
1.Echarts 可以做什么？

Echarts--商业级数据图表

简单来说 echarts 就是一个商业级数据图表工具。

它是一个纯 JavaScript 的图标库，可以兼容绝大部分的浏览器，可以为前端开发提供直观、生动、可交互、可高度个性化定制的数据可视化图表。

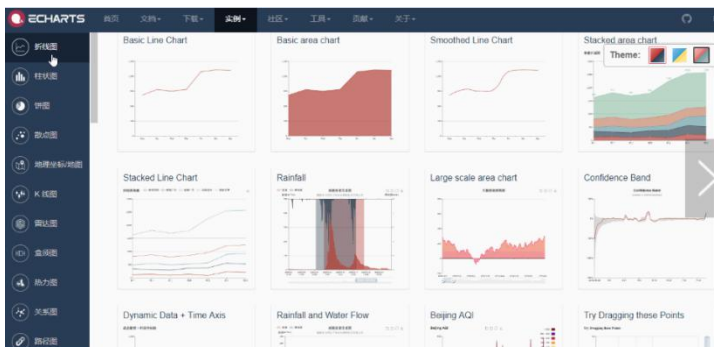
Echarts 支持的图表？

折线图（区域图）、柱状图（条状图）、散点图（气泡图）、K 线图、饼图（环形图）
雷达图（填充雷达图）、和弦图、力导向布局图、地图、仪表盘、漏斗图、事件河流图等
12 类图表

