

This chapter introduces the concepts underlying QuickTime, a set of functions and data structures that you can use in your application to control time-based data. In QuickTime, a set of time-based data is referred to as a *movie*. Your application can allow users to display, edit, cut, copy, and paste movies and movie data in the same way that they can work with text and graphic elements today.

This chapter also introduces the QuickTime architecture, the managers, and the components that constitute QuickTime. It will help you decide what level of QuickTime support your application may need to incorporate.

QuickTime Concepts

To use QuickTime, you need to understand some concepts that are new to most developers of Macintosh applications: movies, media data structures, components, image compression, and time.

Movies and Media Data Structures

A traditional movie, whether stored on film, laser disk, or tape, is a continuous stream of data. A QuickTime movie can be similarly constructed, but it need not be: a QuickTime movie can consist of data in sequences from different forms, such as analog video and CD-ROM. The movie is not the medium; it is the organizing principle.

A QuickTime movie may contain several **tracks**. Each track refers to a media that contains references to the movie data, which may be stored as images or sound on hard disks, floppy disks, compact discs, or other devices. The data references constitute the track's **media**. Each track has a single media data structure.

Note

Throughout this book, the term *media* is used to refer to a Movie Toolbox data structure that contains information that describes the data for a track in a movie. Note that a media does not contain its data; rather, a media contains a reference to its data. If more than one media is being discussed, the term *media structures* is used. ♦

Your application need never work directly with the movie data, as Movie Toolbox functions allow you to manage movie content and characteristics. See the chapter “Movie Toolbox” later in this book for a comprehensive reference to the Movie Toolbox.

Components

QuickTime provides components so that every application doesn't need to know about all possible types of audio, visual, and storage devices. A **component** is a code resource that is registered by the Component Manager. The component's code can be available as a systemwide resource or in a resource that is local to a particular application. Each QuickTime component supports a defined set of features and presents a specified

functional interface to its client applications. Applications are thereby isolated from the details of implementing and managing a given technology. For example, you could create a component that supports a certain data encryption algorithm. Applications could then use your algorithm by connecting to your component through the Component Manager, rather than by implementing the algorithm over again. For comprehensive reference to the QuickTime components supplied by Apple, see the book *Inside Macintosh: QuickTime Components*.

Image Compression

Image data requires a large amount of storage space. Storing a single 640-by-480 pixel image in 32-bit color can require as much as 1.2 MB. Similarly, sequences of images, like those that might be contained in a QuickTime movie, demand substantially more storage than single images. This is true even for sequences that consist of fairly small images, because the movie consists of a large number of those images. Consequently, minimizing the storage requirements for image data is an important consideration for any application that works with images or sequences of images.

The Image Compression Manager provides your application with an interface for compressing and decompressing images and sequences of images that is independent of devices and algorithms. See the chapter “Image Compression Manager” later in this book for details.

Time

Image compression is difficult but worthwhile—images, not to mention long sequences of images, take a lot of memory. Time management in QuickTime is equally essential. You must understand time management to understand the QuickTime functions and data structures.

Seemingly simple issues prove interesting—for example, determining the proper length (duration) of a movie. For many movies, the proper duration is the time required to play them in “real” time—that is, a rate in which human actions appear natural, and objects fall to earth accelerating at 32 feet per second per second. But what is the length of a movie that shows spreadsheet data charted over time, or a map of the earth that recapitulates continental drift? Add to this the differing clock speeds of different platforms, and the need to decompress in real time, and time proves, as ever, complex.

To manage these situations, QuickTime defines **time coordinate systems**, which anchor movies and their media data structures to a common temporal reality, the second. A time coordinate system contains a **time scale** that provides the translation between real time and the time in a movie. Time scales are marked in **time units**. The number of units that pass per second quantifies the scale—that is, a time scale of 26 means that 26 units pass per second and each time unit is 1/26 of a second. A time coordinate system also contains a **duration**, which is the length of a movie or a media in the number of time units it contains. Particular points in a movie can be identified by a time value, the number of time units elapsed to that point.

