



[Nueva guía] Descargue la Guía de 2017 para el desarrollo we...

[Descargar la guía▶](#)

# Las 9 mejores bibliotecas de gráficos de JavaScript

por Anton Shaleynikov · 29 y 17 de noviembre · Web Dev Zone

¿Debería construir su propia solución de experimentación web? Descargue este documento técnico de Optimizely para descubrirlo.

---

Ya que está leyendo esto, seguramente no sería una novedad para usted que la visualización de datos se haya convertido en una parte muy crítica del mundo de TI actual. La gran cantidad de datos generados por diferentes tecnologías web debe ser refinado y visualizado adecuadamente para que el mundo lo utilice y obtenga información valiosa a partir de él.

Las ventajas de comprender los datos son enormes y conduce a tomar decisiones bien informadas. La capacidad de una organización para obtener sus datos extraídos, limpiados, visualizados y analizados podría ser la diferencia entre ganancias o pérdidas.

Como desarrollador web front-end, es un gran conjunto de habilidades para poder visualizar datos tanto como puede hacer hermosas páginas web interactivas. Estas bibliotecas de JavaScript han facilitado la tarea, ya que los desarrolladores no tienen que soportar un

cambio en la sintaxis del lenguaje para convertir esos números mágicos en gráficos comprensibles.

Hay muchas bibliotecas de JavaScript disponibles para este fin. Y esa es la razón por la cual estos nueve han sido elegidos. Se destacan del resto, y servirían mucho para sus propósitos.

Antes de continuar, aquí hay una lista de las bibliotecas que seleccioné:

- Echarts
- Tablas de Tau
- ChartJS
- Chartista
- C3
- Highcharts
- ReCharts
- NVD3
- Flot

Es importante tener en cuenta que este artículo no es un tutorial. Sin embargo, se han agregado enlaces a cada biblioteca para facilitar el trabajo de aprendizaje.

## Echarts





¿Qué más se puede obtener de un producto creado por los peces gordos chinos Baidu, Inc., si no uno con gran calidad?

Echarts es muy útil para visualizar datos en páginas web. Con él, puede crear gráficos intuitivos, personalizables e interactivos, facilitando la interpretación y el análisis de los datos.

Muchas de las bibliotecas de creación de gráficos de JavaScript se han migrado a JavaScript, lo que significa que su uso no fluye a la perfección. Sin embargo, ese no es el caso con Echarts, ya que estaba escrito en JavaScript.

La documentación está disponible para esta biblioteca en su página web oficial , y no tiene que preocuparse por su estado de ser una biblioteca china. Está muy bien documentado en inglés.

La instalación de esta biblioteca se realiza fácilmente con npm:

```
npm install echarts --save
```

Para obtener más información acerca de esta biblioteca, puede verla desde el sitio web oficial y también visitar la página de GitHub .

**Ejemplo** - Diagrama de dispersión de gran saco:

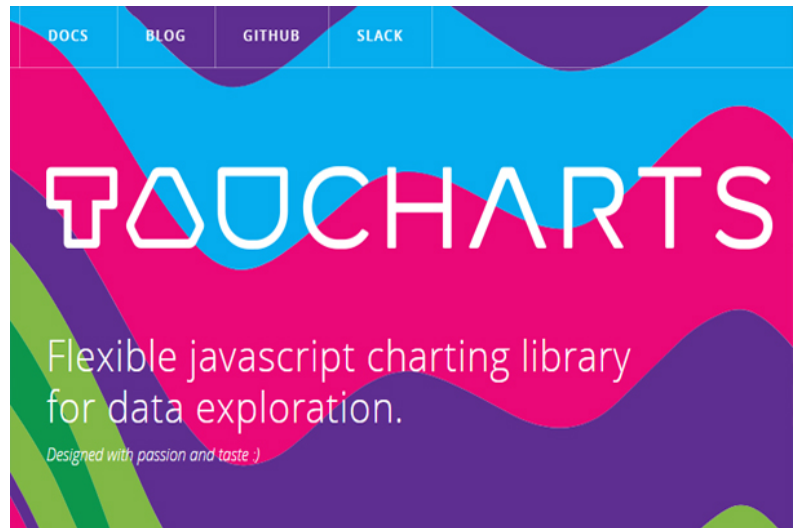
JavaScript   HTML   CSS

[Edit in JSFiddle](#)

```
var dom = document.getElementById("container");
var myChart = echarts.init(dom);
var app = {};
option = null;
option = {
```

```
title: {
  text: 'Large-scale scatterplot'
},
tooltip : {
  trigger: 'axis',
  showDelay : 0,
  autoClose : false
}
```

