11

In This Chapter

Rules

Templates

Pushing Information

Linked Pictures

External Table

Internet Information Services (IIS)

Scripting

GM+View, was new way back in GoldMine 6 BCM, and allows users to view HTML documents directly within GoldMine. GM+Browser was new to GoldMine 6.7, and basically does the same as GM+View, however, in a detached Window, if in Windowed mode, or as an independent Tab, if in the Tabbed mode. There are many, many different presentations that one could define for a given GoldMine installation, and the choice is always yours. You might want to display a location map, a contact resume, the weather at the contacts location, or any of a myriad of other possibilities. One of the more popular GM+Views/GM+Browser is to display linked data from an external table, and I will show you how to accomplish that using GoldMines own **ContHist** table as my external table. However, you can apply this same technique to any external table for which you have an ODBC or OLE connection capability. Just wait, you'll see.

I have supplied you with many different templates that you can just attach to your GM+View/GM+Browser that should work just fine with little or no modification. The zip file containing these is embedded in the zip file that you downloaded when you downloaded the eBook version of this book. If you purchased this book through a print on demand service, you must e-mail me a copy of the **Shipping Order** included within your book shipment. You will then receive an e-mail back from me allowing you to download the eBook version with all of these templates. Buy one, get two. That e-mail address would be: **DJ@DJHunt.US**. Please make sure that the **Subject:** of the e-mail message is **eBook Request**, and I will get them right out to you.

I'll discuss the implementation for most of the templates later in this chapter, however, there are many where the implementation is identical to a previously discussed template, and I will not be discussing these as the process for attaching and utilizing the template will be identical. It will be up to you to make the rest of the templates works.

One thing that I want to discuss with you right off the bat is the **GM+Browser** which, in my opinion, is a great bonus. The GM+Browser is a free-floating browser that defaults the display to the contacts website, however, you can easily add any of your GM+Views to this browser view. Figure 11-1 shows

Figure 11-1

how one could display specific history items for the active GoldMine Contact record in its own screen. Take a look. You'll see this same **Contract History**

GM+View later in this chapter. As a bonus, however, if using the windowed mode for GoldMine, you can leave the GM+Browser window open in your GoldMine frame displaying the current history activity with notes if you would like. Very nice indeed, although you may have to work around the current refresh issues of the GM+Browser.

First, let's discuss some of the configuration features that are behind GM+Views. From the GoldMine menu, click on **Web** | **Setup GM+View...** to bring up the dialog form shown here in Figure 11-2.

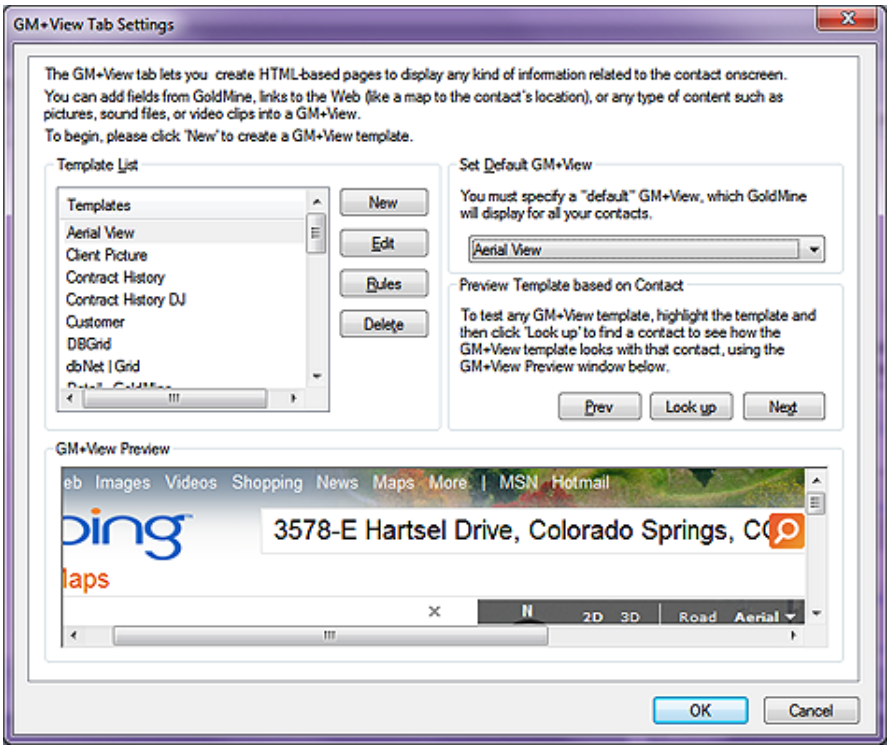

Figure 11-2

Naturally, yours may not look exactly like this one if your GoldMine is not a new installation. For instance, your new GoldMine installation might not have the **Car**, and the **Property** templates displayed, whereas this is a screenshot from my own GoldMine. If you do, however, have these templates, I would immediately ask you to **Delete** the **Car**, and the **Property** templates so that we may begin our discussion of GM+Views with you having a clean slate on your GoldMine. On the other hand, maybe you would want to keep those for a while as they are also good examples for you to fall back on for information.

Rules

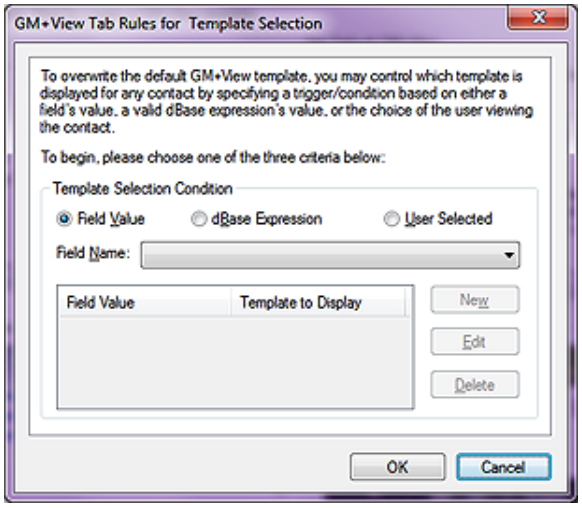

Figure 11-3a

Let's first begin by examining the **Rules** that one may set up. Click on the Rules button to bring forth the dialog forms shown in Figures 11-3a, 11-3b and 11-3c.

You will notice, here in Figure 11-3a, that the first radio button, ◉ **Field Value**, allows you to default display one, and only one, GM+View per Contact record based upon a value contained in any Contact1 or Contact2 table field. The different GM+Views must be pre developed before you can set this rule into action. The activity of creating a rule here is pretty self-explanatory, however, I will take a little time to go over the basics.

You must first select a **Field Name:** from the available fields on the drop list, which as we already know, is all of the fields contained in the Contact1/Contact2 tables. This is the field that must contain the value that will instruct GoldMine as to which GM+View to display when a certain value is contained within this field. You may have read about this earlier in this book. It was called Record Typing.

You would then click on the **New** button, and associate a **Field Value** with a GM+View (**Template to Display**). That's all there is to it. Naturally, you could **Edit** the entry, or **Delete** the entry if the opportunity arises.

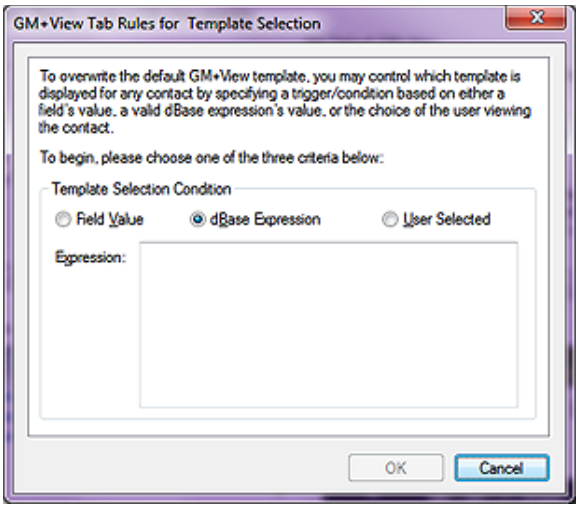

Figure 11-3b

In Figure 11-3b, you can see that an **Expression:** field is exposed when one selects the O **dBase Expression** radio button. It is possible to use nested **iif()** functions (refer to Appendix A), as I described for you in Chapter 4 when we were coloring fields or changing label colors. Again, this type of rule is limiting. You can only display one GM+View per Contact record at a time. This goes back to my discussion on Record Typing that I presented to you in Chapter 4.

An example of an expression that might be utilized in this area would be something on the order of:

```
iif(upper(trim(Contact1.Key1))==[GM DEALER], [Yahoo Map], [Yahoo Weather])
```

I feel that there is too much value under the **GM+View** tab to limit it to just one view per contact record. I advocate creating a number of views, and allowing them to be O **User Selected** by selecting this radio button, as shown in Figure 11-3c. That's all that is required under **GM+View Tab Rules for Template Selection**. Now the user will be able to right-click under the GM+View tab, and select from any of the developed html documents.

