

Creating Mashups with 4D Live Window

By Joseph Resuello, Technical Support Engineer, 4D Inc.
Technical Note 07-07

Abstract

このテクニカルノートではMashupのコンセプトと、4D Web 2.0 Packに含まれる4D Live Windowプラグインを使用することにより、Mashupがどのように4th Dimensionを拡張するかについて説明します。例題データベースはプラグインの多彩な機能を紹介し、他の利用方法についてのヒントを提供します。

What is a Mashup?

エキサイティングに成長を続けるWorld Wide Webの世界において、私たちは情報を提示するまったく新しい方法の目撃者になろうとしています。Web2.0と称されるこの新世代の技術は、Webサイトの内容や機能をサードパーティーデベロッパに提供することで、リッチなユーザエクスペリエンスを提供します。Google, eBay, Yahoo, そしてAmazonなどの有名企業により公開されたApplication Programming Interfaces (API's) は、デベロッパの創造的なアイデアに基づくカスタムWebサイトやアプリケーションの構築を可能とします。

これらの新しいWebサイトやアプリケーションはしばしば“Mashup”を使用します。Mashupは既存の情報を再利用し、必要なカスタマイズを施した上で提示する方法です。4Dを使用して、開発者は彼らのデータベースとこの新しいリッチで再利用可能なコンテンツをMashupさせることができます。

[http://en.wikipedia.org/wiki/Mashup_\(web_application_hybrid\)](http://en.wikipedia.org/wiki/Mashup_(web_application_hybrid))

What is 4D Live Window?

4D Web 2.0 Packの一部である4D Live Windowは4Dのプラグインで、4Dフォームにインターネットブラウザを組み込むことを可能にします。これにより4DデータベースにWebベースのコンテンツを追加できるようになり、フォームにリッチで最新の情報を表示できるようになります。

4D Live Windowにより、4Dデベロッパはデータベース中のレコードを使用して、WebページのURLを組み立て、アクセスすることができます。詳しくは後述しますが、例として顧客の住所を使用して、Google Mapに所在地表示をさせることが考えられます。4D Live Windowを使用すれば、4Dフォームの顧客レコードとGoogle Mapを同居させることができます。関連する情報を表示させるためにアプリケーションを切り替えるのではなく、4DにWebブラウザ機能を持たせることでより効率的に物事を行えるようになるのです。

4D Live WindowのMashupを使用する別の例は、データベースの発送データから発送伝票番号を取り出し、リアルタイムで配送状況を確認することです。常に最新の情報を使用する方法を、4Dデベロッパは手に入れることになります。

4D Live Window Methods

現在4D Live Windowプラグインには6つのコマンドがあります:

Web_Back
Web_Forward
Web_SetPreferences
Web_GetURL
Web_GetContent
Web_SetURL

このテクニカルノートでは、この中で**Web_SetURL**コマンドにフォーカスをあてることにします。(このドキュメントの最後に**Command List Reference**がありますので、そちらも参照してください。)

Web_SetURL

Web_SetURLが最も重要なコマンドです。これを使用して4D Live WindowプラグインエリアにURLで取得したコンテンツをロードします。デベロッパは通常のURLだけでなく、4Dデータベースから構築したURLを使用して、コンテンツをロードすることができます。