2.echarts 有哪些功能

(1) 多样化图表，丰富可视化表达

Echarts 提供多种多样的图表及图表组合，使用者可以根据自己的数据和展示诉求来进行选择。



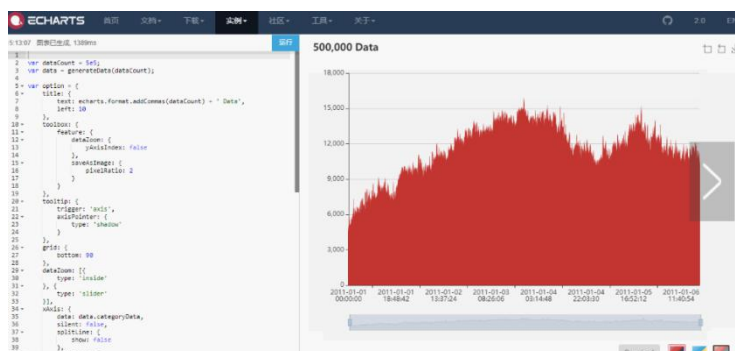
(2) 代码直接引用，效率提升利器

在官网的示例当中，左侧的代码可以直接套用、引用，改变属性，简直太好用了。



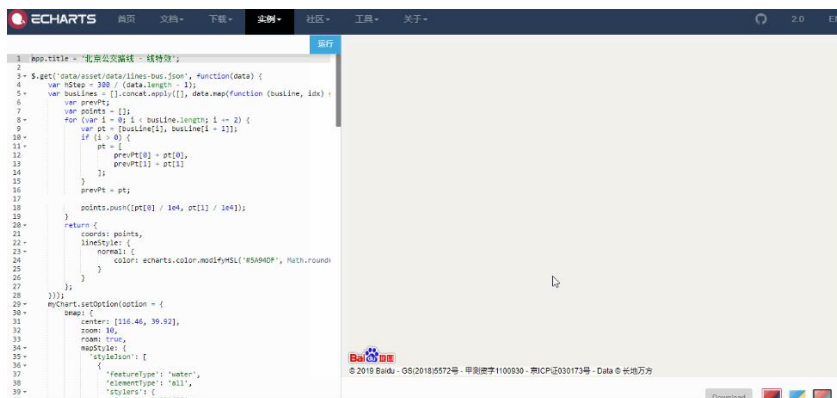
(3) 增量渲染技术，千万级数据量不在话下

通过增量渲染技术，即使展现千万级数据量也能够进行流畅的缩放平移等交互加载。



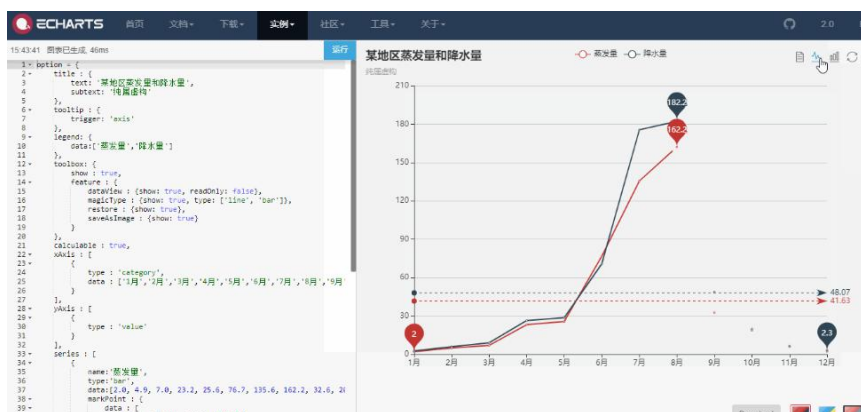
(4) 无缝连接百度地图，实现地图视图

作为百度的开源产品，Echarts 与百度地图无缝连接，结合百度地图的图层，可以做出航线、地铁、公交等动态 3D 线路图，实现超炫的地图视图。



(5) 识别数据差异，让图表动起来

Echarts 会找到两组数据之间的差异，然后通过合适的动画去表现数据的变化，让数据动起来



3.用 echarts 绘制图标步骤

1.1 获取 ECharts

你可以通过以下几种方式获取 ECharts。

1. 从官网下载界面选择你需要的版本下载 <https://echarts.baidu.com/download.html>
2. 通过 npm 获取 echarts, `npm install echarts --save`
3. cdn 引入, 你可以在 cdnjs, npmcdn 或者国内的 bootcdn 上找到 ECharts 的最新版本

1.2 引入 ECharts

```
<script src="echarts.min.js"></script>
```

1.3 准备一个具备大小的 Dom

```
<div id="main" style="width: 600px;height:400px;"></div>
```

1.4 初始化 echarts 实例

```
// 基于准备好的 dom, 初始化 echarts 实例
var myChart = echarts.init(document.getElementById('main'));
```

1.5. 通过 setOption 方法生成一个简单的柱状图

```
myChart.setOption(option);
通过 setOption 方法生成一个简单的柱状图
```

下面是完整代码

```
<!-- 为 ECharts 准备一个具备大小（宽高）的 Dom -->
<div id="main" style="width: 600px;height:400px;"></div>
<script type="text/javascript">
  // 基于准备好的 dom, 初始化 echarts 实例
  var myChart = echarts.init(document.getElementById('main'));

  // 指定图表的配置项和数据
```

```
var option = {
  title: {
    text: 'ECharts 入门示例'
  },
  tooltip: {},
  legend: {
    data:['销量']
  },
  xAxis: {
    data: ["衬衫","羊毛衫","雪纺衫","裤子","高跟鞋","袜子"]
  },
  yAxis: {},
  series: [{
    name: '销量',
    type: 'bar',
    data: [5, 20, 36, 10, 10, 20]
  }]
};

// 使用刚指定的配置项和数据显示图表
myChart.setOption(option);
</script>
```

4.ECharts 中的样式简介

- 颜色主题（Theme）
- 调色盘
- 直接样式设置（itemStyle、lineStyle、areaStyle、label、...）
- 高亮的样式：emphasis
- 视觉映射（visualMap）

调色盘

调色盘，可以在 option 中设置。它给定了一组颜色，图形、系列会自动从其中选择颜色。可以设置全局的调色盘，也可以设置系列自己专属的调色盘。

```
option = {  
  // 全局调色盘。  
  color: ['#c23531', '#2f4554', '#61a0a8', '#d48265', '#91c7ae', '#749f83', '#ca8622', '#bda29a'],  
  series: [{  
    type: 'bar',  
    // 此系列自己的调色盘。  
    color: ['#dd6b66', '#759aa0', '#e69d87', '#8dc1a9', '#ea7e53'],  
    ...  
  }, {  
    type: 'pie',  
    // 此系列自己的调色盘。  
    color: ['#37A2DA', '#32C5E9', '#67E0E3'],  
    ...  
  }]  
}
```

直接的样式设置

itemStyle, lineStyle, areaStyle, label

直接的样式设置是比较常用设置方式。纵观 ECharts 的 option 中，很多地方可以设置 itemStyle、lineStyle、areaStyle、label 等等。这些的地方可以直接设置图形元素的颜色、线宽、点的大小、标签的文字、标签的样式等等。一般来说，ECharts 的各个系列和组件，都遵从这些命名习惯，虽然不同图表和组件中，itemStyle、label 等可能出现在不同的地方

高亮的样式：emphasis

在鼠标悬浮到图形元素上时，一般会出现高亮的样式。默认情况下，高亮的样式是根据普通样式自动生成的。但是高亮的样式也可以自己定义，主要是通过 emphasis 属性来定制。emphasis 中的结构，和普通样式的结构相同，例如：

```
option = {  
  series: {  
    type: 'scatter',  
    // 普通样式。  
    itemStyle: {
```

```
    // 点的颜色。
    color: 'red'
  },
  label: {
    show: true,
    // 标签的文字。
    formatter: 'This is a normal label.'
  },
  // 高亮样式。
  emphasis: {
    itemStyle: {
      // 高亮时点的颜色。
      color: 'blue'
    },
    label: {
      show: true,
      // 高亮时标签的文字。
      formatter: 'This is a emphasis label.'
    }
  }
}
```

