[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html" \l "header)

**Chapter**

**3**

**Managing Systems Projects**

* **3.1**[Overview of Project Management](javascript://)
  + **3.1.1**[What Shapes a Project?](javascript://)
  + **3.1.2**[What Is a Project Triangle?](javascript://)
  + **3.1.3**[What Does a Project Manager Do?](javascript://)
* **3.2**[Creating a Work Breakdown Structure](javascript://)
  + **3.2.1**[Gantt Charts](javascript://)
  + **3.2.2**[PERT/CPM Charts](javascript://)
  + **3.2.3**[Identifying Tasks in a Work Breakdown Structure](javascript://)
  + **3.2.4**[Factors Affecting Duration](javascript://)
  + **3.2.5**[Displaying the Work Breakdown Structure](javascript://)
* **3.3**[Task Patterns](javascript://)
  + **3.3.1**[Using Task Boxes to Create a Model](javascript://)
  + **3.3.2**[Task Pattern Types](javascript://)
  + **3.3.3**[Working with Complex Task Patterns](javascript://)
* **3.4**[The Critical Path](javascript://)
  + **3.4.1**[Calculating the Critical Path](javascript://)
* **3.5**[Project Monitoring and Control](javascript://)
  + **3.5.1**[Monitoring and Control Techniques](javascript://)
  + **3.5.2**[Maintaining a Schedule](javascript://)
  + **3.5.3**[Tasks and the Critical Path](javascript://)
* **3.6**[Reporting](javascript://)
  + **3.6.1**[Project Status Meetings](javascript://)
  + **3.6.2**[Project Status Reports](javascript://)
  + **3.6.3**[Dealing with Problems](javascript://)
* **3.7**[Project Management Software](javascript://)
* **3.8**[Risk Management](javascript://)
* **3.9**[Managing for Success](javascript://)
  + **3.9.1**[Business Issues](javascript://)
  + **3.9.2**[Budget Issues](javascript://)
  + **3.9.3**[Schedule Issues](javascript://)
* **3.10**[Summary](javascript://)
* [Chapter Review](javascript://)
  + [Key Terms](javascript://)

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

**3.1**Overview of Project Management

Many professionals manage business and personal projects every day but do not always give it much thought. The management process for developing an information system or working on a construction project is much the same. The only difference is the nature of the project. To manage a large-scale IT project, specific tools and techniques are needed. A project manager is also needed, someone who is responsible for overseeing all relevant tasks. **Project management** for IT professionals includes planning, scheduling, monitoring and controlling, and reporting on information system development.

A project manager will break the project down into individual tasks, determine the order in which the tasks need to be performed, and figure out how long each task will take. With this information, Gantt charts or PERT/CPM charts can be used to schedule and manage the work. Microsoft Project is a popular project management tool that can help create and then monitor the project plan, report progress, and use risk management to make the whole process easier for everyone.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.1.1What Shapes a Project?

A successful project must be completed on time, be within budget, and deliver a quality product that satisfies users and meets requirements. Project management techniques can be used throughout the SDLC. Systems developers can initiate a formal project as early as the preliminary investigation stage, or later on, as analysis, design, and implementation activities occur.

Systems development projects tend to be dynamic and challenging. There is always a balance between constraints, which was discussed in [Chapter 2](javascript://), and interactive elements such as project cost, scope, and time.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.1.2What Is a Project Triangle?

[Figure 3-1](javascript://) shows a very simple example of a project triangle. For each project, it must be decided what is most important, because the work cannot be good and fast and cheap.

**Figure 3-1**

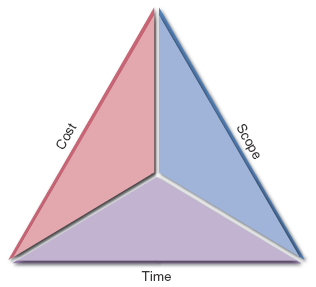
You can’t get everything you want; you have to make choices.



When it comes to project management, things are not quite so simple. Decisions do not need to be all-or-nothing but recognize that any change in one leg of the triangle will affect the other two legs. [Figure 3-2](javascript://) represents a common view of a [**project triangle**](javascript://), where the three legs are cost, scope, and time. The challenge is to find the optimal balance among these factors. Most successful project managers rely on personal experience, communication ability, and resourcefulness. For example, if an extremely time-critical project starts to slip, the project manager might have to trim some features, seek approval for a budget increase, add new personnel, or a combination of all three actions.

**Figure 3-2**

A typical project triangle includes cost, scope, and time.



On its website, Microsoft offers an interesting suggestion for project managers who have a project at risk: Find the “stuck side” of the triangle. Microsoft states that in most projects, at least one side of the triangle is fixed and unlikely to change. It might be a budget cast in stone, a scope that is inflexible, or a schedule driven by factors beyond the firm’s control. Whichever side is fixed is probably critical to the project’s success. The leg where the problem resides must also be identified: cost, scope, or time.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.1.3What Does a Project Manager Do?

Whether a project involves a new office building or an information system, good leadership is essential. In a systems project, the [**project manager**](javascript://), or [**project leader**](javascript://), usually is a senior systems analyst or an IT department manager if the project is large. An analyst or a programmer/analyst might manage smaller projects. In addition to the project manager, most large projects have a project coordinator. A [**project coordinator**](javascript://) handles administrative responsibilities for the team and negotiates with users who might have conflicting requirements or want changes that would require additional time or expense.

Project managers typically perform four activities or functions: planning, scheduling, monitoring, and reporting:

* [**Project planning**](javascript://) includes identifying all project tasks and estimating the completion time and cost of each.
* [**Project scheduling**](javascript://) involves the creation of a specific timetable, usually in the form of charts that show tasks, task dependencies, and critical tasks that might delay the project. Scheduling also involves selecting and staffing the project team and assigning specific tasks to team members. Project scheduling uses Gantt charts and PERT/CPM charts, which are explained in the following sections.
* [**Project monitoring**](javascript://) requires guiding, supervising, and coordinating the project team’s workload. The project manager must monitor the progress, evaluate the results, and take corrective action when necessary to control the project and stay on target.
* [**Project reporting**](javascript://) includes regular progress reports to management, users, and the project team itself. Effective reporting requires strong communication skills and a sense of what others want and need to know about the project.

The following sections describe the project planning and scheduling steps: how to create a work breakdown structure, identify task patterns, and calculate the project’s critical path.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

**3.2**Creating a Work Breakdown Structure

A [**work breakdown structure (WBS)**](javascript://) involves breaking a project down into a series of smaller tasks. Before creating WBSs, the two primary chart types should be understood: Gantt charts and PERT/CPM charts.

Change font size

[**help**](javascript://)

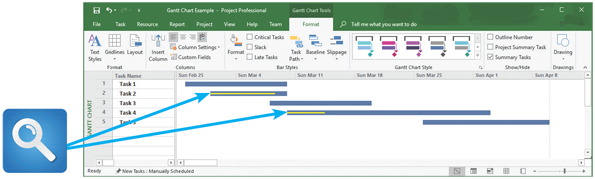
[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.2.1Gantt Charts

Henry L. Gantt, a mechanical engineer and management consultant, developed Gantt charts almost 100 years ago. His goal was to design a chart that could show planned and actual progress on a project. A [**Gantt chart**](javascript://) is a horizontal bar chart that represents a set of tasks. For example, the Gantt chart in [Figure 3-3](javascript://) displays five tasks in a vertical array, with time shown on the horizontal axis. The position of the bar shows the planned starting and ending time of each task, and the length of the bar indicates its duration. On the horizontal axis, time can be shown as elapsed time from a fixed starting point or as actual calendar dates. A Gantt chart also can simplify a complex project by combining several activities into a [**task group**](javascript://) that contains subsidiary tasks. This allows a complex project to be viewed as a set of integrated modules.

