

Gmap4

Delimited Data

By: Joseph Elfelt

Last update to this file: May 15, 2016

For more documentation see the [Gmap4 Help page](#)

Also see the "[What is new](#)" page

Table of contents

Introduction.	2
Build your first map step-by-step	3
Beyond the basics	7
Field delimiter character	8
Add a title to your map	8
Line width and line color are sticky	8
Short symbol names are sticky	8
Write coordinates with the longitude first	9
Labels	9
Make a label that does not have a symbol.	9
Make a label a clickable link.	10
Style your labels with inline HTML and CSS	10
Override the default CSS style for labels	11
Style many labels at once with CSS	12
Move the label location.	13
Two ways to turn labels off for some points.	14
Make your delimited files open faster.	14
Also display KML/KMZ files.	14
Also display GIS data	15
Including photos in a delimited text file	15
A few reminders	16
Summary of syntax for use in delimited text files	17

1. Introduction

Gmap4 can read delimited text files that are prepared as documented below. This delimited file format includes the most commonly used features that a KML file can handle plus additional features not available in a KML file. Some of the things that a Gmap4 delimited text file can do that a KML file cannot do include:

- Specify layers of data hosted on GIS servers
- Lots of control over labels. All labels can be on or off.
- Dashed lines

This delimited text file format is also much easier to understand and edit than GPX or KML files. If you want to build a data file ‘by hand’ that you can display with Gmap4, then the fastest way to do it is by using this delimited text file format. The syntax used for these delimited files is also the syntax that is used for Gmap4’s exclusive **map-in-a-link** feature.

Assume you want to make a map that displays a symbol at some of the parks near where you live. The only thing you need in your delimited data file is the latitude longitude for each park. For example, a delimited text data file containing just these three lines:

```
47.659549,-122.127714  
47.668306,-122.145073  
47.672959,-122.115483
```

is online at: https://sites.google.com/site/gmap4files/p/delimited/delimited_04.txt

These coordinates are inside three different parks located at the City of Redmond, Washington State, USA. The following link displays that data file with Gmap4:

https://mappingsupport.com/p/gmap4.php?q=https://sites.google.com/site/gmap4files/p/delimited/delimited_04.txt

By contrast, you would need well over a 10 lines in either a GPX or KML file in order to make the exact same map. As you learn more about the delimited text file format that Gmap4 can display, then you can add additional features to your maps using a syntax that easy and fast to learn. In addition to the information you are reading here, the Gmap4 Examples page has some maps that display delimited text files.

See: https://mappingsupport.com/p/gmap4_examples.html

You can download the delimited text file used to produce each map, open it in any editor program and notice how the commands in that file control the map that you see.

If you have a PC then the freeware editor Notepad++ (<https://notepad-plus-plus.org/>) is highly recommended for working with delimited text files. This is not Windows notepad. Any editor used for preparing HTML files should also work fine.

Tech tips: Save your delimited text files with **UTF-8 encoding without BOM**. When you make your Gmap4 link remember that the ‘q’ parameter is case sensitive.

Thanks to Robin Tivy of <http://bivouac.com/> for his help in thinking about how this delimited file format should operate and assistance in testing the beta code. Get a free login and check out

Robin's site. The links you see labeled "Topo" will launch Gmap4. Many of the maps you see are produced from delimited text files which Robin's code generates on-the-fly from his database.

2. Build your first map step-by-step

Now let's take the simple map described above in the 'introduction' and add some features to it. Let's start by adding the name of each park.

```
47.659549,-122.127714^ Westside Park
47.668306,-122.145073^ Grass Lawn Park
47.672959,-122.115483^ Anderson Park
```

The above file is online at:

https://sites.google.com/site/gmap4files/p/delimited/delimited_05.txt

and here is the map link:

https://mappingsupport.com/p/gmap4.php?q=https://sites.google.com/site/gmap4files/p/delimited/delimited_05.txt

Each row in this file represents a **waypoint**. If you hover your cursor over a waypoint symbol, then the name appears. Watch what happens when you click **Menu** ==> **Label**. And to have all labels displayed on the map when it first appears on the screen, add the '&label=on' parameter to the Gmap4 link:

https://mappingsupport.com/p/gmap4.php?q=https://sites.google.com/site/gmap4files/p/delimited/delimited_05.txt&label=on

Labels on a map are good for information that you want people to see without having to click on anything. These labels are given a default style (light yellow background, 1 pixel border, etc) by Gmap4. A bit later in this documentation you will learn how to override that default style with your own HTML and CSS.