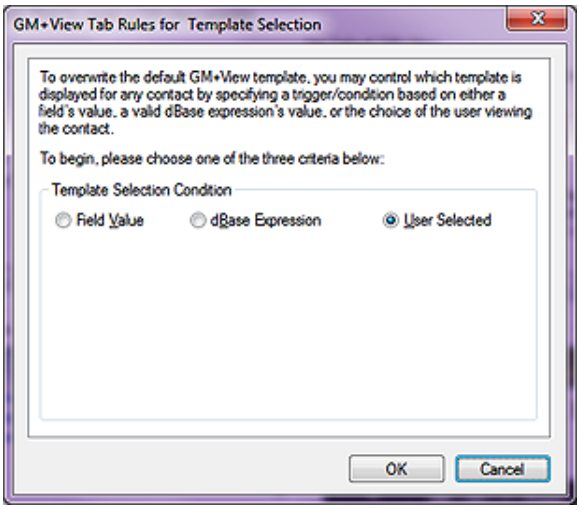

Figure 11-3c

Templates

I have pre developed a handful of GM+Views for you to use, and zipped them into **GM+View HTML.zip** which are included in the eBook download. If you have purchased a Print on Demand or eBook copy of this book, then you may send an e-mail to: **DJ@DJHunt.US** with a copy of your receipt. Please make sure that your **Subject:** is **eBook Request**. I will then send you the download information for the eBook, and all of the templates discussed in this book.

Note

*There are several templates included in **GM+View HTML.zip** that are not discussed in this chapter, however, I highly recommend that you examine them, as they may be useful for your organization. They are all added to the **GM+View** in the same manner as will be discussed in this chapter. Most require no modification.*

Note

*Whenever I talk about editing the HTML code of a view, I mean to do so in anything other than the GoldMine HTML Editor. Even the lowly **NotePad** application is better for editing HTML than is the GoldMine HTML Editor.*

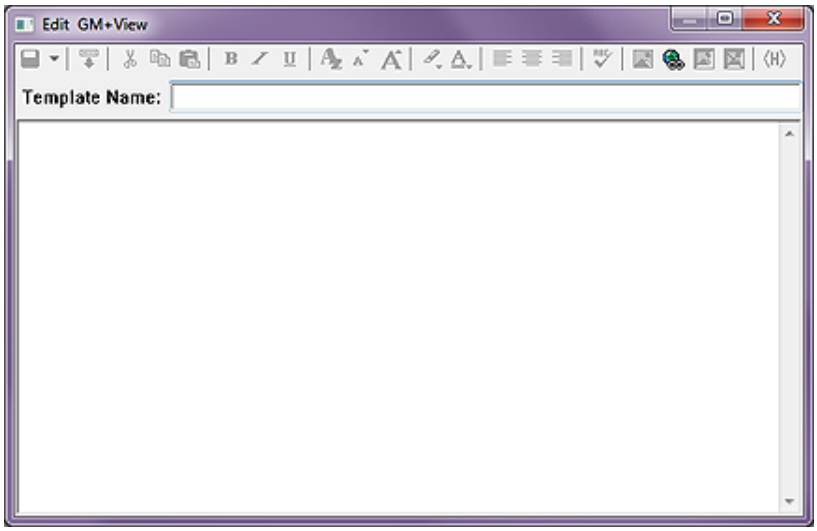

Figure 11-4

I am now going to discuss creating GM+View templates using these examples. For my first creation, I will be displaying a Word document file that exists on the world wide web. For this example I'll display a resume.

First I would ask you to click on the **New** button, as was previously shown to you in Figure 11-2, to bring up the standard GoldMine GM+View Editor as shown on the previous page in Figure 11-4.

Later in this chapter, I will give you one example of creating a GM+View template, within the GoldMine Editor, to display linked pictures. It is my practice to create all of my views in Adobe Dreamweaver, and then to include them into my GoldMine GM+View template. I will discuss that process now, which will be the same for all of the document templates that I have supplied to you. When necessary, I will discuss changes that need to be made to the HTML code in my supplied views for them to function best for you.

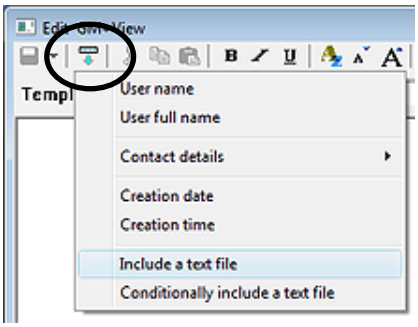

Figure 11-5

Here in Figure 11-5, you will notice that I have highlighted an icon. This icon will become available to you after you have moved your cursor to the body area. Memorize this icon as we will be using it often, as well as the **Include a text file** option that clicking upon it exposes. As I explained in the Tip, you can also use the same technique when creating your E-Mail Templates. This option, under this icon, should be your most utilized tool.

■ Displaying the same document for all contacts

In Figure 11-4 on the previous page, you can see a field called **Template Name:** which is where one would enter, you guessed it, the name of the template. For this one, I ask you to enter **Display Document**, and then tab to the body area. This is the name that will appear in the GM+View list. I will give you a warning, you are limited to 19 characters in this field, and you want to make sure that each GM+View has a unique name.

Notice that, once in the body, certain icons become enabled. I am only interested in one, the second icon from the left, Figure 11-5 (encircled), which, when you hover over it, has a pop up ToolTip that states: **Insert Field**. If you click on this icon, you will be presented with a local menu from which you will want to select: **Include a text file**. This action immediately instantiates an **Open** dialog form from which you would be asked to find: **Display DOC.html**, which I hope that you unzipped to an HTML folder under your GoldMine folder.

In my setup this action inserted:

```
<<file:Z:\Data\Book The Definitive Guide to GoldMine Premium\HTML Docs\Display DOC.html>>
```

It can't get any simpler than that. Now close this template, and save it. That's all, however, I would warn you to read the sidebar which applies to displaying documents or pdfs. When you save this type of template, you may be asked to **Open** the document. Don't. Just **Cancel** that.

■ Displaying a different document for each contact

Should you want to have a different document per contact, then I would pull the relevant string out of my HTML document, and stuff it into a GoldMine **Contact2** field. I used **UserDef01**, and expanded the **UserDef01** field length to 250 characters for this example.

Here is the relevant statement taken from the template:

```
http://www.DJHunt.US/DWSite/eBook/DJ%20Hunt%20Resume.doc
```

I have pasted this string into the **UserDef01** field that I'm displaying on my screen. It is important that you have the string exactly as displayed only inserting your path, and document name with extension. Notice that I am storing my document, in this case, on my website.

I now create a **New** template in GM+Views with a new **Template Name:**, maybe **Display Doc/Contact**. This time, when I enter the body, I simply type click on the **<H>** icon and I paste this code from my HTML Editor:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

        <head>
```

Tip

*This same logic applies to **E-mail Templates** as it does to the **GM+View Templates**. I never use the **GoldMine HTML Editor <H>** for creating my templates. I always create my templates with a real HTML Editor, and simply **Include a text file** in the body of the GoldMine E-mail Template. **Do Not** even consider a **Copy/Paste** from your editor to the GoldMine editor as an option unless you wish to do this on every edit of this template.*

Note

Don't be afraid to look at the coding of the HTML documents that I have supplied you. Use your favorite HTML Editor, Microsoft Expressions, Adobe Dreamweaver, or even Windows Notepad to view the underlying code. You'll be able to create your own documents in the same manner, and then point to them from within GoldMine.

WARNING

If you are using Office 2007, I have noticed that the Internet Plug-in for Word is not being installed, hence, my document, instead of appearing under the GM+View tab, is opening up directly into Microsoft Word 2007. Read that as very slow.

Note

Please note that this could have been a local path as opposed to a URL, and it would have worked just as nicely.

Further, you could have had the Lookup.ini populate this using an expression, and trigger upon NewRecord creation if you wish.

Note

*Did you notice how nicely structured my document is thanks to my HTML Editor? Now compare that to that which you just say when you clicked on the **<H>** icon to get into the GoldMine HTML Editor.*

Oh, wait! There is no real comparison now, is there?

Note

Specifically notice the highlighted text which defines the GoldMine field which contains the variable data per Contact record.

Note

*This structured code was taken from-**Display DOC_2.html** as distributed in the template collection for this book.*

WARNING

If your system is configured to display the file in the Internet Explorer window as opposed to launching the document application, there is a little flaw in the system, one must click off of the GM+View tab when changing records, and then back on to refresh the record. It does work, and that is what counts.

Note

Statement is wrapped for presentation only, and would be one continuous statement within the Lookup.ini.