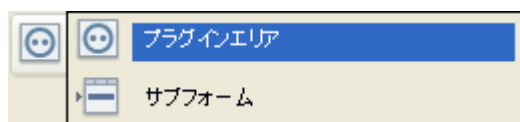
4D Live Window Examples

このセクションでは、4D Live Windowsを使用した例題を紹介します。添付されたサンプルデータベースもあわせてご覧ください。

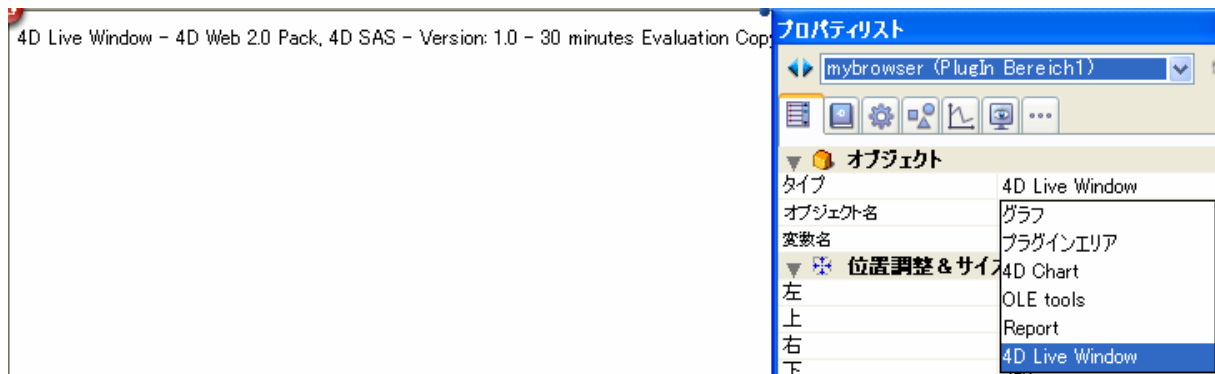
最初の例はGoogle Mapです。データベースの顧客住所情報を使用してURLを構築し、コンテンツをロードする方法を紹介します。

Getting Started

4D Live Windowプラグインエリアを4Dフォームについてする方法は、他の4Dプラグインと同じです。まず'4D Live Window.bundle' プラグインが適切な場所にインストールされていることを確認します。例題データベースではストラクチャと同じ階層の**PlugIns**フォルダに置かれています。データベースを開いたら、デザインモードのフォームエディタを開きます。下図に示すとおり、プラグインエリアツールを選択します:



では4D Live Windowエリアをフォームに置きましょう。プロパティリストでプラグインエリアのタイプを'4D Live Window'に設定します:



