# 大数据可视化方案调研

沈俊焘 软件开发部

### 一、调研目标

大数据可视化，也就是把数据以图表、图形、动画等形式展现出来的技术，需要有多种不同的形式来根据需要进行选择，如折线图、柱状图、饼图、散点图、仪表盘、3D动画等等。为了实现如下效果图，对目前流行的四种前端图表框架HT for Web、Highcharts、Echarts、G2从可行性、上手难易度、实现复杂度三方面进行对比分析。



### 二、介绍

HT for Web <http://www.hightopo.com>

HT是基于[HTML5](http://dev.w3.org/html5/spec/)标准的企业应用图形界面一站式解决方案， 其包含通用组件、拓扑组件和3D渲染引擎等丰富的图形界面开发类库，提供了完全基于HTML5的矢量编辑器、拓扑编辑器及 3D场景编辑器等多套可视化设计工具，和完善的类库开发手册、工具使用手册、及针对HTML5技术如何进行大规模团队开发的客户深度培训手册。

* 易用：易学易用，数小时的学习即可上手，几天的使用即可精通
* 轻量：包含通用组件、2D拓扑和3D引擎的核心包仅有300k左右
* 高性能：所有组件均可承载万以上的数据量，并能保持操作流畅
* 跨平台：可运行于桌面平台和移动终端所有支持的HTML5浏览器



Echarts <https://echarts.baidu.com>

ECharts，由百度开发，一个使用 JavaScript 实现的开源可视化库，可以流畅的运行在 PC 和移动设备上，兼容当前绝大部分浏览器（IE8/9/10/11，Chrome，Firefox，Safari等），底层依赖轻量级的矢量图形库 [ZRender](https://github.com/ecomfe/zrender" \t "https://www.echartsjs.com/_blank)，提供直观，交互丰富，可高度个性化定制的数据可视化图表。

* 丰富的可视化类型
* 多种数据格式无需转换直接使用
* 千万数据的前端展现
* 移动端优化
* 多渲染方案，跨平台使用
* 深度的交互式数据探索
* 多维数据的支持以及丰富的视觉编码手段
* 动态数据
* 绚丽的特效

Highcharts <https://www.highcharts.com.cn>

Highcharts 是一个用纯 JavaScript 编写的一个图表库， 能够很简单便捷的在 Web 网站或是 Web 应用程序添加有交互性的图表，并且免费提供给个人学习、个人网站和非商业用途使用。

Highcharts 支持的图表类型有直线图、曲线图、区域图、柱状图、饼状图、散状点图、仪表图、气泡图、瀑布流图等多达 20 种图表，其中很多图表可以集成在同一个图形中形成混合图。

* 兼容性
* 非商业使用免费
* 开源
* 纯JavaScript
* 丰富的图表类型
* 简单的配置语法
* 动态交互性



G2 <https://antv.alipay.com/zh-cn/g2/3.x/>

G2，由阿里开发，是一套基于可视化编码的图形语法，以数据驱动，具有高度的易用性和扩展性，用户无需关注各种繁琐的实现细节，一条语句即可构建出各种各样的可交互的统计图表。

* 千变万化，自由组合
* 生动、易实现
* 强大的交互语法

### 三、可行性

从效果图上来看，需要能绘制出柱状图、表格、折线图、仪表盘、3D建模以及动画效果。其中3D建模和动画只有HT for Web支持，其他基础类表格这四个框架都支持。从费用上来看，Highcharts是非商业用途免费，HT for Web网站上只有申请试用，是收费的，Echarts和G2均是免费使用。

