# Tech Info Library

## QuickTime: Description of Its Components

```
Revised:        1/22/92
Security:       Everyone
```

QuickTime: Description of Its Components

=========================================================================

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

TOPIC -------------------------------------------------------------

This article describes QuickTime's three major components:
- the Movie Toolbox
- the Component Manager (and its components)
- the Image Compression Manager

The article concludes with a description of graceful degradation as it
applies to QuickTime.

DISCUSSION --------------------------------------------------------

The Movie Toolbox
-----------------
The Movie Toolbox includes the routines you need to play, edit, create, and
otherwise manipulate time-based data in QuickTime files.  If a file
contains compressed image data, the Movie Toolbox automatically interacts
with the Image Compression Manager to decompress the movie in the optimum
way in real time.  The Movie Toolbox includes routines with names such as
CreateMovieFile, CutMovieSelection, IsMovieDone, and AddUserData.

The Component Manager
---------------------
The Component Manager manages multiple components -- software packages that
are somewhat like device drivers.  Like drivers, components can be the
software interface to external hardware.  Unlike drivers, however,
components (for example, a software-only image compressor) can do work by
themselves.  Components are also different from drivers in that components
provide a high-level, device-independent way to get the job done.  (In
contrast, to use a driver you have to know its exact name and
characteristics.)

At startup, the Component Manager searches for all files of type 'thng' and
registers them.  Once they are registered, your program can ask the
Component Manager which components are available and can then use whichever
one you choose.

The most specific way the Component Manager can search for a component is
by type (a four-character ID that characterizes a component as being able
to respond to a specific set of calls).  For example, 'imdc' components
provide a standard set of image-decompression services.  Note, however,
that the 'imdc' component can be either a software-only decompression
routine or the interface to a hardware-decompression NuBus card.  A program
using QuickTime can call FindNextComponent repeatedly to find out what
kinds of decompression are available and can then decide whether to go with
the fastest decompression available or perhaps a slower one that gives a
better image.

The Component Manager also allows a component to describe itself in greater
detail.  The component's description (and a program's query) can also
include a four-character subtype, a four-character manufacturer ID, and a
32-bit field of flags selected by a 32-bit mask field.  By using full
component descriptions and the Component Manager, your program can find the
best component for the job without opening it or knowing its exact
characteristics.  If the component allows multiple simultaneous access, the
Component Manager will allow multiple callers to use multiple instances of
the component at the same time.

QuickTime 1.0 will include six component types:
- image compression ('imco')
- image decompression ('imdc')
- standard movie player ('play')
- component ('thng')
- video digitizer ('vdig')
- clock ('clok')

The Image Compression Manager
-----------------------------
This set of routines is your program's gateway to image-
compression/decompression services.  The Image Compression Manager does so
many things that it is almost impossible to list them all.  The paragraphs
that follow describe the most important ones.

- The Image Compression Manager provides a standard high-level interface
  for the compression and decompression of images.  These services are
  implemented as QuickTime components, so the Image Compression Manager
  uses the Component Manager to get its work done.  These two managers
  work together to give your program easy, device-independent access to
  the best compression and decompression services available on the
  Macintosh running your program.

- The Image Compression Manager ensures that your efforts today will
  automatically work with new compression/decompression solutions in the
  future.  The Component Manager can hide whether a component is software

or hardware.  This means that your QuickTime-aware program coded today
can automatically take advantage of future compression/decompression
solutions without changes in its code.

- The Image Compression Manager enhances the PICT definition in a way that
  benefits existing and new applications.  As mentioned at the beginning
  of this article, when QuickTime is present, any program -- QuickTime-
  aware or not -- can read PICTs that contain QuickTime-compressed
  pixmaps.  (Apple engineers have extended the PICT definition by defining
  a new opcode value for compressed pixmaps.  If a Macintosh is not
  running QuickTime, it simply skips the compressed image data.)  Use this
  new capability to add more color and gray-scale pictures to your
  programs.

- When an application wants something done that the available compressor
  and decompressor components can't do, the Image Compression Manager
  helps the work get done.  Sometimes none of the components can act on a
  request, because the image is at the wrong bit depth, is of the wrong
  type (indexed versus direct color), is too big to be done in one piece,
  or is being displayed across two different monitors.  In such cases, the
  Image Compression Manager automatically steps in to convert the image to
  the proper type, hand it to the appropriate component, and then convert
  (if necessary) and return the result.