Here are a couple ideas for **useful maps you can make with this label feature**. You could make a map that shows the names of other people that live within a short walk of your home. In many places those names are available online via the county or city tax records. Or maybe you could help your child make a map showing where members of their sport team live.

Notice the '^ ('caret') character in the above data file. That character is a 'delimiter' and is used to separate the different fields of information. Let's add another delimiter and then a description for each waypoint:

```
47.659549,-122.127714^ Westside Park^ 5810 156 Ave NE
47.668306,-122.145073^ Grass Lawn Park^ 7031 148 Ave NE
47.672959,-122.115483^ Anderson Park^ 7802 168 Avenue NE
```

The above file is online at:

https://sites.google.com/site/gmap4files/p/delimited/delimited_06.txt

and here is the map link:

https://mappingsupport.com/p/gmap4.php?q=https://sites.google.com/site/gmap4files/p/delimited/delimited_06.txt&label=on

Click a marker and the description appears.

Up to now this series of maps has displayed the default symbol image which is a small red paddle. Instead of continuing to use that default symbol image, the following data file specifies two different images that will be used for waypoint symbols. In order to use an image as a waypoint symbol, the image file must be online and the image must not be larger than 64 pixels by 64 pixels. Here is the revised delimited file which uses some symbol images that are hosted by Google:

```
// This section identifies waypoint images
symbol=https://maps.google.com/mapfiles/kml/pal2/icon4.png name=tree
symbol=https://maps.google.com/mapfiles/kml/pal2/icon2.png name=cabin
// This section lists the waypoint locations and related information
47.659549,-122.127714^ Westside Park^ 5810 156 Ave NE^ tree
47.668306,-122.145073^ Grass Lawn Park^ 7031 148 Ave NE^ tree
47.672959,-122.115483^ Anderson Park^ 7802 168 Avenue NE^ cabin
```

The above file is online at:

https://sites.google.com/site/gmap4files/p/delimited/delimited_07.txt

and here is the map link:

https://mappingsupport.com/p/gmap4.php?q=https://sites.google.com/site/gmap4files/p/delimited/delimited_07.txt&label=on

The lines in the above file that start with // are comment lines. The above file uses the ‘symbol=’ **command** to (1) provide a link to each image and (2) provide a short name for each image. Each **coordinate** then has another delimiter added at the end of the line followed by one of those short names. You can have as many different symbol images on a map as you want.

This page shows many symbols that Google is hosting:

<https://hohonuuli.blogspot.com/2007/09/list-of-paddle-icons-for-kml-z-letters.html>

Here is a large collection of 3rd party symbols. If you wish to use any of these you should download them and put them online yourself by uploading them to Google Sites or somewhere else:

<http://mapicons.nicolasmollet.com>

TIP: You can use Google Sites to build your own library of symbol images.

Symbols are placed on the map such that **middle of the bottom edge** of the image will be at the specified latitude longitude point. A future update to Gmap4 will give you the option to center the symbol at the specified latitude longitude point.