**Figure 3-3**

In this Gantt chart, note the yellow bars that show the percentage of task completion.



Enlarge Image

A Gantt chart can show task status by adding a contrasting color to the horizontal bars. For example, a vertical red arrow marks the current date in [Figure 3-3](javascript://). With a fixed reference point, it is easy to see that Task 1 is way behind schedule; Task 2 is only about 80% done and is running behind schedule; Task 3 should have started, but no work has been done; Task 4 actually is running ahead of schedule; and Task 5 will begin in several weeks.

Gantt charts can present an overview of the project’s status, but they do not provide enough detailed information, which is necessary when managing a complex project. Some project managers may find that PERT/CPM charts, which are discussed in the following section, are better tools for managing large projects.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.2.2PERT/CPM Charts

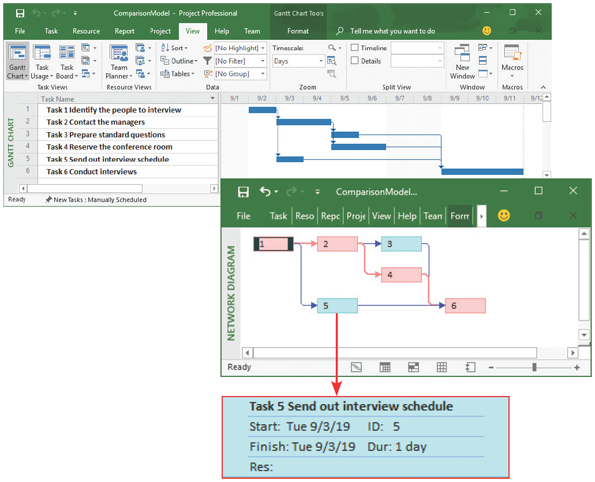
The [**Program Evaluation Review Technique (PERT)**](javascript://) was developed by the U.S. Navy to manage very complex projects, such as the construction of nuclear submarines. At approximately the same time, the [**Critical Path Method (CPM)**](javascript://) was developed by private industry to meet similar project management needs. The distinction between the two methods has disappeared over time, and today the technique is called either PERT, CPM, or [**PERT/CPM**](javascript://). This text will use the term PERT chart.

PERT is a [**bottom-up technique**](javascript://) because it analyzes a large, complex project as a series of individual tasks, just as a pyramid is built from the bottom up using individual blocks. To create a PERT chart, first identify all the project tasks and estimate how much time each task will take to perform. Next, determine the logical order in which the tasks must be performed. For example, some tasks cannot start until other tasks have been completed. In other situations, several tasks can be performed at the same time.

Once the tasks are known, their durations, and the order in which they must be performed, calculate the time that it will take to complete the project. The specific tasks that will be critical to the project’s on-time completion can also be identified. An example of a PERT chart, which Microsoft calls a [**network diagram**](javascript://), is shown in the lower screen in [Figure 3-4](javascript://).

**Figure 3-4**

The top screen shows a Gantt chart with six tasks. The PERT chart in the bottom screen displays an easy-to-follow task pattern for the same project. When the user mouses over the summary box for Task 5, the details become visible.



Enlarge Image

Although a Gantt chart offers a valuable snapshot view of the project, PERT charts are more useful for scheduling, monitoring, and controlling the actual work. With a PERT chart, a project manager can convert task start and finish times to actual dates by laying out the entire project on a calendar. Then, on any given day, the manager can compare what should be happening with what is taking place and react accordingly. Also, a PERT chart displays complex task patterns and relationships. This information is valuable to a manager who is trying to address high priority issues. PERT and Gantt charts are not mutually exclusive techniques, and project managers often use both methods.

[Figure 3-4](javascript://) shows both chart types. The top screen is a Gantt chart with six tasks. The PERT chart below it shows the same project, using a separate box for each task instead of a horizontal bar. Although they both show the task patterns and flow, the PERT chart boxes can provide more information, such as task duration, start date, finish date, and the names of resources assigned to the task. The PERT chart in [Figure 3-4](javascript://) would be too small to view the actual details, which are shown in the expanded text box at the bottom of the figure. How to create PERT charts is explained in a later section.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.2.3Identifying Tasks in a Work Breakdown Structure

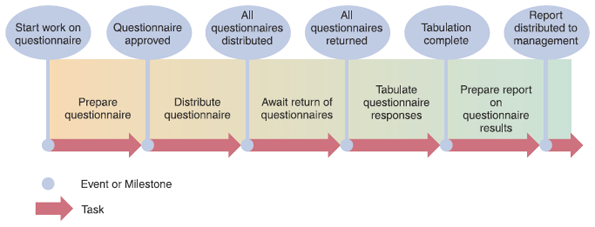
A WBS must clearly identify each task and include an estimated duration. A [**task**](javascript://), or an [**activity**](javascript://), is any work that has a beginning and an end and requires the use of company resources such as people, time, or money. Examples of tasks include conducting interviews, designing a report, selecting software, waiting for the delivery of equipment, or training users. Tasks are basic units of work that the project manager plans, schedules, and monitors—so they should be relatively small and manageable.

In addition to tasks, every project has [**events**](javascript://), or [**milestones**](javascript://). An event, or a milestone, is a recognizable reference point that can be used to monitor progress. For example, an event might be the start of user training, the conversion of system data, or the completion of interviews. A milestone such as “complete 50% of program testing” would not be useful information unless it could be determined exactly when that event will occur.

[Figure 3-5](javascript://) shows tasks and events that might be involved in the creation, distribution, and tabulation of a questionnaire. Note that the beginning and end of each task are marked by a recognizable event. It would be virtually impossible to manage a project as one large task. Instead, the project is broken down into smaller tasks, creating a WBS. The first step in creating a WBS is to list all the tasks.

**Figure 3-5**

Using a questionnaire requires a series of tasks and events to track the progress. The illustration shows the relationship between the tasks and the events, or milestones, which mark the beginning and end of each task.



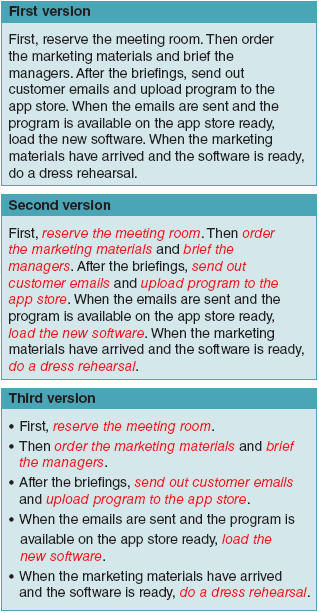
Enlarge Image

### Listing the Tasks

While this step sounds simple, it can be challenging because the tasks might be embedded in a document, such as the one shown in the first version of [Figure 3-6](javascript://). One trick is to start by highlighting the individual tasks, as shown in the second version. Adding bullets makes the tasks stand out more clearly, as shown in the third version. The next step is to number the tasks and create a table, similar to the one shown in [Figure 3-7](javascript://), with columns for task number, description, duration, and [**predecessor tasks**](javascript://), which must be completed before another task can start.