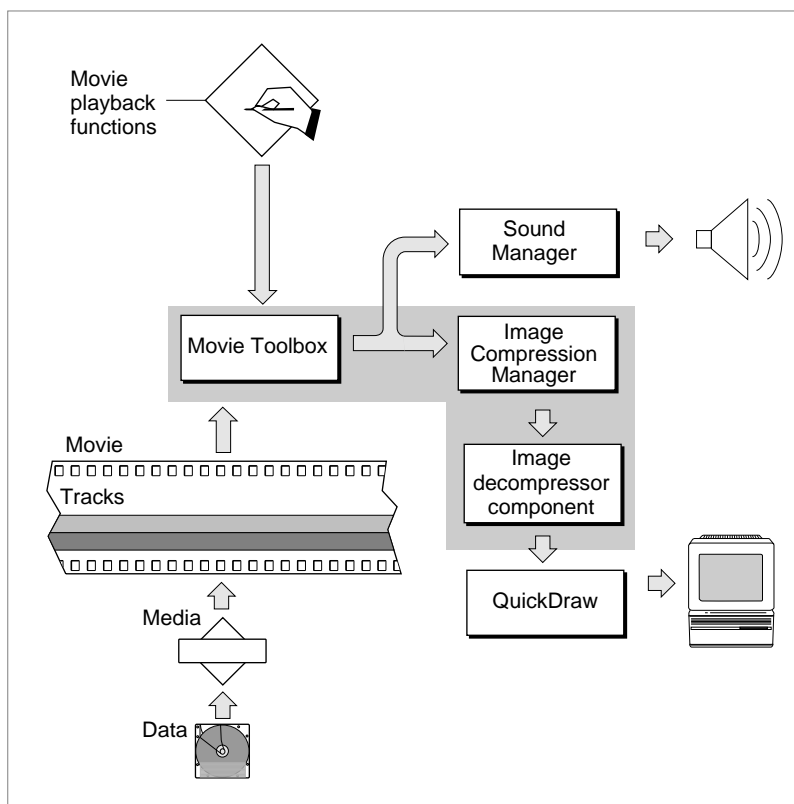
Each media has its own time coordinate system, which starts at time 0. The Movie Toolbox maps each type of media data from the movie's time coordinate system to the media's time coordinate system.

Time bases and time coordinate systems are described in the chapter "Movie Toolbox" later in this book.

The QuickTime Architecture

QuickTime comprises two managers: the Movie Toolbox and the Image Compression Manager. QuickTime also relies on the Component Manager, as well as a set of predefined components. Figure 1-1 shows the relationships of these managers and an application that is playing a movie.

Figure 1-1 QuickTime playing a movie



The following sections discuss these managers in more detail.

The Movie Toolbox

Your application gains access to the capabilities of QuickTime by calling functions in the Movie Toolbox. The Movie Toolbox allows you to store, retrieve, and manipulate time-based data that is stored in QuickTime movies. A single movie may contain several types of data. For example, a movie that contains video information might include both video data and the sound data that accompanies the video.

The Movie Toolbox also provides functions for editing movies. For example, there are editing functions for shortening a movie by removing portions of the video and sound tracks, and there are functions for extending it with the addition of new data from other QuickTime movies.

The Movie Toolbox is described in the chapter “Movie Toolbox” later in this book. That chapter includes code samples that show how to play movies.

The Image Compression Manager

The Image Compression Manager comprises a set of functions that compress and decompress images or sequences of graphic images.

The Image Compression Manager provides a device-independent and driver-independent means of compressing and decompressing images and sequences of images. It also contains a simple interface for implementing software and hardware image-compression algorithms. It provides system integration functions for storing compressed images as part of PICT files, and it offers the ability to automatically decompress compressed PICT files on any QuickTime-capable Macintosh computer.

In most cases, applications use the Image Compression Manager indirectly, by calling Movie Toolbox functions or by displaying a compressed picture. However, if your application compresses images or makes movies with compressed images, you will call Image Compression Manager functions.

The Image Compression Manager is described in the chapter “Image Compression Manager” later in this book. This chapter also includes code samples that show how to compress images or make movies with compressed images.

