

# Customizing Axes

## Overview

---

Dimensions in the data are often displayed on axes, horizontal and vertical. Such is the case for: [Area Chart](https://google-developers.appspot.com/chart/interactive/docs/gallery/areachart) (<https://google-developers.appspot.com/chart/interactive/docs/gallery/areachart>), [Bar Chart](https://google-developers.appspot.com/chart/interactive/docs/gallery/barchart) (<https://google-developers.appspot.com/chart/interactive/docs/gallery/barchart>), [Candlestick Chart](https://google-developers.appspot.com/chart/interactive/docs/gallery/candlestickchart) (<https://google-developers.appspot.com/chart/interactive/docs/gallery/candlestickchart>), [Column Chart](https://google-developers.appspot.com/chart/interactive/docs/gallery/columnchart) (<https://google-developers.appspot.com/chart/interactive/docs/gallery/columnchart>), [Combo Chart](https://google-developers.appspot.com/chart/interactive/docs/gallery/combochart) (<https://google-developers.appspot.com/chart/interactive/docs/gallery/combochart>), [Line Chart](https://google-developers.appspot.com/chart/interactive/docs/gallery/linechart) (<https://google-developers.appspot.com/chart/interactive/docs/gallery/linechart>), [Stepped Area Chart](https://google-developers.appspot.com/chart/interactive/docs/gallery/steppedareachart) (<https://google-developers.appspot.com/chart/interactive/docs/gallery/steppedareachart>) and [Scatter Chart](https://google-developers.appspot.com/chart/interactive/docs/gallery/scatterchart) (<https://google-developers.appspot.com/chart/interactive/docs/gallery/scatterchart>).

When you create a chart with axes you can customize some of their properties:

- [Discrete vs. Continuous](#) (`#Discrete_vs_Continuous`)
- Direction - You can customize the direction using the `hAxis/vAxis.direction` option.
- Label positioning and style - You can customize label positioning and style using the `hAxis/vAxis.textPosition` and `hAxis/vAxis.textStyle` options.
- Axis title text and style - You can customize the axis title text and style using the `hAxis/vAxis.title` and `hAxis/vAxis.titleTextStyle` options.
- [Axis scale](#) (`#axis_scale`) - You can set the scale of an axis to logarithmic (log) scale using the `hAxis/vAxis.logScale` or `hAxis/vAxis.scaleType` options.
- For a complete list of axis configuration options, look at the `hAxis` and `vAxis` options in the documentation for your specific chart.

## Terminology

---

Major/minor axis:

- The *major* axis is the axis along the natural orientation of the chart. For line, area, column, combo, stepped area and candlestick charts, this is the horizontal axis. For a bar chart it is the vertical one. Scatter and pie charts don't have a major axis.

- The *minor* axis is the other axis.

Discrete/continuous axis:

- A *discrete* axis has a finite number of evenly spaced values, called categories.
- A *continuous* axis has an infinite number of possible values.

## Discrete vs Continuous

---

The major axis (#Terminology) of a chart can be either discrete (#Terminology) or continuous (#Terminology). When using a discrete axis, the data points of each series are evenly spaced across the axis, according to their row index. When using a continuous axis, the data points are positioned according to their domain value.

The labeling is also different. In a discrete axis, the names of the categories (specified in the domain column of the data) are used as labels. In a continuous axis, the labels are auto-generated: the chart shows evenly spaced grid lines, where each grid line is labeled according to the value it represents.

The following axes are always continuous:

- Both axes of bubble charts.
- The minor axis (#Terminology).

