

ゲームプログラミング

第9回 MCIによるAVIムービーの再生

MCIでマルチメディアファイルを扱うAPIの中に、関数名の先頭に「MCIWnd」がつくものがあります。この系列のAPIは、マルチメディアファイルを扱うためのウィンドウ(MCIウィンドウ)を生成するのが特徴で、このウィンドウ上でさまざまな操作を行うことができます。

MCIWnd系のAPIとマクロ

MCIWnd系のAPIは、mciSendCommand関数と同じように、MCIでマルチメディアファイルを扱うAPIです。この系列のAPIは、マルチメディアファイル再生用のウィンドウを生成するのが特徴です。生成されるウィンドウの形状やサイズも自由に指定でき、ファイル操作用のツールバーも備えることができます。

MCIWndCreate関数

MCIWndCreate関数は、MCI用のウィンドウクラスを登録し、指定したウィンドウの子ウィンドウとしてMCI用ウィンドウを作成し、マルチメディアファイルを開きます。MCIWndCreate関数で生成したウィンドウは、内部で自動的にメッセージループとウィンドウプロシージャを処理します。

戻り値は、関数が成功するとMCIウィンドウのハンドルが返り、それ以外の場合は0が返ります。

MCIWndCreate(親ウィンドウのハンドル, アプリケーションインスタンスハンドル,
ウィンドウのスタイル, オープンするファイル名)

親ウィンドウのハンドル(HWND)...親ウィンドウのハンドルを指定します。

アプリケーションインスタンスハンドル(HINSTANCE)

...MCIウィンドウと関連付けるプログラムのインスタンスハンドルを指定します。親ウィンドウのハンドルからGetWindowLong関数で取得することもできます。

ウィンドウのスタイル(DWORD)...ウィンドウスタイルをフラグで指定します。CreateWindowEx関数で
使用できるウィンドウスタイルと同じものが指定できます。その他
に、次のスタイルを指定できません(一般的なもののみ掲載します)。

MCIWNDF_NOERRORDLG...MCIエラーを表示しません。

MCIWNDF_NOMENU.....メニューを非表示にし、ポップアップメニューにアクセスできないように
します。

MCIWNDF_NOOPEN.....メニューの「開く」と「閉じる」コマンドを非表示にし、これらの選択
肢にアクセスできないようにします。

MCIWNDF_NOPLAYBAR...ツールバーを非表示にし、アクセスできないようにします。

(デフォルトのスタイルはWS_CHILD、WS_BORDER、WS_VISIBLEです)

オープンするファイル名...オープンするファイルのパスと名前を指定します。

- 使用例 -

```
HWND hMCIWnd = MCIWndCreate(hWnd, hInstance, 0, "D:¥¥¥Movie¥¥¥Prolog.avi");
```

MCIWndをコントロールするマクロ

MCIウィンドウを制御するために、さまざまなマクロが用意されています。これらのマクロは、関数とほとんど同じように使用できます。よく使うのは以下のマクロです(カッコ内はパラメータです)。

MCIWndPlay(MCIWndのハンドル)...ファイルを再生します。

MCIWndStop(MCIWndのハンドル)...再生を停止します。

MCIWndSetSpeed(MCIWndのハンドル, スピード)...スピードを変更します。1000が標準です。

MCIWndSetVolume(MCIWndのハンドル, ボリューム)...ボリュームを変更します。1000が標準です。

MCIWndSetZoom(MCIWndのハンドル, 倍率)...ムービーのサイズを変更します。100が標準サイズです。

MCIWndClose(MCIWndのハンドル)...MCIウィンドウとMCIデバイスをクローズします。

MCIWndDestroy(MCIWndのハンドル)...MCIウィンドウとMCIデバイスを破棄します。

MCIWndGetPosition(MCIWndのハンドル)...再生位置を取得します。

MCIWndGetEnd(MCIWndのハンドル)...終了位置を取得します。

ヘッダファイルとライブラリ

MCIWnd系のAPIやマクロを使用したプログラムをコンパイルするにはヘッダファイル"vfw.h"が必要になります。また、リンク時にスタティックライブラリ"vfw32.lib"が必要になります。

ヘッダファイルは該当するソースファイルでインクルードしてください。ライブラリは、VisualC++

のメニューから「プロジェクト(P) 設定(S) リンク」で指定するか、該当するソースファイルでプリプロセッサ#pragmaを使って、

#pragma comment(lib, "ライブラリ名")
としてビルド時にリンクさせるようにします。

```
#include <vfw.h>  
#pragma comment(lib, "vfw32.lib")
```

MCIウィンドウの表示座標を指定

MCIWndCreate関数では、ムービーファイルの表示座標を指定することはできません。表示座標を指定したい場合は、CreateWindowEx関数でウィンドウを生成します。MCIウィンドウの生成は、通常のウィンドウとは若干手順が異なり、以下のようになります。

- (1) MCIWndRegisterClass関数でMCIウィンドウのクラスを登録
- (2) CreateWindowEx関数でMCIウィンドウを生成(ウィンドウクラス名"MCIWND_WINDOW_CLASS"を使用)
- (3) MCIWndOpen関数でファイルを開く

```
MCIWndRegisterClass(); // MCIウィンドウのクラスを登録  
// MCIウィンドウ生成(表示位置(0, 16))  
HWND hMCIWnd = CreateWindowEx(WSEX_TOPMOST, MCIWND_WINDOW_CLASS, "MCIWND",  
                               WS_CHILD | WS_VISIBLE,  
                               0, 16, 0, 0,  
                               hWnd, NULL, hInstance, NULL);  
  
if(NULL != hMCIWnd) {  
    // MCIウィンドウ生成成功  
    MCIWndOpen(hMCIWnd, "D:¥¥Project¥¥Prolog.avi", 0); // ファイルを開く  
    MCIWndSetZoom(hMCIWnd, 200); // 2倍に拡大  
    MCIWndPlay(hMCIWnd); // 再生開始  
}
```

CreateWindowEx関数の5つ目と6つ目の引数がムービーの表示座標になります。ウィンドウサイズはムービーのサイズとは無関係で、ウィンドウサイズに関係なくムービーはオープンされます。そのため、ここには0を指定してもかまいません。ムービーのサイズはMCIWndSetZoomマクロで変更します。

MCIWndOpen関数で開いたファイルは、使用後にMCIWndCloseマクロで閉じます。また、MCIWndCreate関数とCreateWindowEx関数で生成したMCIウィンドウは、不用になったらMCIWndDestroyマクロで破棄します。

- API解説 -

・MCIWndRegisterClass(パラメータなし)

MCIウィンドウクラス(登録名MCIWND_WINDOW_CLASS)を登録します。成功すると0、それ以外の場合は0以外の値を返します。

・MCIWndOpen(MCIWndのハンドル, オープンファイル名, フラグ)

ファイルとそれにあわせたデバイスを初期化してオープンし、MCIウィンドウと関連付けます。ファイルを再生に使用する場合は、フラグに0を指定します。

課題

(1) MCIWnd系のAPIを使って、マルチメディアファイルを再生してみましょう。

(2) 画面全体を覆うメインウィンドウを作成し、その上にムービー再生用のMCIウィンドウを生成しましょう。次に、ムービーを画面いっぱいに拡大し、自動的に再生が始まるようにしましょう。また、MCIウィンドウ下部の操作ツールバーと右クリックで開くポップアップメニューを非表示にしてみましょう。