What's New in QuickTime for Windows 2.1.1?

What's New in QuickTime for Windows 2.1.1?

Contents

This document provides a brief overview of the new features of QuickTime for Windows. It includes information about

■ QuickTime for Windows 2.1, which was a developer-only release.

■ QuickTime for Windows 2.1.1, which is available to end users as well as developers.

This document is divided into four sections:

■ "New Features" (page 5)

■ "API Changes" (page 9)

■ "Bugs Fixed" (page 11)

■ "Known Issues" (page 14)

The information introduced in this document will be discussed in detail in the revised QuickTime for Windows documentation; in particular, in the book *QuickTime 2.1.1 Developer Guide for Windows*.

New Features

Key new features of QuickTime for Windows include:

■ **New 32-bit version of QuickTime for Windows**

    □ Compatible with Windows 95
    □ Compatible with Windows NT 3.51
    □ Can co-exist with 16-bit version

■ **New audio resampling algorithm**

    □ Solves audio synchronization problems, as discussed in "Audio Resampling" (page 6)

■ **New installer features**

    □ Delivered as a single file
    □ Increases reliability of installations
    □ Includes option for automatic deletion of older versions of QuickTime for Windows

☐ Gives developers greater control over the installation process through an optional `.INI` file, as discussed in "Installation Customization" (page 7)

■ **New control panel features**

☐ Includes an option on the audio page that allows user to turn off the new audio-resampling algorithm

Audio Resampling

QuickTime for Windows 2.1.1 includes a new audio resampling feature ("automatic audio rate adjustment"), which solves the audio synch problems that exist on many of the newest multimedia machines that use the ESS audio chipset.

The audio clock in this chipset is inaccurate (for example, the Compaq Presario 992 gets out of synch by 1/2 second in approximately two minutes of play) and is not stable (the amount of inaccuracy varies from machine to machine, and a single machine's audio rate drifts over time with temperature). The result for users is out-of-synch audio that plays off-pitch, on what was supposed to be a high-end modern multimedia system. QuickTime for Windows 2.1.1 solves this problem by tracking the actual hardware playback rate and adjusting the output rate to match, resulting in accurate synch and accurate pitch.

On some rare configurations, however, the audio hardware queue can cause QuickTime for Windows 2.1.1 to be quite imprecise in observing the actual audio rate. This can cause the adjustment algorithm to lose stability (wild pitch swoops that finally stick at a high-pitched value). If you get a support call describing these symptoms, you should recommend that the user turn off the automatic rate adjustment feature on the control panel's audio page, and reset "Requested Rate" to "default." These two steps set the audio rate back to where it should be (but unadjusted) and turn off any further adjustments.

If lip-synch is not critical for your application, or you want to avoid this rare situation entirely, you can simply turn off the automatic rate adjustment feature in the QTW.INI file from your installer, before installing QuickTime for Windows 2.1.1. This can be done with the following code fragments (pick the one that is appropriate for your installer, depending on whether you have installed 16-bit or 32-bit QuickTime for Windows):

```
// Disable automatic audio rate adjustment in 16-bit QTW
WritePrivateProfileString("Sound", "DisableAutoRateAdjust",
                          "1", "QTW.INI" );

// Disable automatic audio rate adjustment in 32-bit QTW
WritePrivateProfileString("Sound 32", "DisableAutoRateAdjust",
                          "1", "QTW.INI" );
```

Installation Customization

If your program requires QuickTime for Windows, you can install QuickTime for Windows as part of your installation process using the InstallShield interface.

The following example shows the format for the optional configuration file that you can use with QTINSTAL.EXE. If you use an optional configuration file, you must name it QTINSTAL.INI, and you must locate it in the same directory as QTINSTAL.EXE. For all of the options listed except `DialogStyle`, a value of 1 enables the option, a value of 0 disables it.

Although all combinations of options are designed to work (that is, the program will run correctly), not all combinations of options will yield a viable result. For example, creating a program group without unpacking the files would probably not be a good idea.