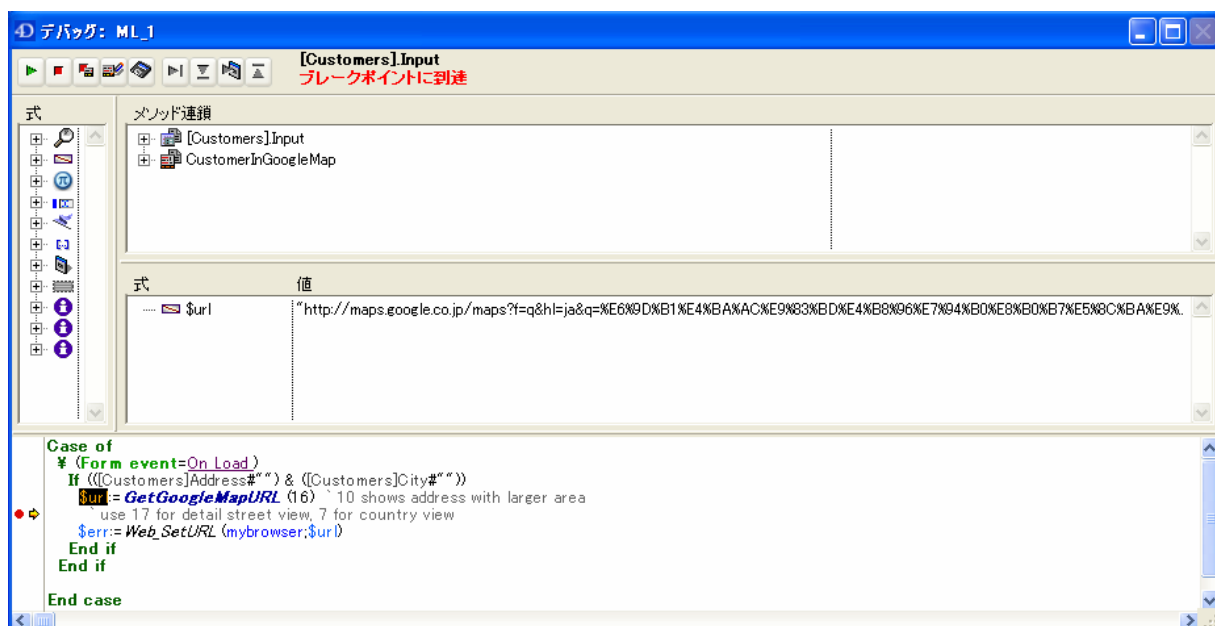
次に4D Live Windowエリアを動作させるためにコードを追加します。

Example 1: Google Maps

Web_SetURL コマンドを使用して4D Live Windowにコンテンツをロードすることができるようになっています。サンプルデータベースでは、**WebSetURL**はOn Loadフォームイベントで呼ばれます。“On Load”イベントは4D live Windowにコンテンツをロードさせる典型的な場所です。その時点で、URLの準備ができていなければなりません。この例題では、URLにデータベースから取り出した住所情報が加えられ、Google Map検索用にフォーマットされています。“On Load”フォームイベントのコードは以下のとおりです：

```
Case of
  ¥(Form event=OnLoad)
    If([(Customer]Address#"")&[(Customers]City#""))
      $url:=GetGoogleMapsURL(10)
      $err:=WebSetURL(mybrowser;$url)
    End if
End case
```

下の4Dデバッガウィンドウには、**WebSetURL**がコールされる時点での\$urlの値を示しています：



URLはGoogle Mapサーバ (maps.google.comやmaps.google.co.jp)と、顧客の住所情報、単語と単語を結ぶ“+”記号、そしてGoogle Map URLを有効なものにするための特別な文字列が含まれています。

データベースから取り出したデータを正しいURLにフォーマットする際には **GetGoogleMapsURL** を使用しています。**GetGoogleMapsURL** はドメイン情報やデータベースのデータから正しいURLを構築するプロジェクトメソッドです。**GetGoogleMapsURL** の中身を見てみましょう:

```
If (Count parameters>=1) ` zoom value
    $z:=$1 ` 10 shows address with larger area, 17 for detail street view, 7 for country view
Else
    $z:=15
End if

$codeLang:=Get indexed string(<>StrLang;39) ` fr, de ou en...
If ($codeLang="")
    $codeLang:="en"
End if

$URL:="http://maps.google.de/maps?f=q&"
$URL:=$URL+"hl="+$codeLang+"&"

$address:=[Customers]Address+", "+[Customers]City+", "
If ([Customers]State# "")
    $address:=$address+[Customers]State
End if

$address:=$address+", "+[Customers]Country
$address:=Replace string($address, " ", "+")

$URL:=$URL+"q="+$address+"&ie=UTF8&z="+String($z) ` zoom
$URL:=$URL+"&iwloc=addr"

$0:=$URL
```

GetGoogleMapsURL メソッドはデータベースの顧客住所データを使用して Google Maps URL を構築します。以下の画像は Google Map による検索結果が表示された 4D フォーム上の 4D Live Window プラグインエリアです:



結果、4D Live Windowを使用するための擬似コードは、2つの手続きからなることになります:

1. データベースのデータをURLで利用可能なようにフォーマットする
2. 結果URLを使用して**WebSetURL**をコールする

これは前述のGoogle Mapの例題で使用されたテクニックです。そしてこれから説明する例題も同じテクニックを使用することになります。

Example 2: Fedex Tracking Information

Google Mapsの例題とまったく同じコンセプトを次の例題、データベースの問い合わせ番号を使用してFedexのWebサイトで荷物を追跡する例題に適用することができます。

Google Mapの例題と同様、Fedex URL情報にデータベースのデータを埋め込み、フォーマットされたURLを返すプロジェクトメソッドを使用します。このメソッドは**GetFedexURL**という名前で、コードは以下のとおりです:

```
$URL:="http://fedex.com/Tracking?ascend_header=1&clienttype=dotcom&cntry_code=us&language=" + "e=english&tracknumbers="
$address:=[Inventory]TrackingNo
$URL:=$URL+$address
$O:=$URL
```

データベースのデータをFedexのURLにリンクするのは簡単なので、**GetFedexURL**は短いものになっています。必要なことは問い合わせ番号をFedex URLの最後に加えることです。

GetFedexURLから返された有効なURLを使用して**WebSetURL**をコールした結果は以下のようになります:

