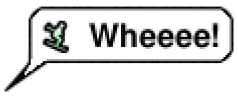
Warning: This API is deprecated. Please use the actively maintained Google Charts (/chart) API instead. See our deprecation policy (/chart/terms) for details.

Dynamic Icons



This page describes how to create a variety of callouts, bubbles, pins, and other graphics that can be requested by URL, or added as markers on top of other charts.

Table of Contents

Introduction

The Chart API enables you to create a variety of interesting callouts, pins, or bubbles that mix text and images. These items are called *dynamic icons*. You can either create a freestanding dynamic icon image, or you can position a dynamic icon on top of your chart as a marker type using the `chem` parameter. This page will describe what dynamic marker types exist, and how to create them as either freestanding images, or as markers on another chart.

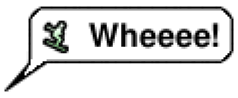
The syntax for creating a dynamic icon depends on whether you want a freestanding icon, or as a dynamic marker in another chart.

Freestanding Icons

You can request a dynamic icon image the same way you request any of the other charts. A freestanding dynamic icon supports a different set of parameters than other charts do:

Parameter	Required or Optional	Description
`chst=<icon_string_constant>`	Required	Describes which kind of icon to create. • *`icon_string_constant`* - A string constant describing the type of icon to create. See the selection of icons (#icon_types) below to choose an icon.
`chld=<icon_data>`	Required	The specific data used to describe the icon's size, rotation, text, and other required data. • *`icon_data`* - A set of pipe-delimited values appropriate to the icon. The documentation on this page describes which values are required by each icon type.
`cht`	NOT USED	Freestanding dynamic icon charts don't use the `cht` parameter.
`chs`	NOT USED	Freestanding dynamic icon charts don't use the `chs` parameter.
`chd`	NOT USED	Use the `chld` parameter to pass data to a freestanding dynamic icon.

Example

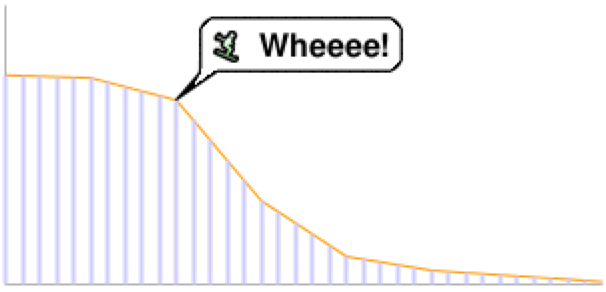


```
https://chart.googleapis.com/chart?chst=d_bubble_icon_text_small&chld=ski|bb|Wheeee!|FFFFFF|000000
```

Dynamic Markers

You can embed a dynamic icon as a marker type on several different types of charts using the `chem` parameter. See the `chem` documentation (https://developers.google.com/chart/image/docs/chart_params?hl=es#gcharts_dynamic_markers) to learn how.

Example



```
https://chart.googleapis.com/chart?
  chs=300x140
  cht=lc&chco=FF9900,224499
  chd=t:75,74,66,30,10,5,3,1
  chls=1|1
  chem=y;s=bubble_icon_text_small;d=ski,bb,Wheeee!,FFFFFF;dp=2;ds=0
  chm=v,ccccFF,0,::.2,2
```

Convert It!

Here's a tool to help you convert between dynamic icon markers and freestanding dynamic icons. Fill in the appropriate boxes and push the button to convert your icon type.

Marker Syntax		Freestanding Icon Syntax
chem		chst
chem=	TO FREESTANDING ICON > < TO MARKER	chst=
		chld
		chld=

Common Icon Features

Most icons can have text strings or shadows associated with them.

Text Strings

All display text passed to the Chart API must be UTF-8 encoded and then URL-encoded. This affects only non-URL-safe characters (URL-safe characters are mostly the English letters a-z both upper and lower case, plus plus a small set of punctuation). For example, the UTF-8 and URL-encoded value for the letter "è" is "`%C3%A8`", and for the Chinese character 駅 is "`%E9%A7%85`". Most browsers will let you use an unencoded value in the URL string (for example, 駅) and will encode it for you behind the scenes. However, it is possible that someone viewing your chart URL is using a browser that doesn't do this, so it is usually best to UTF-8 and URL-encode all non-ASCII characters in text strings. Note that this is only for the text shown in bubbles or pins, not for the &, |, or other characters that are part of the URL syntax.

When using the `chem` parameter to specify dynamic icon markers, you must also escape certain characters in your text, as described in the `chem` (https://developers.google.com/chart/image/docs/chart_params?hl=es#gcharts_dynamic_markers) documentation.

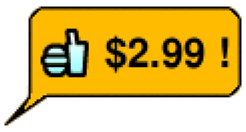
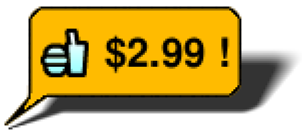
Shadows

You can add shadows to many icons, or even draw shadows for some icons without the icon itself!

Shadowed Icons

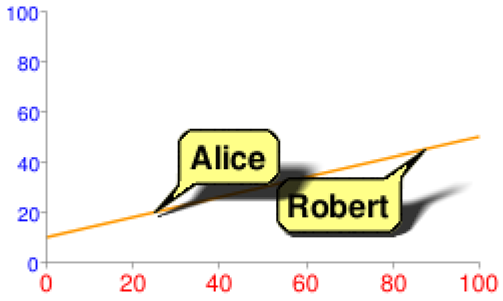
Many of these icons can be drawn with or without shadows. If shadowing is an option, the icon name will have a version that ends in `_withshadow` and another version without that ending. You can specify an icon with either ending, depending on whether you want the shadow or not.

Here's an example of a medium text bubble and a pin with and without shadows:

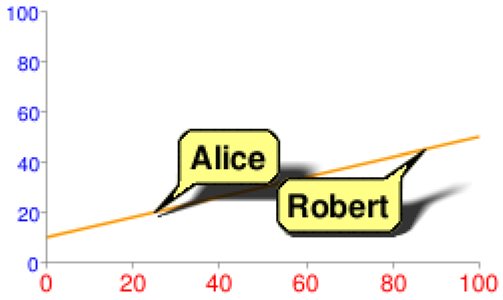
 chst=d_bubble_icon_text_big	 chst=d_bubble_icon_text_big_withshadow
 chst=d_map_pin_icon	 chst=d_map_pin_icon_withshadow

Freestanding Shadows

Some icon types allow you to draw their shadow by itself. You might want to do this if you are using several overlapping shadowed icons on a graphic, and they are so close that the shadow from one falls across another icon. For example, here is two shadowed bubbles, Robert drawn first, then Alice:



Note how Alice's shadow partially covers Robert. To fix this, you can draw the Alice shadow first, then the Robert bubble, then Alice without a shadow. Perhaps not totally realistic in terms of lighting and shadows, but it avoids obscuring one bubble with the shadow of another:



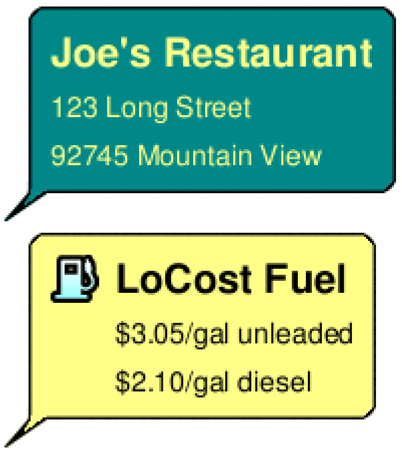
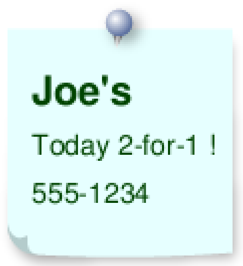
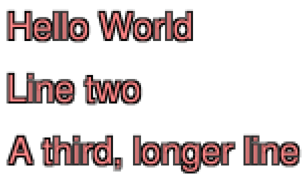
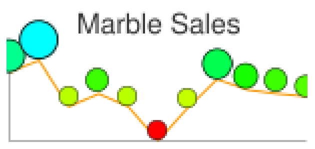
```
chem=
  y;s=bubble_text_small_shadow;d=bb,Alice,FF8,000;ds=0;dp=1;py=1          // Alice, no shadow
  y;s=bubble_text_small_withshadow;d=bbtr,Robert,FF8,000;ds=0;dp=3.5;py=1 // Robert with shadow
  y;s=bubble_text_small;d=bb,Alice,FF8,000;ds=0;dp=1;py=1                 // Alice shadow
```

All markers specify the same z-order of 1 (`py=1`), so they are drawn in the order listed, on top of the chart elements (the chart line). First the Alice shadow is drawn, then the Robert bubble on top of that, and finally the Alice bubble on top.

