

PictContainer Plug-In

By Thomas Maul, General Manager, 4D Germany.

TN 06-24

Introduction

このテクニカルノートには、4th Dimension のピクチャ処理能力を強化する 4D プラグインが収録されています。

プラグインエリアは、画像の読み込み、書き出し、表示ができ、スクロールバーおよびズーム機能がサポートされています。

Windows 版は GDI+API、Mac 版は QuickTime API が使用されています。

プラグインの C ソースコード、およびサンプルデータベースも提供されています。

Main Features

OS native library

PictContainer プラグインは、すべての Macintosh コンピュータにプリインストールされ、Apple 社が推奨するピクチャ処理ライブラリの QuickTime を使用しています。Windows では、Microsoft 社の現行グラフィック API である GDI+を使用しています。GDI+は Windows XP、Windows 2003 にインストールされており、将来のシステムでもサポートされる予定です。注記: Windows 2000 の対応については、本ドキュメント末尾の「GDI+ for Windows 2000」を参照してください。

Windows で QuickTime ではなく GDI+を選んだのは、確実にプリインストールされており、また GDI+のダイナミックリンクライブラリ(DLL)を 4D アプリケーションとバンドルとして出荷すれば、余分なインストールを省くことができるからです。ダメージを受けた画像ファイルの処理能力も、全般的に GDI+のほうが優れています。

Scroll bar support

ピクチャおよびプラグインエリアのサイズ、プラグインのズーム設定に合わせ、自動的にスクロールバーが追加されるようになっています。

Drag & Drop support

PictContainer プラグインは、4D ピクチャフィールド、変数および配列からのドラッグ&ドロップを自動的に処理することができます。4D オブジェクトに対するドラッグ&ドロップについては、プログラムで制御することができます。

Supported picture formats

Mac OS

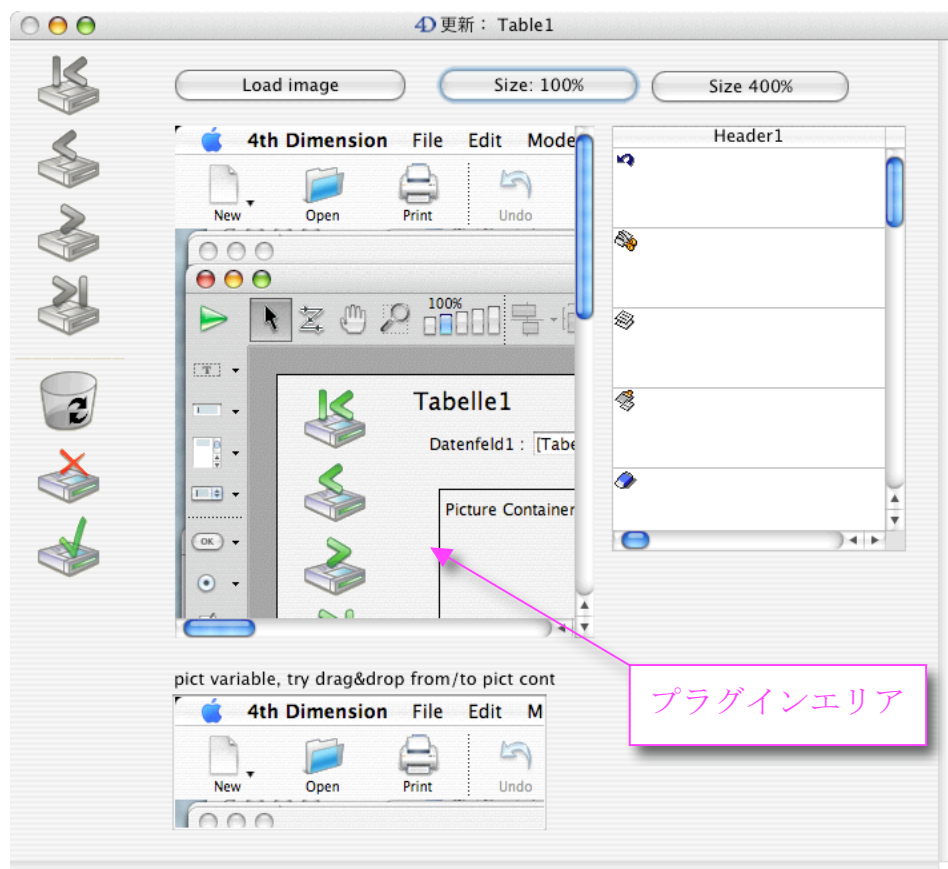
- 読み込み: QuickTime がサポートするすべての画像フォーマット、つまり 4D のピクチャ コマンドで利用できるフォーマット。PICT、BMP、JPG、PNG、TIFF、Photoshop など。
- 書き出し: 開いたファイルはすべて Macintosh ピクチャに変換されるため(つまり 4D のピクチャと一緒に)、ピクチャ変数に保存するか、元のファイルを保持することが勧められています。プラグインは、ピクチャを変換する目的では作られてはいません。既存の 4D コマンドを使用してください(WRITE PICTURE FILE、PICTURE TO BLOB など)。プラグインには、ピクチャを BMP、JPG、PNG、TIFF(非圧縮)で書き出すためのコマンドが用意されています。