4. 相关设置介绍

- title 设置标题
 - legend 设置图例
 - xAxis 设置 x 轴
 - yAxis 设置 y 轴
 - visualMap 设置图上标记的颜色等信息（如迁徙图中航线的颜色）
 - series 为数据项及其相关配置
 - tooltip 配置提示框
 - toolbox 工具框
 - color 可以定义为一个颜色数组，Echarts 图中要使用的颜色会从中取
-

5. 异步数据加载和更新

ECharts 中实现异步数据的更新非常简单，在图表初始化后不管任何时候只要通过 jQuery 等工具异步获取数据后通过 setOption 填入数据和配置项就行。

方法一：

```
var myChart = echarts.init(document.getElementById('main'));
```

在异步获取到数据后，在进行 setoption 渲染

```
$.get('data.json').done(function (data) {  
    myChart.setOption({  
        title: {  
            text: '异步数据加载示例'  
        },  
        tooltip: {},  
        legend: {  
            data:['销量']  
        },  
        xAxis: {  
            data: ["衬衫","羊毛衫","雪纺衫","裤子","高跟鞋","袜子"]  
        },  
        yAxis: {},  
        series: [{  
            name: '销量',  
            type: 'bar',  
            data: [5, 20, 36, 10, 10, 20]  
        }]  
    });  
});
```

方法二：

或者先设置完其它的样式，显示一个空的直角坐标轴，然后获取数据后填入数据。

```
var myChart = echarts.init(document.getElementById('main'));// 显示标题，图例和空的坐标轴
```

1. 先把主要得选项进行渲染

```
myChart.setOption({
```

```
title: {
  text: '异步数据加载示例'
},
tooltip: {},
legend: {
  data:['销量']
},
xAxis: {
  data: []
},
yAxis: {},
series: [{
  name: '销量',
  type: 'bar',
  data: []
}]);
// 2.异步加载数据，再次 setoption 渲染相应的值
$.get('data.json').done(function (data) {
  // 填入数据
  myChart.setOption({
    xAxis: {
      data: data.categories
    },
    series: [{
      // 根据名字对应到相应的系列
      name: '销量',
      data: data.data
    }]
  });
});
```

6.loading 动画

如果数据加载时间较长，一个空的坐标轴放在画布上也会让用户觉得是不是产生 bug 了，因此需要一个 loading 的动画来提示用户数据正在加载

ECharts 默认有提供了一个简单的加载动画。只需要调用 `showLoading` 方法显示。数据加载完成后再调用 `hideLoading` 方法隐藏加载动画。

```
var myChart = echarts.init(document.getElementById('main'));
myChart.showLoading();
```

```
$.get('data.json').done(function (data) {
    myChart.hideLoading();
    myChart.setOption(...);
})
);
```

7. 数据的动态更新

ECharts 由数据驱动，数据的改变驱动图表展现的改变，因此动态数据的实现也变得异常简单。

所有数据的更新都通过 `setOption` 实现，你只需要定时获取数据，`setOption` 填入数据，而不用考虑数据到底产生了那些变化，ECharts 会找到两组数据之间的差异然后通过合适的动画去表现数据的变化。

8. 中国地图和各省地图联动

1. 在配置项找到 `series` 里找到 `type:map`

JavaScript 引入示例

```
<script src="echarts.js"></script>
```

```
<script src="map/js/china.js"></script>
<script>
var chart = echarts.init(document.getElementById('main'));
chart.setOption({
  series: [{
    type: 'map',
    map: 'china'
  }]);
</script>
```

JSON 引入示例

```
$.get('map/json/china.json', function (chinaJson) {
  echarts.registerMap('china', chinaJson);
  var chart = echarts.init(document.getElementById('main'));
  chart.setOption({
    series: [{
      type: 'map',
      map: 'china'
    }]
  });
});
```

2. 基本的中国地图配置如下：

```
var chart = echarts.init(document.getElementById('china-map'));
var seriesData = [{
  name: '北京',
  value: 100
}, {
  name: '天津',
  value: 0
},.....];

var option = {
  series: [{
    type: 'map',
    map: 'china'
  }],
  data:seriesData
}
chart.setOption(option);
```