The Component Manager

Applications gain access to components by calling the Component Manager. The Component Manager allows you to define and register types of components and communicate with components using a standard interface. A component is a code resource that is registered by the Component Manager. The component’s code can be stored in a systemwide resource or in a resource that is local to a particular application.

Once an application has connected to a component, it calls that component directly. If you create your own component class, you define the function-level interface for the component type that you have defined, and all components of that type must support the interface and adhere to those definitions. In this manner, an application can freely choose among components of a given type with absolute confidence that each will work.

The Component Manager is described in *Inside Macintosh: More Macintosh Toolbox*.

QuickTime Components

QuickTime includes several components that are provided by Apple. These components provide essential services to your application and to the managers that make up the QuickTime architecture. The following Apple-defined components are among those used by QuickTime:

- movie controller components, which allow applications to play movies using a standard user interface
- standard image-compression dialog components, which allow the user to specify the parameters for a compression operation by supplying a dialog box or a similar mechanism
- image compressor components, which compress and decompress image data
- sequence grabber components, which allow applications to preview and record video and sound data as QuickTime movies
- video digitizer components, which allow applications to control video digitization by an external device
- media data-exchange components, which allow applications to move various types of data in and out of a QuickTime movie
- derived media handler components, which allow QuickTime to support new types of data in QuickTime movies
- clock components, which provide timing services defined for QuickTime applications
- preview components, which are used by the Movie Toolbox's standard file preview functions to display and create visual previews for files
- sequence grabber components, which allow applications to obtain digitized data from sources that are external to a Macintosh computer
- sequence grabber channel components, which manipulate captured data for a sequence grabber component
- sequence grabber panel components, which allow sequence grabber components to obtain configuration information from the user for a particular sequence grabber channel component

These components and the interfaces they support are discussed in *Inside Macintosh: QuickTime Components*.

Using QuickTime

Applications that use QuickTime fall into two categories: applications that can play existing movies, and applications that can create and edit movies. The following sections describe how applications of both types use QuickTime.

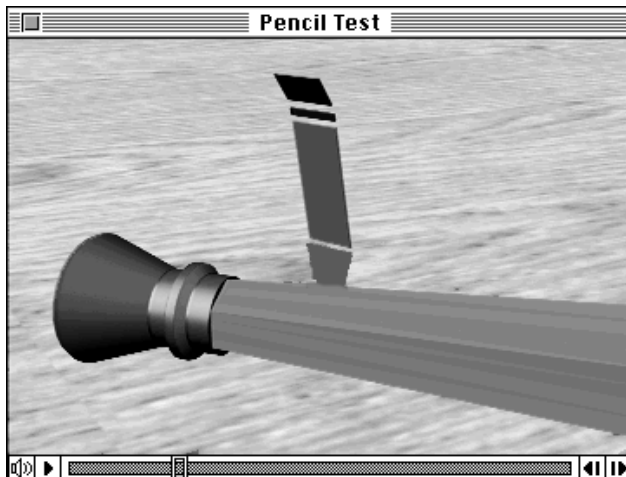
Playing Movies

QuickTime provides a complete set of tools that allow you to play movies in your application. You can also allow the user to position, resize, copy, and paste movies within the documents that your application creates and manipulates.

The Movie Toolbox provides functions that enable you to get a movie into your application; you can either get a movie from a file or from the scrap. Positioning the movie within a document varies with the application. For example, in a text document a movie might be repositioned with tab settings, whereas in a paint document the user might position the movie by selecting and dragging the movie rectangle.

Once you have loaded the movie into your document, you can allow the user to play it by calling the movie controller component provided by Apple. Figure 1-2 shows a sample movie controller.