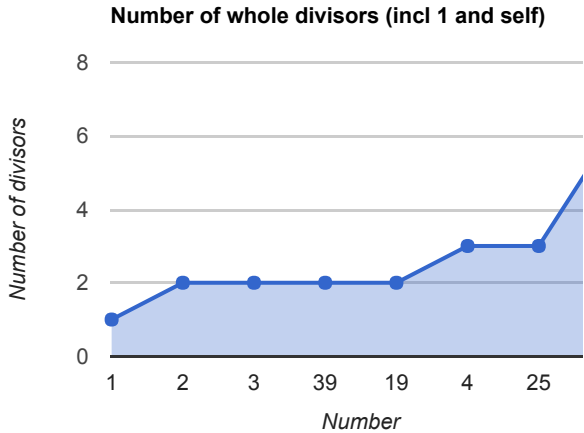
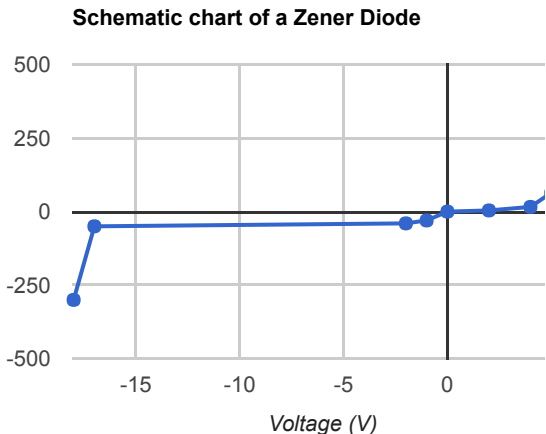
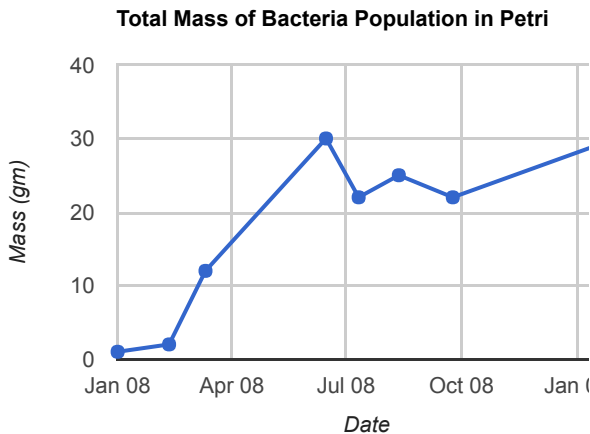
The following axes are always discrete:

- The major axis of stepped area charts (and combo charts containing such series).

In line, area, bar, column and candlestick charts (and combo charts containing only such series), you can control the type of the major axis:

- For a discrete axis, set the data column type to `string`.
- For a continuous axis, set the data column type to one of: `number`, `date`, `datetime` or `timeofday`.

Discrete / Continuous	First column type	Example
Discrete	string	

		<div><div></div><div><p><b>Number of whole divisors (incl 1 and self)</b></p><table><tr><th>Number</th><th>Number of divisors</th></tr><tr><td>1</td><td>1</td></tr><tr><td>2</td><td>2</td></tr><tr><td>3</td><td>2</td></tr><tr><td>39</td><td>2</td></tr><tr><td>19</td><td>2</td></tr><tr><td>4</td><td>3</td></tr><tr><td>25</td><td>3</td></tr><tr><td>26</td><td>4</td></tr></table></div></div>	Number	Number of divisors	1	1	2	2	3	2	39	2	19	2	4	3	25	3	26	4										
Number	Number of divisors																													
1	1																													
2	2																													
3	2																													
39	2																													
19	2																													
4	3																													
25	3																													
26	4																													
Continuous	number	<div><div></div><div><p><b>Schematic chart of a Zener Diode</b></p><table><tr><th>Voltage (V)</th><th>Current</th></tr><tr><td>-18</td><td>-300</td></tr><tr><td>-15</td><td>-100</td></tr><tr><td>-10</td><td>-50</td></tr><tr><td>-5</td><td>-20</td></tr><tr><td>-2</td><td>-10</td></tr><tr><td>-1</td><td>-5</td></tr><tr><td>0</td><td>0</td></tr><tr><td>1</td><td>5</td></tr><tr><td>2</td><td>10</td></tr><tr><td>5</td><td>20</td></tr><tr><td>10</td><td>50</td></tr><tr><td>15</td><td>100</td></tr><tr><td>18</td><td>300</td></tr></table></div></div>	Voltage (V)	Current	-18	-300	-15	-100	-10	-50	-5	-20	-2	-10	-1	-5	0	0	1	5	2	10	5	20	10	50	15	100	18	300
Voltage (V)	Current																													
-18	-300																													
-15	-100																													
-10	-50																													
-5	-20																													
-2	-10																													
-1	-5																													
0	0																													
1	5																													
2	10																													
5	20																													
10	50																													
15	100																													
18	300																													
Continuous	date	<div><div></div><div><p><b>Total Mass of Bacteria Population in Petri</b></p><table><tr><th>Date</th><th>Mass (gm)</th></tr><tr><td>Jan 08</td><td>1</td></tr><tr><td>Feb 08</td><td>2</td></tr><tr><td>Mar 08</td><td>12</td></tr><tr><td>Jul 08</td><td>30</td></tr><tr><td>Aug 08</td><td>22</td></tr><tr><td>Sep 08</td><td>25</td></tr><tr><td>Oct 08</td><td>22</td></tr><tr><td>Jan 09</td><td>28</td></tr></table></div></div>	Date	Mass (gm)	Jan 08	1	Feb 08	2	Mar 08	12	Jul 08	30	Aug 08	22	Sep 08	25	Oct 08	22	Jan 09	28										
Date	Mass (gm)																													
Jan 08	1																													
Feb 08	2																													
Mar 08	12																													
Jul 08	30																													
Aug 08	22																													
Sep 08	25																													
Oct 08	22																													
Jan 09	28																													