```
<!--
  This GM+View Template is provided as part of GoldMine Premium - The Definitive Guide
  series of books, and the author is providing this without guarantee or additional support.
-->

        <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
        <title>GM+View Document</title>

        <SCRIPT language=JavaScript>

                <!--
                function redirect()
                {
                 window.location = "&lt;&lt;trim(Contact2.UserDef01)&gt;&gt;"
                }

                setTimeout("redirect();", 0000)
                // -->

        </SCRIPT>

</head>

<body>
</body>
</html>
```

I then save this template. Give it a try. Again this type of view works best if there is but one view per contact record.

I would like to add that this would be an excellent use for the Lookup.ini. You could update the UserDef01 field string with the Contact name or Company name, as you choose, to repoint to a different document for each contact record. Here is an example of how that specific Lookup.ini code might look:

```
[AutoUpdate]
  NewRecord = UserDef01

[UserDef01]
  Otherwise = &" http://www.DJHunt.US/DWSite/eBook/" + &FirstName + "%20" + &LastName
              + "%20Resume.doc"
```

This could prove to be extremely useful as you could have an entire book per contact record by following the steps as described in ■ **Displaying a different document for each contact.** You might ask, "*Why would this be useful?*". If you had ever used **Notes** to store historical information for a contact, and found that you were losing notes when you exceeded the limit, you wouldn't be asking that question. This is a space to store notes that would be virtually unlimited. When combining this with the Lookup.ini, well, you can just imagine all of the possibilities.

■ Displaying the same pdf for all contacts

The next template that I would like you to try is the **Display PDF.html**. For this template, I simply follow the same steps as discussed previously for ■ **Displaying the same document for all contacts**. This will only work properly, within GM+Views/GM+Browser, if the users have the Adobe PDF plug-in for Internet Explorer 8 installed on their system. I gave mine a **Template Name:** of **Display PDF**. Where as, I used to state: "*Refer to the sidebar warning as it is applicable for a PDF view as well as the DOC views when the plug-in is installed.*", in GoldMine Premium, I no longer need to make this statement. Although you may still right-click to produce your list of available GM+Views, and this is the component that used to conflict with the PDF plug-in, the developers have now added another navigation utility, the drop list in the GM+Views header bar as highlighted in Figure 11-6.

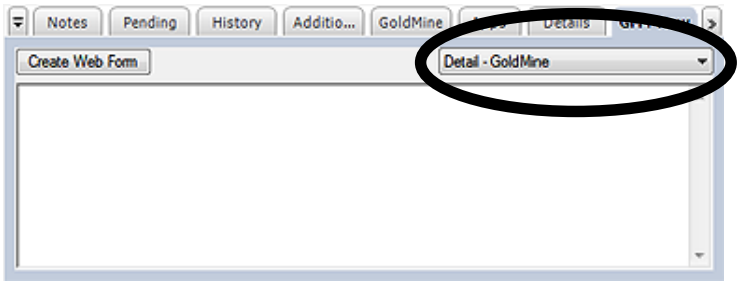

Figure 11-6

Oh, the headaches that this saves.

■ Displaying a different pdf for each contact

Well, if it holds true for Documents then it should hold true for PDFs as well, and it does. Should you want to have a different document per contact, then I would pull the relevant string out of my HTML document, and stuff it into a GoldMine Contact2 field. I used the UserDef01 field again, as its length has already been expanded to 250 characters, for this example.

Here is the relevant statement taken from the template:

```
http://www.DJHunt.US/DWSite/eBook/DJ%20Hunt%20Resume.pdf
```

I now create a **New** template in GM+Views with a new **Template Name:**, maybe **Display PDF/Contact**. This time, when I enter the body, I simply type click on the **<H>** icon and I paste this code from my HTML Editor:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

        <head>

        <!--
          This GM+View Template is provided as part of GoldMine Premium - The Definitive Guide
          series of books, and the author is providing this without guarentee or additional support.
        -->

                <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
                <title>GM+View Document</title>

                <SCRIPT language=JavaScript>

                        <!--
                        function redirect()
                        {
                         window.location = "&lt;&lt;trim(Contact2.UserDef01)&gt;&gt;"
                        }

                        setTimeout("redirect();", 0000)
                        // -->

                </SCRIPT>

        </head>

        <body>
        </body>
</html>
```

I then save this template as before, and give it a try. Refer to all of the previous WARNINGs and Notes governing DOCs. Again this type of view works best if there is but one view per contact record.

I would like to reiterate that this would be an excellent use for the Lookup.ini. You could update the UserDef01 field string with the Contact name or Company name, as you choose, to repoint to a different document for each contact record. See my previous DOC example.

■ Displaying the Primary Website for your contact record

The next template that I would like you to try is the **Display Web.html**. For this template, I will again ask you to follow the same steps as discussed on the previous page for **■ Displaying the same document for all contacts**. There now, wasn't this simple?

■ Displaying Google Driving Directions to the contact location from a fixed location

The next template that we are going to look at is the **Google Directions.html**. For this template I have decided to utilize the registered license information that GoldMine maintains as my starting point. Later, under MapQuest directions, you will see that I have hard coded my starting point. We, again, follow the same steps as discussed previously for **■ Displaying the same document for all contacts**. Here is the driving code section from this template:

```
window.location = "http://maps.google.com/maps?q=&lt;&lt;&amp;licinfo_address1&gt;&gt;%20&lt;&lt;&amp;licinfo_zip&gt;&gt;%20to%20&lt;&lt;&amp;address1&gt;&gt;%20&lt;&lt;&amp;zip&gt;&gt;"
```

You will notice that I have highlighted the code in this statement where the GoldMine registered information is being taken. With Google, we only need to provide minimal information to acquire

Note

***HTML** = Hypertext Markup Language*

Note

I can't emphasize enough the need to test your GM+Views before deploying they. I will have spent well over 10 hours testing all of the GM+Views templates that I have created for this book, however, that does not relieve you of your responsibility to test them within your environment.

WebSites, and their parameters are constantly in flux. You should validate your GM+Views regularly.

maximum results. Very slick, and maybe the reason many prefer Google over the MapQuest solution that I will provide to you shortly.

■ Displaying a Google Location Map

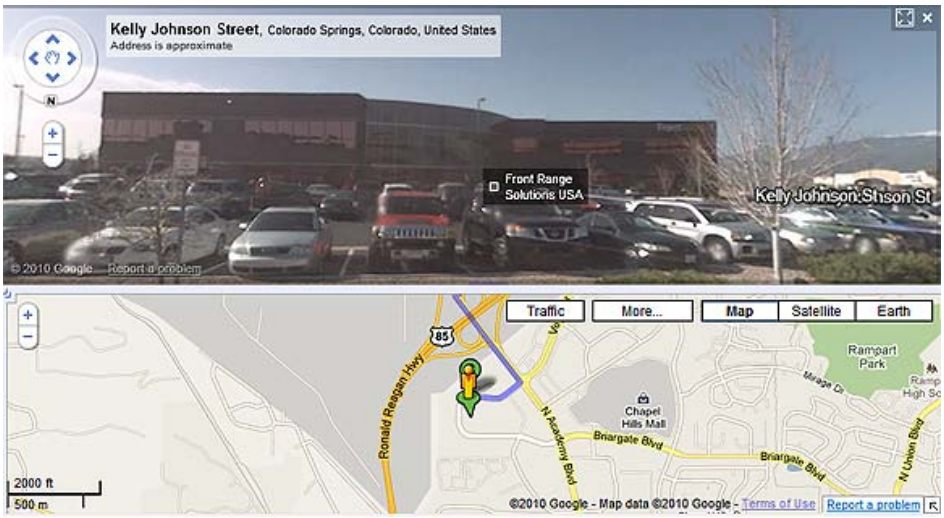

Figure 11-7

In Figure 11-7, I thought that I would show you a screenshot of what one of these templates could produce for you in either GM+View or GM+Browser. Each organization has their own unique needs, and these templates may not be of any use to you at all. I have, however, tested them all, and, at the time of my testing they were working nicely. For **Google Location.html** the control code is:

```
window.location = "http://maps.google.com/maps?oi=map&amp;q=&lt;&lt;&amp;Address1&gt;&gt;%20
&lt;&lt;&amp;City&gt;&gt;%20&lt;&lt;&amp;State&gt;&gt;%20&lt;&lt;&amp;Zip&gt;&gt;"
```

You can see, highlighted, that we are pushing the GoldMine Contact record **Contact1.Address1**, **Contact1.City**, **Contact1.State**, and **Contact1.Zip** to Google for its resolution into a location map.

■ Displaying a Google Search

I know that you have used this Google feature thousands of times in the past. The Search Engine was where Google got its beginnings. From this code:

```
window.location = "http://www.google.com/search?as_q=&num=100&hl=en&ie=UTF-8&oe=UTF-
8&btnG=Google+Search&as_epq=&lt;&lt;&Company&gt;&gt;&as_oq=&as_eq=&lr=&as_ft=i&as_
filetype=&as_qdr=all&as_occt=any&as_dt=i&as_sitesearch=&safe=images"
```

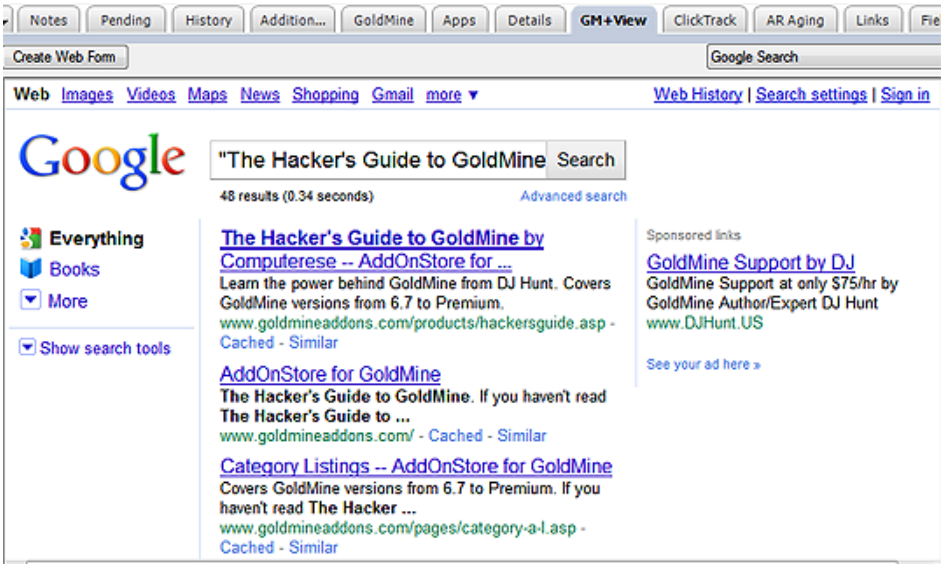

Figure 11-8

You supply only the Contact record Company information to the Google Search Engine, and your GM+View becomes your new active Search Engine, Figure 11-8.