See the documentation of your specific icon type to learn whether you can draw its shadow alone.

Icon Types

Type	Examples
Bubbles (#bubbles) - Choose small or large text bubbles, with or without icons.	

	 Joe's Restaurant 123 Long Street 92745 Mountain View LoCost Fuel $3.05/gal unleaded $2.10/gal diesel
Pins (#pins) - Pin types can be plain, starred, or slanted, and can have an icon, a single letter, or longer text strings.	
Fun style notes with text and optional title (#fun_style_note)	
Weather forecast notes with title, text, and weather icon (#weather_forecast_note)	Barcelona max 25°C min 15°C
Multiline outlined text block, no icon (#outlined_text)	
Single line outlined textblock with icon (#single_line_outlined_text_icon)	Movies Hospital
Contextual icons (#contexticons)	

Back to top (#top)

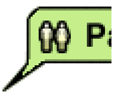
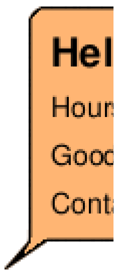
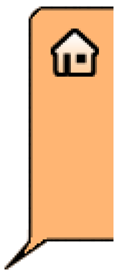
Bubbles

Bubbles come in various combinations of features: large or small size, with or without icon, single line or multi-line text. The following table describes the types of bubbles, and the syntax for each. The description of each parameter value is described below the table.

A few additional notes:

- All bubbles resize to fit the width and height of the text that you enter.
- When using the bubble as a freestanding dynamic icon (#freestanding_icons) use the `chst` bubble constant; when using the bubble as a embedded dynamic icon markers (https://developers.google.com/chart/image/docs/chart_params?hl=es#gcharts_dynamic_markers), use the `chem` bubble constant.
- The bubble constant supports a variation ending in `_withshadow`. If you use the `_withshadow` variation, it adds a shadow (#shadows) below your bubble.
- When using the bubble as an embedded dynamic icon marker, the `chld` string syntax is the same, except that all | delimiters are replaced by , as described in the embedded dynamic icon markers (https://developers.google.com/chart/image/docs/chart_params?hl=es#gcharts_dynamic_markers) documentation.

Text	Icon	`chst` (freestanding) and `chem` (marker) Values	`chld` Syntax	Shadow Only	Exampl

<table>
<tr><td>Single line</td><td>No</td><td>chst=d_bubble_text_small[_withshadow]
chem=bubble_text_small[_withshadow]</td><td>chld=
<frame_style (#frame_style_constants)>|
<text>|
<fill_color>|
<text_color></td><td>chst=
[d_]bubble_text_small_shadow
chld=
<frame_style (#frame_style_constants)>|
<text></td><td>
chst=d_
chld=
bb|
Launc
C6EF8
00000</td></tr>
<tr><td>Single line</td><td>Yes (#icon_list)</td><td>chst=d_bubble_icon_text_small[_withshadow]
chem=bubble_icon_text_small[_withshadow]</td><td>chld=
<icon_string (#icon_list)>|
<frame_style (#frame_style_constants)>|
<text>|
<fill_color>|
<text_color></td><td>chst=
[d_]bubble_icon_text_small_shadow
chld=
<icon_string (#icon_list)>|
<frame_style (#frame_style_constants)>|
<text></td><td>
chst=d_
chld=
wc|
bb|
Pay+T
C6EF8
00000</td></tr>
<tr><td>Single line</td><td>Yes (#icon_list)</td><td>chst=d_bubble_icon_text_big[_withshadow]
chem=bubble_icon_text_big[_withshadow]</td><td>chld=
<icon_string (#icon_list)>|
<frame_style (#frame_style_constants)>|
<text>|
<fill_color>|
<text_color></td><td>chst=
[d_]bubble_icon_text_big_shadow
chld=
<icon_string (#icon_list)>|
<frame_style (#frame_style_constants)>|
<text></td><td>
chst=d_
chld=
snack
bb|
$2.99
FFBB0
00000</td></tr>
<tr><td>Multi-line</td><td>No</td><td>chst=d_bubble_texts_big[_withshadow]
chem=bubble_texts_big[_withshadow]</td><td>chld=
<frame_style (#frame_style_constants)>|
<fill_color>|
<text_color>|
<text_line_1>
|...|
<text_line_n></td><td>chst=
[d_]bubble_texts_big_shadow
chld=
<frame_style (#frame_style_constants)>|
<text_line_1>
|...|
<text_line_n></td><td>
chst=d_
chld=
bb|
FFB57
00000
Help+
Hours
Good+
Conta</td></tr>
<tr><td>Multi-line</td><td>Yes (#icon_list)</td><td>chst=d_bubble_icon_texts_big[_withshadow]
chem=bubble_icon_texts_big[_withshadow]</td><td>chld=
<icon_string (#icon_list)>|
<frame_style (#frame_style_constants)>|
<fill_color>|
<text_color>|
<text_line_1>
|...|
<text_line_n></td><td>chst=
[d_]bubble_icon_texts_big_shadow
chld=
<icon_string (#icon_list)>|
<frame_style (#frame_style_constants)>|
<text_line_1>
|...|
<text_line_n></td><td>
chst=d_
chld=
home
bb|
FFB57
00000
Haunt
Hours
Ghoul
+and+Va
Bring</td></tr>
</table>

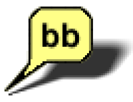
Syntax

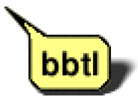
<icon_string>

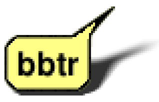
A string specifying one of the icons (#icon_list) described at the end of this document.

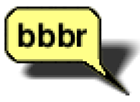
<frame_style>

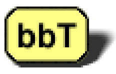
The tail direction. Choose of the following tail direction constants:

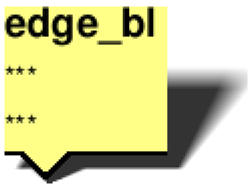
- `bb` - Balloon frame, tail at bottom left 

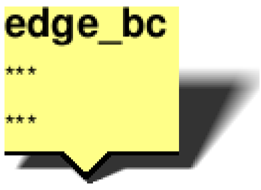
- `bbtl` - Balloon frame, tail at top left 

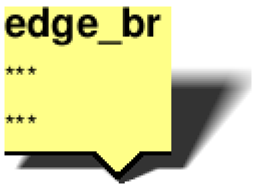
- `bbtr` - Balloon frame, tail at top right 

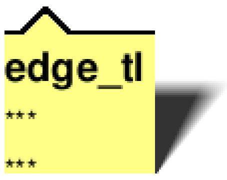
- `bbbr` - Balloon frame, tail at bottom right 

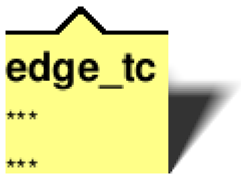
- `bbT` - Balloon frame, no tail 

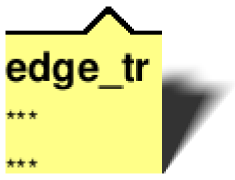
- `edge_bl` - Edge frame, tail at bottom edge, left end 

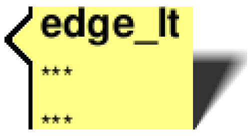
- `edge_bc` - Edge frame, tail at bottom edge, centered 

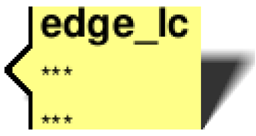
- `edge_br` - Edge frame, tail at bottom edge, right end 

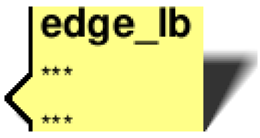
- `edge_tl` - Edge frame, tail at top edge, left end 

- `edge_tc` - Edge frame, tail at top edge, centered 

- `edge_tr` - Edge frame, tail at top edge, right end 

- `edge_lt` - Edge frame, tail at left edge, top end 

- `edge_lc` - Edge frame, tail at left edge, centered 

- `edge_lb` - Edge frame, tail at left edge, bottom end 

- `edge_rt` - Edge frame, tail at right edge, top end

- `edge_rc` - Edge frame, tail at right edge, centered

- `edge_rb` - Edge frame, tail at right edge, bottom end

<fill_color>

The bubble fill color, as a six-digit HTML hexadecimal color.