**Figure 3-6**

The three versions show how to transform a task statement into a list of specific tasks for a work breakdown structure.



**Figure 3-7**

In this table, columns have been added for task number, description, duration, and predecessor tasks, which must be completed before another task can start.

| **Task No.** | **Description** | **Duration (Days)** | **Predecessor Tasks** |
| --- | --- | --- | --- |
| 1 | Reserve the meeting room |  |  |
| 2 | Order the marketing materials |  |  |
| 3 | Brief the managers |  |  |
| 4 | Send out customer emails |  |  |
| 5 | Upload program to the app store |  |  |
| 6 | Load the new software |  |  |
| 7 | Do a dress rehearsal |  |  |

Enlarge Table

### Estimating Task Duration

Task duration can be hours, days, or weeks—depending on the project. Because the following example uses days, the units of measurement are called person-days. A [**person-day**](javascript://) represents the work that one person can complete in one day. This approach, however, can present some problems. For example, if it will take one person 20 days to perform a particular task, it might not be true that two people could complete the same task in 10 days or that 10 people could perform the task in two days. Some tasks can be divided evenly so it is possible to use different combinations of time and people—up to a point—but not all. In most systems analysis tasks, time and people are not interchangeable. If one analyst needs two hours to interview a user, two analysts also will need two hours to do the same interview.

Project managers often use a weighted formula for estimating the duration of each task. The project manager first makes three time estimates for each task: an optimistic, or [**best-case estimate**](javascript://) (B), a [**probable-case estimate**](javascript://) (P), and a pessimistic, or [**worst-case estimate**](javascript://) (W). The manager then assigns a [**weight**](javascript://), which is an importance value, to each estimate. The weight can vary, but a common approach is to use a ratio of , , and . The expected task duration is calculated as follows:

For example, a project manager might estimate that a file-conversion task could be completed in as few as 20 days or could take as many as 34 days, but most likely will require 24 days. Using the formula, the expected task duration is 25 days, calculated as follows:

**Case in Point 3.1**

### Sunrise Software

* A lively discussion is under way at Sunrise Software, where you are a project manager. The main question is whether the person-days concept has limitations. In other words, if a task will require 100 person-days, does it matter whether two people in 50 days, five people in 20 days, ten people in 10 days, or some other combination that adds up to 100 performs the work?
* Two programmers on the project seem to think it doesn’t matter. On the other hand, one of the project’s systems analysts says it is ridiculous to think that any combination would work. To support his point, this extreme example was offered: Could 100 people accomplish a task estimated at 100 person-days in one day?
* Is the systems analyst correct? If so, what are the limits in the “people versus days” equation? Taking the concept a step further, is there an optimum number of people to be assigned to a task? If so, how would that number be determined? You need to offer some guidance at the next project team meeting. What will you say?

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.2.4Factors Affecting Duration

When developing duration estimates, project managers consider four factors:

1. Project size
2. Human resources
3. Experience with similar projects
4. Constraints

### Project Size

As described in [Chapter 1](javascript://), information systems have various characteristics that affect their complexity and cost. In addition to considering those factors, a project manager must estimate the time required to complete each project phase. To develop accurate estimates, a project manager must identify all project tasks, from initial fact-finding to system implementation. Regardless of the systems development methodology used, the project manager must determine how much time will be needed to perform each task. In developing an estimate, the project manager must allow time for meetings, project reviews, training, and any other factors (e.g., scheduled vacations or unscheduled medical leave) that could affect the productivity of the development team.

### Human Resources

Companies must invest heavily in cutting-edge technology to remain competitive in a connected world. In many areas, skilled IT professionals are in great demand, and firms must work hard to attract and retain the talent they need. A project manager must assemble and guide a development team that has the skill and experience to handle the project. If necessary, additional systems analysts or programmers must be hired or trained, and this must be accomplished within a specific time frame. After a project gets under way, the project manager must deal with turnover, job vacancies, and escalating salaries in the technology sector—all of which can affect whether the project can be completed on time and within budget. The project manager also has to accommodate official holidays, family emergencies, and other events that may affect the schedule.

### Experience with Similar Projects

A project manager can develop time and cost estimates based on the resources used for similar, previously developed information systems. The experience method works best for small- or medium-sized projects where the two systems are similar in size, basic content, and operating environment. In large systems with more variables, the estimates are less reliable.

### Constraints

[Chapter 2](javascript://) explained that constraints are defined during the preliminary investigation. A constraint is a condition, restriction, or requirement that the system must satisfy. For example, a constraint might involve maximums for one or more resources, such as time, dollars, or people. A project manager must define system requirements that can be achieved realistically within the required constraints. In the absence of constraints, the project manager simply calculates the resources needed. However, if constraints are present, the project manager must adjust other resources or change the scope of the project. This approach is similar to the what-if analysis described in [Chapter 12](javascript://).

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.2.5Displaying the Work Breakdown Structure

After the task durations are entered, the WBS will look like [Figure 3-8](javascript://). Task groups can be used to manage a complex project with many tasks, just as with a Gantt chart, to simplify the list. Note that the WBS shown in [Figure 3-8](javascript://) is still incomplete: It does not show fields such as Start Date, End Date, Task Name, Duration, and Predecessors—fields that can be key for project managers. With Microsoft Project, the WBS (including some of these missing fields) might resemble [Figure 3-9](javascript://).

**Figure 3-8**

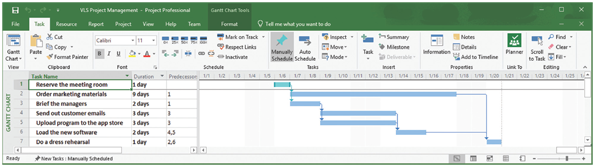
Task durations have been added, and the WBS is complete except for predecessor task information. The predecessor tasks will determine task patterns and sequence of performance.

| **Task No.** | **Description** | **Duration (Days)** | **Predecessor Tasks** |
| --- | --- | --- | --- |
| 1 | Reserve the meeting room | 1 |  |
| 2 | Order the marketing materials | 9 |  |
| 3 | Brief the managers | 2 |  |
| 4 | Send out customer emails | 3 |  |
| 5 | Upload program to the app store | 3 |  |
| 6 | Load the new software | 2 |  |
| 7 | Do a dress rehearsal | 1 |  |

Enlarge Table

**Figure 3-9**

This Microsoft Project screen displays the same WBS, including task number, task name, duration, and predecessor tasks.



Enlarge Image

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

**3.3**Task Patterns

Tasks in a WBS must be arranged in a logical sequence called a [**task pattern**](javascript://). In any project, large or small, tasks depend on each other and must be performed in a sequence, not unlike the commands in a software program. Task patterns can involve dependent tasks, multiple successor tasks, and multiple predecessor tasks. In larger projects, these patterns can be very complex, and an analyst must study the logical flow carefully. This section explains how to understand and create graphical models of these patterns.

Change font size

[**help**](javascript://)

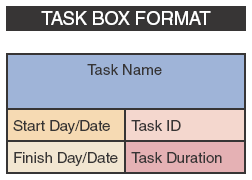
[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.3.1Using Task Boxes to Create a Model

In a PERT/CPM chart, project tasks are shown as rectangular boxes, arranged in the sequence in which they must be performed. Each rectangular box, called a [**task box**](javascript://), has five sections, as shown in [Figure 3-10](javascript://). Each section of the task box contains important information about the task, including the Task Name, Task ID, Task Duration, Start Day/Date, and Finish Day/Date.