## TauCharts



TauCharts es una de las bibliotecas de gráficos JavaScript más flexibles que existen. También está basado en D3 y es una biblioteca de gráficos JavaScript centrada en los datos que permite una mejor visualización de los datos.

Hablando de su flexibilidad, TauCharts permite un fácil acceso a su API. Brindando a los usuarios la oportunidad de mapear y visualizar datos de forma transparente para obtener conocimientos más sorprendentes.

No es solo flexible. También renderiza los gráficos bastante rápido y es fácil de aprender.

Para instalar a través de npm, simplemente escriba:

```
npm install taucharts
```

Para obtener más información al respecto, puede consultar el sitio web oficial y también aprender de algunos tutoriales en la página de npm .

## Ejemplo : línea horizontal

JavaScript HTML CSS Edit in JSFiddle

Resources

```

    "count": 3,
    "priority": "high"
  }];
  var chart = new tauCharts.Chart({
    data: defData,
    type: 'horizontalBar',
    x: 'effort',
    y: 'team',
    color: 'priority'
  });
  chart.renderTo('#bar');
```

## Chart.js



# Chart.js

Simple yet flexible JavaScript charting for designers & developers

[Get Started](#)
[Documentation](#)

Sencillo. Limpiar. Fácil de usar.

Palabras perfectas para describir la biblioteca Chart.js. La biblioteca Chart.js es una biblioteca de JavaScript basada en HTML5 para crear tablas y gráficos animados, interactivos y personalizables.

Con Chart.js, puede visualizar fácilmente sus tipos de gráficos mixtos sin problemas, creando páginas web receptivas de manera predeterminada.

Esta biblioteca le permite comenzar a visualizar sus datos en tiempo rápido. Fácil de configurar y amigable para principiantes. Con él, no tendrá

que involucra en problemas de

que involucrarse en problemas de compatibilidad del navegador, ya que es compatible con navegadores antiguos.

Para instalar Chart.js usando npm:

```
npm install chart.js --save
```

La documentación adecuada y algunos tutoriales en esta biblioteca existen en su sitio web oficial y en la página de GitHub .

## Ejemplo : Radar

JavaScript HTML CSS Edit in JSFiddle

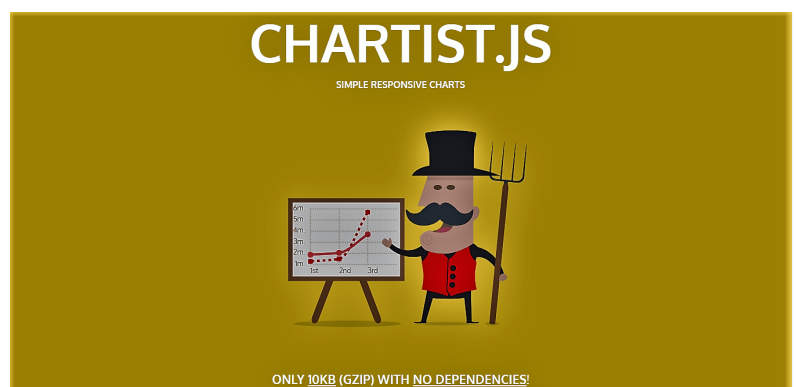
```

    label: 'one',
    borderColor: '#d8463c'
  },
  ],
}

const myRadarChart = new Chart(ctx, {
  data: data,
  type: 'radar',
  options: options
});

```

## Chartista



Si está buscando crear cuadros hermosos, receptivos y de fácil lectura, Chartist es la biblioteca que debe utilizar.

Chartist usa SVG para representar gráficos. También le da la funcionalidad de personalizar sus gráficos usando consultas de medios de CSS y animaciones creativas. Puede traer toda su

y animaciones creativas. Puede hacer toda su creatividad a sus cuadros.