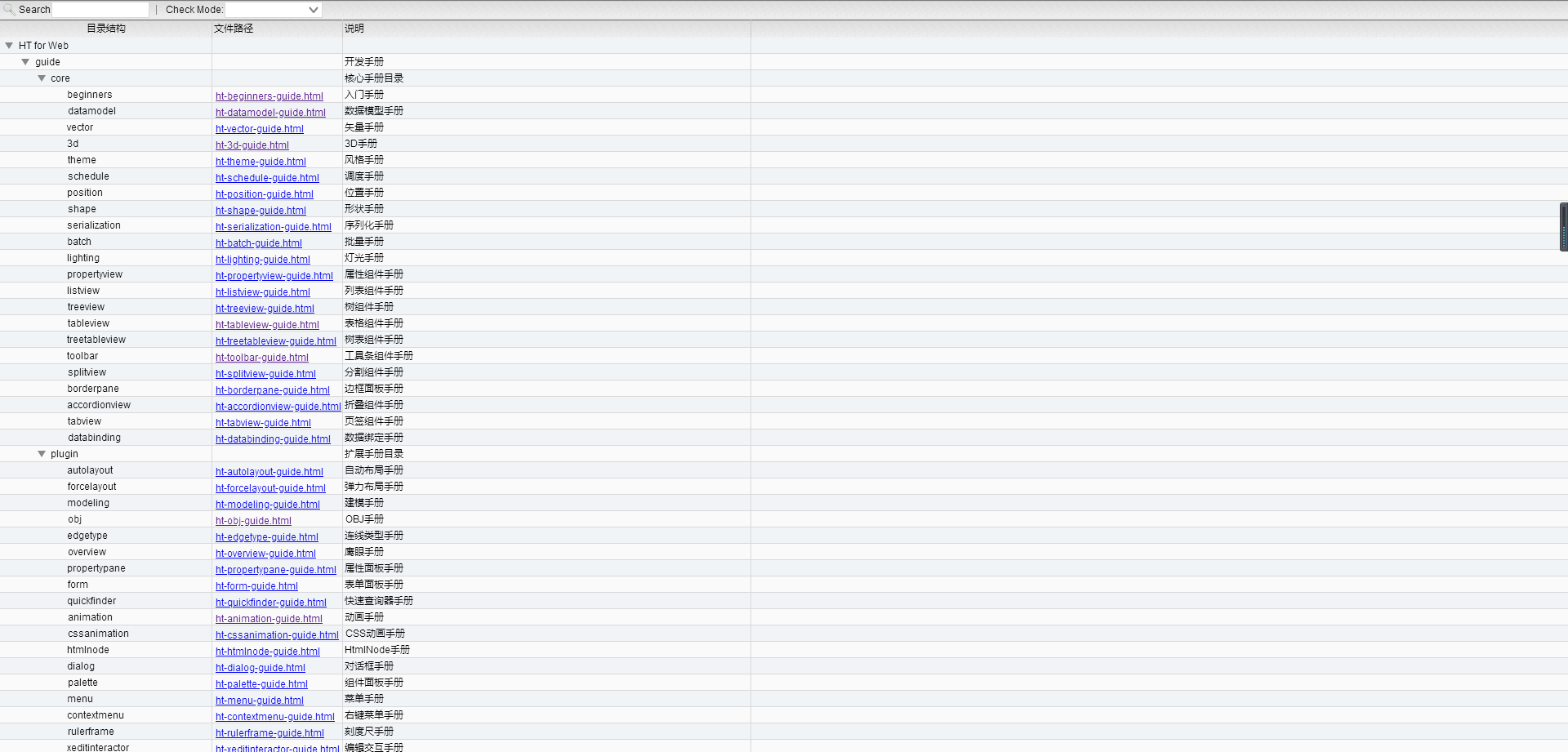
### 四、上手难易度

HT for Web

官网的入门教程文字繁多，阅读起来费时费力。文档中心更是不知道每个链接的内容，查找麻烦，学习曲线较陡峭。



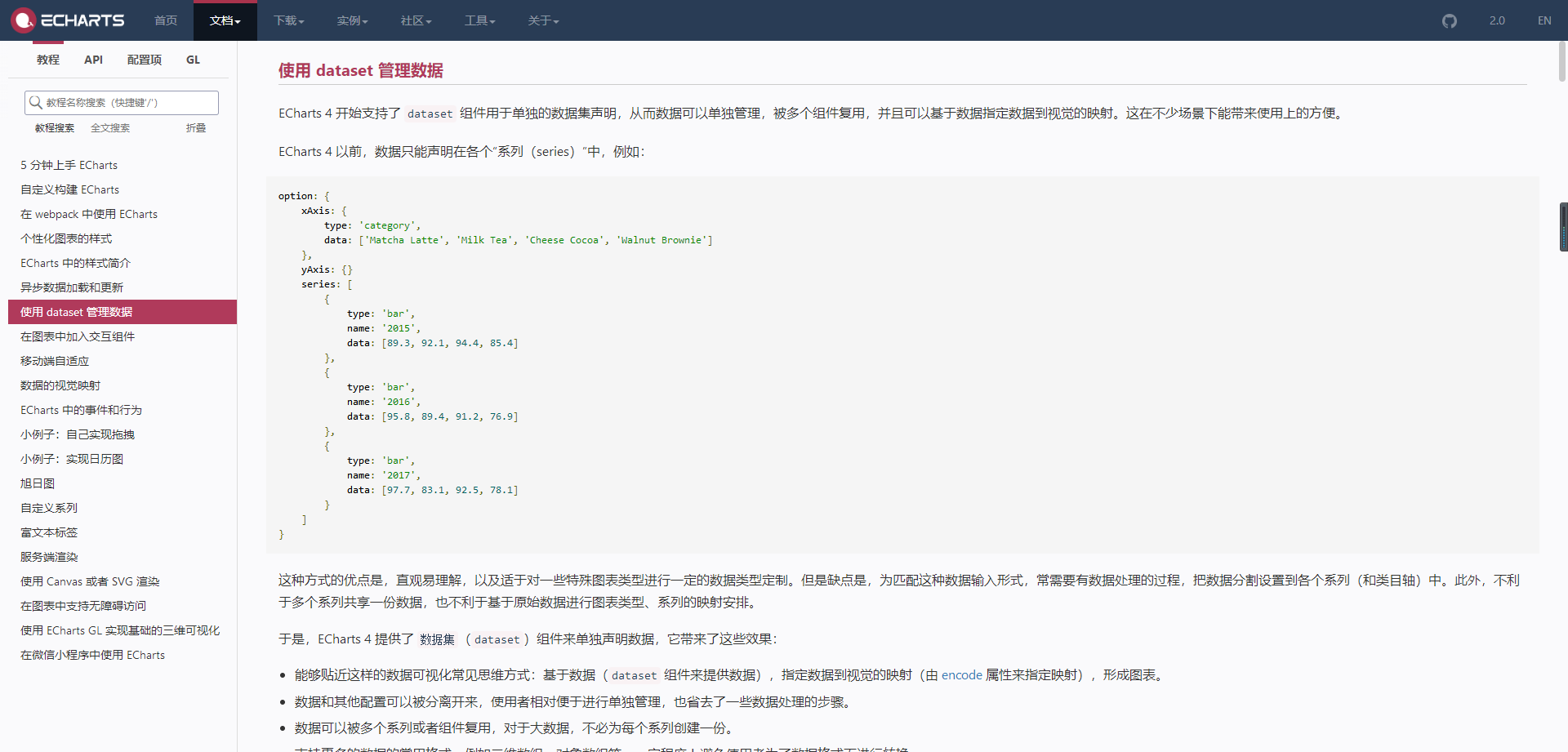
HT for Web入门教程