**Figure 3-10**

Each section of the task box contains important information about the task, including the Task Name, Task ID, Task Duration, Start Day/Date, and Finish Day/Date.



### Task Name

The [**task name**](javascript://) should be brief and descriptive, but it does not have to be unique in the project. For example, a task named Conduct Interviews might occur in several phases of the project.

### Task ID

The [**task ID**](javascript://) can be a number or code that provides unique identification.

### Task Duration

The [**duration**](javascript://) is the amount of time it will take to complete a task, which is not necessarily the same as the elapsed time. For example, a task that takes eight hours of effort to complete would be done in one day by a person dedicated 100%, but if the person assigned this task is only working 50% on this project, the task would take two days elapsed time to complete. All tasks must use the same time units, which can be hours, days, weeks, or months, depending on the project. An actual project starts on a specific date but can also be measured from a point in time, such as Day 1.

### Start Day/Date

The [**start day/date**](javascript://) is the time that a task is scheduled to begin. For example, suppose that a simple project has two tasks: Task 1 and Task 2. Also suppose that Task 2 cannot begin until Task 1 is finished. An analogy might be that a program cannot run until the computer is turned on. If Task 1 begins on Day 1 and has duration of three days, it will finish on Day 3. Because Task 2 cannot begin until Task 1 is completed, the start time for Task 2 is Day 4, which is the day after Task 1 is finished.

### Finish Day/Date

The [**finish day/date**](javascript://) is the time that a task is scheduled for completion. To calculate the finish day or date, add the duration to the start day or date. When doing this, be very careful not to add too many days. For example, if a task starts on Day 10 and has duration of five days, then the finish date would be on Day 14—not Day 15.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.3.2Task Pattern Types

A project is based on a pattern of tasks. In a large project, the overall pattern would be quite complex, but it can be broken down into three basic types of patterns: dependent tasks, multiple successor tasks, and multiple predecessor tasks.

### Dependent Tasks

When tasks must be completed one after another, like the relay race shown in [Figure 3-11](javascript://), they are called [**dependent tasks**](javascript://) because one depends on the other. For example, [Figure 3-12](javascript://) shows that Task 2 depends on Task 1, because Task 2 cannot start until Task 1 is completed. In this example, the finish time of Task 1, Day 5, controls the start date of Task 2, which is Day 6.

**Figure 3-11**

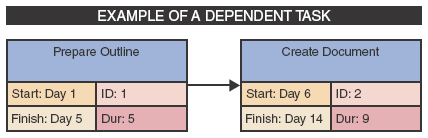
In a relay race, each runner is dependent on the preceding runner and cannot start until the earlier finishes.



**Source:** © William Perugini/ [Shutterstock.com](http://shutterstock.com/" \t "_blank)

**Figure 3-12**

This example of a dependent task shows that the finish time of Task 1, Day 5, controls the start date of Task 2, which is Day 6.

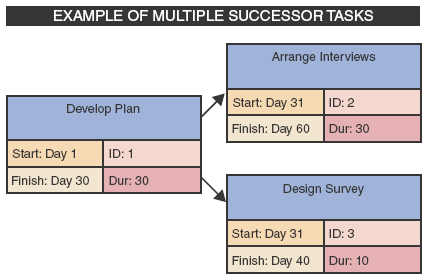


### Multiple Successor Tasks

When several tasks can start at the same time, each is called a [**concurrent task**](javascript://). Often, two or more concurrent tasks depend on a single prior task, which is called a predecessor task. In this situation, each concurrent task is called a [**successor task**](javascript://). In the example shown in [Figure 3-13](javascript://), successor Tasks 2 and 3 both can begin as soon as Task 1 is finished. Note that the finish time for Task 1 determines the start time for both Tasks 2 and 3. In other words, the earliest that Task 1 can finish is Day 30, so Day 31 is the earliest that Tasks 2 and 3 can start.

**Figure 3-13**

This example of multiple successor tasks shows that the finish time for Task 1 determines the start time for both Tasks 2 and 3.

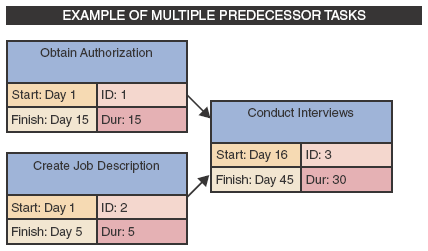


### Multiple Predecessor Tasks

Suppose that a task requires two or more prior tasks to be completed before it can start. [Figure 3-14](javascript://) shows that example because Task 3 cannot begin until Tasks 1 and 2 are both completed. Since the two tasks might not finish at the same time, the longest (latest) predecessor task becomes the controlling factor. Note that the start for Task 3 is Day 16, not Day 6. Why is this so? Because Task 3 depends on two predecessor tasks, Tasks 1 and 2, Task 3 cannot begin until the later of those tasks is complete. Therefore, the start time for a successor task must be the latest (largest) finish time for any of its preceding tasks. In the example shown, Task 1 ends on Day 15, while Task 2 ends on Day 5, so Task 1 controls the start time for Task 3.

**Figure 3-14**

This example of multiple predecessor tasks shows that the start time for a successor task must be the latest (largest) finish time for any of its preceding tasks. In the example shown, Task 1 ends on Day 15, while Task 2 ends on Day 5, so Task 1 controls the start time for Task 3.



Task pattern types are identified by looking carefully at the wording of the task statement. Words like then, when, or and are action words that signal a sequence of events. Here are three simple examples:

* Do Task 1, then do Task 2 describes dependent tasks that must be completed one after the other.
* When Task 2 is finished, start two tasks: Task 3 and Task 4 describe multiple successor tasks that can both start as soon as Task 2 is finished.
* When Tasks 5 and 6 are done, start Task 7 indicates that Task 7 is a multiple predecessor task because it can’t start until two or more previous tasks all are completed.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.3.3Working with Complex Task Patterns

When several task patterns combine, the facts must be studied very carefully to understand the logic and sequence. A project schedule will not be accurate if the underlying task pattern is incorrect. For example, consider the following three fact statements and the task patterns they represent. Examples of the task patterns are shown in [Figures 3-15](javascript://), [3-16](javascript://), and [3-17](javascript://).

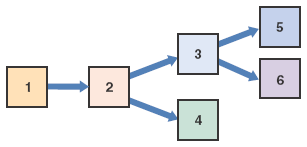
**Figure 3-15**

Dependent tasks.

Task 1 is connected by an arrow pointing to Task 2. This means Task 2 is dependent on Task 1.

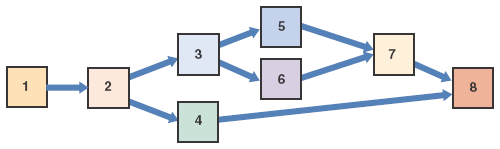
**Figure 3-16**

Dependent tasks and multiple successor tasks.



**Figure 3-17**

Dependent tasks, multiple successor tasks, and multiple predecessor tasks.



### Dependent Tasks

Perform Task 1. When Task 1 is complete, perform Task 2.

### Dependent Tasks and Multiple Successor Tasks

Perform Task 1. When Task 1 is complete, perform Task 2. When Task 2 is finished, start two tasks: Task 3 and Task 4. When Task 3 is complete, start two more tasks: Task 5 and Task 6.