<text_color>

The text color, as a six-digit HTML hexadecimal color.

<text>

A single line of bubble text for one-line bubbles. Spaces must be + marks.

<text_line_1>|...|<text_line_n>

One or more lines of text, for multi-line text bubbles. Each line is separated by a | mark. The first line will be shown larger and boldface. Spaces must be replaced by +.

Shadow Only

For bubbles, you can also draw the freestanding shadow using the syntax shown in the table above. Examples:

```
chst=
  d_bubble_text_small_shadow
chld=
  bb|Launch+site
```

```
chst=
  d_bubble_icon_text_small_shadow
chld=
  wc|bb|Pay+Toilets
```

```
chst=
  d_bubble_icon_text_big_shadow
chld=
  snack|bb|$2.99+!
```

```
chst=
  d_bubble_texts_big_sha
chld=
  bb|Help+Wanted|
  Hours:+Midnight+to+3:0
  Good+pay,+interesting+
  Contact+V.+Helsing
```

Back to top (#top)

Pins

You can make a variety of pins with icons and/or short text strings. Here are the available pin types.

Pin Type	Bubble Constant	Examples

Plain pin with single letter or icon (#plain_pin)	`chst=d_map_pin_letter[_withshadow]`
	`chst=d_map_pin_icon[_withshadow]`
	
	`chem=map_pin_letter[_withshadow]`
	`chem=map_pin_icon[_withshadow]`	

Slanted/starred pin with single letter or icon (#special_pins)	`chst=d_map_xpin_letter[_withshadow]`
	`chst=d_map_xpin_icon[_withshadow]`
	
	`chem=map_xpin_letter[_withshadow]`
	`chem=map_xpin_icon[_withshadow]`	

Scalable, rotatable, multi-line pin (#scalable_pins)	`chst=d_map_spin`	

Plain Pin with Single Letter or Icon

This is a small, upright pin that can hold either a small icon or a single letter.

Letter Syntax

```
chst=d_map_pin_letter[_withshadow(#shadows)]
chld=<character>|<fill_color>|<text_color>
```

Icon Syntax

```
chst=d_map_pin_icon[_withshadow(#shadows)]
chld=<icon_string>|<fill_color>
```

<character>

[*Text pins only*] A single text character.

<icon_string>

[*Icon pins only*] A string specifying one of the icons (#icon_list) described at the end of this document.

<fill_color>

The bubble fill color, as a six-digit HTML hexadecimal color.

<text_color>

[*Text pins only*] The text color, as a six-digit HTML hexadecimal color.

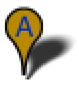
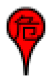
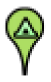
Shadow Only

To draw the shadow only for this pin type, use this syntax:

```
chst=d_map_pin_shadow
```

The `chld` parameter is not required for a shadow only.

Examples


`chst=d_map_pin_letter_withshadow`
`chld=A|FF0000|0000FF`
	
	`chst=d_map_pin_letter`
	`chld=%E5%8D%B1|FF0000|000000`
		
		`chst=d_map_pin_icon`
		`chld=camping|ADDE63`
			
			`chst=d_map_pin_shadow`

Back to top (#top)

Slanted/Starred Pin with Single Letter or Icon

This is a small, pin that can either be tilted left or right, or have a star overlap. The contents of the pin can be either a single character or a small icon.

Letter Syntax

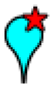
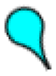
```
chst=d_map_xpin_letter[_withshadow (#shadows)]
chld=<pin_style>|<character>|<fill_color>|<text_color>|<star_fill_color>
```

Icon Syntax

```
chst=d_map_xpin_icon[_withshadow (#shadows)]
chld=<pin_style>|<icon_string>|<fill_color>|<star_fill_color>
```

<pin_style>

The pin style. Choose one of the following constants:

- `pin`
- `pin_star` 
- `pin_sleft` 
- `pin_sright`

<icon_string>

[*Icon pins*] A string specifying one of the icons (#icon_list) described at the end of this document.

<character>

[*Text pins*] A single text character.

<fill_color>

The bubble fill color, as a six-digit HTML hexadecimal color.

<text_color>

[*Text pins*] The text color, as a six-digit HTML hexadecimal color.

<star_fill_color>

[*Star pins*] The fill color of the star, as a six-digit HTML hexadecimal color.

Shadow Only

To draw the shadow only for this pin type, use this syntax:

```
chst=d_map_xpin_shadow

chld=<pin_style>
```

Examples

```
chst=d_map_xpin_letter
chld=pin_star|A|FF0000|
000000
```

```
chst=d_map_xpin_letter
chld=pin_sleft|A|FF0000|
000000
```

```
chst=d_map_xpin_icon
chld=pin_sleft|bank-dollar|
52B552
```

```
chst=d_map_xpin_shadow
chld=pin_sleft
```

Back to top (#top)

Text Pin with Scaling and Rotation

Up
Up
And away!

This is a pin that can you can manually scale to include longer text strings. You can also rotate the pin a custom amount, and control font size and color.

Syntax

```
chst=d_map_spin
chld=<scale_factor>|<rotation_deg>|<fill_color>|<font_size>|<font_style>|<text_line_1>|...|<text_line_n>
```

<scale_factor>

A scaling factor to specify the pin size. This is a positive floating point number where 0.5 is the size of unscaled pins. 0.25 would be half that size, 1 would be twice that size, and so on.

<rotation_deg>

The rotation of the pin, in degrees. Positive and negative values are allowed. Specify 0 for a vertical pin.

<fill_color>

The bubble fill color, as a six-digit HTML hexadecimal color.

<font_size>

The text font size, in pixels.

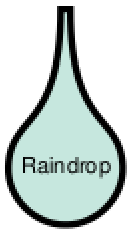
<font_style>

Either '_' (underscore) for normal text or 'b' for boldface text.

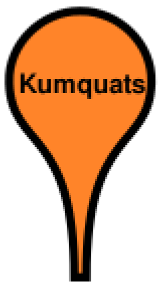
<text_line_1>...<text_line_n>

One or more lines of text, delimited by | characters.

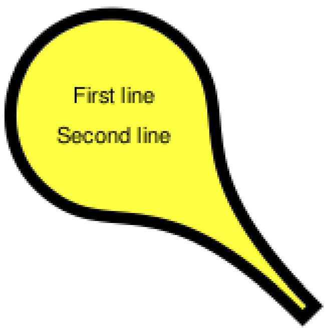
Examples



```
chst=d_map_spin
chld=1.7|180|C6E7DE|11|_|
Raindrop
```



```
chst=d_map_spin
chld=2.1|0|FFFF42|13|b|
Kumquats
```



```
chst=d_map_spin
```

```
chld=3|45|FFFF42|11|_|First+line|
Second+line
```

Back to top (#top)

Fun Style Notes

More information here

You can create a variety of text notes in novelty templates, such as a sticky note or a thought bubble. You can optionally include a title line in the note.

These notes have a fixed size; they will not grow or shrink to accommodate the size of your text.

Syntax

```
chst=d_fnote_title OR chst=d_fnote
chld=<note_type>|<note_size>|<text_color>|<text_alignment>|<text_line_1>|...|<text_line_n>
```

chst

Specify either `d_fnote_title` for a note with a title, or `d_fnote` for a note without a title. In a note with a title, the first line of text will be formatted as a title (larger and boldface).

<note_type>

A string describing the note shape. Choose one of the note type strings in the table below (#note_type_templates).

<note_size>

1 (one) for a large note, or 2 for a small note. Template sizes are fixed; templates do not grow or shrink to fit text. Experiment with both sizes to see which holds your text better.

<text_color>

The text color, as a six-digit hexadecimal color; alpha values are not accepted.

<text_alignment>

The alignment for all text, including the header. Choose one of the following values:

- l - ('L') Left-aligned
- h - Center-aligned
- r - Right-aligned

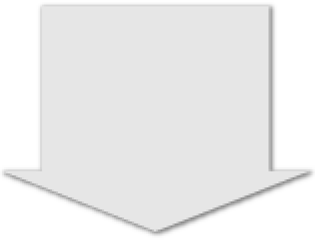
<text_line_1>| ...|<text_line_n>

The text for the note. Use the | character to indicate a newline. When `chst=d_fnote_title`, the first line of text is formatted as a title.