You can also use delimited files to **add lines to your maps**. The following revised data file has two lines. The first one uses the default line color (red) and default line width. The second line specifies the color green and a width of 5 pixels:

```

// This section identifies waypoint images
symbol=https://maps.google.com/mapfiles/kml/pal2/icon4.png name=tree
symbol=https://maps.google.com/mapfiles/kml/pal2/icon2.png name=cabin
// This section lists the waypoints
47.659549,-122.127714^ Westside Park^ 5810 156 Ave NE^ tree
47.668306,-122.145073^ Grass Lawn Park^ 7031 148 Ave NE^ tree
47.672959,-122.115483^ Anderson Park^ 7802 168 Avenue NE^ cabin
// This section contains data for linepoints
line=on
47.658768,-122.132607
47.654202,-122.132477
47.654087,-122.134666
47.654751,-122.140030
line=off
line=on width=5 color=#00FF00 linesymbol=on
47.654751,-122.140030
47.655010,-122.143250
47.663334,-122.143204
47.669060,-122.143120
47.669086,-122.143250

```

The above file is online at:

https://sites.google.com/site/gmap4files/p/delimited/delimited_08.txt

and here is the map link:

https://mappingsupport.com/p/gmap4.php?q=https://sites.google.com/site/gmap4files/p/delimited/delimited_08.txt&label=on

A ‘line=’ command can include the following optional pieces of information that control how the line appears on the map:

width=	default is 2 pixels
color=	default is FF0000 (red)
linesymbol=	default is off
dash=	default is off. See next example map.

The first line in the above file uses the **default width of 2 pixels** and the **default color of red**. The second line in the above file overrides those default values by specifying a line width of 5 pixels and the color green (00FF00).

Color must be in a six character code. For example:

color=#FF0000	red
color=#00FF00	green
color=#0000FF	blue

To learn more about HTML color see http://w3schools.com/html/html_colors.asp. For a color picking tool (there are many others) see: <http://colorpicker.com/>.

Whether or not a symbol is automatically displayed at all of the coordinates along a line is determined by the 'linesymbol' setting. Since the default is 'off', symbols are not displayed at the coordinates that are along the first (red) line. However, the second line=command overrides that default value by saying "linesymbol=on" and a symbol is displayed at each coordinate along the second (green) line. The symbol that is displayed is the most recent one used on the file, which in this case is the 'cabin'.

Notice that the **field delimiter** (^) is not used on the **command** lines. Instead, it is only used on the waypoints and sometimes on the linepoints.

In addition to having solid lines on your map, you can also have **dashed lines**. If you use the dash setting, then it should be the **last one on the line**. The following file has several parallel lines over a random location. The lines appear on the map from **left to right**.

```
line=on
36.0,-111.2
35.8,-111.2
line=off
line=on dash=on           Same as dash=1,20. Compare the next line.
36.0,-111.1
35.8,-111.1
line=off
line=on dash=1,20
36.0,-111.0
35.8,-111.0
line=off
line=on width=3 dash=2,30
36.0,-110.9
35.8,-110.9
line=off
line=on width=4 dash=3,40
36.0,-110.8
35.8,-110.8
line=off
line=on width=5 dash=4,50
36.0,-110.7
35.8,-110.7
line=off
line=on width=6 dash=5,60
36.0,-110.6
35.8,-110.6
line=off
```

The above file is online at:

https://sites.google.com/site/gmap4files/p/delimited/delimited_14.txt

and here is the map link:

https://mappingsupport.com/p/gmap4.php?q=https://sites.google.com/site/gmap4files/p/delimited/delimited_14.txt

Watch what happens to the dashes as you zoom in and out. Notice that sometimes a dash at the end of the line will extend past the last coordinate in that line. Gmap4 simply passes the dash settings to Google and it is Google's code (and not Gmap4) that actually draws the dashes on the map.

Tip: Be careful if you play with custom dash settings. It is easy to pick dash settings that result in a solid line. Please start with the dash settings in the above example and then do your own careful experimentation.

And remember, if you are going to have a dashed line on your map then the "dash=" setting should be the **last thing on the line**.

Summary of key concepts: A row in a delimited text file can be one of four things:

1. Command (Example: symbol=, line=)
2. Waypoint
3. Linepoint (including points on a GPS track)
4. Comment

Congratulations! You now know how to build delimited text files. To view a delimited text file with Gmap4 all you need to do is:

1. Build the text file and give it a file extension of **txt**.
2. Place the text file online (Google Sites works great).
3. Make a Gmap4 link and include the 'q' parameter to point to your text file.
4. Paste your Gmap4 link into a browser bar and admire your map.