### Dependent Tasks, Multiple Successor Tasks, and Multiple Predecessor Tasks

Perform Task 1. When Task 1 is complete, perform Task 2. When Task 2 is finished, start two Tasks: Task 3 and Task 4. When Task 3 is complete, start two more tasks: Task 5 and Task 6. When Tasks 5 and 6 are done, start Task 7. Then, when Tasks 4 and 7 are finished, perform Task 8.

**Case in Point 3.2**

### Parallel Services

* The project management team at Parallel Services is having a debate about how to define tasks in the WBS. The project manager wants to break tasks down into the smallest possible units. For example, she objected to a broad task statement called “Develop a training schedule.” Instead, she suggested three subtasks:
  + (1)

“Determine availability of training room,”

* + (2)

“Determine availability of attendees,” and

* + (3)

“Select specific dates and training times.”

* Another project team member disagrees. He feels that the broader task statement is better because it allows more flexibility and will produce the same result. He says that if you break tasks into pieces that are too small, you risk overmanaging the work and spending more time on monitoring than actually performing the tasks. As a member of the team, which approach do you agree with more? What are the pros and cons of each?

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

**3.4**The Critical Path

Task patterns determine the order in which the tasks are performed. Once the task sequence has been defined, a project manager can schedule the tasks and calculate the critical path. A [**critical path**](javascript://) is a series of tasks that, if delayed, would affect the completion date of the overall project. If any task on the critical path falls behind schedule, the entire project will be delayed.

For example, suppose that Joan and Jim are invited to someone’s home for dinner. Joan arrives on time, but Jim arrives 30 minutes late. Jim’s arrival is part of the critical path because the host does not want to start without him, so the meal will be served 30 minutes later than originally planned.

Project managers always must be aware of the critical path, so they can respond quickly to keep the project on track. Microsoft Project and other project management software can highlight the series of tasks that form the critical path.

Change font size

[**help**](javascript://)

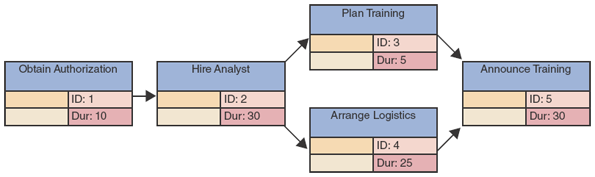
[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.4.1Calculating the Critical Path

[Figure 3-18](javascript://) shows a training project with five tasks. Note that the analyst has arranged the tasks and entered task names, IDs, and durations. The task patterns should be reviewed first. In this example, Task 1 is followed by Task 2, which is a dependent task. Task 2 has two successor tasks: Task 3 and Task 4. Tasks 3 and 4 are predecessor tasks for Task 5.

**Figure 3-18**

Example of a PERT/CPM chart with five tasks. Task 2 is a dependent task that has multiple successor tasks. Task 5 has multiple predecessor tasks. In this figure, the analyst has arranged the tasks and entered task names, IDs, and durations.



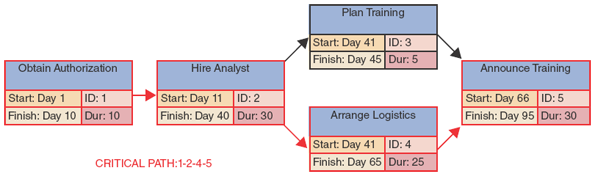
Enlarge Image

The next step is to determine start and finish dates, which will determine the critical path for the project. The following explanation outlines a step-by-step process. The result is shown in [Figure 3-19](javascript://).

* Task 1 starts on Day 1 and has duration of 10 days, so the finish date is Day 10.
* Task 2, which is dependent on Task 1, can start on Day 11—the day after Task 1 ends. With duration of 30 days, Task 2 will end on Day 40.
* Tasks 3 and 4 are multiple successor tasks that can start after Task 2 is done. Task 2 ends on Day 40, so Tasks 3 and 4 both can start on Day 41. Task 3 has duration of five days and will end on Day 45. Task 4 has duration of 25 days and will not end until Day 65.
* Task 5 depends on Tasks 3 and 4, which are multiple predecessors. Because Task 5 depends on both tasks, it cannot start until the later of the two tasks is complete. In this example, Task 3 ends earlier, but Task 4 will not be completed until Day 65, so Task 5 cannot start until Day 66.

**Figure 3-19**

Now the analyst has entered the start and finish times, using the rules explained in this section. Note that the overall project has duration of 95 days.



Enlarge Image

Recall that the critical path is a series of tasks that, if delayed, would affect the final completion date of the overall project. In this example, Tasks 1 and 2 are the first tasks on the critical path. Now look at Task 5, which cannot start until both Tasks 3 and 4 are done. In this case, Task 4 is the controlling factor because Task 4 finishes on Day 65, which is 20 days later than Task 3, which is completed on Day 45. Therefore, the start date for Task 5 is determined by the finish date for Task 4.

In contrast, Task 3 has slack time and could be delayed up to 20 days without affecting Task 5. [**Slack time**](javascript://) is the amount of time that the task could be late without pushing back the completion date of the entire project. Tasks 1, 2, 4, and 5 represent the critical path, which is highlighted with red arrows in [Figure 3-19](javascript://).

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

**3.5**Project Monitoring and Control

Regardless of whether the project was planned and scheduled with project management software or in some other manner, the project manager must keep track of the tasks and progress of team members, compare actual progress with the project plan, verify the completion of project milestones, and set standards and ensure that they are followed.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.5.1Monitoring and Control Techniques

To help ensure that quality standards are met, many project managers institute structured walk-throughs. A [**structured walk-through**](javascript://) is a review of a project team member’s work by other members of the team. Generally, systems analysts review the work of other systems analysts, and programmers review the work of other programmers, as a form of peer review. Structured walk-throughs take place throughout the SDLC and are called [**design reviews**](javascript://), [**code reviews**](javascript://), or [**testing reviews**](javascript://), depending on the phase in which they occur.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.5.2Maintaining a Schedule

Maintaining a project schedule can be challenging, and most projects run into at least some problems or delays. By monitoring and controlling the work, the project manager tries to anticipate problems, avoid them or minimize their impact, identify potential solutions, and select the best way to solve the problem.

The better the original plan, the easier it will be to control the project. If clear, verifiable milestones exist, it will be simple to determine if and when those targets are achieved. If enough milestones and frequent checkpoints exist, problems will be detected rapidly. A project that is planned and scheduled with PERT/CPM or in a WBS with Gantt chart can be tracked and controlled using these same techniques. As work continues, the project manager revises the plan to record actual times for completed tasks and revises times for tasks that are not yet finished.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.5.3Tasks and the Critical Path

Project managers spend most of their time tracking the tasks along the critical path because delays in those tasks have the greatest potential to delay or jeopardize the project. Other tasks cannot be ignored, however. For example, suppose that a task not on the critical path takes too long and depletes the allotted slack time. At that point, the task actually becomes part of the critical path, and any further delay will push back the overall project.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

**3.6**Reporting

Members of the project team regularly report their progress to the project manager, who in turn reports to management and users. The project manager collects, verifies, organizes, and evaluates the information he or she receives from the team. Then the manager decides which information needs to be passed along, prepares a summary that can be understood easily, adds comments and explanations if needed, and submits it to management and users.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.6.1Project Status Meetings

Project managers, like the one shown in [Figure 3-20](javascript://), schedule regular meetings to update the team and discuss project status, issues, problems, and opportunities. Although meetings can be time consuming, most project managers believe they are worth the effort. The sessions give team members an opportunity to share information, discuss common problems, and explain new techniques. The meetings also give the project manager an opportunity to seek input and conduct brainstorming sessions.