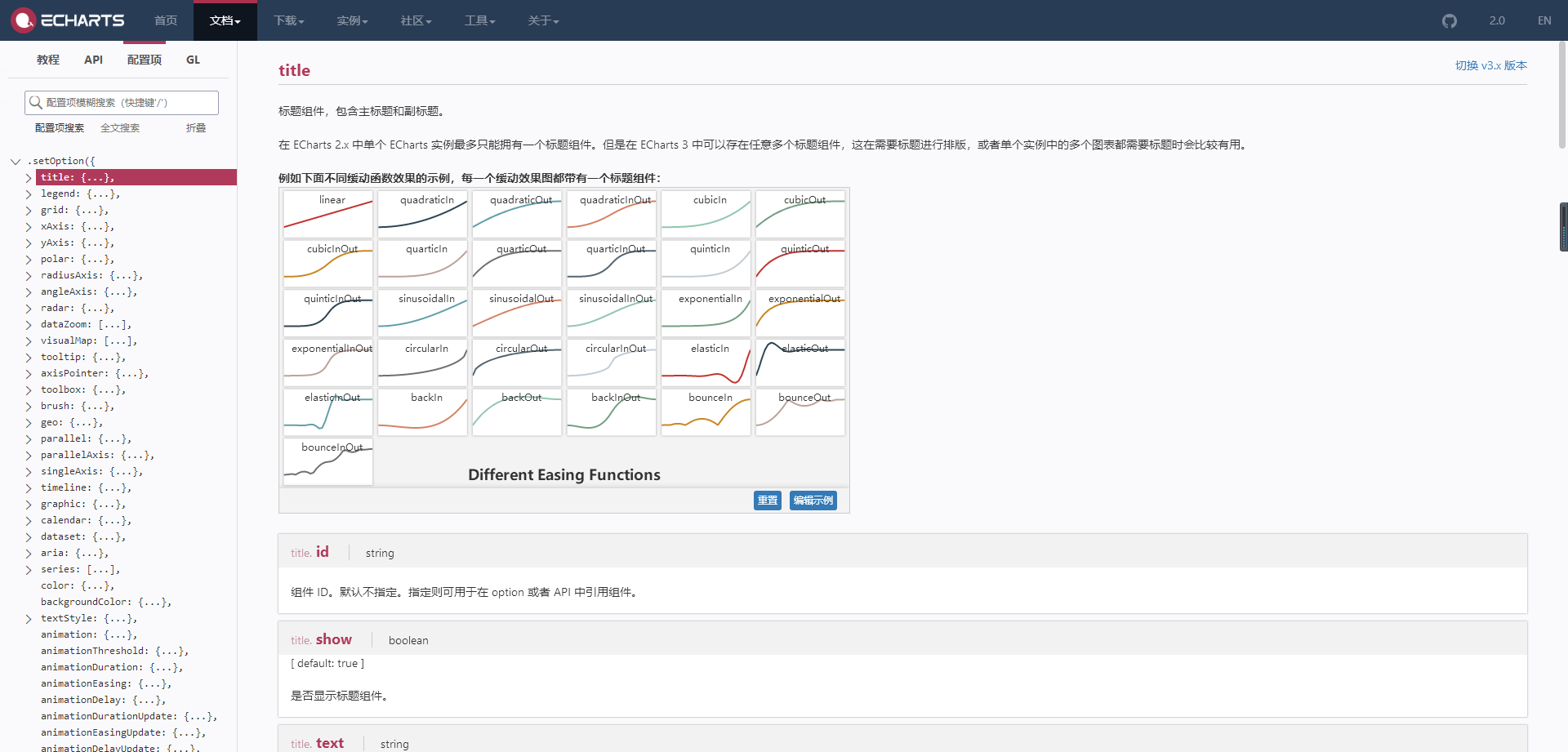
HT for Web文档中心

Echarts

官网有入门教程，API、配置项查找起来很方便，实例中更附带了源码编辑器，可以在线编辑并实时看到效果，很容易入门。



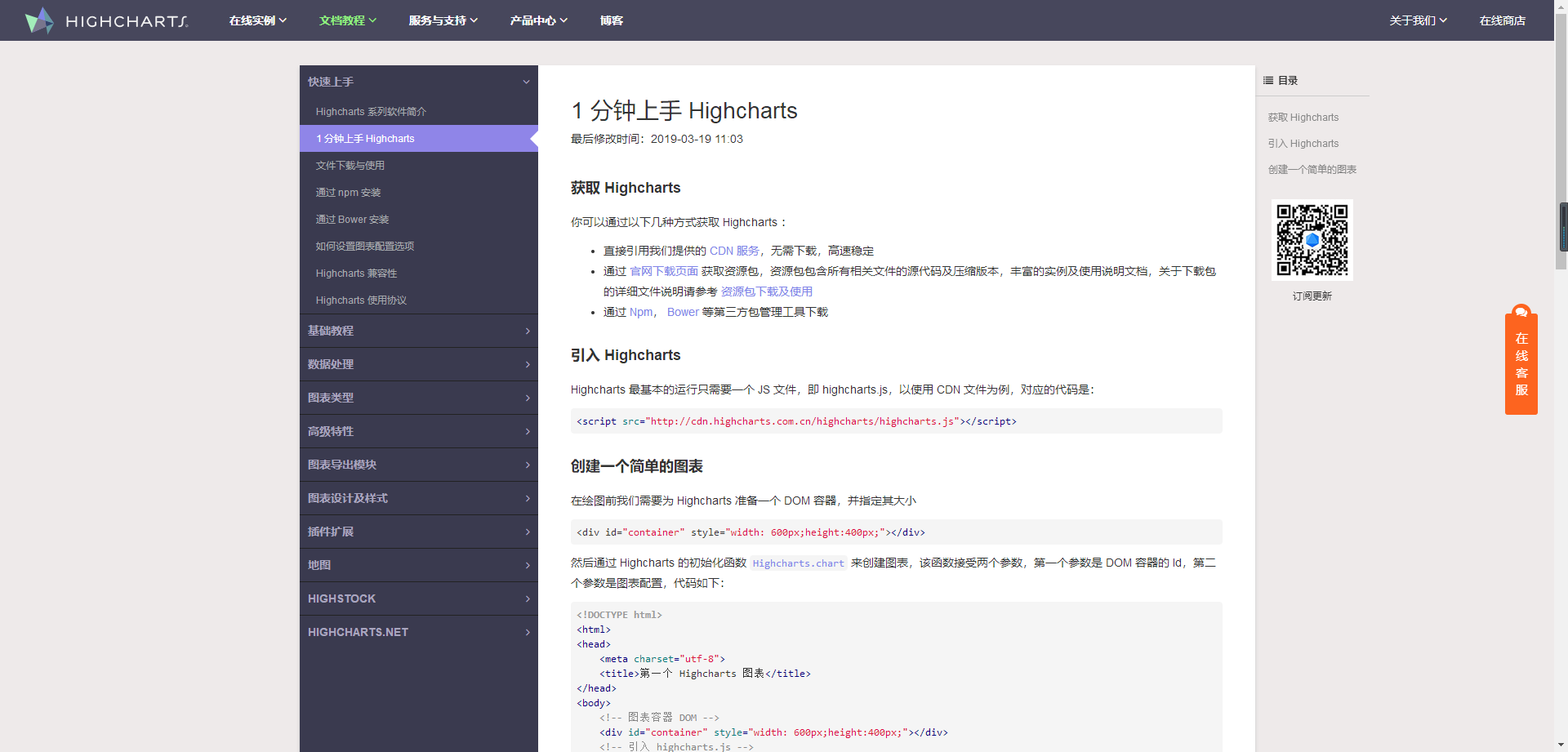
Echarts入门教程



Echarts配置项

Highcharts