Windows

- 読み込み: BMP、GIF、JPG、PNG、TIFF、EMF、WMF、ICO
- 書き出し: BMP、GIF、JPG、PNG、TIFF
-

Using the PictContainer Plug-In

PictContainer プラグインの主要な機能は、このテクニカルノートに収録されているサンプルデータベースで確かめることができます。データベースの名称は「PictContainer.4DB」です。ユーザモードで起動し、1 個のテーブルに 1 件のレコードが登録されています。[Table1]の入力フォームは、次のようになっています：



この入力フォーム上で、次の操作を試すことができます：

- 画像ファイルを選択して **PictContainer** に読み込む。
- **PictContainer** に表示されている画像のズーム率を **100%** に設定する。
- **PictContainer** に表示されている画像のズーム率を **400%** に設定する。
- **PictContainer** の画像を **4D** ピクチャ変数にドラッグ&ドロップする。
- **4D** ピクチャ変数の画像を **PictContainer** にドラッグ&ドロップする。
- **4D** リストボックスの画像を **PictContainer** にドラッグ&ドロップする。

注記：Windows で **PictContainer** を使用する場合、**4D Pack** も必要です。

PictContainer Plug-In Command Reference

PictContainer プラグインコマンドは、次のカテゴリーに分類されています：

- Area Content
- Offscreen Areas
- Drag & Drop
- Area and Picture Properties

Area Content

PictContainer プラグインエリアの内容に関するコマンドです。

PC Load Picture

PC Load Picture(Area; picture) -> error code

パラメータ	タイプ	説明
Area	倍長整数	→ PictContainer リファレンス
picture	4D ピクチャ	→ピクチャフィールド/変数
返り値	倍長整数	←エラーコード(0=エラーなし)

説明

4D(Mac)ピクチャで渡された画像をエリアに読み込みます。**Windows** の場合、ピクチャは **EMF** に変換されます。

PC Load File

PC Load File(Area; Document path) -> error code

パラメータ	タイプ	説明
Area	倍長整数	→ PictContainer リファレンス
Documentpath	テキスト	→フルパス
返り値	倍長整数	←エラーコード(0=エラーなし)

説明

指定されたパスの画像ファイルをエリアに読み込みます。**Windows** の場合、ファイルはディスクから直接表示されます(メモリ使用を抑えるため、メモリには読み込みません)。これはファイルが開かれたままになることを意味します。**Mac OS** では、**QuickTime** 使用によりドキュメントがメモリに読み込まれます。