**Figure 3-20**

Project managers schedule regular meetings to update the project team and discuss project status, issues, problems, and opportunities.



Enlarge Image

Hero Images/Getty Images

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.6.2Project Status Reports

A project manager must report regularly to his or her immediate supervisor, upper management, and users. Although a progress report might be given verbally to an immediate supervisor, reports to management and users usually are written. Gantt charts often are included in progress reports to show project status graphically.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.6.3Dealing with Problems

Deciding how to handle potential problems can be difficult. At what point should management be informed about the possibility of cost overruns, schedule delays, or technical problems? At one extreme is the overly cautious project manager who alerts management to every potential snag and slight delay. The danger here is that the manager loses credibility over a period of time, and management might ignore potentially serious situations. At the other extreme is the project manager who tries to handle all situations single-handedly and does not alert management until a problem is serious. By the time management learns of the problem, little time might remain in which to react or devise a solution.

A project manager’s best course of action lies somewhere between the two extremes but is probably closer to the first. If the consequences are unclear, the analyst should err on the side of caution and warn management about the possibility of a problem.

When the situation is reported, explain what is being done to handle and monitor the problem. If the situation is beyond the analyst’s control, suggest possible actions that management can take to resolve the situation. Most managers recognize that problems do occur on most projects; it is better to alert management sooner rather than later.

Change font size

[**help**](javascript://)

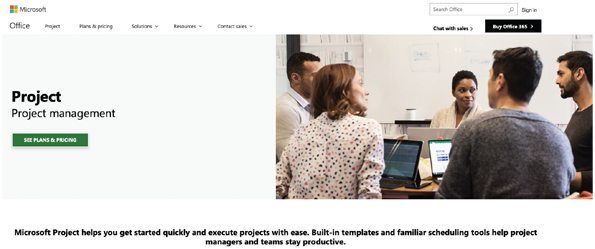
[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

# 3.7Project Management Software

Project managers use software applications to help plan, schedule, monitor, and report on a project. Most programs offer features such as PERT/CPM, Gantt charts, resource scheduling, project calendars, and cost tracking. As shown in [Figure 3-21](javascript://), Microsoft Project is a full-featured program that holds the dominant share of the market. It is available as a software product for Windows and as an add-on online service as part of Microsoft’s Office 365.

**Figure 3-21**

Microsoft Project.



Enlarge Image

**Source:** Microsoft Corporation

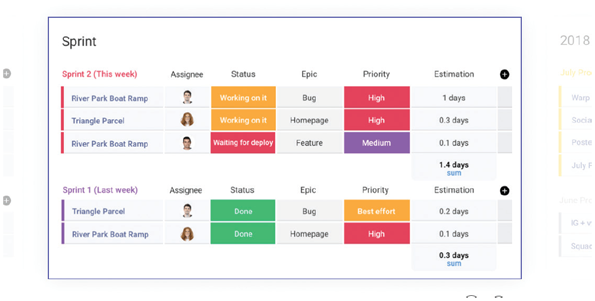
In addition to Microsoft Project, there are a number of other project management tools available. For example, GanttProject is a free [**open-source**](javascript://) Java-based project management tool that is available on multiple platforms (Windows, Mac OS X, and Linux). It can produce Gantt and PERT/CPM charts, calculate the critical path automatically, and read/write Microsoft Project files.

Gantter is a free cloud-based project management tool. It runs in a browser window, so there’s no software to install to use it. Apptivo and smartsheet are other examples of web-based project management tools offering similar capabilities but on a paid subscription model.

Monday is a project management tool that is tailored toward Mac users. As shown in [Figure 3-22](javascript://), it is a highly visual cloud-based tool that supports agile development. Trello, shown in [Figure 3-23](javascript://), is another project management tool that is tailored toward agile development.

**Figure 3-22**

Monday is a project management tool that is tailored toward Mac users.

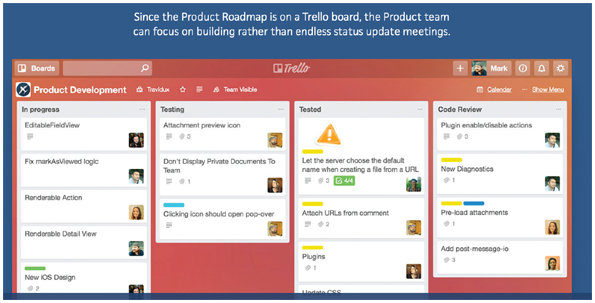


Enlarge Image

**Source:** [monday.com](https://monday.com/" \t "_blank)

**Figure 3-23**

Trello is a project management tool that is tailored toward agile development.



Enlarge Image

**Source:** Atlassian

The websites for all of these tools have more information about their capabilities, including demos, trial versions (where applicable), and training material.

Irrespective of which project management tool used, a step-by-step process is followed to develop a WBS, work with task patterns, and analyze the critical path. The main difference is that the software does most of the work automatically, which enables much more effective management.

The following sections explain how Microsoft Project could be used to handle the sample task summary for a preliminary investigation shown in [Figure 3-24](javascript://). This example illustrates that project management is dynamic and challenging. One significant advantage of integrated project management software is that it allows the project manager to adjust schedules, estimates, and resource assignments rapidly in response to real-world events.

**Figure 3-24**

A sample task summary for a preliminary investigation.

**Please study the following task summary:**

* First, we need to spend one day studying potential problems or opportunities.
* Then, we will define the project scope and constraints. That will take two days.
* Next, we will analyze the organization charts. That will take one day.
* After we analyze the charts, four fact-finding tasks can start at once:
  + Observe operations (two days)
  + Conduct a user survey (three days)
  + Conduct interviews (two days)
  + Review documentation (one day)
* When all four fact-finding tasks are finished, we will spend one day evaluating feasibilty.
* Then we will spend one day presenting the results and recommendations to management.

### Work Breakdown Structure

Creating a WBS using Microsoft Project is much the same as creating it manually. The tasks, durations, and task patterns must still be identified. This information might have to be developed, or a task summary like the one in [Figure 3-24](javascript://) might be used. The goal is to document all tasks, dependencies, dates, and total project duration. The first step is to create a Gantt chart showing the necessary information. As the information for each task is entered into Microsoft Project, the duration and the predecessor tasks, if any, should also be noted.

### Gantt Chart

As tasks are entered, the program automatically performs the calculations, detects the task patterns, and creates a Gantt chart. The chart consists of horizontal bars, connected with arrows that indicate task dependencies. If a typical workweek is selected, tasks will not be scheduled on Saturdays and Sundays. However, for a mission-critical project, a 24/7 calendar might be created. Whatever is specified, the program will handle the tasks accordingly. Microsoft Project offers numerous choices of display settings, formats, and calculation methods.

### Network Diagram

After the Gantt chart is completed, the data can be viewed in the form of a Microsoft Project network diagram, which is similar to a PERT chart. When the Network Diagram option is selected, the project tasks, dependencies, and start and finish dates for each task are shown. A network diagram displays the same information as the Gantt chart, including task dependencies, but use task boxes to include much more detail. Using Microsoft Project, each task can be assigned to one or more people, budgets can be assigned targets, progress reports produced, and schedules and deadlines readjusted as necessary.