官网的教程很详细，具体介绍了每一种图表类型，API也很详细，较容易入门。



Highcharts入门教程

G2

官网的教程详细，API相对来说不易查找。官方还有图表示例，图表示例中更附带了源码编辑器，可以在线编辑并实时看到效果，较容易入门。



G2教程

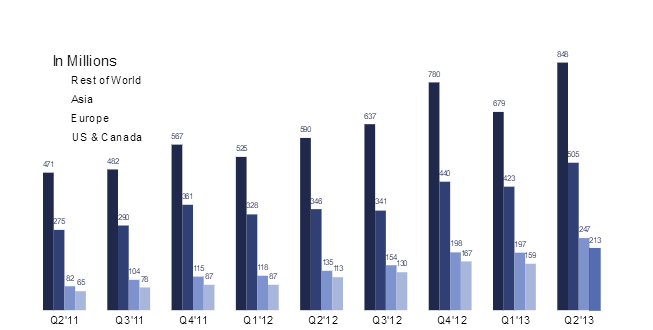


G2图表示例

### 五、实现复杂度

为了测试各个框架的实现复杂度，分别用它们实现一个类似的图表，来比较一下所需行数以及花费的时间。

HT for Web



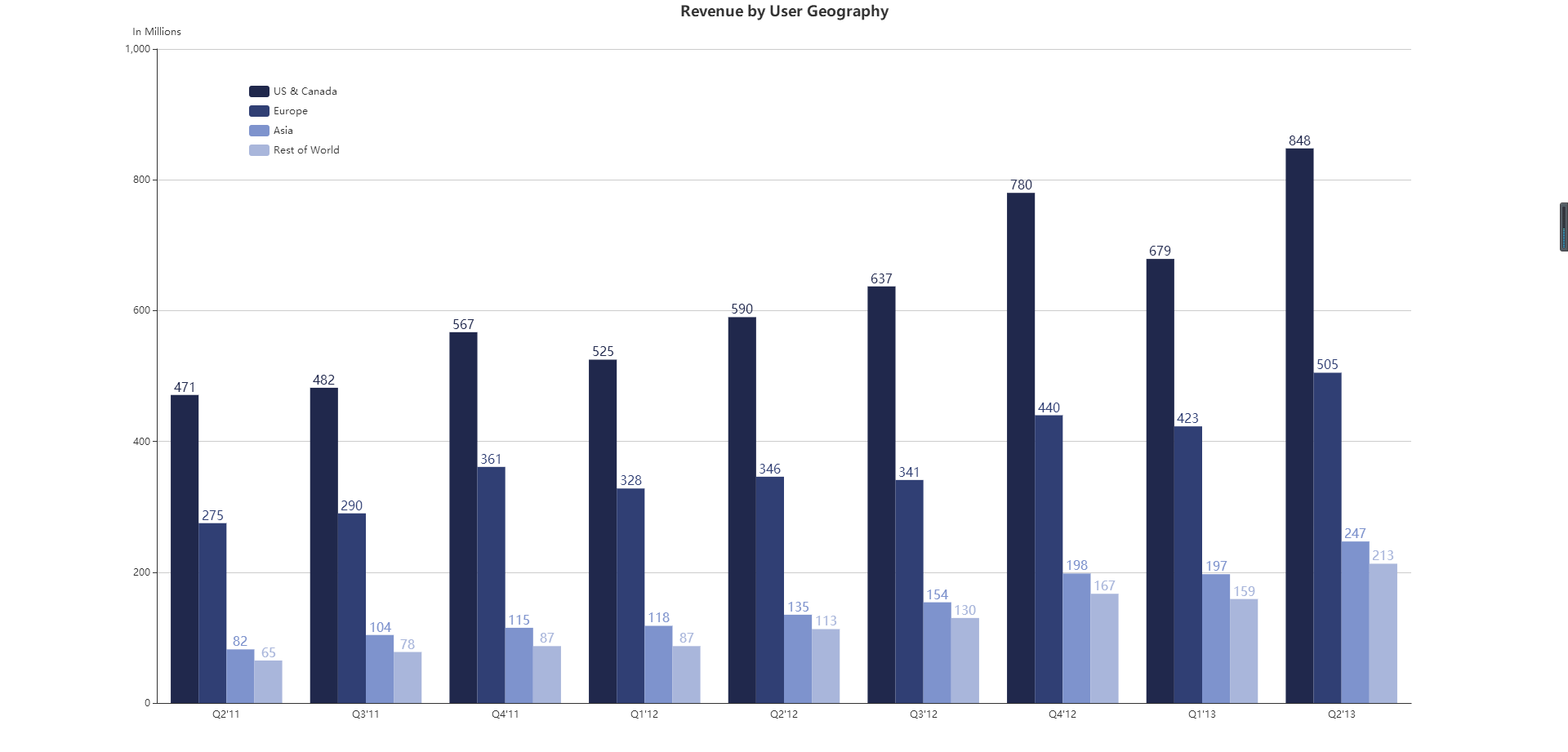
效果图



代码片段截图

实现这张图表，一共用了175行的代码，一个小时的时间。在实现过程中，找官方的例子就花了很久，后来在矢量图的例子里才找到。并且表格中的很多元素包括图例、横纵轴都得自己定义，比较麻烦。