The example follows:

```
    ; All options must be under this section.
[Options]

    ; Specify one of the following values:
    ;    1 = Thin frame
    ;    2 = System menu
    ;    3 = Thin frame and system menu
    ;    default = thick frame, no system menu
DialogStyle=3

    ; 1 shows background window with QuickTime banner.
    ; 0 shows no background window.  0 is useful when
    ; QTINSTAL is to be called from another program.
StandAlone=1
```

```
    ; CTL3D look.  1 = CTL3D, 0 = no CTL3D
Ctl3D=1



    ; Run dialogs that scan disk checking for
    ; existing versions.
CheckExistingVersions=1

    ; Update user INI files or do not
UpdateIniFiles=1

    ; Create user groups or do not.
CreateGroups=1

    ; Unpack files from executable and write them.
    ; Care and consideration should be used before
    ; setting this option to false, since a false
    ; value means no files will be installed.
UnpackFiles=1

    ; Run opening dialog prompting user to start
    ; installation.
PromptToBegin=1

    ; Final "do you want to continue" dialog before
    ; files are written.
PromptToComplete=1

    ; Dialog indicating that installation has completed
    ; successfully.
SuccessDialog=1

    ; If the following option is used, the specified name
    ; is used as the group name (that is, the name as it
    ; displays in the window title bar, not the group
    ; filename) that the installer will use when
    ; installing the QuickTime icons.  For example,
    ; the option shown would use the group name
    ; "My Group."  If the group does not exist it will
```

```
    ; be created.
    ; This option can be used to add the QuickTime
    ; applications to an existing group file.
    ; The string used to specify a group name should
    ; be tested in actual use, since there is a
    ; practical upper limit to the number of
    ; characters Windows will use in a window title.
GroupName=My Group

    ; If the following option is set, the installer will play
    ; the audio calibration movie that sets the initial
    ; state for automatic audio rate adjustment. This
    ; option should be turned on (which it is by default),
    ; unless your installer has disabled automatic audio
    ; rate adjustment, in which case this option must be
    ; turned off.
PlaySampleMovie=1
```

API Changes

Throughout the QuickTime for Windows header files, small changes have been made where necessary to support 32-bit compilers or the Win32 API. Other than enabling 32-bit builds, it's unlikely that these changes will affect you by breaking a build or producing any side effects. Changes that fall into this category are almost always surrounded by preprocessor directives, for example:

```
#ifndef _WIN32
    OSErr QTAPI NewMovieFromDataFork (Movie FAR *, HFILE, LONG, UINT);
#else
    OSErr QTAPI NewMovieFromDataFork (Movie FAR *, HANDLE, LONG, UINT);
#endif
```

In this example, the function NewMovieFromDataFork has both a 16-bit and a 32-bit version. The 32-bit version expects a value of type HANDLE as its second argument, rather than an HFILE value, due to a change in the file-handling interface between Win16 and Win32.

New Typedefs and Enums

QuickTime for Windows 2.1.1 includes three new types and constants:

- **PaletteHandle**

  ```
  typedef Handle PaletteHandle
  ```

  This type was added for use with the new `SetEntryUsage` function.

- **pmAnimated**

  ```
  emum {
      pmAnimated = 0x0004,
  }
  ```

  This enumerator was also added for use with the new `SetEntryUsage` function.

New Functions

There are four new functions in QuickTime for Windows 2.1.1:

- **GetTrackPalette**

  ```
  PaletteHandle GetTrackPalette(Track trkTrack)
  ```

  In the `trkTrack` parameter, you specify the track whose palette you want to retrieve. The `GetTrackPalette` function retrieves the palette and returns a handle to it as the function result. If the track does not have a palette, the function returns `NULL`.

  You can use the functions `GetMovieIndTrack`, `GetMediaTrack`, or `GetMovieIndTrackType` to obtain a valid track.