But wait - there's more. The next section will dig deeper into the delimited file syntax and show you additional features you can add to your map.

3. Beyond the basics

In addition to the information you are reading here, the Gmap4 website has some example maps that use delimited text files.

See: https://mappingsupport.com/p/gmap4_examples.html

Remember, you can always download any data file that Gmap4 is displaying and view the contents of that file with a suitable editor program like Notepad++. So if you see a map that is displaying labels or some other feature in a cool way, it is very easy to learn the secrets for how to do the same thing with your own maps.

a. Field delimiter character

Gmap4 automatically detects the character you are using for the field delimiter. The default field delimiter character is the caret symbol '^'. While you can use any character as the field delimiter, **it is highly recommended that you use the ^ character since that character is unlikely to appear in your data.** Using a semi-colon as the field delimiter might cause conflicts with the enhanced internet security screening in Gmap4. Also, if you plan to either (1) use inline CSS or (2) use the semi-colon character as part of the grammar in your data file, then the semi-colon character **cannot** also be used as the field delimiter and you must either use the default field delimiter character or specify a different field delimiter.

b. Add a title to your map

Use the 'title=' command to add a title to your map. That title will appear near the upper left corner of the screen. Example:

```
title=Some Redmond city parks
```

Remember, you can use inline HTML and CSS with your map title. See below for details.

c. Line width and line color are sticky

The settings for line width and line color are 'sticky' until they are changed in your file. For example, assume your file looks like:

```
line=on width=1 color=#00ff00
47.0,-122.0
47.3,-122.3
line=off
line=on
46.5,-100.9
46.9,-100.7
```

The second line on the map will also be 1 pixel wide and green.

d. Short symbol names are sticky

In prior sections of this documentation you learned that a delimited text file can have four different types of lines. One type of line starts with coordinates. That type of line can have the following fields: Coordinates Name Description Short symbol name

In the following file notice that the last coordinate line does not have a description and **does not have a short symbol name.**

```
// This section has identifies waypoint images
symbol=https://maps.google.com/mapfiles/kml/pal2/icon4.png name=tree
symbol=https://maps.google.com/mapfiles/kml/pal2/icon2.png name=cabin
```



```
// This section lists the waypoints
47.659549,-122.127714^ Westside Park^ 5810 156 Ave NE^ tree
47.668306,-122.145073^ Grass Lawn Park^ 7031 148 Ave NE^ tree
47.672959,-122.115483^ Anderson Park^ 7802 168 Avenue NE^ cabin
47.673425,-122.123143^ Downtown
```

The above file is online at:

https://sites.google.com/site/gmap4files/p/delimited/delimited_09.txt

and here is the map link:

https://mappingsupport.com/p/gmap4.php?q=https://sites.google.com/site/gmap4files/p/delimited/delimited_09.txt&label=on

Examine the above map link. Will the labels appear on the screen when the map opens? Now open the above map and notice that the cabin symbol appears both at Anderson Park and downtown. The **general rule** is:

- Each waypoint will have a symbol on the map
- That symbol will be either (1) the most recently used ‘short symbol name’ in the file or (2) Google’s default symbol.

In the above map the downtown waypoint has a cabin symbol since it is the most recently used short symbol name in the file.

e. Write coordinates with the longitude first

The default way to write a coordinate is latitude, longitude. The “coordinate” command lets you override this default. Example:

```
coordinate=longitude,latitude
```

Later in the same file you could change back to the default way of writing coordinates by including the command:

```
coordinate=latitude,longitude
```

Why might you care? The syntax for a KML file uses the form **longitude**,latitude. If you have a KML file with a GPS track in it, then you can copy the **longitude**,latitude values for that track and paste them into a delimited text file.

