to estimate project duration when there is a high degree of uncertainty about the individual activity duration estimates. PERT uses optimistic, most likely, and pessimistic estimates of activity durations. PERT is seldom used today.

Agile methods like Scrum take a different approach to project schedule management by providing more flexibility. The short time period of each sprint (normally two to four weeks) and daily Scrum meetings provide an environment where team members collaborate to focus on completing specific tasks within that time period. The product owner identifies and prioritizes tasks to be done during each sprint.

Controlling the schedule is the final process in project schedule management. Even though scheduling techniques are very important, most projects fail because of personnel issues, not from a poor network diagram. Project managers must involve all stakeholders in the schedule development process. It is critical to set realistic project schedules and use discipline to meet schedule goals.

Project management software can assist in project scheduling if used properly. With project management software, you can avoid the need to perform cumbersome calculations manually and perform "what if" analysis as activity duration estimates or dependencies change. Many people misuse project management software because they do not understand the concepts behind creating a network diagram, determining the critical path, or setting a schedule baseline. Project managers must also avoid relying too much on sample files or templates when creating their unique project schedules.

Be sure to consider how project schedule management can differ in agile/adaptive environments.

Discussion Questions

1. Why do you think schedule issues often cause the most conflicts on projects?
2. Why is defining activities a process of project schedule management instead of project scope management?
3. Why is it important to determine activity sequencing on projects? Discuss diagrams you

have seen that are similar to network diagrams. Describe their similarities and differences.

4. Explain the difference between estimating activity durations and estimating the effort required to perform an activity.

5. Explain the following schedule development tools and concepts: Gantt charts, critical path method, PERT, critical chain scheduling, and sprints.

6. What do you think about adding slack to individual task estimates (sometimes called *padding estimates*)? What do you think about adding a project buffer for the entire project, as critical chain scheduling suggests? What are some ethical considerations when using slack and buffers?

7. How can you minimize or control changes to project schedules?

8. List some of the reports you can generate with Project 2016 to assist with project schedule management.

9. Why is it difficult to use project management software well?

10. How is schedule management different when using an agile approach?

Quick Quiz

1. Which of the following processes involves determining the policies, procedures, and documentation that will be used for planning, executing, and controlling the project schedule?

    a. Planning schedule management

    b. Defining activities

    c. Estimating activity resources

    d. Activity sequencing

2. Predecessors, successors, logical relationships, leads and lags, resource requirements, constraints, imposed dates, and assumptions are all examples of _____ .

    a. items in an activity list

    b. items on a Gantt chart

    c. milestone attributes

d. activity attributes

3. As the project manager for a software development project, you are helping to develop the project schedule. You decide that writing code for a system should not start until users sign off on the analysis work. What type of dependency is this?

   a. Technical

   b. Mandatory

   c. Discretionary

   d. External

4. You cannot start editing a technical report until someone else completes the first draft. What type of dependency does this represent?

   a. Finish-to-start

   b. Start-to-start

   c. Finish-to-finish

   d. Start-to-finish

5. Which of the following statements is false?

   a. A burndown chart is a tool for schedule control.

   b. Duration and effort are synonymous terms.

   c. A three-point estimate includes an optimistic, a most likely, and a pessimistic estimate.

   d. A Gantt chart is a common tool for displaying project schedule information.

6. What symbol on a Gantt chart represents a slipped milestone?

   a. A black arrow

   b. A white arrow

   c. A black diamond

   d. A white diamond

7. What type of diagram shows planned and actual project schedule information?

    a. A network diagram

    b. A Gantt chart

    c. A Tracking Gantt chart

    d. A milestone chart

8. ____ is a network diagramming technique used to predict total project duration.

    a. PERT

    b. A Gantt chart

    c. Critical path method

    d. Crashing

9. Which of the following statements is false?

    a. Growing grass was on the critical path for a large theme park project.

    b. The critical path is the series of activities that determine the earliest time by which a project can be completed.

    c. A forward pass through a project network diagram determines the early start and early finish dates for each activity.

    d. Fast tracking is a technique for making cost and schedule trade-offs to obtain the greatest amount of schedule compression for the least incremental cost.

10. ____ is a method of scheduling that considers limited resources when creating a project schedule and includes buffers to protect the project completion date.

    a. Parkinson's Law

    b. Scrum

    c. Critical path analysis

    d. Critical chain scheduling

Exercises

1. Using **Figure 6-2**, enter the activities, their durations (in days), and their relationships in Project 2016. Use a project start date of March 26, 2018 if you want the dates to match exactly. View the network diagram. Does it look like Figure 6-4? Print the network diagramon one page. Return to the Gantt chart view. To re-create Table 6-1, right-click the **Select All** button to the left of the Task Mode column heading and select **Schedule**. Alternatively, you can click the **View** tab and click the **Tables** button under the Data group and then select **Schedule**. You may need to move the split bar to the right to reveal all of the table columns. (See Appendix A—available on the Companion website for this text—for detailed information on using Project 2016.) Write a few paragraphs explaining what the network diagram and schedule table show about Project X's schedule.

2. Consider **Table 6-2**. All duration estimates or estimated times are in days, and the network proceeds from Node 1 to Node 9. (Note that you can easily change this table to create multiple exercises.)