PC Load Blob

PC Load Blob(Area; Blob) -> error code

パラメータ	タイプ	説明
Area	倍長整数	→PictContainer リファレンス
Blob	BLOB	→画像の BLOB
返り値	倍長整数	←エラーコード(0=エラーなし)

説明

BLOB 形式の画像をエリアに読み込みます。**BLOB** は、プラグインがサポートするフォーマットの画像を含んでいなければなりません。通常、そのような **BLOB** は **DOCUMENT TO BLOB** で作成します。

PC Save Picture

PC Save Picture(Area; picture) -> error code

パラメータ	タイプ	説明
Area	倍長整数	→PictContainer リファレンス
Picture	4D ピクチャ	←ピクチャ変数
返り値	倍長整数	←エラーコード(0=エラーなし)

説明

エリアのピクチャをピクチャ変数に代入します。

Mac OS の場合、エリアのピクチャは元から **PICT** フォーマットなので、単純に変数へ渡されるだけです。**Windows** の場合、画像は **BMP** に変換された後、**4D Pack** の機能で **PICT** に変換されます。これは **QuickTime** がない環境でも **Mac OS** の **PICT** 形式をサポートするためです。**4D Pack** がインストールされていない場合、エラー#-15002 が返されます。

PC Save File

PC Save File(Area; Document path; Format; Compression) -> error code

パラメータ	タイプ	説明
Area	倍長整数	→PictContainer リファレンス
Documentpath	テキスト	→フルパス
Format	テキスト	→イメージフォーマット
Compression	倍長整数	→圧縮レベル(0-100)
返り値	倍長整数	←エラーコード(0=エラーなし)

説明

指定されたパスに画像を書き出します。

フォーマットは **Format** で特定します。有効な値は **jpg**、**bmp**、**png**、**gif** のいずれかです。**Mac OS** では、**PICT** 形式もサポートされます。

圧縮レベルは **Compression** で設定します。**1** が最高の圧縮率、**100** が最低の圧縮率です。**0** を渡した場合、デフォルトの設定が使用されます。圧縮ができるのは **JPG** 形式だけです。

Windows 版の注記: ファイルを上書き保存することはできません。異なる拡張子(形式)で名前を付けて保存するか、**BLOB** またはピクチャとして保存してください。

Mac OS 版の注記: 読み込まれた画像はすべて **PICT** 形式になっていますが、書き出すことによって元の形式に戻ります。

PC Save Blob

PC Save Blob(Area; Blob; Format; Compression) -> error code

パラメータ	タイプ	説明
Area	倍長整数	→PictContainer リファレンス
Blob	BLOB	←サポート形式の画像
Format	テキスト	→イメージフォーマット
Compression	倍長整数	→圧縮レベル(0-100)
返り値	倍長整数	←エラーコード(0=エラーなし)

説明

指定された形式で **BLOB** に画像を書き出します。

フォーマットは **Format** で特定します。有効な値は **jpg**、**bmp**、**png**、**gif** のいずれかです。**Mac OS** では、**PICT** 形式もサポートされます。

圧縮レベルは **Compression** で設定します。**1** が最高の圧縮率、**100** が最低の圧縮率です。**0** を渡した場合、デフォルトの設定が使用されます。圧縮ができるのは **JPG** 形式だけです。

Windows 版の注記: ファイルを上書き保存することはできません。異なる拡張子(形式)で名前を付けて保存するか、BLOB またはピクチャとして保存してください。

Mac OS 版の注記: 読み込まれた画像はすべて PICT 形式になっていますが、書き出すことによって元の形式に戻ります。

PC Clear Picture

PC Clear Picture(Area)