### Calendar View

Calendar view is a good way to manage day-to-day activity. This view shows the tasks, similar to a PERT chart, as an overlay on the actual calendar. Because the critical path is highlighted in red, it is easy for a project manager to determine priorities at any point in time.

Suppose the project manager wants to view the preliminary investigation in [Figure 3-24](javascript://) as a Gantt chart, a PERT chart, and a day-to-day calendar. All three views are shown in [Figure 3-25](javascript://). Each view shows the tasks, the timing, the dependencies, and the critical path. Note that of the four tasks scheduled for September 25, only the user survey is on the critical path, therefore that should be the project manager’s primary concern.

**Figure 3-25**

Note how each view displays the project and highlights the critical path. If you were the project manager on September 25, your primary concern should be conducting the user survey.



Enlarge Image

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

**3.8**Risk Management

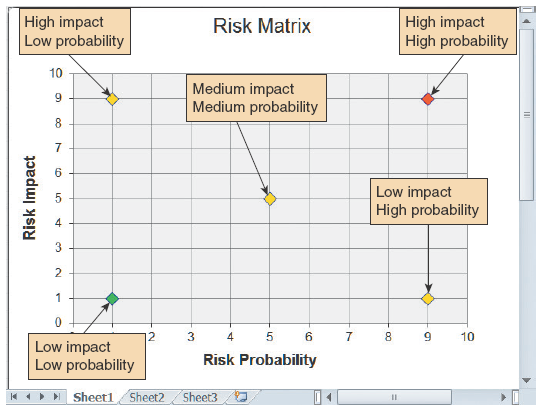
Every IT project involves risks that systems analysts and project managers must address. A [**risk**](javascript://) is an event that could affect the project negatively. [**Risk management**](javascript://) is the process of identifying, analyzing, anticipating, and monitoring risks to minimize their impact on the project.

Although project management experts differ with regard to the number of steps or phases, a basic list of risk management tasks would include the following:

* *Develop a risk management plan*. A [**risk management plan**](javascript://) includes a review of the project’s scope, stakeholders, budget, schedule, and any other internal or external factors that might affect the project. The plan should define project roles and responsibilities, risk management methods and procedures, categories of risks, and contingency plans.
* *Identify the risks*. [**Risk identification**](javascript://) lists each risk and assesses the likelihood that it could affect the project. The details would depend on the specific project, but most lists would include a means of identification, and a brief description of the risk, what might cause it to occur, who would be responsible for responding, and the potential impact of the risk.
* *Analyze the risks*. This typically is a two-step process: Qualitative risk analysis and quantitative risk analysis. [**Qualitative risk analysis**](javascript://) evaluates each risk by estimating the probability that it will occur and the degree of impact. Project managers can use a formula to weigh risk and impact values, or they can display the results in a two-axis grid. For example, a Microsoft Excel XY chart can be used to display the matrix, as shown in [Figure 3-26](javascript://). In the chart, note the various combinations of risk and impact ratings for the five sample values. This tool can help a project manager focus on the most critical areas, where risk probability and potential impact are high.

**Figure 3-26**

You can use a Microsoft Excel XY chart type to display a risk matrix that shows risk probability and potential impact.



The purpose of [**quantitative risk analysis**](javascript://) is to understand the actual impact in terms of dollars, time, project scope, or quality. Quantitative risk analysis can involve a modeling process called what-if analysis, which allows a project manager to vary one or more element(s) in a model to measure the effect on other elements. This topic is discussed in more detail in [Chapter 12](javascript://).

* *Create a risk response plan*. A [**risk response plan**](javascript://) is a proactive effort to anticipate a risk and describe an action plan to deal with it. An effective risk response plan can reduce the overall impact by triggering timely and appropriate action.
* *Monitor risks*. This activity is ongoing throughout the risk management process. It is important to conduct a continuous tracking process that can identify new risks, note changes in existing risks, and update any other areas of the risk management plan.

Fortunately, there is a wide variety of risk management software available to help a project manager with these tasks. Most packages allow a project manager to assign specific dates as constraints, align task dependencies, note external factors that might affect a task, track progress, and display tasks that are behind schedule. Armed with this information, the IT team can quantify the project’s risks, just as they use financial analysis tools to quantify costs and benefits.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

# 3.9Managing for Success

Project management is a challenging task. Project managers must be alert, technically competent, and highly resourceful. They also must be good communicators with strong human resource skills. Project managers can be proud when they handle a successful project that helps the company achieve its business objectives.

Unfortunately, projects can and do get derailed for a wide variety of reasons. When problems occur, the project manager’s ability to handle the situation becomes the critical factor. When a project manager first recognizes that a project is in trouble, what options are available? Alternatives can include trimming the project requirements, adding to the project resources, delaying the project deadline, and improving management controls and procedures. Sometimes, when a project experiences delays or cost overruns, the system still can be delivered on time and within budget if several less critical requirements are trimmed. The system can be delivered to satisfy the most necessary requirements, and additional features can be added later as a part of a maintenance or enhancement project.

If a project is in trouble because of a lack of resources or organizational support, management might be willing to give the project more commitment and higher priority. For example, management might agree to add more people to a project that is behind schedule. Adding staff, however, will reduce the project’s completion time only if the additional people can be integrated effectively into the development team. If team members lack experience with certain aspects of the required technology, temporary help might be obtained from IT consultants or part-time staff. Adding staff can mean training and orienting the new people, however. In some situations, adding more people to a project actually might increase the time necessary to complete the project because of a principle called [**Brooks’ law**](javascript://). Frederick Brooks, Jr., then an IBM engineer, observed that adding manpower to a late software project only makes it later. Brooks reached this conclusion when he saw that new workers on a project first had to be educated and instructed by existing employees whose own productivity was reduced accordingly.

To be successful, an information system must satisfy business requirements, stay within budget, be completed on time, and—most important of all—be managed effectively. As stated earlier and detailed next, when a project develops problems, the reasons typically involve business, budget, or schedule issues, as explained in the following sections. In addition to planning and managing the project, a project manager must be able to recognize these problems and deal with them effectively.

**Case in Point 3.3**

### Just-in-Time Software

* You are a systems analyst at Just-in-Time Software, a company that specializes in short delivery cycles for its products. The current project is running behind schedule, and the project manager wants to bring a few extra programmers onboard to help with the work.

You are familiar with Brook’s Law. How can you best explain to the project manager that adding more people to the project at this late stage may make things worse? You don’t want to be seen as a negative team player, but you’re convinced that if you don’t speak up, the project’s schedule will slip even more.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.9.1Business Issues

The major objective of every system is to provide a solution to a business problem or opportunity. If the system does not do this, then it is a failure—regardless of positive reaction from users, acceptable budget performance, or timely delivery. When the information system does not meet business requirements, causes can include unidentified or unclear requirements, inadequately defined scope, imprecise targets, shortcuts or sloppy work during systems analysis, poor design choices, insufficient testing or inadequate testing procedures, and lack of change control procedures. Systems also fail because of changes in the organization’s culture, funding, or objectives. A system that falls short of business needs also produces problems for users and reduces employee morale and productivity.

