



Tech Info Library

QuickTime: How Movies Work

Revised: 1/22/92
Security: Everyone

QuickTime: How Movies Work

=====
Article Created: 13 December 1991
Article Last Reviewed:
Article Last Updated:

TOPIC -----

The most important thing about QuickTime is that it allows users to create, play, and edit sets of time-based data. If you want to capture new video footage, you will need extra equipment -- a digitizer card, video camera or controllable VCR, and associated components. But you will be able to play and edit movies and other time-based data without extra hardware.

This article describes the movie feature of QuickTime, and provides some human-interface suggestions.

DISCUSSION -----

Movies, Tracks, and Media -----

A movie file contains a resource of type 'moov' (pronounce it "moo-vee"), which describes the movie in terms of its tracks, its media, and its references to the raw data the movie is to play.

Assume the movie contains all the knowledge needed to run the movie (the tracks and the media), but not the actual contents of the movie (the data). The data can easily run into multiple megabytes, whereas the movie itself is short enough to fit on the 2 Clipboard. By factoring out the large amount of data needed to play a movie, Apple engineers have made the movie file small enough to be moved, edited, cut, copied, or pasted easily.

You can think of a QuickTime movie as a container for tracks. The movie starts at time 0 and plays until it is finished. Depending on how they are defined, one or more tracks (each containing one kind of data: video, animation, or sound, for example) are active and playing at any instant.

To keep the concept of a QuickTime media general enough to be useful with vastly different data types, a media contains not the data but a one-to-one, linear mapping between itself and the data. (Yes, we do call it "a media," largely because QuickTime documentation, routine names, and variables exclusively use the word media instead of medium.) The data, for now, is always in a file somewhere on the network. Someday the data may be outside the Macintosh and its network -- for example, on a videotape in a VCR the Macintosh controls.

Each track has exactly one media associated with it, and vice versa. The track, through the media and code called the media handler, points to whatever data should be playing at any given time. This means that a track can omit part of the media, play segments out of sequence, play the same segment multiple times, or even play sequences that overlap within the media. This means that a movie-editing program can rearrange or delete movie footage simply by adding or changing a handful of pointers within a track. (If such a program had to cut and paste the actual footage, the program would be prohibitively slow and the delay would be proportional to the amount of footage moved.)

Movies, tracks, and media all have time scales; and they can all be the same, or they can be different. If the time scale of a track is 10, then ten such units make one second of playing time, and the track can start at any 1/10-second interval. Fortunately, QuickTime takes care of synchronizing all these time scales with the others.

Human Interface Suggestions

For QuickTime to be successful, movies must have certain basic behavior that users can learn (or, better, already know) and then forget. A program should be able to select a movie, cut or copy it to the Clipboard, and paste it from the Clipboard into a document.

A program should also implement an easy, common-sense way of playing and stopping movies and controlling the sound output (a sound-mute control is mandatory, but a volume control is even better). Apple created a new Macintosh human-interface element, the standard movie controller.

The standard movie controller can give basic movie control to users in the same way every other QuickTime program will. With a few Movie Toolbox routines, the standard movie controller gives users basic control of the movie's playback.

Here are some enhancements to movie behavior to implement:

- A poster is a single image that best represents the movie. It might be used when the movie is visible in its document but isn't playing or when the document containing the movie is printed. The poster can come from one of the movie's video tracks (that is, the frame at time t from track x), or it can be an arbitrary image in its own track. (Apple recommends that you store a poster as a PICT with an uncompressed pixmap so that the user can see it in the Preview dialog box even if the proper

decompressor isn't present.)

- A preview is a short subsequence of the movie that also represents the movie in an appropriate context; it might play, for example, when a user is previewing a long movie file.
- Another useful option for movies is that of alternate tracks. With them, you can specify alternative representations of the same data, with the understanding that only one is active at a time. A simple example is an instructional movie that has vocal instructions in Spanish, English, and French, only one of which the user chooses when playing the movie. Another example is a movie that contains two alternative video tracks, one (of lower quality) meant to be played by a software-only decompressor and the other (of higher quality) requiring hardware decompression. The program playing the movie could then automatically show the movie at the best quality level available to the program and its hardware.

Copyright 1991, Apple Computer, Inc.

Tech Info Library Article Number:9415