Compressor and Decompressor Components
--------------------------------------
QuickTime 1.0 will ship with compressor and decompressor components for
three data types: still-image, video, and animation.

Use these components for still and moving pixmap images only; they aren't
meant to be used on audio, text, or any other data type.  Some compressors
can be either "lossless" (the decompressed data is identical to the
original) or "lossy" (the decompressed version has lost some of the
original information).  Many lossy compressed images can be much smaller
than a lossless version could ever be and still decompress to an image that
shows little or no difference from the original.  (This is especially true
for video and animation movies, where each image is seen for only a
fraction of a second.)

Apple will supply three compressors (and their accompanying decompressors)
with QuickTime 1.0: the Photo, Animation, and Video Compressors.  Because
these compressors are a part of QuickTime, they are a part of Macintosh
system software, and this means they are available to any color-capable
Macintosh.

Although these software-only compressors are very useful just as they are,
they don't constitute Apple's full video-compression/decompression
architecture.

QuickTime 1.0 Compressors
-------------------------
The following paragraphs describe each of the compressors that come with
QuickTime 1.0.  Please keep in mind that the "average" compression ratios

quoted below vary widely with the content of the image and the nature
(lossy or lossless) of the compression.

- Photo Compressor
  The Photo Compressor is a software implementation of the baseline JPEG
  (Joint Photographic Experts Group) still-image standard.  It will work
  with images (as many as 24 bits of color per pixel) and can, depending
  on the image content, achieve 5:1 to 10:1 compression ratios with little
  or no visible change to the decompressed image.  (Higher compression
  ratios are also possible, often with very good results.)

- Animation Compressor
  The Animation Compressor is best at compressing and decompressing
  computer-generated animations—graphics sequences in which much of the
  image is "pure" and doesn't change between successive frames.
  (Animation is usually created by programs such as Electronic Arts'
  Studio/1, MacroMind's Director, and Paracomp's FilmMaker).  The
  Animation Compressor has the very important quality of being able to
  decompress and play a movie (along with its synchronized sound) in real
  time from hard disk or CD-ROM.

  Apple's Animation Compressor is based on run-length encoding (RLE)
  compression, which is very effective on images that contain "runs" of
  identical pixels.  The "noise" of a digitized video image (where pixels
  change value slightly from frame to frame) usually defeats the
  advantages of RLE compression.

  The Animation Compressor can work in either a lossless or a lossy mode.
  In addition, it can store movie frames by their differences —- that is,
  store only those pixels that have changed between successive frames.
  This method, called frame differencing or temporal compression, can
  achieve very high levels of compression.  Compression ratios are often
  in the range 10:1 to 50:1.

- Video Compressor
  The Video Compressor works well on digitized video or scanned imagery.
  Like the Animation Compressor, it can decompress and play a movie and
  its sound in real time. (Without the ability to play movies directly
  from a disk, QuickTime would be able to show only very short movies that
  could be stored in and played from memory.) This compressor can also use
  frame differencing; when it does, it can produce compression ratios in
  the range 5:1 to 25:1.

For still-image compression and decompression, trade-offs occur among five
factors: compression time, decompression time, image quality, processing
power of the computer used, and compression ratio.  Along with the above
trade-offs, movie compression and decompression add two more: the frame
rate (the number of frames displayed per second) and the presence or
absence of audio.  Although video quality can vary with circumstances, the
quality of the audio itself is usually assured: QuickTime gives audio
higher priority than video.

Graceful Degradation

--------------------
One very important property of QuickTime is its "graceful degradation."
That is, when given a larger task than it can handle, QuickTime
approximates or drops less important attributes and retains the most
important ones for as long as it can.  For example, if QuickTime can't do
32-bit colors, it uses the 8-bit colors it has and approximates more colors
by dithering.  Or, if QuickTime can't show a movie at its internal frame
rate, it drops frames to keep up but always keeps the audio running
smoothly.

Copyright 1991, Apple Computer, Inc.


Tech Info Library Article Number:9414