QTAudioPlayer - Version 1.0.1 - released: 01/14/2002

This RB class is open source. A few people have been a great deal of help to my efforts, but I would like to offer special thanks to the guys working on TBFinder (who've made this project much simpler because of that great tool), Doug Holton, Joe Huber, Matthijs van Duin and Erick Tejkowski who have answered questions and helped me to find documentation for some of the methods.

If you want to contact me, my name is Benjamin J. Schneider (Ben) and my email is schneidb@ohio.edu. Feel free to drop me a post to let me know if you've improved on the class so I can incorporate your changes into the module that will be posted on my website.

QTAudioPlayer - What is it?

This class is a subclass of the MoviePlayer class with enhanced audio functions. QTAudioPlayer supports Mac OS 8.5 through Mac OS X with QuckTime 4.0 or higher.

Conventions:

Many of these methods return integer values. With methods that return an integer value, a value returned that is less than 0 often means an error has occurred and the error returned will be documented in Inside Macintosh. So, if we are trying to set a parameter, the only thing the return value represents is the error code. 0 often means no error has occurred. If we are trying to retrieve data from a device, a negative value will still often represent an error while a nil (0) value or a positive value will be a result we can use. If I needed to return a string of characters, I usually called a msgBox within the method to report the error instead of returning the error code from the method. As far as the exceptions to this go, they are noted in each function's description.

Instructions:

(Sub) - SetMovieVolume (theVolume As Integer) - Pass the Volume parameter to set the playback level of the movie in code. The range is from 0 to 255.

(Function) - GetMovieVolume () As Integer - Returns the current playback volume of the movie. The range is from 0 to 255.

(Function) - GetMediaHandler (theTrack As Integer) As Integer - Returns the media handler for the soundtrack of the movie (passed in "theTrack" parameter) loaded in the QTAudioPlayer. You can use this integer to call the other functions in the class.

(Sub) - GetSoundLevelMeterInfo (ByRef leftCh As Integer, ByRef rightCh As Integer) - This subroutine will set the leftCh and rightCh parameters to the current left and right audio levels of the movie that is loaded in the QTAudioPlayer. To use this for a VU Meter, you'll have to call this over and over again very quickly, either in a thread with a loop or with a timer with a vey short period. The range of values that leftCh and rightCh will be changed to is from 0 to 255. If you're unsure how to use ByRef parameters, see the project file or refer to the RealBasic documentation.

(Function) - GetSoundLevelMeteringEnabled (theTrack As Integer) As Integer - Use this call to see if level metering is on or off. A return value of 0 represents that level metering is off and a value greater than 0 represents that level metering is on.

(Sub) - SetSoundLevelMeteringEnabled (theTrack As Integer, theState As Integer) - Use this subroutine to turn level metering on of off. Set "State" to 1 to turn on level metering and set it to 0 to turn off level metering. You will not need to use this method if you always want level metering enabled when a movie is playing because I have already put this call in the play (enable) and stop (disable) events of the QTAudioPlayer.

(Function) - GetSoundBalance (theTrack As Integer) As Integer - This method will return a value between -128 and 128. It will return a value of 0 if the balance between the left and right channels are equal. It will return -128 if the balance is all the way to the left and it will return 128 if the balance is all the way to the right.

(Sub) - SetSoundBalance (theTrack As Integer, theBalance As Integer) - Use this subroutine to set the balance of a track. Send an integer value between -128 and 128 where -128 is all the way to the left, 0 is centered and 128 is all the way to the right.

(Sub) - GetSoundBassAndTreble (ByRef bassLevel As Integer, ByRef trebleLevel As Integer) - This subroutine will set the bassLevel and trebleLevel parameters to the current bass and treble volume of the track. The range for each parameter is -256 to 256. If you're unsure how to use ByRef parameters, see the project file or refer to the RealBasic documentation.

(Sub) - SetSoundBassAndTreble (theTrack As Integer, bassLevel As Integer, trebleLevel As Integer) - Use this subroutine to set the bass and treble volume. The range for each parameter is from -256 to 256.