f. Labels

i. Make a label that does not have a symbol

This section describes an **exception** to the general rule that each waypoint has to have a symbol. Use this exception when you want to have a label on your map without an associated waypoint symbol. To use this exception, put the keyword ‘**nosymbol**’ in the field where the short symbol name would go:

```
// This section has identifies waypoint images
symbol=https://maps.google.com/mapfiles/kml/pal2/icon4.png name=tree
```

```

symbol=https://maps.google.com/mapfiles/kml/pal2/icon2.png name=cabin
// This section lists the waypoints
47.659549,-122.127714^ Westside Park^ 5810 156 Ave NE^ tree
47.668306,-122.145073^ Grass Lawn Park^ 7031 148 Ave NE^ tree
47.672959,-122.115483^ Anderson Park^ 7802 168 Avenue NE^ cabin
47.673425,-122.123143^ Downtown^ ^ nosymbol

```

The above file is online at:

https://sites.google.com/site/gmap4files/p/delimited/delimited_10.txt

and here is the map link:

https://mappingsupport.com/p/gmap4.php?q=https://sites.google.com/site/gmap4files/p/delimited/delimited_10.txt

When you open the above map you will not see any labels. That is because the default for the label link parameter is label=off. After the map opens, turn labels on by clicking Menu ==> Label. The downtown label appears even though there is not any waypoint symbol at that location.

ii. Make a label a clickable link

The following code shows a good way to turn a label into a clickable link. This link will not be underlined and it will not change color. The cursor will still change appearance when it is over the link. Everything that has a yellow background goes into the 'label' field in your delimited text file.

```

47.672959,-122.115483^ Anderson Park^ 7802 168 Avenue NE^ cabin^ <a
href=http://ci.redmond.wa.us/cms/one.aspx?portalId=169&pageId=4077
style="text-decoration:none;" target="_blank"><font color=black>Anderson
Park</font></a>

```

iii. Style your labels with inline HTML and CSS

Now let's make the label for Anderson Park and Downtown use bold text.

```

// This section has identifies waypoint images
symbol=https://maps.google.com/mapfiles/kml/pal2/icon4.png name=tree
symbol=https://maps.google.com/mapfiles/kml/pal2/icon2.png name=cabin
// This section lists the waypoints
47.659549,-122.127714^ Westside Park^ 5810 156 Ave NE^ tree
47.668306,-122.145073^ Grass Lawn Park^ 7031 148 Ave NE^ tree
47.672959,-122.115483^ <b>Anderson Park</b>^ 7802 168 Avenue NE^ cabin
47.673425,-122.123143^ <b>Downtown</b>^ ^ nosymbol

```

The above file is online at:

https://sites.google.com/site/gmap4files/p/delimited/delimited_11.txt

and here is the map link:

https://mappingsupport.com/p/gmap4.php?q=https://sites.google.com/site/gmap4files/p/delimited/delimited_11.txt

Now hover your cursor over the symbol for Anderson Park. The waypoint name will be displayed along with the raw HTML codes. That is certainly not attractive! Fortunately there is a **better way** to add **inline** HTML and CSS to labels.

In addition to the four fields already described (coordinate, name, description and short symbol name) a **fifth field can be used to hold a label**. That 5th field can either have the same text as the 'name' field or completely different text. Now we can revise the prior data file to look like:

```
// This section has identifies waypoint images
symbol=https://maps.google.com/mapfiles/kml/pal2/icon4.png name=tree
symbol=https://maps.google.com/mapfiles/kml/pal2/icon2.png name=cabin
// This section lists the waypoints
47.659549,-122.127714^ Westside Park^ 5810 156 Ave NE^ tree
47.668306,-122.145073^ Grass Lawn Park^ 7031 148 Ave NE^ tree
47.672959,-122.115483^ Anderson Park^ 7802 168 Avenue NE^ cabin^ <b>Anderson</b>
47.673425,-122.123143^ ^ ^ nosymbol ^ <b>Downtown</b>
```

Here is the map link:

https://mappingsupport.com/p/gmap4.php?q=https://sites.google.com/site/gmap4files/p/delimited/delimited_12.txt

Label rule: If there is data in the 'label' field, then that data will be used for the label. If there is no data in the 'label' field, then the 'name' field will be used for the label.