Echarts



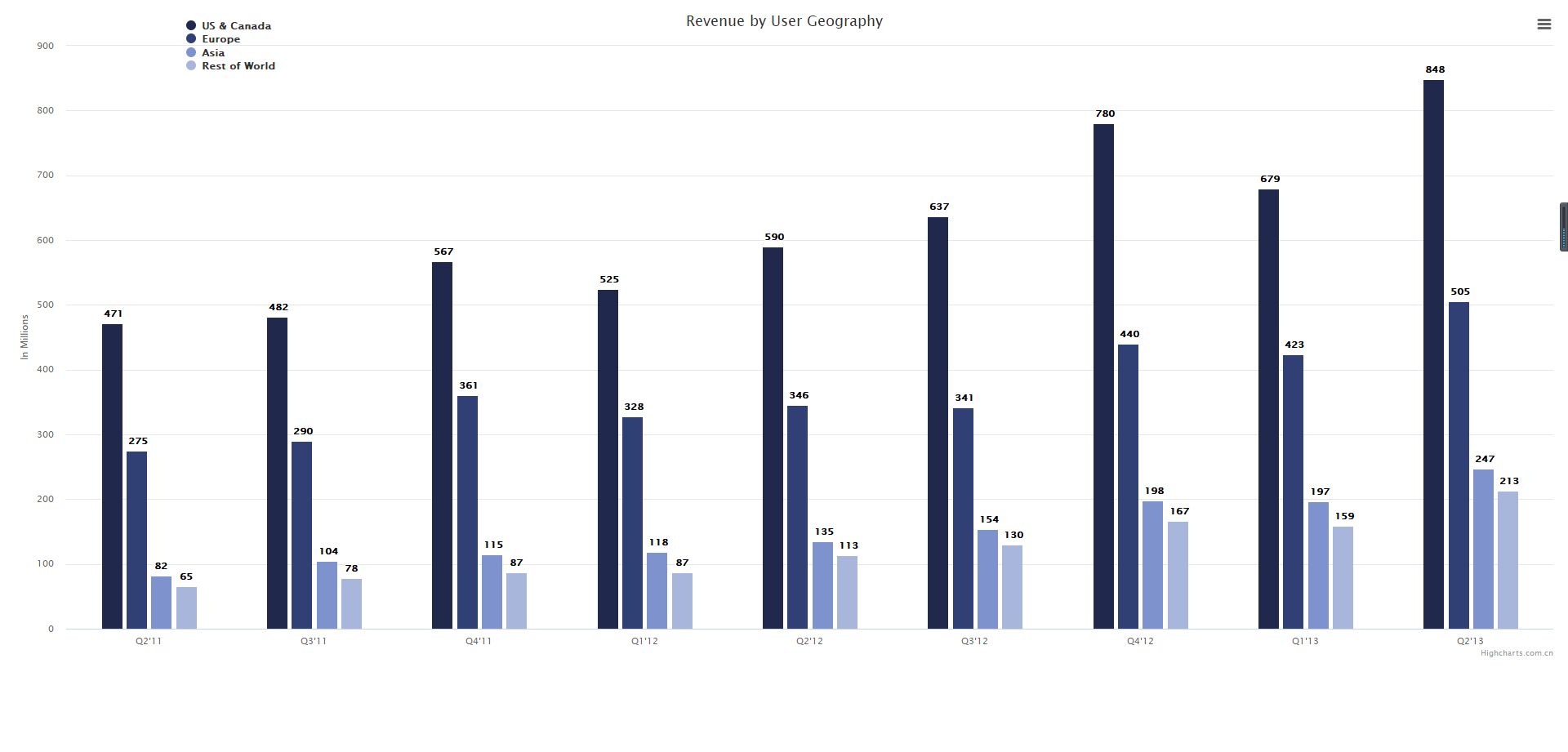
效果图



代码片段截图

实现这张图表，一共用了146行代码，二十分钟的时间。在官方的示例里很容易就找到了差不多的图表，修改一些数据，再照着API文档修改一些配置，就达到了差不多的效果。