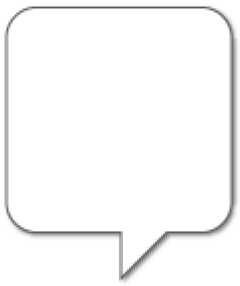
Note Type Templates

The following templates are supported for note templates. The *note_type* string for each is shown below the template.

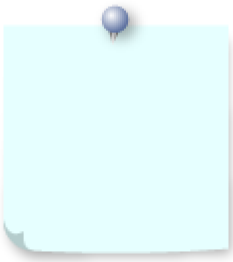
Note: Text strings are case-sensitive!



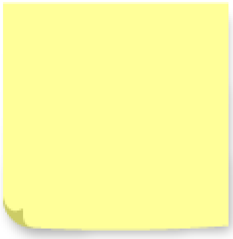
`arrow_d`



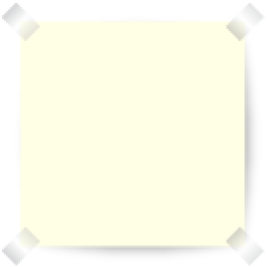
`balloon`



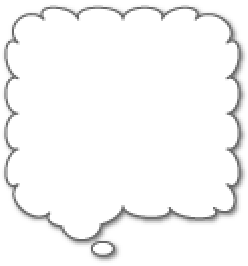
`pinned_c`



`sticky_y`



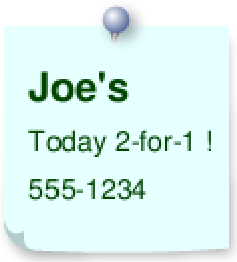
`taped_y`

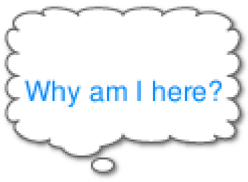


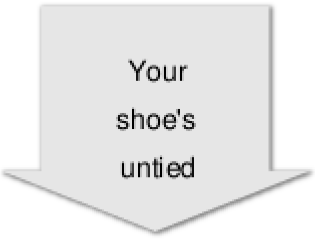
`thought`

Examples

Description	Example

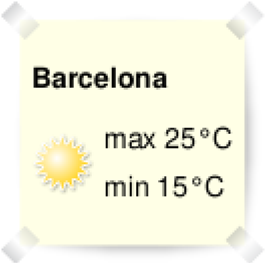
Note with a title: `chst=d_fnote_title`. The first line of text is used as the title. Size is large note (1).	
	`chst=d_fnote_title`
	`chld=pinned_c|1|004400|l|Joe's|Today+2-for-1+!|555-1234`

Note without a title: `chst=d_fnote.` Text is centered.	
	`chst=d_fnote`
	`chld=thought|1|0088FF|h|Why+am+I+here?`

A useful template for an old joke.	
	`chst=d_fnote`
	`chld=arrow_d|1|000000|h|Your|shoe's|untied`

Back to top (#top)

Weather Forecast Notes



You can create a note with a weather indicator icon. Similar layout to fun style notes, but text color is always black, text is always left-aligned, and you can only have from one to three lines of text.

These notes have a fixed size; they will not grow or shrink to accommodate the size of your text.

Syntax

```
chst=d_weather
chld=<note_type>|<weather_icon>|<title>|<line_2>|<line_3>
```

chst

`d_weather` indicates a weather note.

<note_type>

The template to use for this note. Use one of the note strings listed in the fun style note list (#note_type_templates) above.

<weather_icon>

One of the weather icon strings in the table below. A note can take a single weather icon.

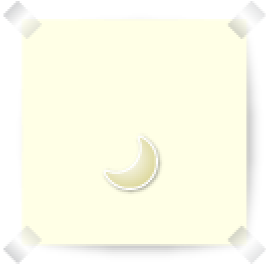
<title>|<line_2>|<line_3>

The title, plus up to two more lines of text (these extra text lines are optional). The title is larger and boldface.

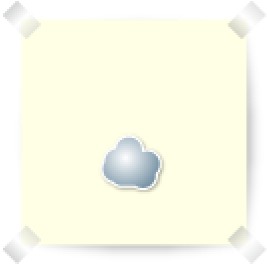
Weather Icons

Here is a list of supported weather icons, showing the *weather_icon* string for each.

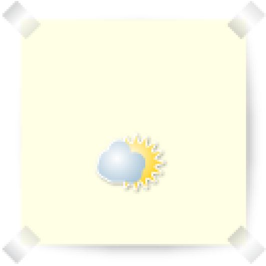
Note: Text strings are case-sensitive!



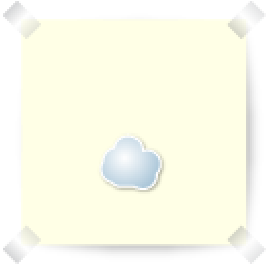
`clear-night-moon`



`cloudy-heavy`



`cloudy-sunny`



`cloudy`

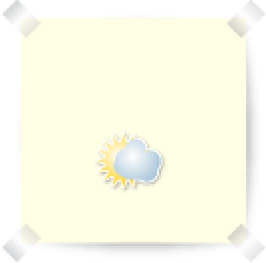
`rain`

`rainy-sunny`

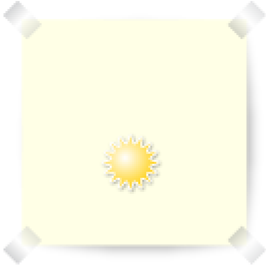
`snow`

`snowflake`

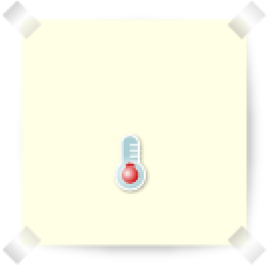
`snowy-sunny`



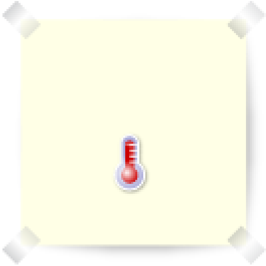
`sunny-cloudy`



`sunny`

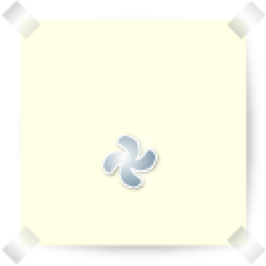


`thermometer-cold`



`thermometer-hot`

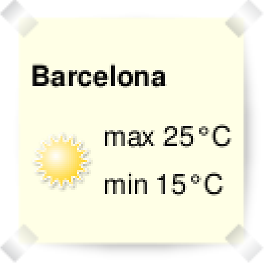
`thunder`



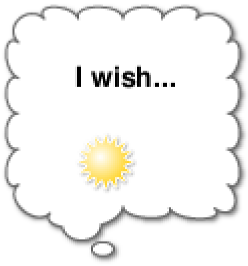
`windy`

Examples

Description	Example

A weather note with a title and two lines on a `taped_y` style template.	
	`chst=d_weather`
	`chld=taped_y|sunny|Barcelona|max+25°C|min+15°C`

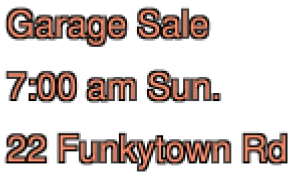
A wishful thought for winter.	



```
chst=d_fnote
chld=thought|sunny|I+wish...
```

Back to top (#top)

Outlined Font Text Blocks (no icon)


Garage Sale
7:00 am Sun.
22 Funkytown Rd

You can create a block of outlined text with a white background. If you want an icon, consider using the single line variety (#single_line_outlined_text_icon), which supports an icon.

Syntax

```
chst=d_text_outline
chld=<text_fill_color>|<font_size>|<text_alignment>|<outline_color>|<font_weight>|<text_line_1>|...|<text_line_n>
```

<text_fill_color>

The text fill color. This is a six-digit hexadecimal color; alpha values are not accepted.

<font_size>

A number specifying the font size, in pixels.

<text_alignment>

The alignment for all text, including the header. Choose one of the following values:

- l - ('L')Left-aligned
- h - Center-aligned
- r - Right-aligned

<outline_color>

The text outline color. This is a six-digit hexadecimal color; alpha values are not accepted.