The screenshot shows the FedEx Tracking website interface. At the top, there are navigation links for "US Home", "Español", "Information Center", "Customer Support", and "Site Map". A search bar is present with the text "Search" and a "Go!" button. Below the search bar, there are tabs for "Package / Envelope Services", "Office / Print Services", "Freight Services", and "Expedited Services". Under "Package / Envelope Services", there are sub-tabs for "Ship", "Track", "Manage My Account", and "International Tools". The "Track" tab is selected. The main content area displays a message: "More than one shipment was found with this tracking number." Below this message, it says "Please select the correct shipment and click Resubmit". A table of tracking results is shown with columns: Tracking number, Ship date, Destination, and Service. The first row shows tracking number "999999999999", ship date "Oct 27, 2006", destination, and service "FedEx Ground".

Tracking number	Ship date	Destination	Service
999999999999	Oct 27, 2006		FedEx Ground

Example 3: UPC Lookup

品物の在庫管理において、バーコードで品物の管理をすることができれば効率的です。次の例題ではUPCDatabase.comをバーコードを使用して検索します。URLを構築するGetUpcURLプロジェクトメソッドは以下のとおりです:

```
$URL:="http://www.upcdatabase.com/item.asp?upc="
$UPC:=[Inventory]UPC
$URL:=$URL+$UPC

$0:=$URL
```

Fedexの例のように、URLを構築するメソッドは非常に単純です。バーコード番号がUPCdatabase.com URLの最後に付加されます。

Web_SetURLを実行した結果は以下のようになります:

The screenshot shows a web application titled "UPC Code Lookup". At the top, there are navigation buttons (back, forward, etc.) and status icons (error, warning, success). The main form has two input fields: "Item Name" with the value "The Godfather DVD" and "UPC" with the value "097361564746". Below the form, there is a section for "Item Record" which displays a barcode, the UPC-A number "0 97361 56474 6", and the following details: "The Godfather Collection", "5 DVD", "PARAMOUNT HOME VIDEO (0097361) more items", "United States", "Sun, 14 Mar 2004 18:22:03 UTC", and "0" under "Pending Updates". On the left side of the application, there is a login section with fields for "Username:" and "Password:", a "Login" button, and links for "Join" and "Forgot Password". At the bottom left, there is a "Look Up" button. On the right side, there are advertisements for "Barcode .NET Controls" and "Barcode Search".

Conclusion

4D Live Windowを使用してMashupを作成することは、通常のURL文字列を使用してWebサイトを呼び出すことであり、非常に簡単な作業です。きちんと定められた定義があれば、データベースのデータをそれに埋め込むだけです。このテクニカルノートの実例では、4Dデータベースのウィンドウに4D Live Windowエリアを組み込む、コンセプト実証のための実際の方法をお見せしました。デベロッパーは4D Live Windowプラグインを使用して、Mashupの世界においてより創造的で革新的な方法を発見することができます。

Command List Reference

Web_SetURL

Web_SetURL (Area; URL; MacOSPath) → error code

Parameter	Type		Description
Area	Longint	→	4D Live Windowエリア
URL	Text	→	フルURL (http://www.4d.com/)
MacOSPath	Longint	→	0: URL 1: ローカルファイルパス Windowsでは使用しない
Function Result	Longint	←	エラーコード (0 = エラーなし)

Web_SetURLはエリアに対して通常最初に発行するコマンドです。開くURLやドキュメントを指定します。URLは完全に有効なURLでなければなりません。(i.e. <http://www.4D.com> or <file:///C:/my%20Document.pdf>)

Mac OS上では、ファイル名は有効なUnixパス名でなければなりません:

<file:///localhost/Users/thomas/Documents/image%C3.jpg>.