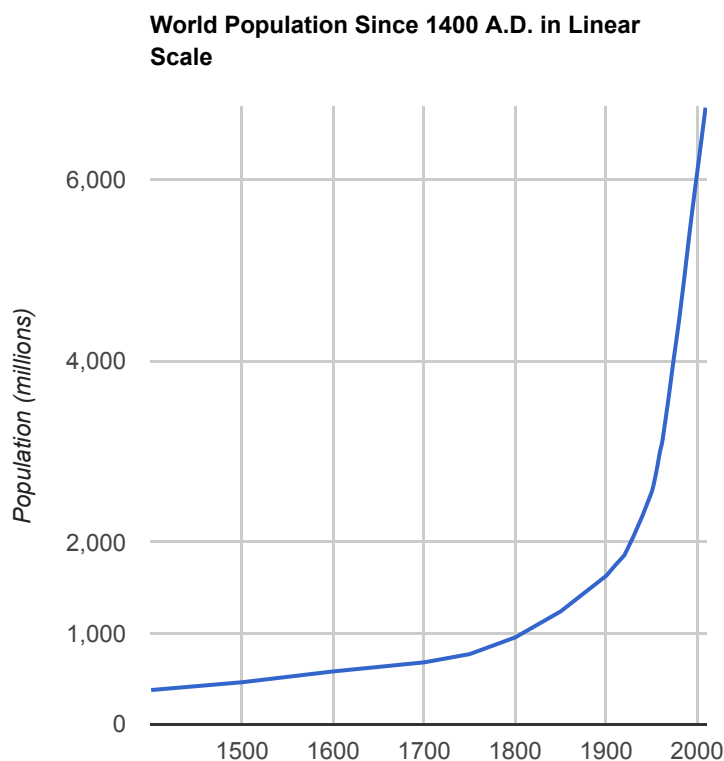
# Axis Scale

You can set the scale of an axis using the `scaleType` option. For example, to set the scale of the vertical axis to log scale, use the following option:

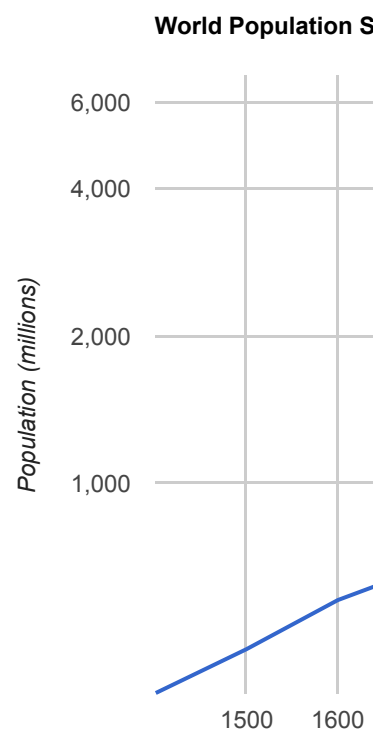
```
vAxis: {  
    scaleType: 'log'  
}
```

The following line chart shows the growth of the world population in both linear (standard) scale and log scale.