■ Displaying a Google Group Search

This is much the same as a Google Search, only this time we are searching with slightly different criterion, and using a slightly different part of the Google Search Engine. The primary code is:

```
window.location = "http://groups.google.com/groups/search?lr=&safe=off&num=10&q=&lt;&lt;&amp;
Company&gt;&gt;+&lt;&lt;&amp;City&gt;&gt;&safe=off&qt_s=Search"
```

■ Displaying MapQuest Driving Directions - Fixed Starting Location

The next template that I would like you to try is the **MapQuest Directions.html**. For this template, we again follow the same steps as discussed previously for ■ **Displaying the same document for all contacts**, however, you will need to modify the MapQuest Directions.html document slightly. That is unless you want to know how to get from my location to your clients location. Open MapQuest Directions.html in your favorite HTML editor (**not** Microsoft Word or Publisher). The line that you would be looking for is:

```
window.location = "http://www.mapquest.com/directions/main.adp?go=1&1a=150+PRATT+RD
&1c=FITCHBURG&1s=MA&1z=01420-4142&2a=&lt;&lt;&amp;&Address1&gt;&gt;&amp;&2c=&lt;&lt;
&amp;&City&gt;&gt;&amp;&2s=&lt;&lt;&amp;&State&gt;&gt;&amp;&2z=&lt;&lt;&amp;&Zip&gt;&gt;"
```

It is important to note that the above is a single line of text. I have highlighted the items that you will need to modify before saving the document, however, I would like to point out some things. The plus sign (**+**) is the equivalent of a space between words for MapQuest. There should not be spaces, carriage returns or line feeds in this statement anywhere. Thus, my information, in the above statement, which is:

```
150 PRATT ROAD
FITCHBURG
MA
01420-4142
```

...is represented as:

```
150+PRATT+ROAD
FITCHBURG
MA
01420-4142
```

Of course, you would replace my addressing information with your own MapQuest Starting location. Being in the United States, I have not had the opportunity to test this in other countries, however, I suspect that you would certainly need to add the country code to this script if you were in a different country. There are other services available for other countries, and you may be able to use this code as the basis for setting up your own script. After you've made your changes, save them, and then test this document in your GM+Views.

Note

*To **Copy**, and **Paste** location maps in GoldMine E-mail, you must be have to Option selected to **Use HTML when creating E-mail**.*

■ Displaying a MapQuest Location Map

The HTML document that you will use for this one is **MapQuest Location.html**. Again, no changes should be required for this document to produce the desired results. Just follow the steps that I discussed previously under ■ **Displaying the same document for all contacts**. I gave this template a **Template Name:** of **MapQuest Local Map**. This is rather neat in a way. Under the GM+View tab, with this view selected, I click on the Printer-Friendly hotlink. Then I right-click on the results, choosing **Select All**, and then right-clicking again, choosing **Copy** from the local menu. I can then **Paste** this map into an E-mail to be sent to a representative or anyone requiring this information. Additionally, as of today anyway, I could display this as a Street map, Aerial map, I could show Gas Stations, Parking Garages, and more.

■ Displaying Yahoo Driving Directions

So as not to show favoritism, I have included some Yahoo GM+Views as well. The first of which is the HTML document that you will use for this exercise is **Yahoo Directions.html**. No changes should be required for this document to produce the desired results if your starting point is to be the same point as indicated in your GoldMine Registration information. You may want to use hard coded starting locations, and this should be an easy modification for you at this point. Make sure that you utilize a good HTML Editor, and that you **do not** use Microsoft Word or Microsoft Publisher. Just follow the steps that I discussed previously under ■ **Displaying the same document for all contacts**. I gave this template a **Template Name:** of **Yahoo Directions** within my GoldMine.

The key script line in this document is:

```
window.location = "http://maps.yahoo.com/py/ddResults.py?Pyt=Tmap^&tarname=^&tardesc=^&newname=^&newdesc=^&newHash=^&newTHash=^&tlt=^&tln=^&slt=^&sln=^&newFL=Use+Address+Below^&newaddr=&lt;&lt;&LicInfo_Address1&gt;&gt;+^&newcsz=&lt;&lt;&LicInfo_City&gt;&gt;%2C+&lt;&lt;&LicInfo_State&gt;&gt;+&lt;&lt;&LicInfo_Zip&gt;&gt;+&newcountry=us^&newTFL=Use+Address+Below^&newtaddr=&lt;&lt;contact1-&gt;address1&gt;&gt;+^&tcsz=&lt;&lt;contact1-&gt;city&gt;&gt;%2C+-&lt;&lt;contact1-&gt;state&gt;&gt;+&lt;&lt;contact1-&gt;zip&gt;&gt;+&tcountry=us^&Submit=Get+Directions"
```

The words that are highlighted (all begin with &LicInfo for you printed edition readers) are the portions of the code that would need to be changed if you were to hard code the starting point. See the example that I previously discussed back under the ■ **Displaying MapQuest Driving Directions - Fixed Starting Location** section of this chapter.

■ Displaying Yahoo Location Map

The second Yahoo GM+View that I would like to present, and is the HTML document that you will use for this exercise is **Yahoo Location.html**. No changes should be required for this document to produce the desired results. I gave this template a **Template Name:** of **Yahoo Local Map** within my GoldMine.

The key script line in this document is:

```
window.location = "http://us.rd.yahoo.com/maps/us/insert/Tmap/extmap/*-http://maps.yahoo.com/maps_result?addr=&lt;&lt;&amp;address&gt;&gt;&amp;csz=&lt;&lt;&amp;CityStateZip&gt;&gt;&amp;country=us"
```

Those of you who are not located in the US, may want to change the country designation.

■ Displaying Yahoo Yellow Pages

Note

With the merger of Bing and Yahoo Searches, the Yahoo Yellow Pages script is pulling in aerial shots of this location. You may want to try this on your own to see what it will pull when you are ready to utilize it.

Remember, these sites are in constant flux, and you will have to modify your GM+Views occasionally.

The next Yahoo GM+View that I would like to present, and is the HTML document that you will use for this exercise is **Yahoo Yellow Pages.html**. No changes should be required for this document to produce the desired results. I gave this template a **Template Name:** of **Yahoo Yellow Pages** in my GoldMine.

The key script line in this document is:

```
window.location = "http://yp.yahoo.com/py/ypResults.py?stx=Savings+Bank&stp=a&tab=B2C&addr=&lt;&lt;&amp;Address1&gt;&gt;=&lt;&lt;&amp;City&gt;&gt;&state=&lt;&lt;&amp;State&gt;&gt;&zip=&lt;&lt;&amp;Zip&gt;&gt;&uzip=&lt;&lt;&amp;Zip&gt;&gt;&country=&lt;&lt;&amp;Country&gt;&gt;"
```

This script was pushing that we want to search for Savings Banks that are local to the Contacts address. As Country is a parameter, I suspect that this script should work nicely outside of the US as well as in the US.

■ Displaying Location Aerial Photo

The HTML document that you will use for this exercise is **Aerial Map.html**. No changes should be required for this document to produce the desired results. Just follow the steps that I discussed previously under ■ **Displaying the same document for all contacts**. I gave this template a **Template Name:** of **Aerial Photo** within my GoldMine.

Note

I would mention that the Hoover's template does require that you have a subscription to Hoover's as this is a paid service.

■ Displaying Hoovers Information

The HTML document that you will use for this exercise is **Hoovers.html**. No changes should be required for this document to produce the desired results. Just follow the steps that I discussed previously under ■ **Displaying the same document for all contacts**. I gave this template a **Template Name:** of **Display Hoovers** within my GoldMine.

■ Displaying the Weather for a Location

The HTML document that you will use for this one is **Local Weather.html**. Again, no changes should be required for this document to produce the desired results. Just follow the steps we discussed previously under ■ **Displaying the same document for all contacts**. I gave this template a **Template Name:** of **Display Local Weather**.

■ Displaying MultiMap Information (specifically designed for my UK readers)

The HTML document that you will use for this exercise is **UK MultiMap.html**. No changes should be required for this document to produce the desired results. Just follow the steps that I discussed previ-

Note

As of todays testing, MultiMap appears to have been gobbled up by Bing, and is now utilizing the Microsoft Maps.

ously under ■ **Displaying the same document for all contacts**. I gave this template a **Template Name:** of **UK MultiMap** within my GoldMine.