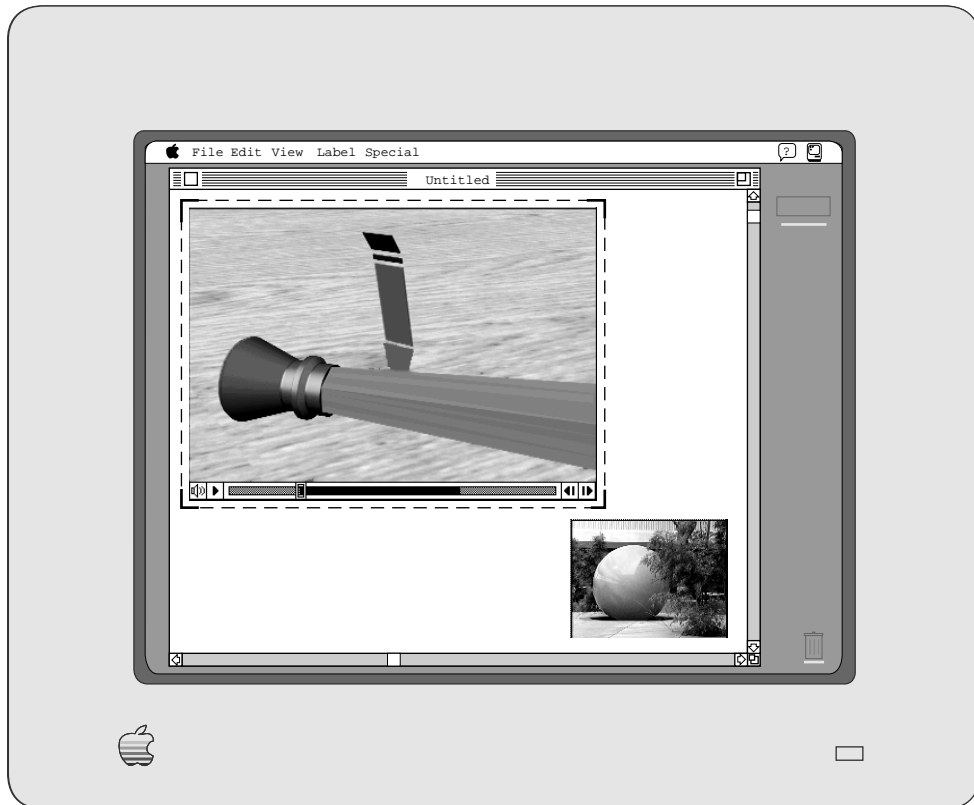
Figure 1-2 A QuickTime movie with Apple's movie controller



Introduction to QuickTime

Resizing the movie's rectangle is the same as resizing PICT rectangles within a text or paint document. When the user selects the movie, a selection rectangle appears with resizing handles at the corners of the rectangle, like those shown in Figure 1-3. The user can drag the handles to resize the movie rectangle.

Figure 1-3 A QuickTime movie with an active selection rectangle



Changing the size of a movie window may affect the performance of the video during playback as well as its appearance on the display.