- **SetEntryUsage**

  ```
  void SetEntryUsage (PaletteHandle plthDestPalette, short srcEntry,
                   short srcUsage, short srcTolerance)
  ```

  This function allows you to set the state for specific palette entries so that flagged entries in the color palette will be animated automatically by the hardware. It applies only to movies with 8-bit color.

  In the `plthDestPalette` parameter you supply a handle to the palette, which must be retrieved from `GetTrackPalette`.

  In the `srcEntry` parameter you supply a value from 0 to 255 that represents an entry in the color lookup table (CLUT) to apply the usage flag to.

You can set the `srcUsage` parameter to `pmAnimated` to animate the entry or to 0 to turn off animation. The `srcTolerance` parameter is reserved; you must set it to 0.

■ **SetTrackFollowCLUT**

```
void SetTrackFollowCLUT (Track trTrack, Boolean fFollow)
```

This function allows you to enable or disable redithering of the palette of a specified track. You use the `trTrack` parameter to specify the track and the `fFollow` parameter to indicate whether redithering should be enabled or disabled.

■ **GetMediaSampleReference**

```
OSErr GetMediaSampleReference (Media, long FAR *, long FAR *,
                               Time Value, TimeValue FAR *,
                               TimeValue FAR *,
                               SampleDescriptionHandle, long FAR *,
                               long, long FAR *, short FAR *)
```

This function, which was included in the original version of QuickTime for Macintosh, has been added to QuickTime for Windows. It is similar to the `GetMediaSample` function, which returns a sample from a movie data file. This function returns reference information about a sample from a movie data file, but does not return the actual sample data.

Bugs Fixed

The following bugs have been fixed in QuickTime for Windows 2.1.1:

■ **Audio resampling**

   □ Fixed sound synchronization by automatically adjusting audio output sample rate to exactly match the actual rate at which the sound card is playing samples, even if that actual rate changes over time. This includes changes to the sound code, the control panel audio page, and the installer, which now plays a special calibration movie (instead of the old sample movie) that does the initial QuickTime calibration to match the sound card.

   □ Fixed bug in audio resampling for rates above 32 KHz.

- **Installer**
  - □ Fixed an installer bug where it might delete the reinstaller just after installing it.
  - □ Fixed the 32-bit installer where it might report an earlier 32-bit version where there was only a 16-bit version. It wouldn't ever delete the 16-bit version, fortunately, but the message was incorrect. A similar bug was fixed in the 16-bit installer.
  - □ Updated the installer's "required disk space" numbers to match reality.
  - □ Fixed the installer so it no longer fails when the installer temp directory already exists.
  - □ Modified the installer to delete "Optimize=Driver" entry from QTW.INI if it found that Win95 put it there.
  - □ Fixed the installer so it now deletes obsolete icons from the QuickTime program group.
  - □ Fixed the uninstaller so it now correctly undoes MCI and extension associations.

- **Cinepak**
  - □ Fixed 0.5x blits direct to a hard-banked display.
  - □ Fixed 2x and 1x blits direct to a hard-banked 32-bit pixel display.

- **CIRRUS component**
  - □ Fixed 32-bit pixels.
  - □ Fixed screen larger than 1 MB.
  - □ Fixed conflict with Tseng chipset.
  - □ Worked around Compaq PCI driver bug.
  - □ Worked around QTVRW bug.

- **QTVHDW component**
  - □ Worked around QTVRW bug, which was causing control panel crashes.
  - □ Worked around S3 Trio 64+ DCI provider bug (claims to be banked, but isn't).
  - □ Worked around Trident DCI provider bug (part of returned data structure is missing).
  - □ Eliminated "DCIMAN.DLL missing" error message.

■ **DCI32**

☐ Worked around S3 Trio 64+ DCI provider bug (screen pointer was 16-bit selector instead of 32-bit linear address).

☐ Eliminated "DCIMAN32.DLL missing" error message.