パラメータ	タイプ	説明
Area	倍長整数	→PictContainer リファレンス

説明

Area で指定された PictContainer プラグインエリアをクリアし、メモリを解放します。

Windows の場合、PC Open File で開かれたファイルが閉じられます。エリアの内容もクリアされます。

Offscreen Areas

オフスクリーンエリアの使用に関するコマンドです。

PC Create Offscreen Area

PC Create Offscreen Area -> Area

パラメータ	タイプ	説明
返り値	倍長整数	←PictContainer リファレンス

PictContainer プラグインのオフスクリーンエリアを作成し、そのリファレンス ID を返します。

受け取った ID は、リファレンスを必要とする他のコマンドに渡すことができます。

作成したオフスクリーンエリアは、最後に PC DeleteOffscreen Area でクリアしてください。

Example

ピクチャをオフスクリーンに展開し、情報を取得します:

C_LONGINT(IAreaRef)

IAreaRef:=PC Create Offscreen Area

tPath2File := "C:¥¥mypicture.jpg"

IMovieFileLastError:=PC Load File (IAreaRef;tPath2File)

IMovieFileLastError:=PC Get Picture Propertes (IAreaRef; \$ResX; \$ResY; \$Width; \$Height;\$Format)

PC Delete Offscreen Area (IAreaRef)

PC Delete Offscreen Area

PC Delete Offscreen Area(Area)

パラメータ	タイプ	説明
返り値	倍長整数	→PictContainer リファレンス

説明

PC Create Offscreen Area で作成された PictContainer プラグインのオフスクリーンエリアをクリアし、メモリを解放します。

Area and Picture Properties

PC Set Display Properties

PC Set Display Properties(Area; ZoomX, ZoomY, ScrollX, ScrollY) -> error code

パラメータ	タイプ	説明
Area	倍長整数	→PictContainer リファレンス
ZoomX	倍長整数	→水平ズーム
ZoomY	倍長整数	→垂直ズーム
ScrollX	倍長整数	→水平スクロール位置
ScrollY	倍長整数	→垂直スクロール位置
返り値	倍長整数	←エラーコード(0=エラーなし)

説明

PictContainer プラグインエリアの表示プロパティを変更するためのコマンドです。

画像のズーム率は、ZoomX と ZoomY で操作します。正の値はパーセント値、つまり 100 が元のサイズで 200 が倍のサイズ、50 が半分のサイズを意味します。

次の定数を使用して自動リサイズを適用することができます：

PC_ScaledToFitProp	0	比率を維持してリサイズ
PC_ScaledToFitPropCentered	-1	比率を維持してリサイズ、中央揃え
PC_ScaledToFit	-2	エリアに合わせる
PC_ReducedToFitPropCentered	-3	比率を維持して縮小
PC_ReducedToFitProp	-4	比率を維持して縮小、中央揃え

スクロール位置は、ScrollX と ScrollY で操作します。値は 0 から画像サイズとエリアサイズの差の範囲内のピクセル数で渡します。

PC Get Display Properties

PC Get Display Properties(Area; ZoomX, ZoomY, ScrollX, ScrollY) -> error code

パラメータ	タイプ	説明
Area	倍長整数	→PictContainer リファレンス
ZoomX	倍長整数	←水平ズーム
ZoomY	倍長整数	←垂直ズーム
ScrollX	倍長整数	←水平スクロール位置
ScrollY	倍長整数	←垂直スクロール位置
返り値	倍長整数	←エラーコード(0=エラーなし)

説明

現行の表示プロパティを取得します。詳細は **PC Set DisplayProperties** を参照してください。

PC Get Picture Properties

PC Get Picture Properties(Area; ResX; ResY; SizeX; SizeY; Format) -> error code

パラメータ	タイプ	説明
Area	倍長整数	→PictContainer リファレンス
ResX	倍長整数	←水平解像度
ResY	倍長整数	←垂直解像度
SizeX	倍長整数	←ピクチャ幅
SizeY	倍長整数	←ピクチャ高さ
Format	テキスト	←ピクチャフォーマット
返り値	倍長整数	←エラーコード(0=エラーなし)

説明

エリアの画像に関する情報を取得します。**ResX**、**ResY**、**SizeX**、**SizeY** はそれぞれピクセル数で返されます。**Format** は元の画像フォーマットです。