Chartist es especialmente fácil de configurar, además de fácil de personalizar con Sass. Sin embargo, no es compatible con navegadores antiguos como su "hermana" Chart.js.

Puede imaginarse cuánta perfección se le puede ocurrir al embellecer su SVG diseñando con CSS y creando todas las encantadoras animaciones que desee.

Esta biblioteca también se puede instalar usando npm:

```
npm install chartist --save
```

Hay mucha información sobre el uso de esta biblioteca en su sitio web oficial .

**Ejemplo :** gráfico circular con etiquetas personalizadas

JavaScript HTML CSS Edit in JSFiddle

```

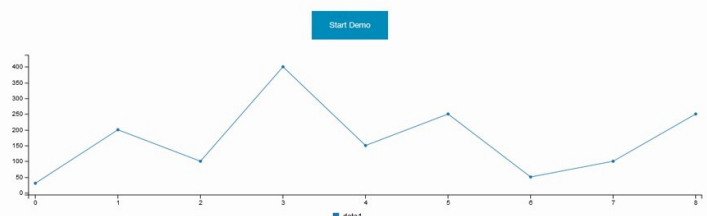
labelInterpolation: 'explode',
labelInterpolationFnc: function(value) {
  Resources Result
  return value;
},
}],
['screen and (min-width: 1024px)', {
  labelOffset: 80,
  chartPadding: 20
}]
];

new Chartist.Pie('.ct-chart', data, options, responsive

```

## C3.js

C3.js D3-based reusable chart library



## Why C3?

Comfortable

Customizable

Controllable

Al igual que TauCharts, C3 es otra biblioteca de visualización de gráficos basada en D3 muy eficiente. Como ventaja, le permite crear clases que pueden personalizarse fácilmente según el gusto o el estilo personal.

C3, en la superficie, parece una biblioteca difícil de usar. Sin embargo, en el momento en que obtienes sus técnicas, no hay nada que te detenga.

Con esta biblioteca de gráficos, puede actualizar gráficos incluso después de la primera representación mediante la creación de devoluciones de llamadas. Es fácil encontrar el camino para usar esta biblioteca, ya que también reduce el trabajo que se debe hacer al permitirle crear gráficos reutilizables para sus aplicaciones web.

Para instalar la biblioteca de gráficos C3 utilizando npm, escriba el siguiente comando:

```
npm install c3
```

Se puede leer más sobre esta biblioteca desde el sitio web oficial y la página de GitHub .

### Ejemplo : cuadro combinado

JavaScript

HTML

CSS

[Edit in JSFiddle](#)

```
var chart = c3.generate({
  data: {
    columns: [
      ['data1', 30, 20, 50, 40, 60, 50],
      ['data2', 200, 130, 90, 240, 130, 220],
      ['data3', 300, 200, 160, 400, 250, 250],
      ['data4', 200, 130, 90, 240, 130, 220],
      ['data5', 130, 120, 150, 140, 160, 150],
      ['data6', 90, 70, 20, 50, 60, 120],
    ],
  },
});
```



type: 'bar',

dataset: 0



# HighCharts



Haz que tus datos cobren vida con HighCharts.js.

Con HighCharts, puede crear gráficos interactivos usando JavaScript puro. Se procesa fácilmente con SVG y funciona perfectamente con HTML5 a medida que intenta agregar gráficos interactivos a sus aplicaciones web.

Es muy ligero, ya que no requiere ningún complemento. Esto no lo hace menos eficiente que las otras bibliotecas de gráficos. Puedes crear cualquier gráfico que puedas imaginar, ya sea la columna, la barra, el pie o el gráfico polar sin poner memoria en la línea.

A pesar de su simplicidad, HighCharts también es muy compatible con navegadores antiguos, por lo que puede elegirlo si no necesita representar datos utilizando estilos de gráficos avanzados.

También se puede instalar usando npm:

```
npm install highcharts --save
```

Para saber más sobre esta biblioteca, consulte el sitio web oficial y la página de GitHub .