One way to style a label with **inline css** is to use a span tag:

```
47.672959,-122.115483^ Anderson Park^ 7802 168 Avenue NE^ cabin^ <span
style="font-weight:bold;">Anderson</span>
```

Note the semi-colon after the word "bold." **Because a semi-colon is part of the CSS syntax, that character cannot also be used as the field delimiter when you have CSS in your file.**

iv. Override the default CSS style for labels

Here is the default CSS style that Gmap4 applies to labels:

```
position:relative; left:-20px; top:0px; white-space:nowrap; border:1px solid black;
padding:2px; background-color:#FFFFCC; font-size:1em;
```

You can override that default with your own **inline** CSS. For example, to make the label for Anderson Park appear on a light blue background and with a larger font, you might write:

```
47.672959,-122.115483^ Anderson Park^ 7802 168 Avenue NE^ cabin^ <span
style="background-color:#0000CC; font-size:1.5em;">Anderson</span>
```

v. Style many labels at once with CSS

You have already learned that the label command can turn the display of labels off and on for waypoints that come after the command. In addition, the label command can specify two different sets of CSS statements that will be applied to all of the following labels. You can have many label commands in your file. The basic syntax is:

```
label=on css=_____ satellitecss=_____
```

Why are there two sets of CSS? This approach solves a problem that pops up if you want to have labels on your map with a **transparent background**. When you shift the map to an aerial view then a label with black text and a transparent background is impossible to see if the aerial image happens to have a black shadow where your label is located. The satellitecss= statement can be used to give your labels a **solid background** when the map is displaying one of the aerial views. Perhaps you will think of another use for this feature.

Both of these CSS statements are optional. If you use both CSS statements, then **they must be in the order shown**. Each of these css statements is limited to 150 characters.

To make all labels bold:

```
label=on css=font-weight:bold;
```

To make all labels have bold font, smaller font, no border, no padding, transparent background for non-aerial and yellow background for aerial. This all goes on one line in your file (**see next section for example map**):

```
label=on css=font-weight:bold;font-size:0.8em;border:none;padding:0px;  
background-color:transparent; satellitecss=background-color:#ffffcc;
```

To use the default CSS that is built into Gmap4 for the following labels, put this command in your file:

```
label=on
```

Here's a peek under the Gmap4 hood to help you understand how this works. Each time a 'label' command appears in your file, Gmap4 generates two groups of css statements:

- #1. Default CSS in Gmap4 (already described) + css
- #2. Default CSS in Gmap4 (already described) + css + satellitecss

The first group of CSS is used when the map view being displayed **is not** one of the aerial views. The second group of CSS is used when the map view being displayed **is** one of the aerial views. If the same CSS statement appears in a group, then **the last statement overrides earlier statements**. An example of CSS to be used with one of the aerial map views might look like:

```
position:relative; left:-20px; top:0px; white-space:nowrap; border:1px solid black;  
padding:2px; background-color:#FFFFCC; font-size:1em; font-weight:bold;  
font-size:0.8em; border:none; padding:0px; background-color:transparent;  
background-color:#ffffcc;
```

The red part is the default CSS built into Gmap4. The black part would come from `css=` and the blue part would come from `satellitecss=`. The statement “background-color” appears three times in this group. Since the **last statement overrides the earlier ones**, any labels styled with this group of CSS will have a light yellow background.

Each ‘label’ command stands on its own and does not inherit any CSS statements from previous label commands in your file.

vi. Move the label location

The default CSS contained in Gmap4 will place labels such that the upper left corner of the label is 20 pixels to the left of the coordinates stated on that same line of the delimited text file. You can override that position by adding your own positioning CSS to the ‘label’ command. (Inline CSS will not work for changing the label position.) Use the following CSS statements to change the label position:

left: __px; Number > 0 move label right. Number < 0 move label left.
top: __px; Number > 0 move label down. Number < 0 move label up.

Below is the content of a delimited text file that demonstrates:

- Labels moved so they are next to (instead of below) their symbols.
- Labels that have either a transparent or solid background depending on which map view is being displayed.
- A label (Downtown) that goes back to using only the default CSS built in to Gmap4.