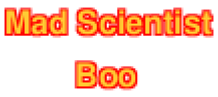
<font_weight>

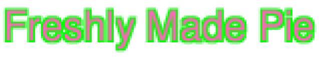
Normal or bold text. Underscore ' _' for normal text, and 'b' for bold text.

<text_line_1>| ...|<text_line_n>

The text for the note. Use the | character to indicate a newline. When `chst=d_fnote_title`, the first line of text is formatted as a title.

Examples


Mad Scientist
Boo


Freshly Made Pie

```
chst=d_text_outline
chld=FFCC33|16|h|FF0000|b|Mad+Scientist|Boo
```

```
chst=d_text_outline
chld=FF65BB|20|h|00FF00|_|Freshly+Made+Pie
```

Back to top (#top)

Outlined Font Text Blocks (icon)

You can create a single line of outlined text plus an icon on the top, left, bottom, or right of the block. If you don't need an icon, use the outlined font text block (#outlined_text).

Syntax

```
chst=<icon_position_string>
chld=<text>|<font_size>|<font_fill_color>|<icon_name>|<icon_size>|<icon_fill_color>|<icon_and_text_border_color>
```

<icon_position_string>

Specifies where the icon appears in the text box. Choose one of the following values:

- `d_simple_text_icon_below` - Places the icon on the bottom of the box:
- `d_simple_text_icon_above` - Places the icon on the top of the box:
- `d_simple_text_icon_left` - Places the icon on the left of the box:
- `d_simple_text_icon_right` - Places the icon on the right of the box:

<text>

The text to show. One line only; use + for spaces.

<font_size>

A number specifying the font size, in pixels.

<font_fill_color>

The fill color for the text, as a six-digit string. Alpha values are not supported.

<icon_name>

One of the icon names listed at the bottom of this document.

<icon_size>

The height of the icon, in pixels. The following values are supported: 12, 16, 24.

<icon_fill_color>

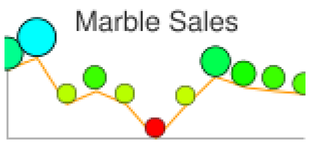
The color of the icon, as a six-digit string. Alpha values are not supported.

<icon_and_text_border_color>

The color of the border around the icon and the text, as a six-digit string. Alpha values are not supported.

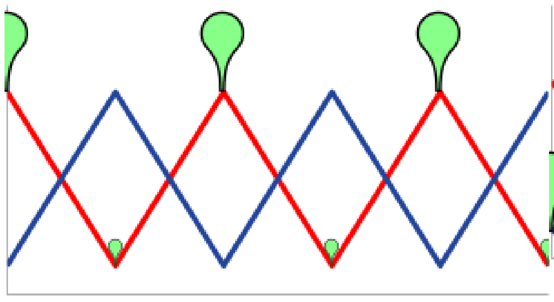
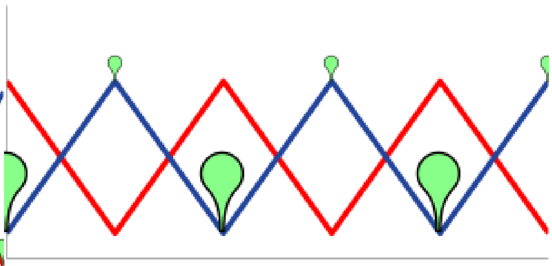
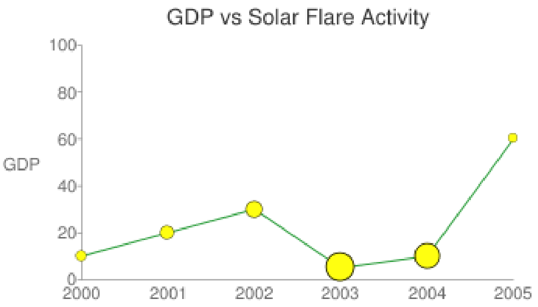
Back to top (#top)

Contextual Icons (Dynamic Markers Only)



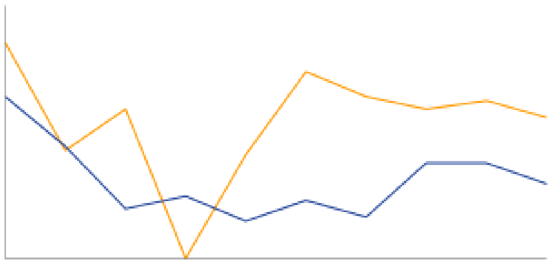
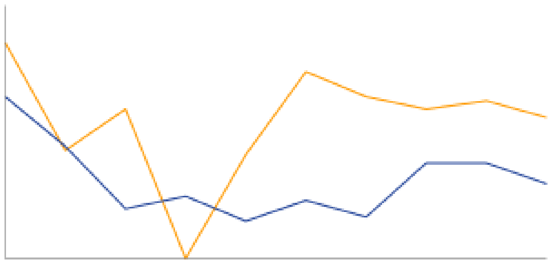
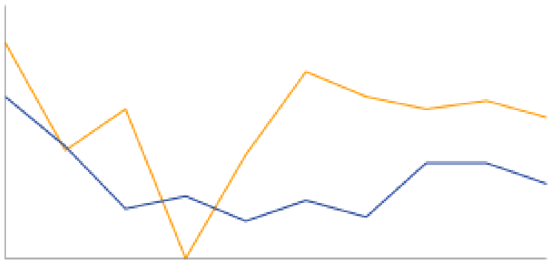
You can specify an icon that varies its color, size, or stacking according to the point they are assigned to. These icon types are available as dynamic icon markers only (`chem` parameter), not as freestanding icons.

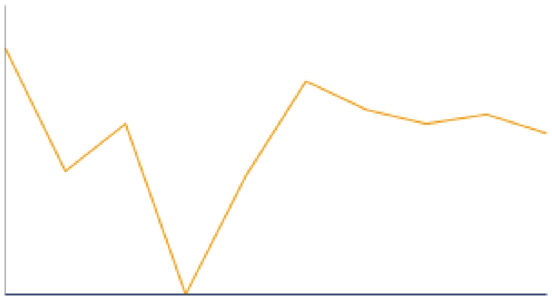
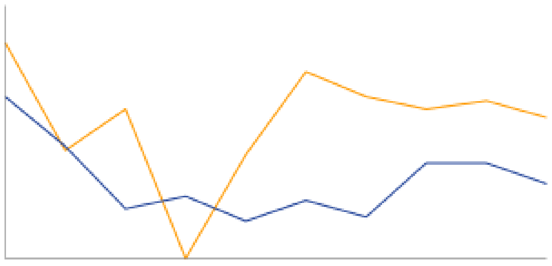
These icons can be rendered on a series other than the series that specifies their color, size, or stacking information. This means that the `chem` parameter's `ds` value specifies the series on which to render the icon, but the values for determining the size or color of the icon are specified in the parameters given below. One good use of this is to use a hidden data series for icon data, but render the icons on a visible line or bar. Here are some examples:

Icon rendered on source series	Icon rendered on non-source series	Icon using hidden series
		

`chem=`
`  y;s=cm_size;ds=0;...;`
`  d=maps_pin,0,10,50,10,8F8,000,`
`hb`
• `ds=0` - Render on series 0 (red line)
• `d=maps_pin,0,...` - Data from series 0
	`chem=`
	`  y;s=cm_size;ds=1;...;`
	`  d=maps_pin,0,10,50,10,8F8,000,`
	`hb`
	• `ds=1` - Render on series 1 (blue line)
	• `d=maps_pin,0,...` - Data from series 0
		`chd=t1:`
		`  10,20,30,5,10,60`
		`  10,20,30,70,60,5`
		`chem=`
		`  y;s=cm_size;ds=0;dp=all;`
		`  d=disk,1,5,20,5,FFFF10,000`
		• `t1:` - First series is shown and used line data. All later series are hidden.
		• `ds=0` - Markers rendered on series 0.
		• `d=disk,1`,... - Disk data comes from hidden series 1.