**Ejemplo :** burbujas Highcharts con relleno de degradado radial

JavaScript

HTML

CSS

Edit in JSFiddle

```

Highcharts.chart('container', {
  chart: {
    type: 'bubble',
    plotBorderWidth: 1,
    zoomType: 'xy'
  },

  title: {
    text: 'Highcharts bubbles with radial gradient'
  },

```

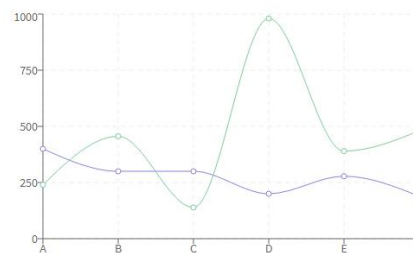
# Recharts

## Recharts

A composable charting library built on React components

Install v0.22.1

Star 6,056



ReCharts es una biblioteca de gráficos creada con React y basada en la biblioteca de gráficos D3.

Se redefine para facilitar las cosas a los amantes de React, ya que puede escribir gráficos en sus aplicaciones web de React sin problemas.

Recharts es muy liviano y representa el uso de elementos SVG para crear gráficos sensacionales.

Se puede instalar usando npm:

```
npm install recharts
```

Recharts no tiene una documentación extensa, ya que es directo al grano y también es fácil

obtener soluciones a problemas cuando se enfrenta a una pared de ladrillos.

El sitio web oficial tiene mucha más información sobre esta biblioteca y la página de GitHub .

## Ejemplo : mapa de árbol de contenido personalizado

JavaScript 1.7

HTML

CSS

[Edit in JSFiddle](#)

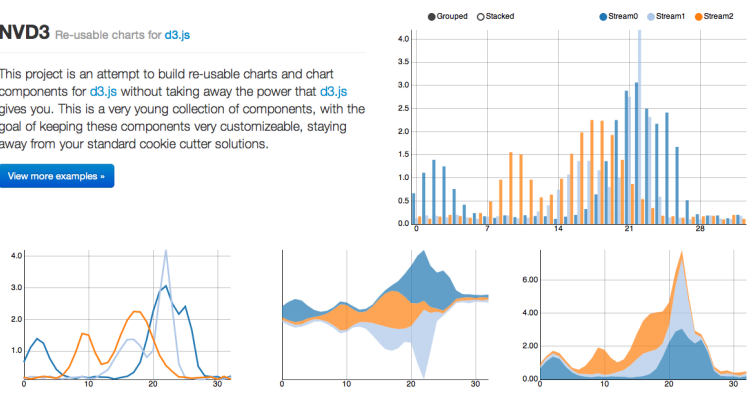
```
const {Treemap} = Recharts;

const data = [
  {
    name: 'axis',
    children: [
      { name: 'Axes', size: 1302 },
      { name: 'Axis', size: 24593 },
      { name: 'AxisGridLine', size: 652 },
      { name: 'AxisLabel', size: 636 },
      { name: 'CartesianAxes', size: 6703 },
    ]
  }
]
```

## NVD3

### NVD3 Re-usable charts for d3.js

This project is an attempt to build re-usable charts and chart components for [d3.js](#) without taking away the power that [d3.js](#) gives you. This is a very young collection of components, with the goal of keeping these components very customizable, staying away from your standard cookie cutter solutions.

[View more examples »](#)


Escrito por Mike Bostock, NVD3 es otra biblioteca de JavaScript basada en D3 de calidad. Permitiéndole crear hermosos gráficos reutilizables en sus aplicaciones web.

Tiene excelentes funciones para visualizar datos con gráficos preciosos como los diagramas de caja, rayos de sol y candeleros. Si está buscando una gran cantidad de funcionalidades en una biblioteca de gráficos de JavaScript, NVD3 es el **que debe tener en cuenta**

que debe tener en cuenta.

Por mucho que la velocidad a veces sea un problema con esta biblioteca, usarla con Fastdom instalado lo hace mucho más rápido.

Puede obtener más información acerca de esta biblioteca desde el sitio web y la página de GitHub .