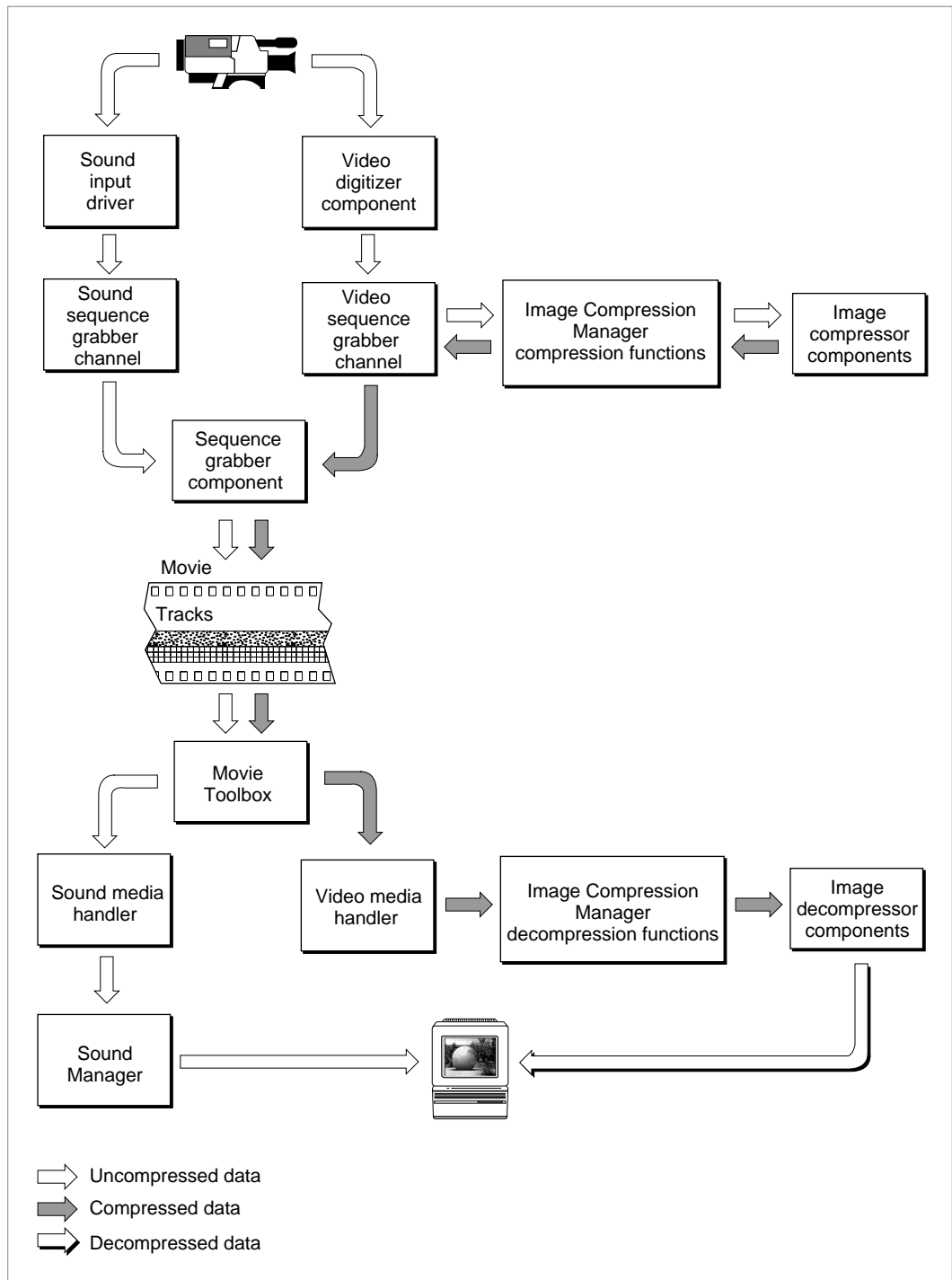
Creating and Editing Movies

More sophisticated applications allow the user to create new movies and edit existing ones. An example of a movie-creating application is an electronic mail system that supports the creation and transmission of video memos. Other examples are an application that might be included in a video digitizer card package, an architectural walk-through program, or an application that creates animation sequences that can be saved as QuickTime movies.

Movie-creating applications fall into two categories:

- those that use a sequence grabber component and the compression functions of the Image Compression Manager to obtain movie data
- those that make a movie and then use the Movie Toolbox and the decompression functions of the Image Compression Manager to work with the movie data

If you are creating an application that creates or edits movies, you are going to use more of the capabilities of the Movie Toolbox and the other managers that make up QuickTime. Figure 1-4 shows some of these other elements in an expanded view of the QuickTime architecture. For comprehensive information on the video digitizer component, the sequence grabber channel component, the sequence grabber component, and video and media handlers, see *Inside Macintosh: QuickTime Components*.