Context Marker Types

Marker Type	chem s Value	Example
Color variation (#contextcolorvariation)	`s=cm_color`	
Size variation (#contextsizevariation)	`s=cm_size`	
Color and size variation (#contextcolorandsizevariation)	`s=cm_color_size`	
Stacking variation (#contextstacking)	`s=cm_repeat`	

		
Stacking and color variation (#contextstackingandcolor)	s=cm_repeat_color	

Alignment Strings for Contextual Icons

The contextual icons support an optional alignment string to specify an alignment and offset of the icon to the data point. This string has the following syntax:

```
<alignment>[+/-<h_anchor_offset>+/-<v_anchor_offset>]
```

alignment

Two letters describing the alignment of the icon to the point. Examples include `tl` (top left), and `rb` (bottom right). See the *alignment_string* parameter description of the `chem` parameter (https://developers.google.com/chart/image/docs/chart_params?hl=es#embedded_charts) for a complete listing and description.

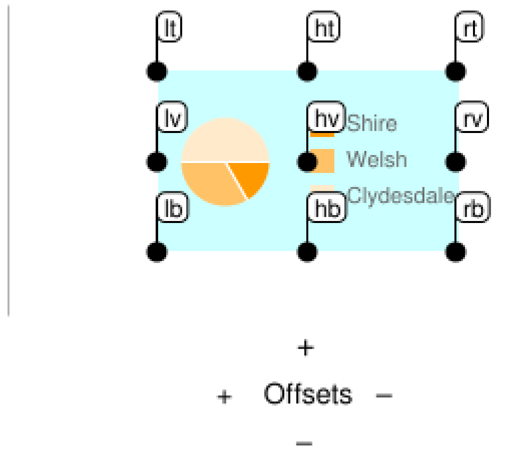
h_anchor_offset

[*Optional*] The horizontal offset of the anchor point, in pixels. Values *including zero* must be preceded by either + or -. **Important:** You must url-encode + as %2B.

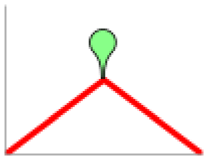
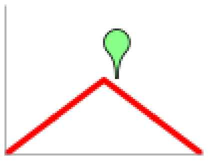
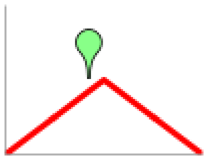
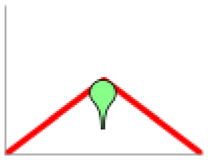
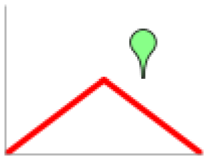
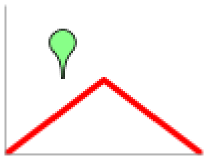
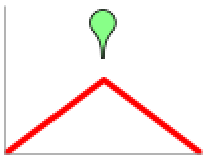
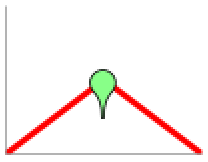
v_anchor_offset

[*Optional*] The vertical offset of the anchor point, in pixels. Values *including zero* must be preceded by either + or -. **Important:** You must url-encode + as %2B.

Note that you can also use the `of` component of the `chem` parameter to specify horizontal and vertical offsets. If you specify both the `of` component and `h_anchor_offset v_anchor_offset` values, all offsets will be applied to your icon.



Examples:

 `hb-0-0` Horizontal center bottom No offsets	 `lb-0-0` Bottom left No offsets	 `rb-0-0` Bottom right No offsets	 `ht-0-0` Horizontal top No offsets
			

`hb-20-0`	`hb%2b20-0`	`hb-0%2b10`	`hb-0-20`
Horizontal center bottom	Horizontal center bottom	Horizontal center bottom	Horizontal center bottom
-20 horizontal	+20 horizontal	0 horizontal	0 horizontal
0 vertical	0 vertical	+10 vertical	-20 vertical

Color Variation (`cm_color`)

You can vary the color of a contextual chart marker according to the point that it represents. You must specify a color range, and the data value will be scaled to a corresponding color within that range.

Syntax

```
chem=y;s=cm_color;ds=<series_rendering_index>; ...other_values... (https://developers.google.com/chart/image/docs/chart_params
  d=<icon_shape>,<color_data_series>,<low_color>,<middle_color>,<high_color>,<icon_size>,<outline_color>,<alignment>
```

<icon_shape>

The icon to use. Specify an ID string identifying one of the images listed at the end of the page (#icon_list).

<color_data_series>

The zero-based index of the data series used to vary the color of the icons.

<low_color>

The low color value in the range, as a three- or six-digit HTML hexadecimal color (no # mark). This is associated with the lowest possible value in the available data range.

<middle_color>

The middle color value in the range, as a three- or six-digit HTML hexadecimal color (no # mark). This is associated with the middle value in the available data range.

<high_color>

The high color value in the range, as a three- or six-digit HTML hexadecimal color (no # mark). This is associated with the highest possible value in the available data range.

<icon_size>

The size of the icon, in pixels. The following values are supported: 12, 16, 24.

<outline_color>

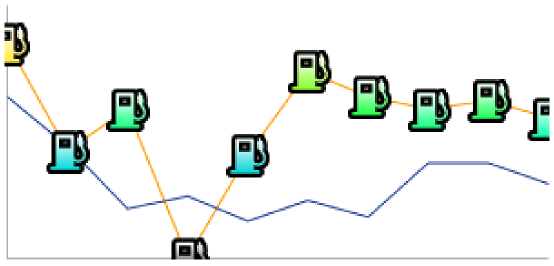
The outline color for the icon, as a three- or six-digit HTML hexadecimal color (no # mark).

<alignment>

An optional string (#contextualaligmentstrings) describing the icon alignment and offset.

Example

- `s=cm_color` - color variation icon
- `ds=0` - Render them on data series 0
- `dp=all` - Put an icon on all points.
- d values:
  - `petrol` - Icon identifier
  - `0` - Color from data series 0
  - `000,0FF,F55` - Definition of the color scale
  - `24` - Size of icon
  - `000` - Black outlines
  - `hv` - Align icons on center with the point both horizontally and vertically.



```
chem=y;s=cm_color;
  ds=0;
  dp=all;
  d=petrol,0,000,0FF,F55,24,000,hv
```

Back to top (#top)

Size Variation (`cm_size`)

You can vary the size alone of a contextual chart marker, according to the data series of your choice.

Syntax

`chem=y;s=cm_size;ds=<series_rendering_index>;` *`...other_values...`* (https://developers.google.com/chart/image/docs/chart_params?
`d=<icon_type>,<size_data_series>,<zero_value_size>,<size_multiplier>,<min_size>,<outline_color>,<fill_color>,<aligı`

<icon_type>

The shape of the icon. **Choose one of the following values: `maps_pin`, `disk`, or `square`.**

<size_data_series>

The zero-based index of the data series used to vary the size of the icons.

<zero_value_size>

The base size of the icon, at the minimum data value for the series.

<size_multiplier>

A size scaling factor. This value is multiplied against the difference between each icon's data value and the minimum series value, to calculate the final icon size. Therefore, an icon at the 0 data value will not be affected by this multiplier.

<min_size>

The minimum size for any icon, in pixels.

<outline_color>

The outline color for the icon, as a three- or six-digit HTML hexadecimal color (no # mark).

<fill_color>

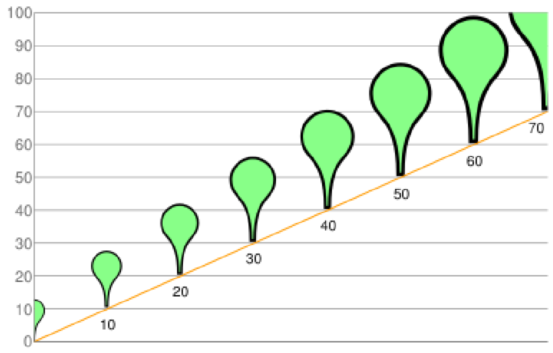
The fill color for the icon, as a three- or six-digit HTML hexadecimal color (no # mark).

<alignment>

An optional string (#contextualaligmentstrings) describing the icon alignment and offset.

Examples

A basic example. The icon with zero value is rendered at the zero value size, which is 30 pixels. Sizes increase along with the data.