## Linear Scale



## Log Scale

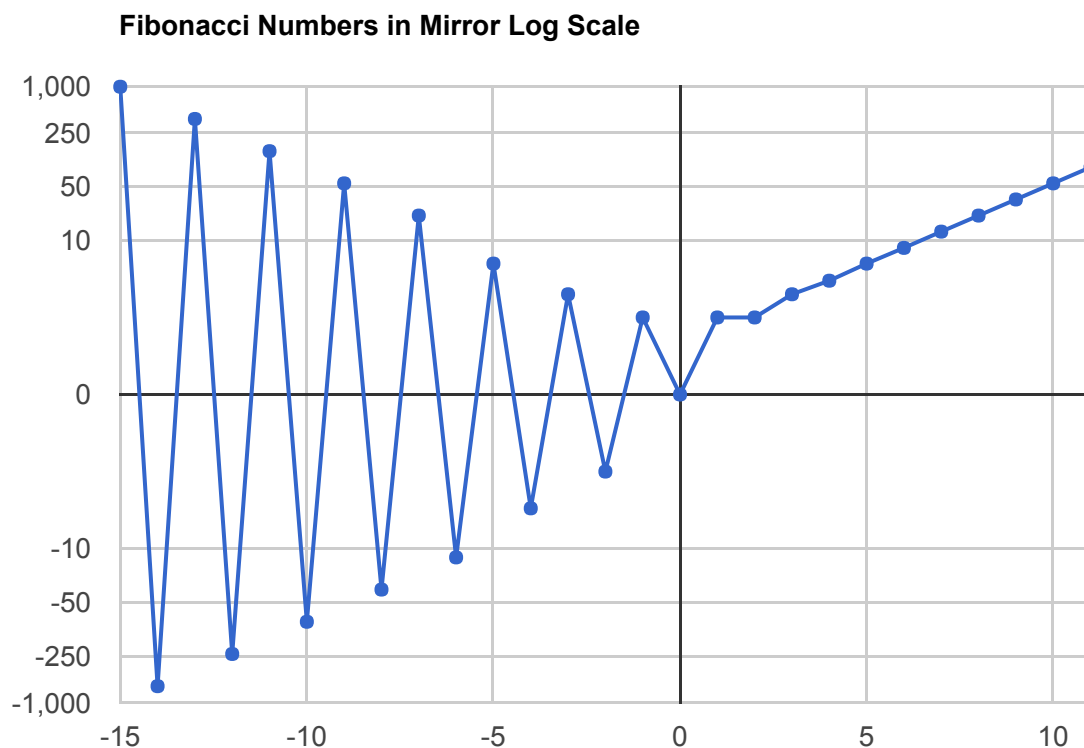


[CODE IT YOURSELF ON JSFIDDLE](#)

If your data contains zero or negative values, you can plot the points using the `scaleType`: `'mirrorLog'` option. In mirror log scale, the plotted value of a negative number is minus the log of the absolute value of the number. Values close to 0 are plotted on a linear scale.

The following example shows positive and negative Fibonacci numbers plotted in both mirror log scale and linear scale.

[CHANGE TO LINEAR SCALE](#)



[CODE IT YOURSELF ON JSFIDDLE](#)