(Function) - GetMovieDuration () As Integer - This Function will return the Movie's duration in seconds.

(Sub) - SetSoundEqualizerBands (theTrack As Integer, frequencyList As String) - Use this subroutine to set the bands of the equalizer for the track passed in "theTrack" parameter. Pass a comma delimited list of frequncies. See the code in the demo if you're not sure what this means. Note, you cannot adjust the EQ's levels. QuickTime's EQ is for display purposes only. Use the SetSoundBassAndTreble subroutine to make playback adjustments.

(Function) - GetSoundEqualizerBandLevels (theTrack As Integer) As String - This subroutine will return a comma delimited list of the levels of the bands in the QuickTime equalizer. This works just as the GetSoundLevelMeterInfo, except that it returns band values rather than a summed total. The range of the returned values is 0 through 255.

(Function) - GetMovieRate () As Integer - Use this function to determine the current speed of playback for the movie loaded into the player. A return value of 65536 represents normal playback speed and a return value of 0 is stopped. This value is a signed integer, which means the range is from -2147483647 to 2147483647.

(Sub) - SetMovieRate (theRate As Integer) - Use this subroutine to set the current playback rate of the movie. Because normal playback is 65536, you can pass a value slightly higher or lower to make precise playback adjustments. Alternatively, you can pass an extremely large value (between -2147483647 and 2147483647) to mimick the search functions on CD players or the rewind / fast forward functions on tape machines.

(Sub) - LoadMovieIntoRam () - This will tell the player to load the movie into ram to decrease problems with playback on slow harddrives etc. If there isn't enough ram to store the file, it will first try to load the movie into ram, then fail gracefully and play the file from disk. As of A6, there is no preflighting (to check if it'll fit into ram before even trying) and there is no threading, so you'll get the watch cursor when first loading the movie. Don't assume your Mac has crashed, it's just loading a large file into ram.

(Function) - SetSoundOutputComponent (theTrack As Integer, outComp As Integer) As Integer - Use this function to set the movie's output component. Get the component reference number from the calls below and then pass this function the reference number of the device you want the movie to play out through.

(Function) - CountComponents (compType As String) As Integer - Use this function to determine the number of components of the type that is passed that are currently registered with the component manager. To get sound output devices, pass "sdev" to this function.

(Function) - FindNextComponent (compType As String, startingAt As Integer) As Integer - Use this function to find the reference number for the next component of the component type that is passed in the compType parameter from the component reference that is passed in the startingAt parameter. For Sound Output Devices, pass "sdev" in the compType parameter. You can also pass 0 in the startingAt parameter for the first call of this function, then subsequentially pass the reference number returned from the previous call to this function. See the demo project for a working implementation of this.

(Function) - GetSoundOutputComponent (theTrack As Integer) As Integer - Use this function to get the current SoundOutput component reference number. Please keep in mind that if you haven't yet used the SetSoundOutputComponent function, you'll get 0 returned from this call.

(Function) - GetComponentInfo (theComp As Integer, ByRef compType As String, ByRef compSubType As String, ByRef compMan As String, ByRef compFlags As Integer, ByRef compFlMask As Integer) As String - Use this function to get all of the pertinant information about the component so you can build a list of devices. Pass the reference number of the component you want to get info on in the "theComp" parameter. All other parameters are passed by reference. This means their values will be set when this function is called. Additionally, the name of the component will be the returned result. See the demo project for a working implementation of this.

(Function) - GetTrackVolume (theTrack As Integer) As Integer - Use this function just as you would use the GetMovieVolume function with the exception that in this call, the range is from 0 to 512 instead of 255.

(Function) - GetTrackCount () As Integer - Use this function to get the number of soundtracks in the movie. This will need to be called prior to the other functins that require a track parameter.

(Function) - SetTrackVolume (theTrack As Integer, theVolume As Integer) As Integer - Use this function to set the volume of the track. The range is between 0 and 512.