Highcharts



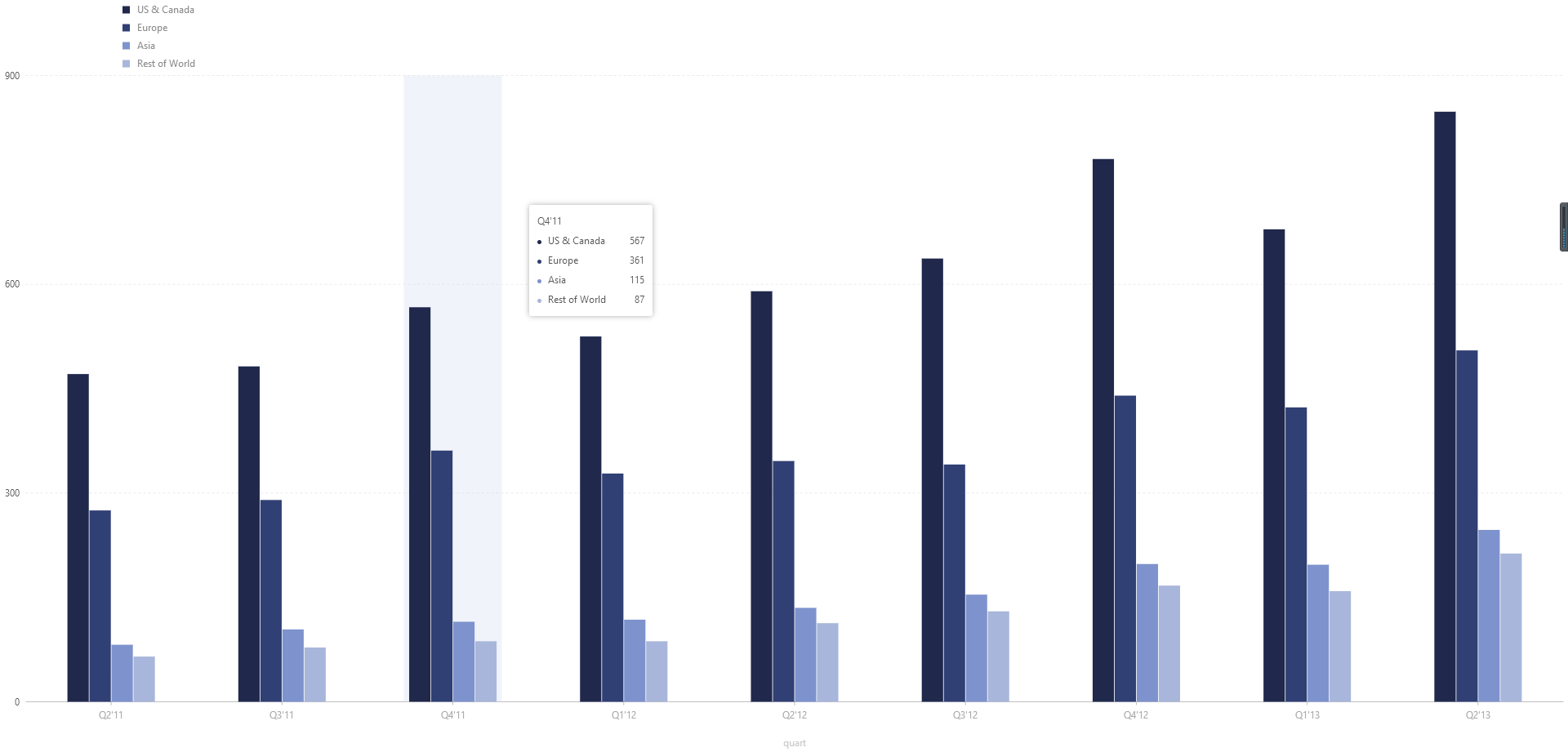
效果图



代码片段截图

实现这张图表，一共用了63行代码，三十分钟的时间。官方示例中很容易就找到了类似的图表，然后照着API修改一些配置，再改改数据就做出相似的效果了。可以看到所用代码行数非常少，相当简洁。