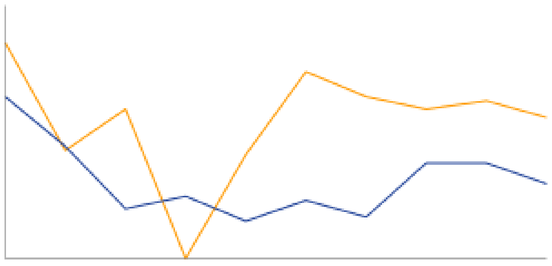


```
chd=t:0,10,20,30,40,50,60,70
chem=y;s=cm_size;ds=0;dp=all;py=-1;d=maps_pin,0,
30,100,10,8F8,000,hb
```

In this example, the icons take their size data from the yellow line, but are rendered on the blue line.

- `chem=y` - Dynamic marker

- `s=cm_size` - Size variance
- `ds=1` - Rendered on data series 1 (the blue line)
- `dp=all` - Rendered on all points.
- `d=`
    - `maps_pin` - Use a map pin icon
    - `0` - Scale the size according to data series 0 (the yellow line)
    - `10` - Pin size at value 0
    - `90` - Size multiplier
    - `10` - Minimum size
    - `8F8` - Fill color
    - `000` - Outline color
    - `hb` - Center them horizontally at the bottom of the pin.



`chem=chem=y;s=cm_size;ds=1;dp=all;d=maps_pin,0,10,90,10,8F8,000,hb`

Back to top (#top)

Color and Size Variation (`cm_color_size`)

You can vary both color and size of a contextual chart marker, according to the data series of your choice.

Syntax

```
chem=y;s=cm_color_size;ds=<series_rendering_index>; ...other_values... (https://developers.google.com/chart/image/docs/chart_p
  d=<icon_type>,<color_data_series>,<low_color>,<middle_color>,<high_color>,<size_data_series>,<zero_value_size>,<si
```

<icon_type>

The shape of the icon. **Choose one of the following values: `maps_pin`, `disk`, or `square`.**

<color_data_series>

The zero-based index of the data series used to vary the color of the icons.

<low_color>

The low color value in the range, as a three- or six-digit HTML hexadecimal color (no # mark). This is associated with the lowest possible value in the available data range.

<middle_color>

The middle color value in the range, as a three- or six-digit HTML hexadecimal color (no # mark). This is associated with the middle value in the available data range.

<high_color>

The high color value in the range, as a three- or six-digit HTML hexadecimal color (no # mark). This is associated with the highest possible value in the available data range.

<size_data_series>

The zero-based index of the data series used to vary the size of the icons.

<zero_value_size>

The base size of the icon, at the minimum data value for the series.

<size_multiplier>

A size scaling factor. This value is multiplied against the difference between each icon's data value and the minimum series value, to calculate the final icon size. Therefore, an icon at the 0 data value will not be affected by this multiplier.

<min_size>

The minimum size for any icon, in pixels.

<outline_color>

The outline color for the icon, as a three- or six-digit HTML hexadecimal color (no # mark).

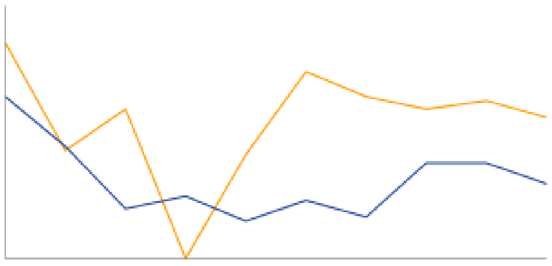
<alignment>

An optional string (#contextualaligmentstrings) describing the icon alignment and offset.

Examples

This example uses two lines. The pins use color data from the series on which they're rendered, but use size data from the other series.

- `chd=s:0akAZtnkmi,nbMPJOKXXS` - First series is used to plot the yellow line and determine the pin color. The second series is used for the blue line, and the pin size.
- `chem=y` - Render a dynamic icon
- `s=cm_color_size` - Use the color and size variation context icon.
- `ds=1` - Render the items on data series 1, the blue line.
- `dp=all` - Put the icon on all points.
- `d=`
  - `maps_pin` - Use the map pin symbol.
  - `1` - Use data series 1 for the pin color.
  - `000,0FF,F55` - The low, medium, and high colors.
  - `0` - Use data series 0 for the pin size.
  - `10` - Pin is 10 pixels at value 0.
  - `90` - A size multiplier of 90.
  - `10` - Minimum pin size of 10 pixels.
  - `000` - Black outline color.
  - `hb` - Center the pins horizontally along the bottom edge to each data point.



```
chd=s:0akAZtnkmi,nbMPJOKXXS
chem=y;s=cm_color_size;ds=1;dp=all;d=maps_pin,
1,000,0FF,F55,0,10,90,10,000,hb
```

Back to top (#top)

Stacking Variation (`cm_repeat`)

You can vary the height of a stack of icons according to the data value at a specific point.

Syntax

```
chem=y;s=cm_repeat;ds=<series_rendering_index>; ...other_values... (https://developers.google.com/chart/image/docs/chart_param
  d=<icon_shape>,<repeat_series_index>,<scaling_factor>,<stacking_direction>,<icon_size>,<fill_color>,<outline_color
```

<icon_shape>

The icon to use. Specify an ID string identifying one of the images listed at the end of the page (#icon_list).

<repeat_series_index>

The zero-based index of the data series used to calculate how many icons to place at this point.

<scaling_factor>

The source data series value is scaled to a value from 0 to 1 and multiplied by this value to determine how many markers to place on this point. Partial values are truncated.

<stacking_direction>

Stacking direction: either "h" (lowercase) for horizontal or "V" (uppercase) for vertical.

<icon_size>

The size of each marker, in pixels. The following values are supported: 12, 16, 24.

<fill_color>

The fill color for the icon, as a three- or six-digit HTML hexadecimal color (no # mark).

<outline_color>

The outline color for the icon, as a three- or six-digit HTML hexadecimal color (no # mark).

<spacing>

How much space to put between each marker in a stack, in pixels.

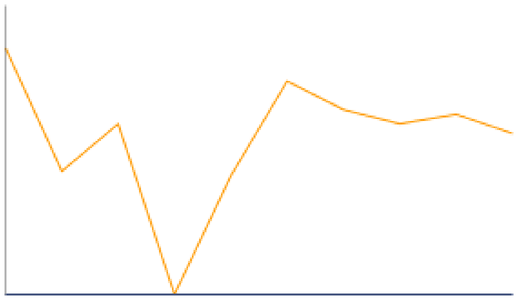
<alignment>

An optional string (#contextualaligmentstrings) describing the icon alignment and offset.

Example

This example uses a second dummy data series. It is not rendered on the chart, but used as a way to space all the stacks evenly, starting at the bottom of the chart.

- `chd=s1:0akAZtnkmi,AAAAAAAAAA` - First series is used to plot the line and determine the stack height. The second is used specify the base of each icon stack, at the 0 value.
- `chem=y` - Render a dynamic icon
- `s=cm_repeat` - Use the stacking variation context icon.
- `ds=1` - Render the items on data series 1.
- `dp=all` - Put the icon on all points.
- `d=`
    - `d=petrol` - Use the petrol symbol.
    - `0` - Use data series 0 for the repeat count.
    - `9` - Use a scaling factor of 6.
    - `V` - Stack vertically.
    - `16` - Make each marker 16 pixels high.
    - `F00` - The icon fill color.
    - `000` - The icon outline color.
    - `2` - Two pixels between each icon in the stack.
    - `hb` - Center the stacks on the bottom



```
chd=s1:0akAZtnkmi,AAAAAAAAAA
chem=
  y;s=cm_repeat;ds=1;dp=all;
  d=petrol,0,9,V,16,F00,000,2,hb
```

Back to top (#top)

Stacking and Color Variation (`cm_repeat_color`)

You can vary the both the height and color of an icon stack, according to the data value at a specific point.

Syntax

```
chem=y;s=cm_repeat_color;ds=<series_rendering_index>; ...other_values... (https://developers.google.com/chart/image/docs/char
  d=<icon_shape>,<repeat_series_index>,<scaling_factor>,<stacking_direction>,<icon_size>,<color_data_series>,<low_co.
```

<icon_shape>

The icon to use. Specify an ID string identifying one of the images listed at the end of the page (#icon_list).

<repeat_series_index>

The zero-based index of the data series used to calculate how many icons to place at this point.

<scaling_factor>

The source data series value is scaled to a value from 0 to 1 and multiplied by this value to determine how many markers to place on this point. Partial values are truncated.

<stacking_direction>

Stacking direction: either "h" (lowercase) for horizontal or "V" (uppercase) for vertical.