The relevant script is:

```
window.location="http://www.multimap.co.uk/map/browse.cgi?client=public&db=pc&addr1=&client=public&addr2=&advanced=&addr3=&pc=&lt;&lt;&amp;Zip&gt;&gt;&quicksearch=&lt;&lt;&amp;Zip&gt;&gt;";
```

You could easily modify this script to be more specific as I am only passing it the Zip Code (Don't you call this Postal Code?) information.

■ Displaying Your Social Networking Site Login

The HTML document that you will use for this one is **LinkedIn.html**. Again, no changes should be required for this document to produce the desired results. Just follow the steps we discussed previously under ■ **Displaying the same document for all contacts**. I gave this template a **Template Name:** of **LinkedIn**.

■ LinkedIn Search

Once you have logged in utilizing the ■ **Displaying Your Social Networking Site Login**, you may then utilize this LinkedIn Search for the Active Contact until you have logged back out. The HTML document that you will use for this one is **LinkedInSearch.html**. Again, no changes should be required for this document to produce the desired results. Just follow the steps we discussed previously under ■ **Displaying the same document for all contacts**. I gave this template a **Template Name:** of **LinkedIn Search**.

■ Displaying Active Windows Explorer

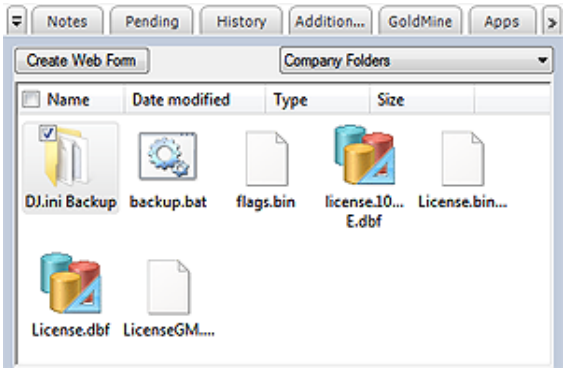

Figure 11-9

Scenario: You maintain a folder for each contact. In this folder are subfolders as well as documents and other applicable information about your contact. Each main folder is using the same name as a contacts name. You want to be able to view the different folders, and information contained in those folders from within your GM+Views. You want to actively open those documents from within your GM+Views.

The HTML document that you will use for this exercise is **Contact Explorer.html**. I gave this template a **Template Name:** of **Contact Explorer** in my GoldMine.

The relevant script is:

```
<meta http-equiv=REFRESH content="0;url=File:///\\Dell-Server\Apps\GoldMine\Temp\&lt;&lt;&amp;Contact&gt;&gt;\">
```

You may have noticed that the UNC path is: **\\Dell-Server\Apps\GoldMine\Temp**. It is important to understand that this must be a shared, and fully accessible path for everyone. You may then have noticed that I am pulling the contact name from the active record, therefore, against my record, the full UNC path when developed would be: **\\Dell-Server\Apps\GoldMine\Temp\DJ Hunt**. It is important to understand that the contact name may not contain any special characters that are not acceptable Windows pathing characters.

Figure 11-10

Passed Information

Number to Dial: (978)342-3333
Calling E-mail: DJH@DJ-Hunt.com
Caller Alias:
Send E-mail:
Number to Conference: (978)342-3333
Caller E-mail: DJH@DJ-Hunt.com
Caller Alias: GoldMine Guru

OrgID: 12345
PinID: 67890

Note

*These are older screenshots that I am reusing and my E-mail Address has since changed to: **DJ@DJHunt.US**.*

Pushing Information

This concludes the setup of the our predefined HTML documents. That wasn't too bad now, was it? This supplies you with a good taste for the potential of the GM+View/GM+Browser. I have also used GM+Views to **Pull** GoldMine data, and then, with the use of the **HTML Post** capability, **Pushed** that data to an ASP file to be processed at another location. In Figure 11-10 on the previous page, I show you a screen shot of this in action. This is one of several of the **HTML Post** uses that I have developed for clients.

Note

*Please note that I have not had the opportunity to test the **Submit** on this within GoldMine Premium 9.0.1.27 Beta build in a Windows 7 Ultimate environment, however, I have no reason to believe that it would still function as it had in the past.*

One of my clients' needed to push information from GoldMine to an Internet dialer, so I had to come up with a form that would do just that. The form shown previously in Figure 11-10 was the result of that exercise. The form itself is displayed here in Figure 11-11. Now I do want to emphasize that this is an **HTML Form**, and the **Dial Number** button has a form **Submit** action of:

Figure 11-11

"http://www.DJHunt.US/GMPush.asp"

Basically, all I am doing is taking the information from GoldMine, and placing it into this form. There are a couple of editable fields, however, the gist of the exercise is just to push the contact information out to an ASP file for processing.

You can see in Figure 11-11 just how I elected to capture the information, and in Figure 11-10, exactly what is being passed to the ASP file for the example given. There is nothing exotic about this exercise, however, it does show nicely how you can simply pass information from GoldMine to any website.

Linked Images

Are you a visual person? Are you a recruiter? Have you pictures that you would like to see of and for your candidates? For this next GM+View exercise, I will walk you through the simple creation of a GM+View template that will have some GoldMine fields, as well as a linked picture. I will do this directly in GoldMine, using the GoldMine tools available, just as an exercise for this book.

As I stated earlier, I would normally create my documents outside of GoldMine, and then incorporate them into the GM+View template. However, for you, the reader, I will work on an example directly within GoldMine.

I ask that you begin by creating a new GM+View document, and giving it a **Template Name:** of **Contacts Image**. Then tab to the body of the template.

The first thing that I will attempt to have you accomplish is to insert a picture place holder. I should explain this a little first. As this is a template, you can insert a place holder for a picture into the template. The place holder will include a variable by which the picture can be linked to a contact record under the GoldMine **Links** tab. When viewing the template under the GM+View tab, if there is a linked picture for that variable it will be displayed, else you will see a box with a red **X** in the upper left corner. See example in Figure 11-13 on the next page.

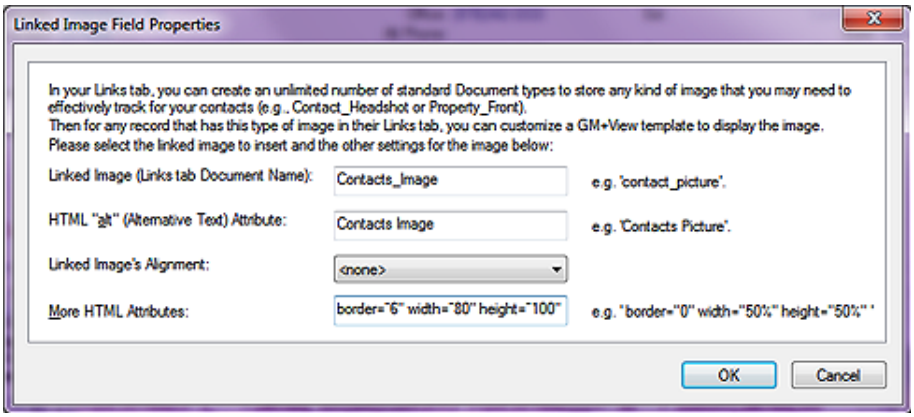

Figure 11-12

So then, let's begin by clicking the Insert field icon, refer to Figure 11-5. From the local menu select **Contact details ►** | **Linked Image** which will bring up the dialog shown above in Figure 11-12. You will notice that I have filled in some of the information in this dialog form already. For this GM+View Linked Image Field Properties dialog form, I intend the linked image to be a portrait of the contact. I

have, therefore, given the image the variable name of **Contacts_Image**. Additionally, I have included the alternate text of, **Contacts Image**, in case there is no linked image. The alternate text would be displayed identifying the place holder for the picture within GM+View, see Figure 11-13. I have chosen to do nothing about the place holder picture alignment as I just accept the default left location. I did include a border of **6**, a height of **100** and a width of **80** pixels. The height and width could have been entered as a percentage of the displayed page instead of using a fixed pixel height and width (see the examples to the right of each field in the dialog form). Actually, displaying examples is something that FrontRange is including in some of its newer dialog forms. There are examples right on the dialog form such that you should not need to select F1 for Help. When I click on the **OK** button, this is the string that I see in my GoldMine GM+View Editor:

<<LinkedImage:Contacts_Image:alt="Contacts Image" border="1" width="80" height="100">>

I could re-edit this information directly now to tweak it in, and I did, and/or I could add a new linked image by hand now that I know the syntax.

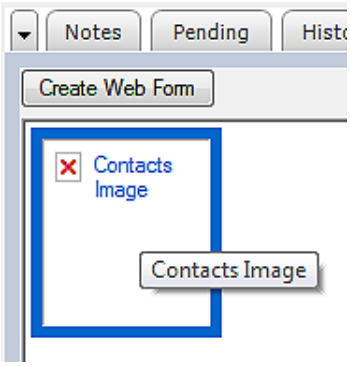

Figure 11-13

Close the **GM+View Tab Settings** dialog now so that you can put the place holder into action. I want to show you how to add a linked image to a specific contact record. Select the **GM+View** tab, and then select **Contacts Image** from the GM+Views drop list. Figure 11-13 displays what I see in my GoldMine, minus the cursor. Hopefully, yours is close to being the same. From here, I would ask you to move the cursor over the box. Notice that the cursor changes to a pointing fingered hand indicating that some action could take place with a simple click of the left mouse button. Click that mouse button now.