Windows の場合、**Format** に返されるのは JPEG、BMP、EMF、WMF、PNG、GIF、TIF、ICN のいずれかです。

Mac OS の場合、値は QuickTime から返されます。

PC Get Picture Properties

PC Get Thumbnail(Area; Picture; Size) -> error code

パラメータ	タイプ	説明
Area	倍長整数	→PictContainer リファレンス
Picture	ピクチャ	→サムネイル画像
Size	倍長整数	→サイズ指定、Windows のみ

説明

4D ピクチャ形式でサムネイルを返すコマンドです。**Mac OS** の場合、**QuickTime** がサイズを設定(通常は **80** ピクセル)します。

Windows の場合、**Size** で希望のサイズが指定できます。デフォルトは **48** です。画像の縦横比率は維持されます。ピクチャにサムネイルが含まれていれば、その画像が必要に応じてリサイズされ、返されます。大きなサイズを指定した場合、元のサムネイルが拡大されるため、画質は劣ったものになります。

Windows の場合、実行には **4D Pack** が必要です。**4D Pack** がインストールされていなければ、エラー#**15002** が返されます。

PC Set Area Properties

PC Set Area Properties(Area; AllowCopy, AllowPaste, AutoDrop)

パラメータ	タイプ	説明
Area	倍長整数	→PictContainer リファレンス
AllowCopy	倍長整数	→コピーを許可する
AllowPaste	倍長整数	→カット、消去を許可する
AutoDrop	倍長整数	→ピクチャのドロップを許可する

説明

編集操作の許可/禁止をするコマンドです。

デフォルトの設定では、すべての操作が許可されています。プラグインエリアに画像が入っている場合、ユーザは編集メニューのコピーコマンドで画像をクリップボードに取り込むことができますが、**AllowCopy** に **0** を渡すことにより、この操作を禁止することができます。同じように、**AllowPaste** に **0** を渡せばペースト、クリア、カットを不許可にすることができます。編集メニューのカットが許可されるためには、コピーとペーストのプロパティが両方とも有効になっていなければなりません。(カットはコピーとクリアを複合した操作です。)

AutoDrop は、ピクチャオブジェクトの自動的なドラッグ&ドロップ処理を停止するためのプロパティです。無効にした場合でも、プログラミングによるドラッグ&ドロップ処理は、引き続き可能です。ドラッグ&ドロップコマンドの記述を参照してください。

PC Get modified

PC Get modified(Area) -> modified

パラメータ	タイプ	説明
Area	倍長整数	→PictContainer リファレンス
A 返り値	倍長整数	←0=変更なし/1=変更された

説明

ユーザが編集メニューまたはドラッグ&ドロップ操作でプラグインエリアのピクチャを変更した場合、1 が返されます。コマンドでピクチャが変更された場合は 1 が返されません。

Drag & Drop

PictContainer プラグインのプロパティリストで「ドラッグ可」が有効にされている場合、ピクチャのドラッグイベントが発生するようになります。DRAG AND DROP PROPERTIES コマンドにはプラグインオブジェクトが返されるので、それをもとに PC Save Picture でピクチャを取得し、対象オブジェクトに代入することができます。

同一フォーム(プロセス)内におけるピクチャ変数、フィールド、配列からのドロップ操作はランゲージなしで自動的に処理することができます。PictContainer プラグインのプロパティリストで「ドロップ可」を有効にしておいてください。

別プロセスからのドロップはもう少し複雑です。変数の値は GET PROCESS VARIABLE で取得できますが、フィールドの場合は相手プロセスのカレントレコードが分からなければなりません。多くの場合、そのために CALL PROCESS が使用されます。複数のピクチャ列が存在する可能性のあるリストボックスからのドロップは、同一プロセス内であっても自動では処理できません。このようなケースを想定し、プラグインは PC Allow Drop コマンドによる高度なドロップ処理をサポートしています。このオプションを選択する場合、汎用的なコードを用意し、標準的なピクチャドロップもすべてプログラムで処理することが勧められています。自動ドロップの処理は PC Set Area Properties コマンドで無効にすることができます。