2バイト文字や8-bit文字、空白のような特別文字はエンコードしなければなりません。無効な文字はクラッシュの原因となります。

MacOSPath引数は自動でファイルパスをエンコードします。

"MyDisk:Users:thomas:Documents:image.jpg"のようなHFSパスを使用するには、1を渡します。

Windowsでは"C:¥my Document.pdf"や"Straße.jpg"のような、エンコードなしの実際のファイル名を受け入れます。また"www.4D.com"のような単純なURLも利用可能です。ただし完全なパスを指定することをお勧めします。

Function result

コマンドの実行に成功すると0を返します。0以外の値が返った場合はエラーがあったことを示します。

Web_GetURL

Web_GetURL (Area; URL) → error code

Parameter	Type		Description
Area	Longint	→	4D Live Windowエリア
URL	Text	→	現在のURL
Function Result	Longint	←	Error code (0 = エラーなし)

Web_GetURLコマンドは現在表示されているURLを返します。

Function result

コマンドの実行に成功すると0を返します。0以外の値が返った場合はエラーがあったことを示します。

Web_Back

Web_Back (Area) → error code

Parameter	Type		Description
Area	Longint	→	4D Live Windowエリア
Function Result	Longint	←	エラーコード (0 = エラーなし)

Web_Backコマンドは、ブラウザの「戻る」機能呼び出し、前に表示されていたURLをロードします。

Function result

コマンドの実行に成功すると0を返します。0以外の値が返った場合はエラーがあったことを示します。

Web_Forward

Web_Forward (Area) → error code

Parameter	Type		Description
Area	Longint	→	4D Live Windowエリア
Function Result	Longint	←	エラーコード (0 = エラーなし)

Web_Forwardコマンドは、ブラウザの「進む」機能呼び出します。「戻る」機能が呼び出されていない場合、このコマンドは何も行いません。

Function result

コマンドの実行に成功すると0を返します。0以外の値が返った場合はエラーがあったことを示します。

Web_GetContent

Web_GetContent (Area; content) → error code

Parameter	Type		Description
Area	Longint	→	4D Live Windowエリア
Content	Blob	→	ページの内容
Function Result	Longint	←	エラーコード (0 = エラーなし)

Web_GetContentコマンドはメインフレーム中のコンテンツをBlobに返します。

Function result

コマンドの実行に成功すると0を返します。0以外の値が返った場合はエラーがあったことを示します。

Web_SetPreferences

Web_SetPreferences (Area; Selector; NumValue, StringValue) → error code

Parameter	Type		Description
Area	Longint	→	4D Live Windowエリア

Selector	Longint	→	環境設定セクタ
NumValue	Longint	→	設定する数値
StringValue	Alpha	→	設定する文字列値
Function Result	Longint	←	エラーコード (0 = エラーなし)

Web_SetPreferencesコマンドを使用して4D Live Windowの動作を変更することができます。

Release 1.0では4番目の引数StringValueは使用されていません。将来の互換性のため常に""を渡してください。

Notes:

(1) Web_kResize Mac OSのみ: プラグインエリアがリサイズされると、いくつかのWebサイト(特にmaps.google.com)において、JavaScriptを使用していくつかの要素の再配置が行われます。これが原因でクラッシュが発生することがあります。この現象を避けるには、ブラウザエリアをリサイズ不可に設定してください。

(2) いかなる場合でも、(ウィンドウが開かれてからWebサイトが読み込まれる前に) それが始めに表示されるときを除き、Mac上でブラウザエリアは自動でリサイズされません。JavaScriptなしのHTMLやPDFファイルなどを表示するときは、このオプションを使用したリサイズ機能が使用できるかもしれません。そのような場合、あなたはエンドユーザに対しURLの入力を禁止すべきです。このオプションはWindowsでは効果ありません。

(3) Web_kVisible 4DのSET VISIBLEコマンドはブラウザエリアを完全には非表示にしません。なぜならブラウザはマウスオーバーやJavaScriptなど再描画を起こすイベントに直接応答するからです。そのためブラウザエリアの表示/非表示を切り替えるにはSET VISIBLEとWeb_SetPreferences両方を使用する必要があります。

Function result

コマンドの実行に成功すると0を返します。0以外の値が返った場合はエラーがあったことを示します。

Example

```
$err:=Web_SetPreferences(mybrowser; 1; 1; "") ` to enable resizing
```

Error Codes

4D Live Windowプラグインのすべてのコマンドは、実行に成功すると0を返します。

返される可能性のあるエラーコードは以下のとおりです:

- 15001 指定されたエリアは4D Live Windowエリアではありません。
- 15002 無効な引数が渡されました。
- 15003 内部エラー