Do you recognize this dialog form, Figure 11-14? That's right, it's just the standard **Linked Document** dialog form with one exception, the **Document Name:** field has been populated for you with, in this case, **Contacts_Image**, and is uneditable. Click on the browse button (**...**), and browse to the image associated with your Contact that you want to have linked, a jpg file, I would hope. Now go ahead and click on the **OK** button. You can see that I used **C:\Inetpub\wwwroot\DJ Hunt\DWSite\Images\DJ_Cruise.jpg** as my personal image.

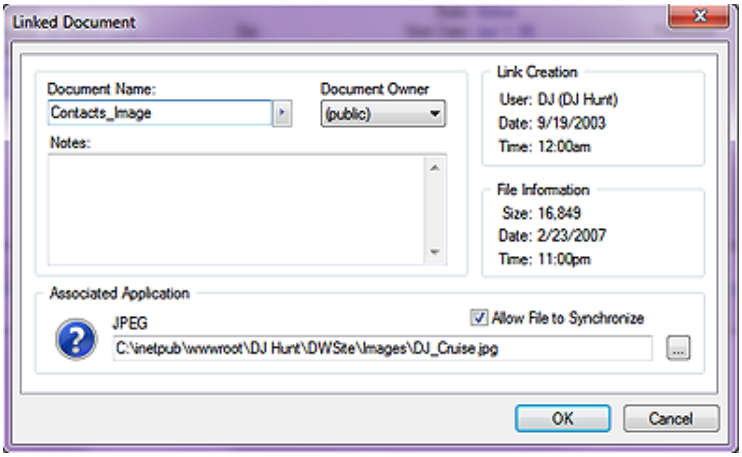

Figure 11-14

That's all there is to it. Now there is a Linked image for this contact record only, and, as important, the image will synchronize as long as you had selected to ☑ **Allow File to Synchronize** on the **Linked Document** dialog form. Figure 11-15 shows the results of this exercise based on my activity so far.

Figure 11-15

I feel that this is one of the really valuable bonuses represented by the addition of the GM+View to the GoldMine application. Imagine you were in Real Estate, and using the GoldMine Record Typing feature. You could have one Record Type for Clients, and another Record Type for Properties. On the properties record, you could be displaying images of the property, location maps, etcetera, easily, and quickly accessible for your prospective owners or your representatives to preview. Also, they are easily, and quickly accessible for e-mail to your client. Oh, the possibilities that the GM+View feature in GoldMine affords are virtually limitless. Add to this the GM+Browser, and you have a combination that cannot be beaten. When GM+View/GM+Browser was first introduced, FrontRange held a contest among its GoldMine Partners to design an award winning GM+View/GM+Browser. My Real Estate view, incorporating that which I discussed above, won second place, and was used as the GoldMine GM+View/GM+Browser Marketing brochure image for years.

Note

Actually, I'm going to tell you a little secret. After I created the screenshot for this book, I decided that my image looked weird so I modified the settings. They are actually:

border="6" width="90" height="100"

It was imperative that the text match the image, but I was just too skinny with the original settings.

Tip

If your organization will be synchronizing these linked images, I recommend that you create a folder under the GoldMine folder on the server, and then again on the remote installation alike. I then recommend that you place all of your linked images in this folder, and that the link to the image be made from this location.

Tip

Once you have the pointer HTML Code, in this case:

<<LinkedImage:Contacts_Image:alt="Contacts Image" border="6" width="90" height="100">>

...you could easily copy that into your hand built HTML document which might show 4 or 5 linked images along with formatted text as I discuss in the Properties GM+View.

You are limited only by your own design capabilities.

Let's go back in and edit the Contact Image a little more. This time I'll show you how to insert conditional text into the view. I have created, ahead of time, two text files, and saved them to my Temporary folder. They are **Fitchburg.txt**, and **Boston.txt**. Each text file contains some information relative to that particular city. I will use these in my example. You should create two similar text files to follow along or use the two that I have included in the accompanying template package.

Now, from the body of my template, I select the Insert field icon. This time I am selecting **Conditionally include a text file**. This causes the **Open** file dialog to pop up from which I select the first of my two files, Fitchburg.txt.

Once that is done, our template now contains:

```
<<LinkedImage:Contact_Picture:alt="Contact Picture" border="0">>

<<file: ?C:\Temporary\Fitchburg.txt>>
```

I position our cursor before the question mark (?), and enter my conditional expression such that the coding now looks like this:

```
<<LinkedImage:Contact_Picture:alt="Contact Picture" border="0">>

<<file: upper(&City) = [FITCHBURG]?C:\Temporary\Fitchburg.txt>>
```

Next I copy, and paste this statement again changing the conditional expression, and the text file that I want to use. The resulting coding now looks like:

```
<<LinkedImage:Contact_Picture:alt="Contact Picture" border="0">>

<<file: upper(&City) = [FITCHBURG]?C:\Temporary\Fitchburg.txt>>
<<file: upper(&City) = [BOSTON]?C:\Temporary\Boston.txt>>
```

I save this template, and close the **GM+View Tab Settings** dialog form. Now let's switch to the GM+View tab, if you're not already there, and look at the results.

Figure 11-16 shows my record for a person that resides in Fitchburg, Massachusetts. Notice that only one of the two conditional statements has been displayed, and, in fact, the correct txt file was displayed based on my conditional expression.

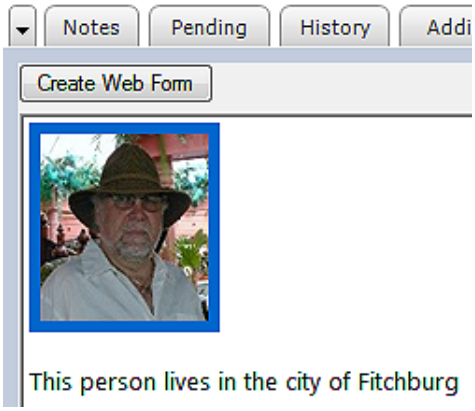

Figure 11-16

The possibilities are endless, and stretch as far as your imagination. Let's go back to a Real Estate example. You could have a text file with the property description, and have it display for the appropriate property.

How about security on a GM+View? Would you like to restrict a view to a certain user or users? That is just as easily accomplished using the conditional statement as I showed you previously, but with the condition modified. Something like this should do nicely:

```
<<file: &UserName $ [BG DJ RJ]?C:\GoldMine\HTMLTemplates\ThisHTMLFile.htm>>
```

Would you rather restrict it to a User Group? Something like this should serve you well:

```
<<file: &UserInGroup([Sales])?C:\GoldMine\HTMLTemplates\ThisHTMLFile.htm>>
```

Again, you are only bound by your imagination, and any bugs that you may come across in your experimentation.

Okay, let's continue on with one last item in our Contact Image template that I want to cover. I won't be covering the basic field item entries that are possible as you should have no trouble entering those. You should be testing those out on your own. I am going to enter a humongous expression, just to show you how this can be accomplished. This opens your GM+Views up to yet another set of possibilities.

I am going to use a rather extensive expression so I don't expect you to necessarily understand this expression, but you should have a general understanding of its results. I will explain that later. Again, I just want to show you that it can be done.

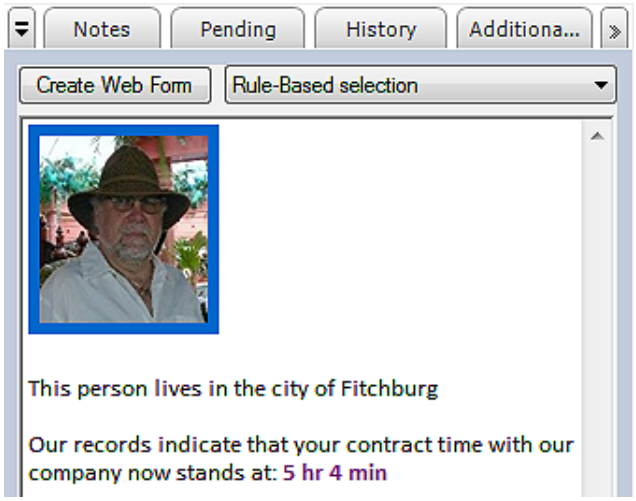

Figure 11-17

Here is the expression that I am entering at the bottom of the template:

Our records indicate that your contract time with our company now stands at: **<<&iif(Contact2.unCntTime>=0,alltrim(str(int(Contact2.unCntTime)))+" hr " +left(alltrim(str(60*(Contact2.unCntTime-(int(Contact2.unCntTime))))),2)+" min", iif(Contact2.unCntTime>-1,"-"+alltrim(str(int(Contact2.unCntTime)))+" hr " + substr(alltrim(str(60*(Contact2.unCntTime-(int(Contact2.unCntTime))))),2,2)+" min",alltrim(str(int(Contact2.unCntTime)))+" hr " + substr(alltrim(str(60*(Contact2.unCntTime-(int(Contact2.unCntTime))))),2,2)+" min"))>>**

Is that enough of an expression for you?

