

ゲームプログラミング

DirectDraw - 第18回 画像ファイルの読み込み

OleLoadPicture関数は、ビットマップのほかにJPEGやGIF形式の画像を読み込むことができます。

OleLoadPicture関数

OleLoadPicture関数を使った画像ファイルの読み込みは、以下のようになります。

```
/*
 *                                     画像読み込み
 */
bool DDLoadFromFile(const DDSRFC ddsDest, LPCTSTR lpszFileName)
{
#ifdef _DEBUG
    if(NULL == g_lpDDSurface7[DDS_BACKBUF]) {
        OutputDebugString("**** Error - バックバッファ未生成(DDLoadFromFile)\n");
        return false;
    }
    if(DDS_TEXTURE1 <= ddsDest) {
        OutputDebugString("**** Error - サーフェイス指定エラー(DDLoadFromFile)\n");
        return false;
    }
#endif

    // ファイルオープン
    HANDLE hFile = CreateFile(lpszFileName, GENERIC_READ, 0, NULL, OPEN_EXISTING,
        FILE_ATTRIBUTE_NORMAL, NULL);
    if(INVALID_HANDLE_VALUE == hFile) {
        OutputDebugString("**** Error - ファイルオープン失敗(DDLoadFromFile)\n");
        return false;
    }

    // 画像を読み込むためのグローバルメモリを確保
    const DWORD FILE_SIZE = GetFileSize(hFile, NULL); // ファイルサイズ取得
    HGLOBAL hGlobal = GlobalAlloc(GPTR, FILE_SIZE); // グローバルメモリ確保
    if(NULL == hGlobal) {
        OutputDebugString("**** Error - グローバルメモリ確保失敗(DDLoadFromFile)\n");
        CloseHandle(hFile);
        return false;
    }

    // グローバルメモリに画像を読み込む
    DWORD dwActual;
    if(0 == ReadFile(hFile, hGlobal, FILE_SIZE, &dwActual, NULL) || FILE_SIZE != dwActual) {
        OutputDebugString("**** Error - 画像ファイル読み込み失敗(DDLoadFromFile)\n");
        CloseHandle(hFile);
        GlobalFree(hGlobal);
        return false;
    }
    CloseHandle(hFile);

    // グローバルメモリからストリームオブジェクトを生成
    IStream* piStream; // ストリームオブジェクト
    CreateStreamOnHGlobal(hGlobal, TRUE, &piStream);

    // ピクチャーオブジェクトを生成し、ストリームオブジェクトから画像を読み込む
    IPicture* piPicture; // ピクチャーオブジェクト
    OleLoadPicture(piStream, FILE_SIZE, TRUE, IID_IPicture, (LPVOID*)&piPicture);

    // 画像サイズ取得
    OLE_XSIZE_HIMETRIC oxhWidth;
    OLE_YSIZE_HIMETRIC oyhHeight;
    piPicture->get_Width(&oxhWidth);
    piPicture->get_Height(&oyhHeight);
}
```

```

// 取得したサイズは、HiMetric(1/100mm)という単位なので、ピクセル単位に変換
HDC hDC = DDGetDC(DDS_BACKBUF);
const long lDestWidth = MulDiv(oxhWidth, GetDeviceCaps(hDC, LOGPIXELSX), HIMETRIC_PER_INCH);
const long lDestHeight = MulDiv(oyhHeight, GetDeviceCaps(hDC, LOGPIXELSY), HIMETRIC_PER_INCH);
DDReleaseDC(DDS_BACKBUF);

// 画像サイズでサーフェイスを生成
if(false == DDCreateSurface(ddsDest, lDestWidth, lDestHeight, 0)) {
    piPicture->Release();
    GlobalFree(hGlobal);
    return false;
}

// 描画
hDC = DDGetDC(ddsDest);
piPicture->Render(hDC, 0, 0, lDestWidth, lDestHeight, 0, oyhHeight, oxhWidth, -oyhHeight, NULL);
DDReleaseDC(ddsDest);

// イメージ解放
piPicture->Release();
GlobalFree(hGlobal);

return true;
}

```

この関数をコンパイルするには、ヘッダファイル<olecl.h>と<olestd.h>が必要になります。