Table 6-2. Network diagram data for a small project

Activity	Initial Node	Final Node	Estimated Duration
A	1	2	2
B	2	3	2
C	2	4	3
D	2	5	4
E	3	6	2
F	4	6	3
G	5	7	6
H	6	8	2
I	6	7	5
J	7	8	1
K	8	9	2

a. Draw an AOA network diagram representing the project. Put the node numbers in circles and draw arrows from node to node, labeling each arrow with the activity letter and estimated time.

b. Identify all of the paths on the network diagram and note how long they are, using **Figure 6-8** as a guide for how to represent each path.

c. What is the critical path for this project and how long is it?

d. What is the shortest possible time needed to complete this project?

3. Consider **Table 6-3**. All duration estimates or estimated times are in weeks, and the network proceeds from Node 1 to Node 8. (Note that you can easily change this table to create multiple exercises.)

    a. Draw an AOA network diagram representing the project. Put the node numbers in circles and draw arrows from node to node, labeling each arrow with the activity letter and estimated time.

    b. Identify all of the paths on the network diagram and note how long they are, using **Figure 6-8** as a guide for how to represent each path.

    c. What is the critical path for this project and how long is it?

    d. What is the shortest possible time needed to complete this project?

Table 6-3. Network diagram data for a large project

Activity	Initial Node	Final Node	Estimated Duration
A	1	2	10
B	1	3	12
C	1	4	8
D	2	3	4
E	2	5	8
F	3	4	6
G	4	5	4
H	4	6	8
I	5	6	6
J	5	8	12
K	6	7	8
L	7	8	10

4. Enter the information from **Exercise 2** into Project 2016. View the network diagram and task schedule table to see the critical path and float or slack for each activity. Print the Gantt chart and network diagram views and the task schedule table. Write a short paper that interprets this information for someone unfamiliar with project schedule management.

5. Enter the information from **Exercise 3** into Project 2016. View the network diagram and task schedule table to see the critical path and float or slack for each activity. Print the Gantt chart and network diagram views and the task schedule table. Write a short paper

that interprets this information for someone unfamiliar with project schedule management.

6. Create a scenario and schedule for a 6-month or 12-month project. Include the five process groups, but focus on key tasks for executing the project. For example, you could develop a schedule for starting a business, renovating a house, developing an app, or writing a book. Give only your scenario information to a classmate to work on, and compare the results. Create a schedule based on your classmate's scenario. Write a short paper summarizing the similarities and differences in what each of you created and the challenges you faced.

7. Find at least three different sample schedules created in Microsoft Project, MindView, or other project management software. Analyze the schedules, focusing on how complete the task lists are, how realistic the durations are, whether dependencies are included, and so on. Document your results in a short paper, including screenshots of the schedules and references.

8. Interview someone who uses some of the techniques discussed in this chapter. How does the person feel about network diagrams, critical path analysis, Gantt charts, critical chain scheduling, Scrum, using project management software, and managing the personnel issues involved in project schedule management? Write a short paper describing the responses.

9. Review two different articles about critical chain scheduling. Write a short paper describing how this technique can help improve project schedule management.

10. Search for videos that explain how and why you would want to find the critical path for a project. Write a short paper describing your findings. Summarize the best two videos; provide screenshots and explain why you liked the videos. (Note: Instructors can change the topic from critical path to other concepts in this chapter as well.)

11. Research how schedule management is performed on agile projects. Summarize your findings in a short paper or presentation.

Running Case

You are the project manager for the Global Treps Project, sponsored by Dr. K. Team members include Bobby, your IT guy; Kim, a new college grad now working for a non profit group in Vietnam; Ashok, a business student in India; and Alfreda, a student in the United States originally from Ethiopia. You plan to outsource some of the work (e.g., purchasing laptops, developing a website feature for accepting donations, and creating videos for the website). Recall that your schedule and cost goals are to complete the project in six months for under $120,000.

Tasks

1. Review the WBS and Gantt chart you created for **Tasks 3** and 4 in Chapter 5. Propose three to five additional activities that would help you estimate resources and durations. Write a one-page paper describing these new activities.

2. Identify at least five milestones for the Global Treps Project. Write a short paper describing each milestone using the SMART criteria. Discuss how determining the details of these milestones might add activities or tasks to the Gantt chart. Remember that milestones normally have no duration, so you must have tasks that will lead to completing them.

3. Using the Gantt chart you created for **Task 4** in Chapter 5 and the new activities and milestones you proposed in Tasks 1 and 2 above, create a new Gantt chart using Project 2016 or another tool. Estimate the task durations and enter dependencies as appropriate. Remember that your schedule goal for the project is six months. Print the Gantt chart and network diagram, each on one page.

Key Terms

activity p.223

activity attributes p.226

activity list p.226

activity-on-arrow (AOA) p.229

arrow diagramming method (ADM) p.229

backward pass p.241

baseline dates p.238

buffer p.245

burst p.230

crashing p.243

critical chain scheduling p.244

critical path p.238

critical path method (CPM) or critical path analysis p.238

dependency p.228

discretionary dependencies p.228

dummy activities p.232