My GoldMine has a user defined field, **Contact2.unCntTime**, **N**, **10**, **2** in which I record the GoldMine Support Contract Time remaining on a customers contract. For this particular record, the contract time remaining is 5.07. I needed to convert that into the more readable hours and minutes format for my customers & users. The above expression comes right out of my GoldMine HTML E-mail Templates, however, it can be just as easily used in GM+Views. Figure 11-17 above, shows the resulting GM+View tab for this contact.

WOW! There is just so much that one can do with GM+Views that it just boggles the mind.

External Table

I would like to now show you how to display any **External Table** within GoldMines GM+Views. This feature has been available since the GM+View/GM+Browser was added to GoldMine, hence, this section is applicable to all versions of GoldMine that contain the GM+View/GM+Browser feature.

Let's see if I can think of some place to use this feature. Hmmm! How about displaying data from GoldMine History for the active Contact record? Yes, I know that the History table is not truly an external table, however, as I plan to use it, it is in effect an external table. You will be able to use this same methodology to access any SQL table in the GM+Views/GM+Browser. I'm going to show you how to link back to the GoldMine SQL ContHist table, and to display formatted information from that table. As I use GoldMine Premium for Contracted Support (I am not yet using the Services module), I am going to use the ContHist table, and segregate out just those GoldMine history records that pertain to any Contract activity that has occurred for the active Contact record. I would remind you again, however, that you can connect to any external table(s) for which you have a bridge (ODBC or OLE DB). You should be able to take this discussion, and apply it to any external data source that you wish to access.

Internet Information Services (IIS)

To achieve my goal, I am going to have to make use of the **Active Server Pages** (**ASP**), hence, the need to have something on my server that will allow for the processing of asp files. My weapon of choice for internal asp processing is **Internet Information Services** (**IIS**). This automatically installs as part of your operating system, however, it is not usually configured automatically. This, then, may need to be done by hand.

Here are the steps to assure that you have this configured on, let's say, your GoldMine server. These steps are taken from my Windows 7 Ultimate system:

Step 1

In the Pearl Start Search area type in Program. You should see, appearing at the top of your list, Programs and Features. Click upon that option to bring forth the dialog form shown in Figure 11-18 on the next page.

Note

You are reminded that this is one continuous expression, and that there are no linefeeds or carriage returns anywhere in this expression. You should be able to just copy & paste this into your example if you want to follow along.

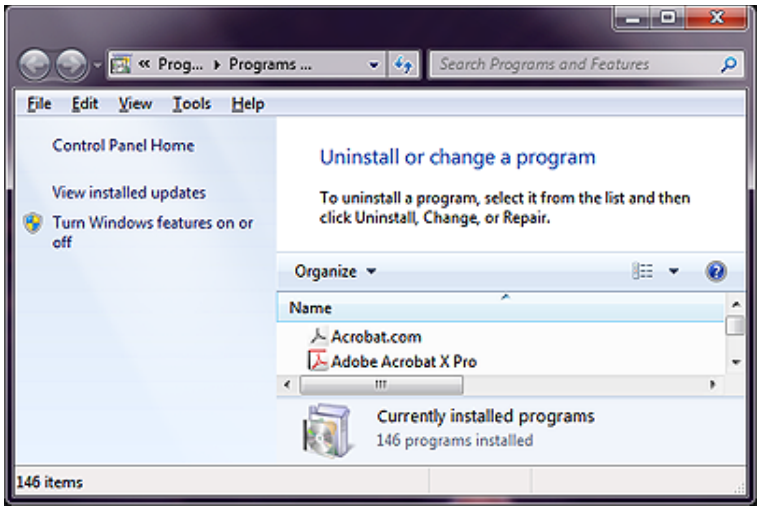

Figure 11-18

Step 2

Click on the option to **Turn Windows features on or off**.

Step 3

This should bring up the **Windows Features** dialog form shown here in Figure 11-19. Locate the checkbox for **Internet Information Services** (highlighted in Figure 11-19), and make certain that it is selected.

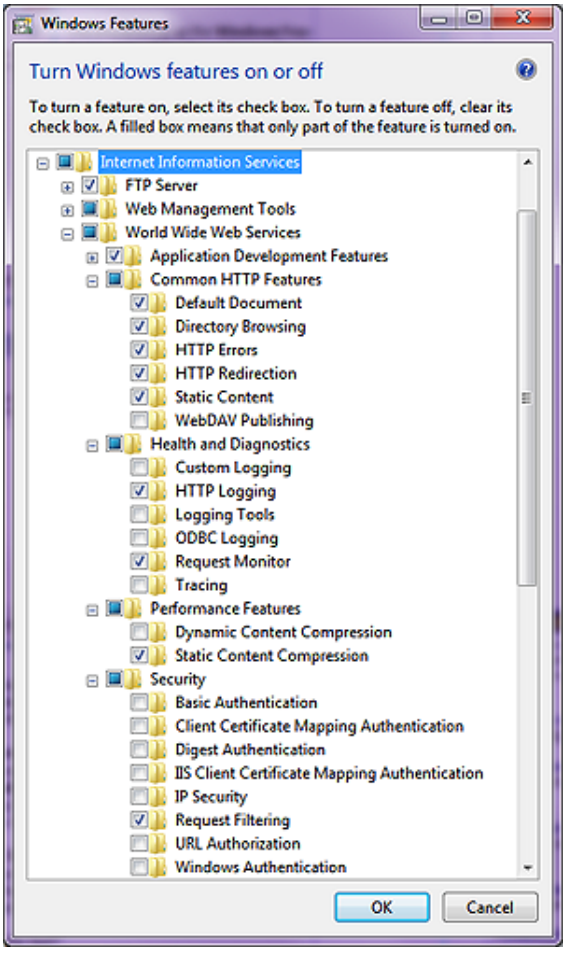

Figure 11-19

Step 4

If the box is checked, as shown here, then IIS has already been implemented for this system. On the other hand, if the option is not selected or partially selected, then place a check in the checkbox, and click the tree plus sign (**+**) to expand the tree as seen in the dialog form shown.

Here you would want to make certain that you have **World Wide Web Services** enabled. You may click upon the plus sign (**+**) to expand this tree as well. This should expose the tree further as seen in the dialog form displayed. In my configuration, not everything is checked. Naturally, your mileage may vary. Click on the **OK** button on this dialog form when finished.

After all that, click on the **OK** button, you may be asked to reinsert your **Windows DVD** so that the installer can capture some of the required components from the DVD. Alternatively, as in my case, the dialog form may simply close as I already have everything selected installed on my Window 7 Ultimate system. During the process you may be prompted to insert your Windows Installation disk, so be sure to have that handy when performing this action. Obviously, I have no idea which version your system will have installed. What I do know is that the build will be appropriate to your operating system, and that is what I care about.

Once the installation has finished, **Web Services** will be running automatically, and thereafter every time that you boot up your system. IIS installs mostly in the **C:\Windows\System32\inetsrv** folder, however, our interest is in the **C:\Inetpub\wwwroot** folder that is created at the same time. Under this folder I have created my working folder, **DJ_Asp_Book**, and would ask that you create a folder also as this is where I will ask you to put any work produced in this section.

As far as **Internet Information Services** (**IIS**) is concerned, that's all that there is to it. However, for safety, what say we test the service? Open your browser, and type in the address: **http://127.0.0.1/LocalStart.asp** or **http://LocalHost/LocalStart.asp** as either should work. Hopefully the page displayed correctly in your browser. You now have a web server, and are ready to proceed with your programming adventure.

Are you ready to put your programming skills to the test?

Scripting

Actually, it is not as hard as you may think. I will supply you with, and comment here on my **GM-ViewTable.asp** file. You will need a tool with which you can create your own script file. I use Adobe DreamWeaver, however, NotePad will work just as well. The only caveat that I would have for you is to not utilize Microsofts Word or Publisher for this effort. I have heard of too many cases where people have trouble with the resulting code from those applications. Remember, this is not a graphical page that we are creating, but script to send a graphical page of output to be displayed within, in our case, GM+Views.