<icon_size>

The size of each marker, in pixels. The following values are supported: 12, 16, 24.

<color_data_series>

The zero-based index of the data series used to vary the color of the icons.

<low_color>

The low color value in the range, as a three- or six-digit HTML hexadecimal color (no # mark). This is associated with the lowest possible value in the available data range.

<middle_color>

The middle color value in the range, as a three- or six-digit HTML hexadecimal color (no # mark). This is associated with the middle value in the available data range.

<high_color>

The high color value in the range, as a three- or six-digit HTML hexadecimal color (no # mark). This is associated with the highest possible value in the available data range.

<outline_color>

The outline color for the icon, as a three- or six-digit HTML hexadecimal color (no # mark).

<spacing>

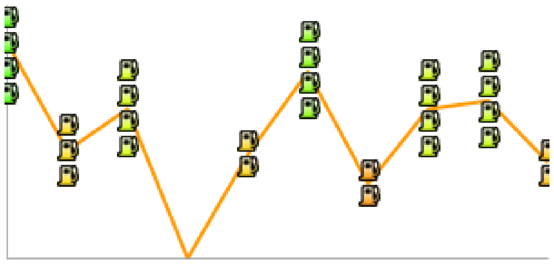
How much space to put between each marker in a stack, in pixels.

<alignment>

An optional string (#contextualaligmentstrings) describing the icon alignment and offset.

Example

- `chem=y` - Render a dynamic icon
- `s=cm_repeat_color` - Use the stacking and color variation context icon.
- `ds=0` - Render the items on data series 0.
- `dp=all` - Put the icon on all points.
- `d=`
  - `petrol` - Use the petrol symbol.
  - `0` - Use data series 0 for the repeat count.
  - `6` - Use a scaling factor of 6.
  - `V` - Stack vertically.
  - `12` - Make each marker 12 pixels high.
  - `0` - Use series 0 to specify the color.
  - `F00,0F0,00F` - The low, middle, and high color values.
  - `000` - The icon outline color.



```
chem=
  y;s=cm_repeat_color;ds=0;dp=all;
  d=petrol,0,6,V,12,0,F00,0F0,00F,000,2,hv
```

- **2** - Two pixels between each icon in the stack.
- **hv** - Center the stacks vertically and horizontally on each data point.

Back to top (#top)

Image List

The following images can be used in dynamic icons using the appropriate parameter.

Note: Text strings are case-sensitive

Note: Icons are available only in the following sizes: 12, 16, 24.

Basic icons

academy activities airport amusement aquarium art-gallery atm baby bank-dollar bank-euro bank-intl
bank-pound bank-yen bar barber beach beer bicycle books bowling bus cafe camping car-dealer
car-rental car-repair casino caution cemetery-grave cemetery-tomb cinema civic-building computer corporate
courthouse fire flag floral helicopter home info landslide legal location locomotive medical mobile
motorcycle music parking pet petrol phone picnic postal repair restaurant sail school scissors ship
shoppingbag shoppingcart ski snack snow sport star swim taxi train truck wc-female wc-male wc
wheelchair

Icons from Glyphish.com

These icons are by Joseph Wain / glyphish.com (http://www.glyphish.com). This work is licensed under the Creative Commons Attribution 3.0 United States License (http://creativecommons.org/licenses/by/3.0/us/).

glyphish_airplane glyphish_baby glyphish_badge glyphish_balloon glyphish_bandaid glyphish_bar-chart glyphish_battery
glyphish_beaker glyphish_beaker2 glyphish_beermug glyphish_bird glyphish_bone glyphish_book glyphish_bookmark
glyphish_bow-and-arrow glyphish_brightness glyphish_bug glyphish_bullhorn glyphish_cabinet glyphish_calendar glyphish_camera
glyphish_chat glyphish_chat2 glyphish_chicken glyphish_clock glyphish_cloud glyphish_coathanger glyphish_coffee
glyphish_compass glyphish_contrast glyphish_controller glyphish_dashboard glyphish_dice glyphish_display glyphish_dogpaw
glyphish_download glyphish_dumbbell glyphish_ekg glyphish_envelope glyphish_equalizer glyphish_eye glyphish_film-roll
glyphish_flag glyphish_food glyphish_fork-and-knife glyphish_fuel glyphish_gameplan glyphish_gear glyphish_gear2 glyphish_gift
glyphish_golf glyphish_group glyphish_headphones glyphish_heart glyphish_house glyphish_id-card glyphish_inbox
glyphish_index-cards glyphish_iphone glyphish_ipod glyphish_key glyphish_lanscape glyphish_lifebuoy glyphish_lightbulb
glyphish_line-chart glyphish_location glyphish_lock glyphish_loopback glyphish_magnifying-glass glyphish_map glyphish_map-marker
glyphish_medical glyphish_medical-bag glyphish_microphone glyphish_moon glyphish_movie1 glyphish_movie2 glyphish_navigation
glyphish_network glyphish_note glyphish_outlet glyphish_palette glyphish_paperclip glyphish_phone glyphish_photos glyphish_piano
glyphish_picture-frame glyphish_piggybank glyphish_pill glyphish_pin glyphish_pinetree glyphish_planet glyphish_puzzle
glyphish_radar glyphish_redo glyphish_refresh glyphish_runner glyphish_shoebox glyphish_shopping-bag glyphish_shopping-cart
glyphish_shuffle glyphish_signpost glyphish_skull glyphish_skull-n-crossbones glyphish_sliders glyphish_sock glyphish_spraycan
glyphish_squiggle glyphish_star glyphish_stats glyphish_stopwatch glyphish_suitcase glyphish_tag glyphish_tags glyphish_target
glyphish_testtube glyphish_thermometer glyphish_todo glyphish_toolbox glyphish_trophy glyphish_tshirt glyphish_tv glyphish_umbrella
glyphish_user glyphish_walk glyphish_weather glyphish_widescreen glyphish_wineglass glyphish_zap

Flags

Here are flags for some countries and territories.

flag_ad flag_ae flag_af flag_ag flag_ai flag_am flag_ao flag_ar flag_as flag_at flag_au
flag_az flag_ba flag_bd flag_be flag_bf flag_bg flag_bh flag_bi flag_bj flag_bn flag_bo
flag_br flag_bs flag_bw flag_by flag_bz flag_ca flag_cat flag_cc flag_cd flag_cf flag_cg
flag_ch flag_ci flag_ck flag_cl flag_cm flag_cn flag_co flag_cr flag_cu flag_cz flag_de
flag_dj flag_dk flag_dm flag_do flag_dz flag_ec flag_ee flag_eg flag_es flag_et flag_fi
flag_fj flag_fm flag_fr flag_ga flag_ge flag_gg flag_gh flag_gi flag_gl flag_gm flag_gp
flag_gr flag_gt flag_gy flag_hk flag_hn flag_hr flag_ht flag_hu flag_id flag_ie flag_il
flag_im flag_in flag_is flag_it flag_ja flag_je flag_jm flag_jo flag_jp flag_ke flag_kg
flag_kh flag_ki flag_kr flag_kw flag_kz flag_la flag_lb flag_li flag_lk flag_ls flag_lt
flag_lu flag_lv flag_ly flag_ma flag_md flag_me flag_mg flag_mk flag_ml flag_mn
flag_mo flag_ms flag_mt flag_mu flag_mv flag_mw flag_mx flag_my flag_mz flag_na

flag_ne flag_nf flag_ng flag_ni flag_nl flag_no flag_np flag_nr flag_nu flag_nz flag_om
flag_pa flag_pe flag_pg flag_ph flag_pk flag_pl flag_pn flag_pr flag_ps flag_pt flag_py
flag_qa flag_ro flag_rs flag_ru flag_rw flag_sa flag_sb flag_sc flag_se flag_sg flag_sh
flag_si flag_sk flag_sl flag_sm flag_sn flag_sr flag_st flag_sv flag_td flag_tg flag_th
flag_tj flag_tk flag_tl flag_tm flag_tn flag_to flag_tp flag_tr flag_tt flag_tv flag_tw
flag_tz flag_ua flag_ug flag_uk flag_us flag_uy flag_uz flag_vc flag_ve flag_vg flag_vi
flag_vn flag_vu flag_ws flag_yu flag_za flag_zm flag_zw

Back to top (#top)

Última actualización: Marzo 26, 2012.