## Ejemplo : gráfico de líneas simple

JavaScript HTML CSS

Edit in JSFiddle


```

//These lines are all chart setup. Pick and choose what you want
nv.addGraph(function() {
    var chart = nv.models.lineChart()
        .margin({left: 100}) //Adjust chart margins to give the
        .useInteractiveGuideline(true) //We want interactive guide
        .transitionDuration(350) //how fast chart transitions
        .showLegend(true) //Show the legend
        .showYAxis(true) //Show the y-axis
        .showXAxis(true) //Show the x-axis

    //Create the chart
    chart.update();
    chart.xAxis.update(); //Chart x-axis settings

```

## Flot


[Documentation](#)
[Examples](#)
[Plugins](#)
[Blog](#)
[Forum](#)
[Bugs](#)

Attractive JavaScript plotting for jQuery

Flot is a pure JavaScript plotting library for jQuery, with a focus on simple usage, attractive looks and interactive features.

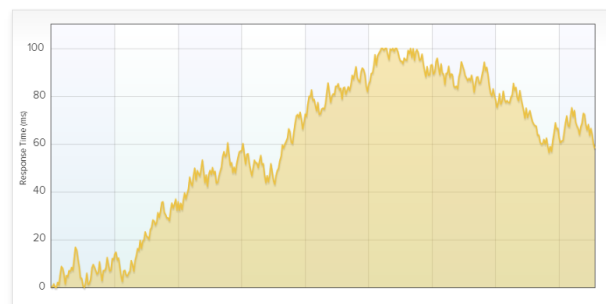
Works with Internet Explorer 6+, Chrome, Firefox 2+, Safari 3+ and Opera 9.5+

ZIP

Version 0.8.3

ZIP

Development



jQuery se ha convertido en una herramienta muy importante para los desarrolladores web a lo largo del tiempo. Con Flot.js, las cosas se han vuelto más fáciles.

Flot es una de las bibliotecas de gráficos más antiguas del mundo de JavaScript de hoy. Sin

embargo, esto de ninguna manera reduce su funcionalidad, ya que admite gráficos de líneas, gráficos circulares, gráficos de barras, gráficos de áreas e incluso gráficos apilados.

Flot ha desarrollado un gran tiempo extra de seguimiento y también tiene una documentación maravillosa. Por lo tanto, es más fácil encontrar soluciones a los problemas cuando se encuentra con una pared de ladrillos.

También da soporte a navegadores más antiguos.

Con Flot, no necesariamente necesita instalarlo usando npm. Todo lo que necesita es incluir el archivo JavaScript en HTML5 después de incluir jQuery.

Para obtener más información sobre Flot.js, simplemente consulte el sitio web oficial y la página de GitHub .

## Ejemplo - Uso básico

JavaScript HTML CSS Edit in JSFiddle

```
function () {  
    var d1 = [];  
    for (var i = 0; i < 14; i += 0.5)  
        d1.push([i, Math.sin(i)]);  
  
    var d2 = [[0, 3], [4, 8], [8, 5], [9, 13]];  
  
    // a null signifies separate line segments  
    var d3 = [[0, 12], [7, 12], null, [7, 2.5], [12, 2.5]];  
  
    $.plot($("#placeholder"), [ d1, d2, d3 ]);  
};
```

## Envolviendolo

Al final, sería difícil elegir la mejor de estas bibliotecas, ya que todas son bibliotecas de calidad. Sin embargo, podría encontrar que algunos son más difíciles de aprender, ya sea


debido a una curva de aprendizaje abrupta o debido a la falta de materiales de aprendizaje en la web. Por lo tanto, sería mejor complementar una biblioteca con otra. También hay muchas otras bibliotecas, aparte de las nueve que se analizan en este artículo, por lo que, en caso de que no encuentre lo que quiere de esta gran lista de nueve, puede investigar más.

Sin embargo, no todas estas bibliotecas son gratuitas, por lo que es importante que consulte sus sitios web oficiales para ver los términos y condiciones relacionados con el uso de estas herramientas. Sin embargo, esto no debe impedir que visualice sus datos de la manera más hermosa posible. Adelante, represente esos números con gráficos.