Here is my script:

■ **This section is required for the HTML document. Simply enter it as it is.**

```
<%Option Explicit%>
<html>
<head>
<title>External Table in GM+Views</title>
</head>
```

■ **This statement sets the background color for the GM+View displayed page, and you may change the color to match your schema**

```
<body bgcolor="#ECE9D8">
```

■ **And it begins for real. This marks the beginning of our script**

```
<%
```

■ **Here I am instantiating the variables that will be used throughout the script**

```
Dim adOpenForwardOnly, adLockReadOnly, adCmdTable
Dim objConn, objRS
Dim strAccountNo, strSQL
```

■ **Here I am presetting three constants to make them more readable in the script**

```
adOpenForwardOnly = 0
adLockReadOnly = 1
adCmdTable = 2
```

■ **Here I am creating two new objects to be used by the script**

```
Set objConn = Server.CreateObject("ADODB.Connection")
Set objRS = Server.CreateObject("ADODB.Recordset")
```

■ **Here I am populating the string AccountNo variable with the AccountNo as it was sent by GM+View in a QueryString. i.e.**

...asp?AcctNo='A0101955535'&Company='Computerese'

```
strAccountNo = Request.QueryString("AcctNo")
```

■ **Here I am actually making the connection to my SQL database. You will wish to set these properties specific to your SQL database.**

```
objConn.Open "Provider=SQLOLEDB;" & _
             "Persist Security Info=False;" & _
             "Data Source=Server;" & _
             "Database=SQLGoldMine;" & _
             "User ID=sa;" & _
             "Password=SAPassword"
```

WARNING

*This section **must** be modified with the configuration for your system or the .asp will not function. Specifically:*

***Provider=SQLOLEDB** is for a Microsoft SQL backend.*

***Data Source=Server** requires the name of your SQL Server computer.*

***Database=SQLGoldMine** requires the name of your MSSQL database.*

***User ID=sa** is the System Administrator login for your SQL Server.*

***Password=SAPassword** where you would replace SAPassword with the password belonging to your System Administrators login.*

Note

*Pay attention the portion of the statement highlighted. I am only taking the first 11 significant characters of the **Contact1.AccountNo** field as my match. This avoids special characters that GoldMine uses, but at the cost of possibly retrieving more records than desired as this may mean that there is a better chance of duplication of AccountNos.*

*You may be able to get away with using the entire **Contact1.AccountNo** today. I haven't tried it, but I'm sure that you, my reader, will let me know if it doesn't function as expected.*

Note

Be careful of line wrapping for eBook presentation. Look at the original .asp file to see the proper syntax.

■ **Here I am defining a SQL Query, and placing it in a variable that I will execute against my connection string to my SQL database**

```
strSQL = "select Userid, convert(varchar,OnDate,7) as CallDate," & _
    "OnTime, " & _
    "Duration, " & _
    "Notes, " & _
    "left(Ref,charindex('(oc:',Ref)-1) as Reference, " & _
    "substring(Ref,charindex('oc:',Ref)+3, len(Ref)-charindex('oc:',Ref)-3) as Contact " & _
    "from ContHist " & _
    "where ActvCode like 'C%' and " & _
    "ResultCode = 'COM' and " & _
    "OnDate+30 >= getdate() and " & _
    "'''' & left(strAccountNo, 11) & "'' = left(ContHist.AccountNo, 11)" & _
    "order by OnDate desc"
```

■ **This is where the query is actually executed, and the results of the query is stored in the Record Set aliased as objRS**

```
Set objRS = objConn.Execute (strSQL)
```

■ **Here I begin to develop the HTML code that will be returned to the requestor, GM+View. I begin by defining a table.**

```
Response.Write "<P><H2><Font color='#000080'>QuickBooks History</Font></H2></P>"
Response.Write "<table id='myTable' cellspacing='0' border='0'>"
Response.Write "<tbody>"
Response.Write "<tr>"
```

■ **Here I add the first row, or header row to the table**

```
Response.Write "<H5><Font color='#000080'>"
Response.Write "Contract Time Remaining: " & Request.QueryString("ContractTime")
Response.Write "</Font></H5>"
Response.Write "<td><H5><u><font color='#000080'>User_ID</font></u></H5></td>" & _
               "<td><H5><u><font color='#000080'>Call_Date</font></u></H5></td>" & _
               "<td><H5><u><font color='#000080'>Duration</font></u></H5></td>" & _
               "<td><H5><u><font color='#000080'>Reference</font></u></H5></td>"
Response.Write "</tr>"
```

■ **Here I begin to populate my columns by looping through the returned Record Set. With each pass I am actually populating three rows of the table.**

```
If Not objRS.EOF then
        Do While Not objRS.EOF
```

■ **This is the first row in my GM+View, and it is populated with the UserID, Call Date, Duration, and the Reference.**

```
Response.Write "<tr>"
Response.Write "<td width=50><font size='2' font color='#000080'>" & objRS("Userid")
        & "</font></td>"
Response.Write "<td width=65><font size='2' font color='#000080'>" &
        objRS("CallDate") & "</font></td>"
Response.Write "<td width=50><font size='2' font color='#000080'>" &
        objRS("Duration") & "</font></td>"
Response.Write "<td><font size='2' font color='#000080'>" & objRS("Reference") &
        "</font></td>"
Response.Write "</tr>"
```

■ **This is the second row in my GM+View, and it is populated only with the Notes information.**

```
Response.Write "<tr>"
Response.Write "<td><font size='2' font color='#993333'>" & " " & "</font></td>"
Response.Write "<td><font size='2' font color='#993333'>" & " " & "</font></td>"
Response.Write "<td><font size='2' font color='#993333'>" & " " & "</font></td>"
Response.Write "<td><font size='2' font color='#993333'>" & objRS("Notes")
        & "</font></td>"
Response.Write "</tr>"
```

■ This is the third row in my GM+View, and it is populated with a period in the last column, in effect, creating a blank row between records.

```
Response.Write "<tr>"
Response.Write "<td><font size='2' font color='#993333'>" & " " & "</font></td>"
Response.Write "<td><font size='2' font color='#993333'>" & " " & "</font></td>"
Response.Write "<td><font size='2' font color='#993333'>" & " " & "</font></td>"
Response.Write "<td><font size='2' font color='#993333'>" & "." & "</font></td>"
Response.Write "</tr>"
objRS.MoveNext
Loop
End If
```

■ After I have finished looping through my record set to End of File, eof(), I have some programming chores to accomplish, like closing my objects in reverse of their creation.

```
objRS.Close
objConn.Close
```

■ Adding the closing markers for my table.

```
Response.Write ("</tbody>")
Response.Write ("</table>")
```

■ Setting the objects to nothing releases their memory allocation.

```
Set objRS = Nothing
set objConn = Nothing
```

■ And, lastly, my closing script marker, and my closing document markers.

```
%>
</body>
</html>
```

WARNING

*Notice, in the **Contract Usage.html** document, that the path is to my server, and the location of the .asp which resides on my WebServer. This path must be modified in your document prior to employing it in your GM+View.*

This is but an explanation of the **GMViewTable.asp** document which is called by the **Contract Usage.html** document, both of which I have supplied to you in a zip file included within the download version of this eBook. You could just drop that script into your **C:\Inetpub\wwwroot\Asp_Book** folder, modify the connection string to match your system, and it should work with the proper call from GoldMine GM+Views. Review the sidebar **WARNING** on this.

Next we must have a GM+View template that calls our .asp, see included **Contract Usage.html**. Remember that I advocate creating your templates outside of GoldMine, and then using the GM+View **Include a text file** option in the body of the template.

My template code is very simple, but the statements are very long, and wrap for eBook presentation. Please review the actual document supplied while I am covering it here.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/
xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
    <head>

        <!--
            This GM+View Template is provided as part of GoldMine Premium - The Definitive Guide
            series of books, and the author is providing this without guarantee or additional support.
        -->

    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

    <title>Contract Usage Document</title>

    <Script language=JavaScript>

        <!--
        function redirect()
        {
```

■ These next 3 lines are wrapped, but in the document the line is one continuous line without spaces.

```
        window.location = "http://DJ-Workstation/DJ_Asp_Book/GMViewTable.asp?AcctNo=&lt;
                &lt;&amp;left(AccountNo, 11)&gt;&gt;&Company=&lt;&lt;&amp;Company
                &gt;&gt;&ContractTime=&lt;&lt;contact2->unCntTime&gt;&gt;"
```

```
        }
        setTimeout ("redirect();", 0000)
        // -->

    </Script>

</head>

<body>
</body>

</html>
```

Once you have modified **Contract Usage.html** to your specifications, you may include it in the body of your GM+View template as I have done for the many other templates that I have discussed in this chapter.

Well now the acid test. Let's try it out. Selecting this view from the GM+View should produce a view similar to that shown here in Figure 11-20.

Figure 11-20

Does yours look similar to Figure 11-20? My guess would be not. You may have noticed that my SQL Query was looking for a specific type of Activity Code having a specific type of Result Code, neither of which may be applicable to your situation. This brings up a good point. You must make your SQL Query draw the data that you want to display. Try your SQL Queries out first in GoldMine to make certain that they are pulling the data that you expect to display in GM+Views, and then modify your SQL Query in your html document appropriately.

Let me explain a little about what this query is filtered against. My structure makes pulling out the where clause statements easy, as each is on its own line.

- **Where the ContHist.ActvCode begins with C**

  ```
  "where ActvCode like 'C%' and " & _
  ```

- **Where the ContHist.ResultCode equals COM**

  ```
  "ResultCode = 'COM' and " & _
  ```

- **Where the ContHist.OnDate is not older than 30 days ago**

  ```
  "OnDate+30 >= getdate() and " & _
  ```

- **Where the first 11 characters of the ContHist.AccountNo equals the sent value**

  ```
  "'" & left(strAccountNo, 11) & "' = left(ContHist.AccountNo, 11)" & _
  ```

With this as the basis of your work, you should be able to pull data from any source into your GM+View for which you have an **ODBC** or **OLE DB** driver. There are endless possibilities, and I hope that you will let your imagination run wild. If you do create some nice GM+Views that you would like to share with my audience, please send them along to DJ@DJHunt.US.