Thee underlined portions of this data file are all on one line. Note that the Anderson Park label is a clickable link.

```
// This section has identifies waypoint images
symbol=https://maps.google.com/mapfiles/kml/pal2/icon4.png name=tree
symbol=https://maps.google.com/mapfiles/kml/pal2/icon2.png name=cabin
// This section lists the waypoints
label=on css=left:20px;top:-25px;font-weight:bold;font-size:0.8em;border:none;padding:0px;
background-color:transparent; satellitecss=background-color:#ffffcc;
47.659549,-122.127714^ Westside Park^ 5810 156 Ave NE^ tree
47.668306,-122.145073^ Grass Lawn Park^ 7031 148 Ave NE^ tree
47.672959,-122.115483^ Anderson Park^ 7802 168 Avenue NE^ cabin^ <a
href=http://ci.redmond.wa.us/cms/one.aspx?portalId=169&pageId=4077
style="text-decoration:none;" target=" blank"><font color=black>Anderson Park</font></a>
label=on
47.673425,-122.123143^ ^ ^ nosymbol ^ Downtown
```

The above file is online at:

https://sites.google.com/site/gmap4files/p/delimited/delimited_13.txt

and here is the map link:

https://mappingsupport.com/p/gmap4.php?q=https://sites.google.com/site/gmap4files/p/delimited/delimited_13.txt

vii. Two ways to turn labels off for some points

By default, all labels are ‘on’ and will be displayed when the map opens. If you want to turn **all labels** “off” then either include &label=off in the Gmap4 link or click Menu ==> Label.

If you are using a delimited text file then there are two ways to turn off just **some of the labels**.

You can use the ‘label’ command to turn labels off for some points. If your file includes the line:

label=off

then points after that command line will not have any labels. You can turn labels back on for subsequent points by including the command line:

label=on

You can also put the keyword ‘nolabel’ in the label field in order to turn the label off for just that point.

g. Make your delimited files open faster

When Gmap4 reads a file that uses the delimited format it does a lot of error checking. It part, it makes sure that every symbol image listed in the file actually exists. It turns out that this check takes from 1/5th to 1/2 of a second per image. Yikes! (The timing difference appears to depend on the configuration of the server hosting the image file.) This check can now be suppressed by adding **&checkheader=off** to the Gmap4 link. Of course if you are building delimited files for use with Gmap4, then you should not suppress this check until your file is debugged and ready to go into production. Thanks to Robin Tivy of <http://bivouac.com> for bringing this to my attention.

h. Also display KML/KMZ files

A delimited text file can identify up to five KML or KMZ files that will be displayed on the map along with whatever data is in the delimited text file. Here is the syntax for identifying such a file:

file=_____

Replace the underline with the full http or https link to a KML or KMZ file. Be careful not to include any spaces. The maximum number of KML/KMZ files that can be displayed on a map is five.

The data in the KML/KMZ files will be stacked on the basemap in order as the files are listed in the delimited text file with data from the first file appearing on the bottom and data from the last file appearing on the top of the stack.

i. Also display GIS data

You can use the same syntax to specify GIS data in two different ways. If your map is going to be able to display one or two GIS layers then you can specify those layers by adding parameters to the Gmap4 link. But if you want your map to be able to display more GIS layers then it likely is more convenient to specify those GIS layers in a text file, put that text file online and make a Gmap4 link that points to your text file. In both cases, the syntax is the same and usually looks like:

ArcGIS **MapServer** layer(s):

rest=_____&name=_____&layers=_____&transparent=_____

ArcGIS **ImageServer** data:

rest=_____&name=_____

WMS layer(s):

wms=_____&name=_____&layers=_____&transparent=_____

Replace the underlines with appropriate values for the GIS servers you want to access and the layers you want to display.

You can have a maximum of 25 “rest” and/or “wms” commands in a delimited text file

You can put your text file online at Google Sites for free. The Gmap4 Help page has a pdf file titled “Working With Files” that contains step-by-step instructions for working with Google Sites.

