

PLAYWAVE

Dynamic Software Solutions

5860 Picadilly Lane

Beaumont, Tx. 77708

BBS (409) 899-1709

2400,n,8,1

USERNAME == PLAYWAVE

PASSWORD == REGISTER

PLAYWAVE is the Share Ware version of my full object oriented multi-media library. This public domain library contained here in allows you to use sound wave files (*.WAV) from disc directly, or as resource objects imbedded into an applications resource file.

Included in this documentation are examples of how to use the ***PLAYWAVE*** libraries. Also included along with this document in the zipped file are example usage of the library contained in the SNDPLAY example. These examples are used as the study aid in this document.

The file *MMSYSTEM.DLL*, (shipped with WINDOWS 3.1) is required in your windows directory before these functions will operate.

PLAYWAVE.HPP

```
#include <windows.h>

typedef BOOL (FAR PASCAL *_sndPlaySound_) (LPSTR,WORD);

class PLAYWAVE
{
private:
    WORD wError;
    _sndPlaySound_ sndPlaySound;
    HANDLE hMemMMSYSTEM;
    void Error(void);
public:
    PLAYWAVE();
    ~PLAYWAVE();
    void Play(LPSTR,WORD);
    void Resource(HANDLE,LPSTR);
};

#define SOUND_SYNC          0x0000          // play synchronously (default)
#define SOUND_ASYNC        0x0001          // play asynchronously
#define SOUND_NO_DEFAULT   0x0002          // don't use default sound
#define SOUND_MEMORY       0x0004          // lpszSoundName points to a memory file
#define SOUND_LOOP         0x0008          // loop the sound until next
sndPlaySound
#define SOUND_NOSTOP       0x0010          // don't stop any currently playing sound

#define WAVE_ERROR_BAD_FORMAT 0x0020          // Wave format not supported
#define WAVE_ERROR_PLAYING   0x0021          // Still something playing
#define WAVE_ERROR_NOT_READY 0x0022          // Header not ready
#define WAVE_ERROR_SYNC_ONLY 0x0023          // Device is synchronous only
```

The above header file, *PLAYWAVE.HPP*, is a C++ header with function prototype definitions and control constants defined. This file is to be included in any applications that you develop.

PLAYWAVE

Dynamic Software Solutions
5860 Picadilly Lane
Beaumont, Tx. 77708
BBS (409) 899-1709
2400,n,8,1
USERNAME == PLAYWAVE
PASSWORD == REGISTER

PLAYWAVE.LIB

PLAYWAVE.LIB is an object library containing the code to implement the *PLAYWAVE* object. This file must be linked to the application for the function to operate. Please remember that the functions that make up the object rely on *MMSYSTEM.DLL* to be present in the *WINDOWS* directory.

SNDPLAY.CPP

```
#include "playwave.hpp"

PLAYWAVE    PlayWave;

int PASCAL WinMain(HANDLE hInstance,
                  HANDLE hPrevInstance,
                  LPSTR lpstrCmdLine,
                  int nCmdShow)
{
    PlayWave.Resource(hInstance, "SOUND");
    PlayWave.Play("SPOCKD.WAV", SOUND_SYNC);
}
```

SNDPLAY.CPP is the example usage program listing. This very simple and short program demonstrates how easy it is to play wave files using this object. Both disk file and resource file methods are demonstrated.

In line 1, the C++ header file, *PLAYWAVE.HPP* is included. The format shown is when the header file is located in the current directory. Of course the < and > symbols should enclose the file name if you build a separate \CPP\INCLnn and \CPP\LIBRnn directories for your sound library.

In line 3, an instance of the **PLAYWAVE** object is created. Once this object successfully creates an instance and initializes without error, sound capability is now enabled.

Lines 5 through 8 comprise the opening of the WinMain windows application entry point. Note the absence of a message loop. *One is not needed when you do not wish to process window messages.*

Line 10 plays a sound resource imbedded inside the resource file. The object looks for objects of type ID_WAVE matching the name specified by the LPSTR parameter. If no matching sound resource is found, then the request is ignored.

PLAYWAVE

Dynamic Software Solutions
5860 Picadilly Lane
Beaumont, Tx. 77708
BBS (409) 899-1709
2400,n,8,1

USERNAME == PLAYWAVE
PASSWORD == REGISTER

Line 11 plays a sound file (*.WAV) file from the disk. Note the use of the SOUND_SYNC directive. Any of the directives listed in the *PLAYWAVE.HPP* header file can be used.

This simple program demonstrates the use of the object in it's entirety. Please note the syntax required, and that when a resource sound is used, a handle to the instance of the application must be passed. This can be obtained from the WinMain procedure parameters.

SNDPLAY.RC

WAV_WHISTLE ID_WAVE "WHISTLE.WAV"

The above single line entry in the resource script file demonstrates how to insert a wave file resource into the resource file. The first string is the resource name identifying the sound resource inside the resource file. The second string identifies the type of resource, in this case a wave file. The third string (be sure to use the quotes) identifies the full path to the wave file to be imbedded into the resource file.

DISCLAIMER

The data contained here in is intended for educational purposes only. Any loss or damage from using these functions are the responsibility of the individual programmers or hobbyist. As with any software system, bugs do creep into the workings when least expected, so use and enjoy.