As explained in [Chapter 2](javascript://), projects without clear scope definitions are risky because they tend to expand gradually, without specific authorization, in a process called project creep. However, even when a project is clearly described, it must be managed constantly.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.9.2Budget Issues

Cost overruns typically result from one or more of the following:

* Unrealistic estimates that are too optimistic or based on incomplete information
* Failure to develop an accurate forecast that considers all costs over the life of the project
* Poor monitoring of progress and slow response to early warning signs of problems
* Schedule delays due to factors that were not foreseen
* Human resource issues, including turnover, inadequate training, and motivation

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

## 3.9.3Schedule Issues

Problems with timetables and project milestones can indicate a failure to recognize task dependencies, confusion between effort and progress, poor monitoring and control methods, personality conflicts among team members, or turnover of project personnel. The failure of an IT project also can be caused by poor project management techniques.

If the project manager fails to plan, staff, organize, supervise, communicate, motivate, evaluate, direct, and control properly, then the project is certain to fail. Even when factors outside his or her control contribute to the failure, the project manager is responsible for recognizing the early warning signs and handling them effectively.

### A Question of Ethics

* [iStock.com](https://istock.com/" \t "_blank)/faberfoto\_it“Better blow the whistle,” says your friend and project teammate at work. “The project is out of control, and you know it!” “Maybe so,” you respond, “But that’s not my call—I’m not the project manager.” What you don’t say is that the project manager feels like her career is on the line, and she is reluctant to bring bad news to management at this time. She honestly believes that the project can catch up and says that a bad report on a major project could result in bad publicity for the firm and frighten potential customers.

To be fair, the next management progress report is scheduled in three weeks. It is possible that the team could catch up, but you doubt it. You wonder if there is an ethical question here: Even though the report isn’t due yet, should a significant problem be reported to management as soon as possible? You are concerned about the issue, and you decide to discuss it with Stephanie. What will you say to her?

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

**3.10**Summary

Project management is the process of planning, scheduling, monitoring, and reporting on the development of an information system. Planning includes identifying all project tasks and estimating the completion time and cost of each. Project scheduling involves the creation of a specific timetable, usually in the form of charts that show tasks, task dependencies, and critical tasks that might delay the project. Project monitoring requires guiding, supervising, and coordinating the project team’s workload. The project manager must monitor the progress, evaluate the results, and take corrective action when necessary to control the project and stay on target. Project reporting includes regular progress reports to management, users, and the project team itself. Effective reporting requires strong communication skills and a sense of what others want and need to know about the project. A successful project must be completed on time, be within budget, and deliver a quality product that satisfies users and meets requirements.

A project triangle shows three legs: project cost, scope, and time. A project manager must find the best balance among these elements because a change in any leg of the triangle will affect the other two legs. Project management techniques can be used throughout the SDLC.

Planning, scheduling, monitoring, and reporting—all take place within a larger project development framework, which includes three key steps: creating a WBS, identifying task patterns, and calculating the critical path. A WBS must clearly identify each task and include an estimated duration. A task, or activity, is any work that has a beginning and an end and requires the use of company resources such as people, time, or money. Time and cost estimates for tasks usually are made in person-days. A person-day represents the work that one person can accomplish in one day. Estimating the time for project activities is more difficult with larger systems. Project managers must consider the project size and scope, IT resources, prior experience with similar projects or systems, and applicable constraints. In addition to tasks, every project has events, or milestones. An event, or a milestone, is a recognizable reference point that can be used to monitor progress.

Task patterns establish the sequence of work in a project. Task patterns involve dependent tasks, multiple successor tasks, and multiple predecessor tasks. In larger projects, these patterns can be very complex.

A critical path is a series of tasks that, if delayed, would affect the completion date of the overall project. If any task on the critical path falls behind schedule, the entire project will be delayed. Tasks on the critical path cannot have slack time. To identify the critical path, calculate the start and finish date for each task, which will determine the critical path for the project.

In project scheduling, the project manager develops a specific time for each task, based on available resources and whether or not the task is dependent on other predecessor tasks. The manager can use graphical tools such as Gantt charts and PERT charts to assist in the scheduling process.

A Gantt chart is a horizontal bar chart that represents the project schedule with time on the horizontal axis and tasks arranged vertically. It shows individual tasks and task groups, which include several tasks. In a Gantt chart, the length of the bar indicates the duration of the tasks. A Gantt chart can display progress but does not show task dependency details or resource assignment unless the chart was created with a project management program that supports dependency linking and the entry of other information.

A PERT/CPM chart shows the project as a network diagram with tasks connected by arrows. Using a prescribed calculation method, the project manager uses a PERT chart to determine the overall duration of the project and provide specific information for each task, including the task IDs, their durations, start and finish times, and the order in which they must be performed. With this information, the manager can determine the critical path, which is the sequence of tasks that has no slack time and must be performed on schedule in order to meet the overall project deadline.

Most project managers use software applications such as Microsoft Project to plan, schedule, and monitor projects. Project managers are responsible for risk management, which is the process of identifying, analyzing, anticipating, and monitoring risks to minimize their impact on the project.

In the end, project management involves the same skills as any other management. The project manager must be perceptive, analytical, well organized, and a good communicator. If the project manager senses that the project is off-track, he or she must take immediate steps to diagnose and solve the problem. If the project manager fails to plan, staff, organize, supervise, communicate, motivate, evaluate, direct, and control properly, then the project is certain to fail. Even when factors outside his or her control contribute to the failure, the project manager is responsible for recognizing the early warning signs and handling them effectively.

Change font size

[**help**](javascript://)

[Main content](https://ng.cengage.com/static/nbreader/ui/apps/nbreader/nbreader.html#header)

# Chapter Review

## **Key Terms**

* [**activity**](javascript://)
* [**best-case estimate**](javascript://)
* [**bottom-up technique**](javascript://)
* [**Brooks’ law**](javascript://)
* [**code reviews**](javascript://)
* [**concurrent task**](javascript://)
* [**critical path**](javascript://)
* [**Critical Path Method (CPM)**](javascript://)
* [**dependent tasks**](javascript://)
* [**design reviews**](javascript://)
* [**duration**](javascript://)
* [**events**](javascript://)
* [**finish day/date**](javascript://)
* [**Gantt chart**](javascript://)
* [**milestones**](javascript://)
* [**network diagram**](javascript://)
* [**open-source**](javascript://)
* [**person-day**](javascript://)
* [**PERT/CPM**](javascript://)
* [**predecessor tasks**](javascript://)
* [**probable-case estimate**](javascript://)
* [**Program Evaluation Review Technique (PERT)**](javascript://)
* [**project coordinator**](javascript://)
* [**project leader**](javascript://)
* [**project management**](javascript://)
* [**project manager**](javascript://)
* [**project monitoring**](javascript://)
* [**project planning**](javascript://)
* [**project reporting**](javascript://)
* [**project scheduling**](javascript://)
* [**project triangle**](javascript://)
* [**qualitative risk analysis**](javascript://)
* [**quantitative risk analysis**](javascript://)
* [**risk**](javascript://)
* [**Risk identification**](javascript://)
* [**Risk management**](javascript://)
* [**risk management plan**](javascript://)
* [**risk response plan**](javascript://)
* [**slack time**](javascript://)
* [**start day/date**](javascript://)
* [**structured walk-through**](javascript://)
* [**successor task**](javascript://)
* [**task**](javascript://)
* [**task box**](javascript://)
* [**task group**](javascript://)
* [**task ID**](javascript://)
* [**task name**](javascript://)
* [**task pattern**](javascript://)
* [**testing reviews**](javascript://)
* [**weight**](javascript://)
* [**work breakdown structure (WBS)**](javascript://)
* [**worst-case estimate**](javascript://)

Change font size

[**help**](javascript://)