## Number Formats

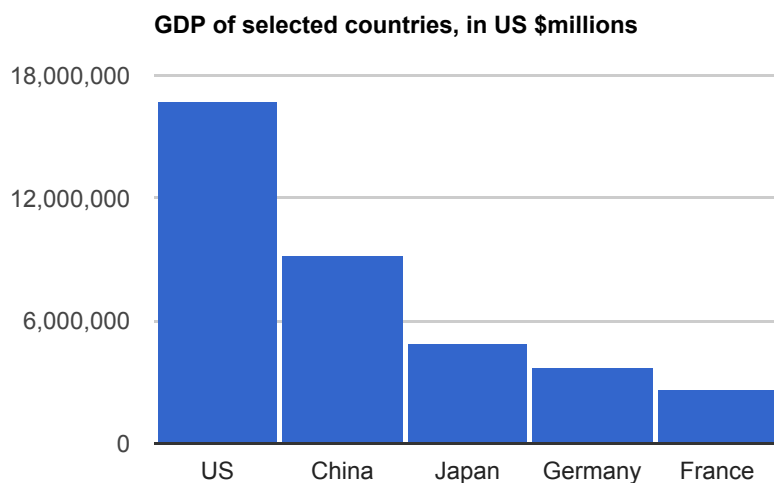
You can control the formatting of label numbers with `hAxis.format` and `vAxis.format`. For instance, `{hAxis: { format: '#,###%' } }` displays the values "1,000%", "750%", and "50%" for values 10, 7.5, and 0.5. You can also supply any of the following presets:

- `{format: 'none'}`: displays numbers with no formatting (e.g., 8000000)
- `{format: 'decimal'}`: displays numbers with thousands separators (e.g., 8,000,000)

- `{format: 'scientific'}`: displays numbers in scientific notation (e.g., 8e6)
- `{format: 'currency'}`: displays numbers in the local currency (e.g., \$8,000,000.00)
- `{format: 'percent'}`: displays numbers as percentages (e.g., 800,000,000%)
- `{format: 'short'}`: displays abbreviated numbers (e.g., 8M)
- `{format: 'long'}`: displays numbers as full words (e.g., 8 million)

In the chart below, you can select among the presets:

decimal ▼



A complete web page for the above chart follows.

[CODE IT YOURSELF ON JSFIDDLE](#)

```
<html>
<head>
  <script type="text/javascript" src="https://www.gstatic.com/charts/loader">
  <script type="text/javascript">
    google.charts.load('current', {packages:['corechart']});
    google.charts.setOnLoadCallback(drawStuff);

    function drawStuff() {
      var data = new google.visualization.DataTable();
      data.addColumn('string', 'Country');
      data.addColumn('number', 'GDP');
      data.addRows([
        ['US', 16768100],
        ['China', 9181204],
        ['Japan', 4898532],
        ['Germany', 3730261],
```

```

        ['France', 2678455]
    ]);

    var options = {
        title: 'GDP of selected countries, in US $millions',
        width: 500,
        height: 300,
        legend: 'none',
        bar: {groupWidth: '95%'},
        vAxis: { gridlines: { count: 4 } }
    };

    var chart = new google.visualization.ColumnChart(document.getElementById('number_format_chart'));
    chart.draw(data, options);

    document.getElementById('format-select').onchange = function() {
        options['vAxis']['format'] = this.value;
        chart.draw(data, options);
    };
};
</script>
</head>
<body>
    <select id="format-select">
        <option value="">none</option>
        <option value="decimal" selected>decimal</option>
        <option value="scientific">scientific</option>
        <option value="percent">percent</option>
        <option value="currency">currency</option>
        <option value="short">short</option>
        <option value="long">long</option>
    </select>
    <div id="number_format_chart">
</body>
</html>

```

When using a format that employs text (e.g., the `long` format, which will render 8,000,000 as "8 million", you can localize the strings into other languages by specifying a language code when you load the Google Charts library. For instance, to change "8 million" into the Russian "8 миллиона", call the library like so:

```

<script type='text/javascript' src='https://www.gstatic.com/charts/loader.js'>
<script type='text/javascript'>
    google.charts.load('current', {'packages': ['corechart'], 'language': 'ru'})
    google.charts.setOnLoadCallback(drawMarkersMap);
</script>

```

---

*Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 3.0 License](http://creativecommons.org/licenses/by/3.0/) (<http://creativecommons.org/licenses/by/3.0/>), and code samples are licensed under the [Apache 2.0 License](http://www.apache.org/licenses/LICENSE-2.0) (<http://www.apache.org/licenses/LICENSE-2.0>). For details, see our [Site Policies](https://developers.google.com/terms/site-policies) (<https://developers.google.com/terms/site-policies>). Java is a registered trademark of Oracle and/or its affiliates.*

*上次更新日期: 二月 23, 2017*