For more information, please visit:

GIS Help page: https://mappingsupport.com/p/help_files/gmap4_gis_help.html

GIS Viewer page:

https://mappingsupport.com/p/map4-arcgis-layers-on-google-maps.html#part_3

j. Including photos in a delimited text file

The ‘description’ field on a coordinate line can include photo thumbnails that can be clicked to display the photo full size. There are two important things to keep in mind:

1. You do not need to create a thumbnail image. Google will do that on-the-fly. Both the href= and src= must point to your photo.

2. You should specify both `height=` and `width=` for the thumbnail size you want. Otherwise the balloon will likely not be seized correctly. **These values should have the same ratio as your photo.**

One approach is to make a table with 1 row that has 3 cells. The first cell has some text, the second cell is just a spacer and the third cell holds the photo thumbnail. **All of this gets entered as a waypoint 'description' on a single line.** Here is an example of a waypoint description. It is shown on multiple lines for clarity:

```
<table><tr>
<td>Trail junction<br>at about 5,000'<br><br><font color="red">Click the
pic<font><br><br></td>
<td width="5px"></td>
<td><a
href="https://farm3.static.flickr.com/2519/4241732074_2f05842a56_o.jpg"></a></td>
</tr></table>
```

k. A few reminders

When you make a delimited text file to display with Gmap4, you can decide which of these building blocks you want to include:

- Map title
- Lines
- Symbols (i.e. markers/icons)
- Balloons that open when a symbol is clicked and contain text/images/links
- Labels
- KML/KMZ files
- GIS data

All latitude longitude values must be in the WGS84 datum. For points in the western hemisphere, longitude must have a minus sign. For points south of the equator, latitude must have a minus sign.

A **waypoint** can have up to five fields of information. These fields are called coordinate, name, description, short symbol name, label. A **linepoint** usually will only have a coordinate field but the other fields are also allowed if you wish to use them.

Maybe you do not need to use the label field at all. You only need to use the label field if either (1) the text for the label is different than the text that is already in the name field or (2) you are going to style the label with any **inline** HTML or CSS.

Do not use any **inline** HTML or CSS in the name field. You will see the raw HTML code on the map when you hover over such a symbol.

A label does not have to be text. The content of a label could be an **image**. The image does have to be online somewhere. To use an image as a label, the label field should look like:

```

```

You need to tweak the items inside the quotes.

If you want to assign a name, description and/or symbol short name to a **linepoint**, you can certainly do so. Remember to use a delimiter character to separate the different fields even if a field is empty.

To add a comment line in your file, start that line with either two slashes (//) or a semi-colon (;). Blank lines in the file are OK.

I. Summary of syntax for use in delimited text files

Command lines

coordinate=latitude,longitude (default) coordinate=longitude,latitude
Specify whether latitude or longitude comes first for the following coordinates.

file=_____ Link to a KML/KMZ file. Up to five 'file' lines can be used.

label=on (default) label=off
Specify whether or not labels will be displayed for the following coordinates.

line=off width=2 color=#FF0000 linesymbol=off dash=off (defaults)
When line=on then the following coordinates will be joined with a line. Width is in pixels. Color is any valid 6 character color code.
If linesymbol=off then symbols on a line are **not** automatically displayed.
If dash=off the line will be solid.

rest= Specify one or more layers of data from a GIS server that uses the REST interface.

symbol= name= Symbol is the link to an image not exceeding 64x64 pixels. Google seems to prefer png images, but jpgs will work. Name is a short name which will also be used on the coordinate lines.

wms= Specify one or more layers of data from a GIS server that uses the WMS interface.

Waypoint fields

Coordinate (required), name, description, short symbol name, label

Linepoint fields

Coordinate (required), name, description, short symbol name, label
Most people will likely just use the coordinate field.

Special keywords

Nosymbol	Prevent one waypoint from showing a symbol image. One use is to have a label on the map without any related symbol. Such a label could be used for a logo, your mug shot, or anything else.
Nolabel	Prevent one waypoint from showing a label