■ **Error messages**

☐ Removed "Library not freed ... call Microsoft" debug message, although the Microsoft bug remains (we are unable to free QTIM32.DLL during `QTTerminate` due to undocumented resource locks that can sometimes cause deadlock).

☐ Removed incorrect and confusing Movie Player error message, "Movie contains data reference. Data references are not supported in this version of QuickTime for Windows."

■ **Miscellaneous**

☐ Fixed 16-color display crash caused by odd-width movies.

☐ Fixed crash when opening sprite movie (we don't yet support sprites in QuickTime for Windows, but "Unable to open movie" is a more desirable behavior).

☐ Fixed Indeo codec to no longer crash on 32-bit pixel displays.

☐ Fixed MCI bug that prevented more than one movie being opened at once.

☐ Fixed bug when multiple 16-bit applications all played movies simultaneously.

☐ Fixed bug where QuickTime for Windows paints the movie frame at client offset (0,0) just before the Movie Controller positions it somewhere else.

☐ Fixed "occasional garbage noise" bug in 32-bit QTW's IMA audio support.

☐ Fixed "non-1.0 play rates play audio at 1.0 rate" bug.

☐ Fixed the `GetTrackEnabled` function (was returning garbage values).

☐ Modified the player and viewer About boxes to indicate 16-bit or 32-bit.

☐ Rebased all 32-bit DLLs for faster loading.

☐ Fixed handling of long file names and file names containing spaces.

☐ Simplified the user READ ME significantly.

Known Issues

The following list describes known issues in QuickTime for Windows 2.1.1:

- **Incorrect error codes**

  The following functions return incorrect error codes:

  □ The `PictureToDib` function sets the internal error number to 0x6fc62a32, rather than `noErr`, when it succeeds.

  □ The `GetMoviePict` function returns -2161, rather than `noErr`, when it succeeds.

  □ The `OpenMovieFile` function, when passed the name of a nonexistent file, returns the `invalidDataRef` error code, rather than the `wfFileNotFound` error code.

- **Audio rate**

  □ QuickTime for Windows may set the audio Requested Rate to the rate of the first movie that you play (which will be 22050 Hz if you play the calibration movie during installation). You may want to change Requested Rate on the QTW control panel Audio page. QTW will not change Requested Rate once you have set it.

- **32-bit versus 16-bit QuickTime for Windows**

  □ 32-bit QuickTime for Windows runs on Windows 95 and Windows NT only. It will not run on Windows 3.1, even if you have Win32s installed.

  □ 32-bit QuickTime for Windows does not have an MCI driver for Windows 95, since Windows 95 does not support 32-bit MCI drivers. 32-bit applications can still make MCI calls, but they will be serviced by 16-bit QTW. (You can install both 16- and 32-bit QTW on your system, and 16-bit QTW will automatically be used to support MCI.)

  □ On Windows NT, 16-bit MCI drivers are not supported; 32-bit QTW must be installed to get MCI support for both 16-bit and 32-bit applications.

  □ When you are running Windows 95 and you assign a sound to play when a program is launched, you may get a system halt when you launch the 16-bit Movie Player for the very first time. The problem will not recur after the initial system halt and reboot.

        □ QuickTime VR (distributed separately) is currently only a 16-bit component and therefore only works with 16-bit QuickTime for Windows. As a workaround, you can install both 16- and 32-bit QuickTime for Windows on your system, and use the 16-bit Movie Player when you want to view QTVR scenes or objects.

■ **Miscellaneous issues**

        □ MACE compressed sound is not supported.

        □ The recently released Real Magic MPEG driver (v2.20) from Sigma Designs is incompatible with QuickTime for Windows. As a workaround on Windows 3.1, you can reinstall the older Real Magic driver (v2.01), but on Windows 95 there is no other version to install. Sigma Designs and Apple are currently working on a solution.

        □ The Cirrus display driver v1.23 (aka Compaq/CL3X v1.23) can cause QuickTime for Windows control panel crashes. Upgrading to v1.24 fixes the problem.