PC Set Allow Drop

PC Set Allow Drop(Area)

パラメータ	タイプ	説明
Area	倍長整数	→PictContainer リファレンス

説明

発生中のドロップ処理を受け付けるコマンドです。プラグインエリアのオブジェクトメソッド、**On Drag Over** フォームイベントのコンテキストで使用する必要があります。

通常、カレントプロセス内におけるピクチャ変数、フィールド、配列からのドロップは自動的に処理されます(**PC Set Area Properties** で無効にされていない場合)。このコマンドを使用すれば、リストボックスや他のプラグインエリアからのドロップも処理できるようになります。

On Drag Over イベントの最中にこのコマンドが使用されると、プラグインはドラッグオーバーを受け入れ、プラグインエリアの上でマウスボタンがリリースされると、オブジェクトメソッドの **On Drop** イベントが発生します。この場合、ドラッグ&ドロップは、標準の **4D コマンド** で処理することができます。

Example

PictContainer の変数名が **pictarea**、リストボックスの変数名が **list box1** であるとします。リストボックスの列 **column1** がピクチャの配列です。リストボックスからプラグインエリアに対するドラッグ&ドロップは、次のようなオブジェクトメソッドで処理することができます：

Case of

¥ (**Form event=On Drag Over**)

DRAG AND DROP PROPERTIES(\$source;\$sourceelement;\$sourceprocess)

If (\$source=(->List box1))

PC Set Allow Drop (pictarea;1)

End if

¥ (**Form event=On Drop**)

DRAG AND DROP PROPERTIES(\$source;\$sourceelement;\$sourceprocess)

If (\$source=(->List box1))

\$err:=PC Load Picture (pictarea;column1{\$sourceelement})

End if

End case

Conclusion

PictContainer は、4D のピクチャ処理を強化するプラグインです。Windows および mac OS コンピュータのネイティブ API を活用しており、特別なインストール要件はありません。

GDI+ for Windows 2000

GDI+ は、Windows XP、Windows 2003 および将来のシステムでサポートされる API ですが、Windows 2000 には含まれていません。配布先に DLL がない可能性は低いと思われますが、一応、アプリケーションと一緒に出荷するのが無難です。インストールは簡単で、**4D.exe**(または **4D Client.exe** など)と同じ場所に DLL を配置しておけば OK です。再配布できる DLL は、次の場所からダウンロードできます：

PictContainer Error codes

プラグインコマンドは、次のような特殊なエラーコードを返します：

- 15001 エリア ID が無効です。
- 15002 4D Pack がインストールされていません。(Windows)
- 15003 サムネイルが作成できません。(Windows)

Windows Error Codes

次のエラーコード(正の値)は GDI+ から返されるものです：

- 1 GenericError
- 2 InvalidParameter
- 3 OutOfMemory
- 4 ObjectBusy
- 5 InsufficientBuffer
- 6 NotImplemented
- 7 Win32Error
- 8 WrongState
- 9 Aborted
- 10 FileNotFound
- 11 ValueOverflow
- 12 AccessDenied
- 13 UnknownImageFormat
- 14 FontFamilyNotFound
- 15 FontStyleNotFound
- 16 NotTrueTypeFont
- 17 UnsupportedGdiplusVersion
- 18 GdiplusNotInitialized
- 19 PropertyNotFound
- 20 PropertyNotSupported
- 21 ProfileNotFound

注記：読み込みおよび書き出しに失敗した場合、実際にはメモリ不足ではなく、パス名が間違っている場合などであっても、エラーコードには **OutOfMemory** が返されるかもしれません。

Mac OS Error Codes

標準の QuickTime/Mac OS エラーコードが返されます。