Figure 1-4 Capturing and playing back movies

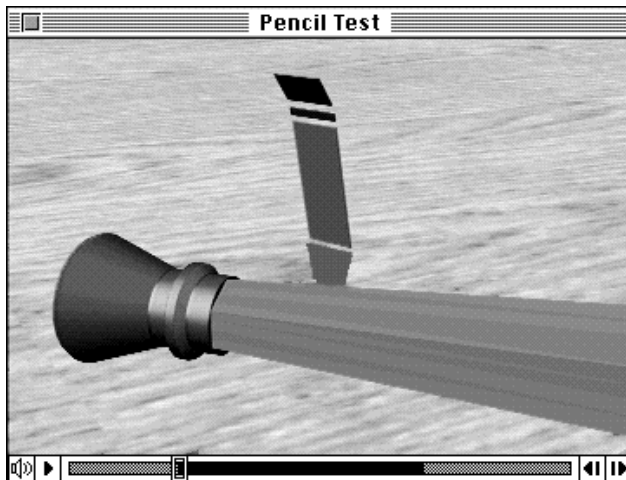
Movie-Editing Applications

The Movie Toolbox includes functions that help your application provide movie-editing capabilities to the user. The easiest way to allow the user to edit a movie is to use the movie controller component provided by Apple.

Alternatively, you can use QuickTime's editing functions to remove, copy, replace, rearrange, or extend the content of movies. The user interface for editing is up to you, as long as you observe the guidelines suggested by Apple (see the chapter "Movie Toolbox" later in this book for more information on human interface guidelines for movie applications).

To give a user some simple editing tools, you could use the movie controller component to create a movie-editing window similar to the one shown in Figure 1-5.

Figure 1-5 Apple's movie controller with a portion of the movie selected for editing



This window gives the user access to various viewing and editing controls. These controls include a real-time position controller that allows random access over the length of the movie, single-step controls in both forward and reverse directions, visual feedback for selecting a sequence of frames in the movie, and a rectangular marker highlighting the currently displayed frame.

Movie-Creating Applications

Applications that create QuickTime movies can capture the movie's data from an external source and store it in a media. As with any movie, this data may be digitized video, digitized sound, computer animation, MIDI (Musical Instrument Digital Interface) data, external data such as an audio CD or videotape, and so on. Each type of data in a movie has an associated movie track. Movie tracks contain an edit list that sequences the data stored in the media.

The Movie Toolbox supplies functions that allow you to modify the edit list of the tracks in a movie to rearrange, remove, and extend the playback display sequence of the data in the movie. You can use these functions to create an application that captures external video and creates movies.

Figure 1-6 shows a sample user interface for a video-capture application. Before the user digitizes the data, the application displays an editing window (called a *monitor window*) to help preview the information prior to capturing it.

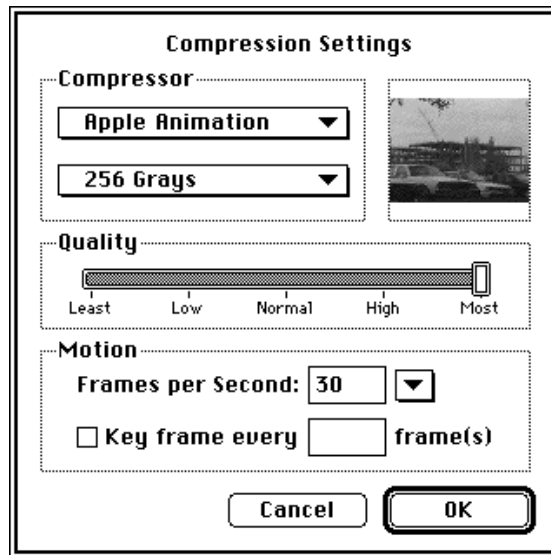
Figure 1-6 A monitor window



Introduction to QuickTime

Figure 1-7 shows a dialog box that this application provides to allow the user to select compression methods for video using the standard image-compression dialog component.

Figure 1-7 Compression settings



The remainder of this book provides the technical reference you need to develop an application that lets users display, edit, cut, copy, and paste movies and movie data in the same way that they currently manipulate text and graphic elements.

Chapter 2 discusses the Movie Toolbox, the set of functions with which you can create and modify movies and movie files.

Chapter 3 describes the Image Compression Manager, with which your application can compress and decompress still images and video sequences.

Chapter 4 describes the format and content of movie resources and movie files.

This chapter is of interest only to developers of QuickTime components.

The book concludes with a glossary and an index.