G2



效果图



代码片段截图

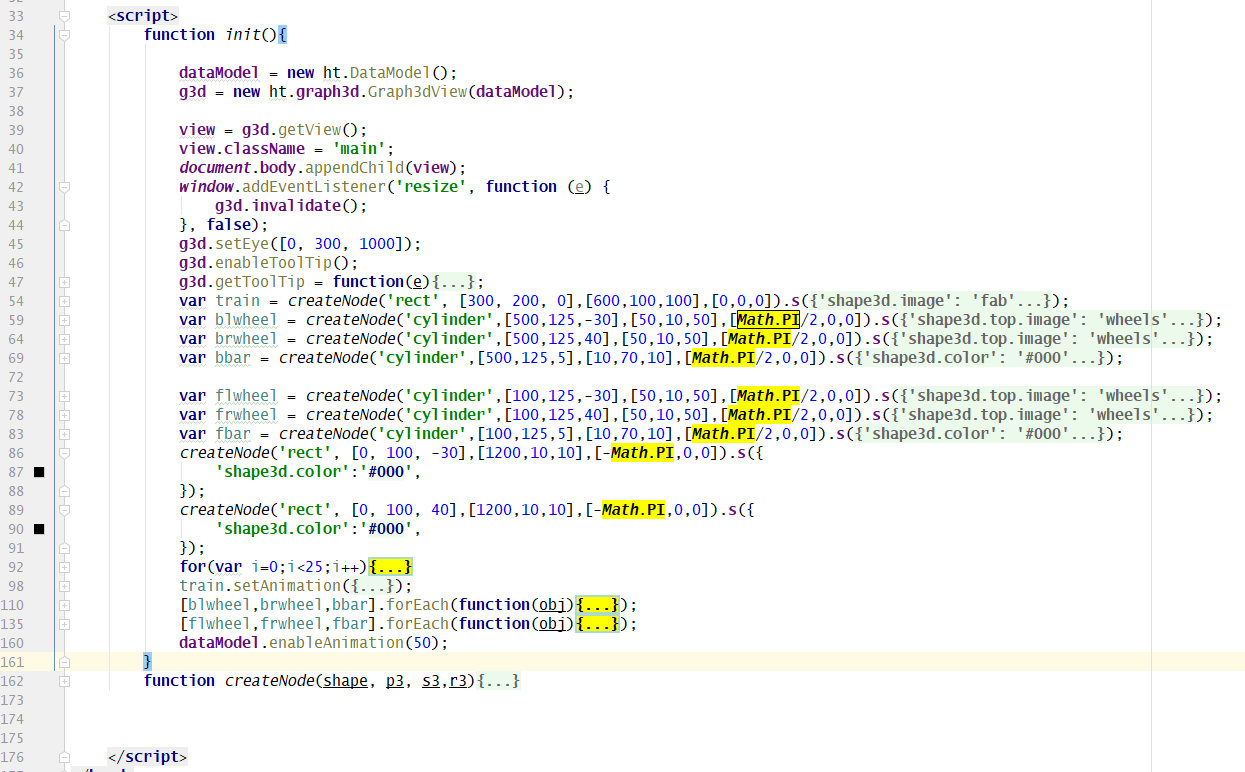
实现这张图表，一共用了196行代码，一个小时的时间。主要花费时间的地方在于G2的数据组织和其他的框架都不一样，其他的框架只需要一个数组就可以，他需要用json的格式来组织数据，每个json要包括横纵轴名称和值，就需要用java代码去遍历数组然后重新组装数据。G2还不支持给表格标题，这一点很费解。文档查找起来也比较麻烦，分类不是很清楚，导航也不方便。

下面再用它们分别实现一个动车行进的动画。

HT for Web







用了127行代码，花费四个小时的时间。

### 总结

除了3D建模和动画，HT for Web、Echarts、Highcharts、G2都可以实现需求。从上手难易度和实现复杂度来看，Echarts和Highcharts是最合适的框架。

3D建模和动画只有HT for Web可以实现。