3. 添加样式

```
title: {
  text: '中国地图',
  left: 'center'
},
tooltip: {
  trigger: 'item',
  formatter: '{b}<br/>{c} (p / km2)'
},

series: [{
  type: 'map',
  map: 'china',

  roam: true, // 是否开启鼠标缩放和平移漫游
  top: "3%", // 组件距离容器的距离
  zoom: 1.1,
  selectedMode: 'single',
  label: {
    show: true, // 显示省份标签
    color: '#fbfdfe' // 省份标签字体颜色
  },
  itemStyle: {
    borderWidth: 0.5, // 区域边框宽度
    borderColor: '#0550c3', // 区域边框颜色
    areaColor: '#4ea397', // 区域颜色
  },
  emphasis: { // 对应的鼠标悬浮效果
    borderWidth: 2,
    borderColor: '#4b0082',
    areaColor: '#ece39e',
  }
}],
```

封装在一个函数 `initEcharts` 里，把 `seriesData` 数据移到一个单独的文件里去。

中国地图已经定制完成，如何实现点击某个省出现省的地图呢？

4. 点击加载省地图

```
myChart.on('click', function (param) {  
    console.log(param.name);  
    //这里是省的名称，得到了省的名称如何去加载相应省的 js，  
    // 省的所有 js 都放在了 province 文件夹里，动态加载，但是这里获取的是中文名  
    // 得 根据中文找到对应 province 文件夹里的相应的省， 所以中文 和 英文得一一对应
```

可以考虑在 seriesData 数据里增加一个 字段 alias 来放置省的英文名

```
var provinceEnName = seriesData.filter(function(option,index){  
    return option.name==param.name;  
    })[0].alias;
```

//获取到英文后， 加载相应省的 js

```
loadRelativeProvince(provinceEnName,function(data){  
    //这里得从接口里获取到相应的数据  
    initEcharts(param.name,seriesHebeiData);  
});  
  
})
```

//加载省的地图

```
function loadRelativeProvince(provinceVal,callback){
```

//这里可以判断下， 是否已经加载了某个对应的省的 js， 如果已经加载过， 就不要加载了
// 思路一： 找到所有的 script 的 src 去循环判断， 目前加载的 js 是否已经加载

```
// 思路二： 加一个 全局变量， 每加载一个省的 js, 先判断是否在这个全局变量里，  
// 在的就停止加载， 不在就继续后面的加载， 并把加载的省追加到这个全局对象里  
if(provinceLoaded.indexOf(provinceVal)!=-1){  
    callback && callback(provinceVal);  
    return;  
}
```

```
var script = document.createElement('script');  
script.id = provinceVal;  
script.type = 'text/javascript';  
script.src = './js/province/' + provinceVal+'.js';
```

```
if(script.readyState) {
```

```
    console.log('ie10 以下，包括 IE10 会走这里');
    script.onreadystatechange = function(){
        if (script.readyState === "loaded" || script.readyState === "complete") {
            script.onreadystatechange = null;
            callback && callback(provinceVal);
        }
    }
} else {
    script.onload = function(){
        console.log('别的浏览器会走这里');
        callback && callback(provinceVal);
    }
}
document.getElementsByTagName("head")[0].appendChild(script);
}
```

2D 级联省完成

9. 如何做 3d 的 map

引入 echarts-gl.min.js

主要是把 type 改成 map3D

map:'china', //山东，如果是中国地图就是 china, 是省地图 就用中文 省的名字

具体代码如下：

```
var myChart = echarts.init(document.getElementById('china-map'));
function initEcharts(){
    var option = {
        title: {
            text: '中国地图',
            left: 'center'
        },
        tooltip: {
            type: 'item',
            formatter: '{b}:{c}'
        },
    },
```

```
series: [{
  data: seriesData,
  type: 'map3D', //默认是 2d 的 map
  map: 'china',
  boxDepth: 50,
  regionHeight: 3,
  boxWidth: 90,
  label: {
    show: true, //显示省份标签
    color: '#fbfdfe', //省份标签字体颜色
    textStyle: {
      fontSize: '12'
    }
  },
  itemStyle: {
    color: '#B1D0EC', //图形的颜色
    borderColor: '#dadfde', //边线的颜色
    borderWidth: 1
  }
}]

}
myChart.setOption(option);
}
initEcharts();
```

10. 相关 js 源码下载:

1. 世界地图.js, 中国地图.js, 中国各省份地图.js

地址: <https://github.com/ecomfe/echarts/tree/master/map/js>

2. echarts 基础报表 js 源文件 (echarts.min.js)

地址: <https://github.com/ecomfe/echarts/tree/master/dist>

3. 3D 地图所需的 js 文件 (echarts-gl.min.js)

地址: <https://github.com/ecomfe/echarts-gl/tree/master/dist>