	ECharts	Tau Charts	Chart.js	Chartist	C3.js	Highcharts	ReCharts	Nvd3	Flot
<b>Description</b>	A powerful charting and visualization library for browser.	Unique syntax that lets the developer describe the data using D3.js, making the ability to reuse data.	Tiny library including core chart types.	Simple, lightweight chart library that uses D3.js to render the chart, and CSS to style it.	D3.js based reusable chart library.	A series of charting libraries for a variety of uses. Can be compatible back to IE6.	React-based chart library built with React and D3.js.	Re-usable charts for D3.js.	A jQuery charting library for jQuery. Contains several chart types but looks like it may no longer be in active development.
<b>Current version</b>	3.8.5	1.2.2	2.0	0.11.0	0.4.18	6.0.3	10.0 - beta.0	1.8.1	0.8.3
<b>GitHub stars</b>	21 775	1 538	31 402	10 148	7 285	6 878	7 140	6 241	5 474
<b>Canvas</b>	☑	☑	☑	☑	☑	☑	☑	☑	☑
<b>SVG</b>	☑	☑	☑	☑	☑	☑	☑	☑	☑
<b>Line</b>	☑	☑	☑	☑	☑	☑	☑	☑	☑
<b>Bar</b>	☑	☑	☑	☑	☑	☑	☑	☑	☑
<b>Radar</b>	☑	☑	☑	☑	☑	☑	☑	☑	☑
<b>Horizontal Bar</b>	☑	☑	☑	☑	☑	☑	☑	☑	☑
<b>Pie</b>	☑	☑	☑	☑	☑	☑	☑	☑	☑
<b>Scatter</b>	☑	☑	☑	☑	☑	☑	☑	☑	☑
<b>Gauges</b>	☑	☑	☑	☑	☑	☑	☑	☑	☑
<b>Maps</b>	☑	☑	☑	☑	☑	☑	☑	☑	☑
<b>Legends</b>	☑	☑	☑	☑	☑	☑	☑	☑	☑
<b>Dependency</b>	jQuery	D3.js, TopoJSON	Chart.js, color, Moment	No dependencies	D3.js	No dependencies	classnames, Core.js, d3-interpolate, d3-scale, d3-shape, lodash, prop-types, react-resize-detector, react-smooth, react-transition-state, reduce-css-calc	No dependencies	No dependencies
<b>Browser support</b>	IE6, IE7, IE8, IE9+, Chrome, Firefox, Safari, Opera	Chrome, Firefox, Opera, Safari, Internet Explorer 10	Chrome 50+, Firefox 40+, Internet Explorer 11, Edge 14+, Safari 9+	IE9, IE10, IE11, Firefox 31, Chrome 35, Safari 7, Safari 8, Android 4.3, Android 4.4, iOS Safari 9, iOS 7	Chrome, Firefox, Opera, Safari, Internet Explorer	Internet Explorer 8+, Chrome 31+, Firefox 24+, Safari 4+, and Opera 9.4+, iOS Safari 3.0+, Android Browser 2+	Chrome, Firefox, Opera, Safari, Internet Explorer 10	Chrome, Firefox, Opera, Safari, Internet Explorer 10	Internet Explorer 6+, Chrome Firefox 2+, Safari 3+ and Opera 9.5+
<b>License</b>	Open-source. Free for all uses.	Open-source. Free for all uses.	Open-source. Free for all uses.	Open-source. Free for all uses.	Open-source. Free for all uses.	Free for non-commercial, paid for commercial use.	Open-source. Free for all uses.	Open-source. Free for all uses.	Open-source. Free for all uses.

Implementar una solución de experimentación: ¿elegir si construir o comprar?

Temas: JAVASCRIPT, LIBRERÍAS JAVASCRIPT, VISUALIZACIÓN DE DATOS, DESARROLLO WEB

Publicado en DZone con el permiso de Anton Shaleynikov. [Vea el artículo original aquí.](#)  Las opiniones expresadas por los contribuidores de DZone son suyas.

# Obtenga lo mejor de Web Dev en su bandeja de entrada.

Manténgase actualizado con el boletín quincenal Web Dev de DZone. VER UN EJEMPLO

SUSCRIBIR

## Web Dev Partner Resources

Learn how to experiment server-side with Optimizely Full Stack. Watch a six-minute demo today.

Optimizely



A Guide to Modern Java Web Development with Crafter CMS  
Crafter Software



How do you choose the right experimentation solution?  
Optimizely



Try Qlik Playground: a free programming environment to test the data-driven app of your dreams

Qlik



