

使用 ECharts 绘制网址径向树状图

作者: [RustFisher](#)

原文链接: <https://ld246.com/article/1630574051737>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

an.rustfisher.com有很多内容，很多页面。如果用一个树状图把所有页面展示出来会是什么效果？第一时间想到了ECharts。

最后效果：<https://an.rustfisher.com/an-tree.html>

数据处理

数据来源于配置文件。我们只需要把内容读出来，整理成ECharts需要的格式并保存为文件。

yml

Android站用的框架是Mkdocs，网址都设置在mkdocs.yml里。关键配置如下

```
nav:  
- Android basic:  
  - 开始:  
    - 新建工程: android/startProject/start_new_project.md  
    - 使用Kotlin: android/startProject/as-use-kotlin.md  
  - Activity:  
    - Activity综述: android/activity/overview.md
```

我们需要把这些数据读出来，弄成ECharts可以用的格式。

打开PyCharm，用python来处理数据。

python处理数据

我们使用3.7版本python。要读取yml的内容，不想用读文本的方式一行行来读，还得自己解析。

python中有一个库PyYAML专门用于读取yml格式的数据。

在PyCharms里装上这个包。用它来读取yml里的数据。

```
import yaml  
  
with open(md_file, encoding='utf8') as a_yaml_file:  
    # 解析yaml  
    yaml_data = yaml.load(a_yaml_file, Loader=yaml.FullLoader)  
    nav = yaml_data['nav']
```

ECharts需要的格式是name和children。同时可以添加别的字段。

```
{  
  "name": "Android教程",  
  "children": [  
    {  
      "name": "Android basic",  
      "children": []  
    }  
  ]  
}
```

读取到的数据层级比较多，用递归方法来处理它们。把所有的数据都遍历一遍，整理成需要的格式。

递归的时候，遇到数组，则继续递归；遇到字符串，则表示可以拼接网址了。

完整脚本如下

```
import json
import os
import yaml

def gen_url_map(main_site, md_file):
    with open(md_file, encoding='utf8') as a_yaml_file:
        # 解析yaml
        yaml_data = yaml.load(a_yaml_file, Loader=yaml.FullLoader)

        nav = yaml_data['nav']

        sorted_data = {"name": "Android教程", "children": [], "collapsed": False}

        for big_dict in nav:
            load_url(main_site, big_dict, sorted_data)

        url_file = 'android-url-map.json'
        if os.path.exists(url_file):
            os.remove(url_file)
        with open(url_file, 'w') as s:
            s.write(str(json.dumps(sorted_data, ensure_ascii=False)))

def load_url(main_site, input_dict, res_data):
    """
    递归处理数据
    """
    for k1 in input_dict:
        v1 = input_dict[k1]
        if isinstance(v1, list):
            item1 = {"name": k1, "children": []}
            res_data['children'].append(item1)
            for v1_child in v1:
                load_url(main_site, v1_child, item1)
        elif isinstance(v1, str):
            # 此时已经是url了
            final_item = {"name": k1, "url": main_site + v1[0:-3]}
            res_data['children'].append(final_item)
            return final_item

if __name__ == '__main__':
    print("生成android站的sitemap")
    gen_url_map('https://www.an.rustfisher.com/',
                '/Users/rustfisher/mkdocs.yml')
```

最后写一个**android-url-map.json**文件。

制作网页

经过观察，我们选用径向树状图（tree-radial）。

查看ECharts的径向树状图[使用示例](#)，下载一个示例html，在它的基础上修改。

添加上[jquery](#)的引用。

```
<script src="https://cdn.bootcss.com/jquery/3.4.1/jquery.js"></script>
```

修改要读取的json文件名

```
$.getJSON('./android-url-map.json', function (data) { /* ... */ }
```

把前面得到的json文件和这个html文件放在同一层级，上传服务器。

打开：<https://an.rustfisher.com/an-tree.html> 可以看到效果。

扩大显示范围

这个时候的网页体验不是很好，页面不够大，图表超出上下范围了。可以给它添个竖直的滚动条 `overflow-y: scroll`。

```
<html style="height: 100%; width: 100%; overflow-y: scroll;">
```

修改一下网页的 `body`。外层的 `container` 占100%宽高，里面的 `chartContainer` 给一个 `min-height: 1600px`。

```
<body style="height: 100%; margin: 0; width: 100%;">
  <div id="container" style="height: 100%; width: 100%;">
    <div id="chartContainer" style="width: 100%; min-height: 1600px"></div>
  </div>
```

script里加上对 `resize` 的监听

```
$(window).on('resize', function () {
  if (myChart != null && myChart != undefined) {
    myChart.resize();
  }
});
```

这时再看浏览器的效果，图表有了更多的空间显示。

参考

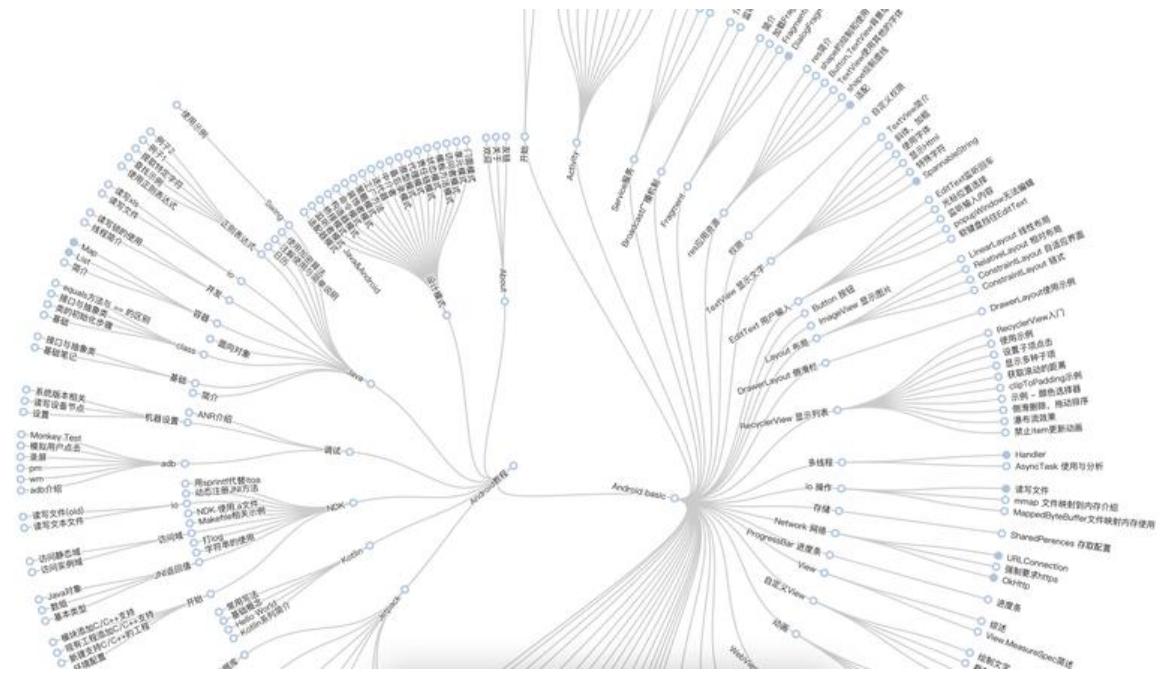
- <https://stackoverflow.com/questions/33454946/echarts-and-bootstrap-responsive-width>
- <https://stackoverflow.com/questions/12575251/why-is-my-page-not-scrollable>

点击打开页面

默认点击效果是折叠和展开。对于最末尾的叶子节点，加一个点击事件让它打开相应的界面。

```
myChart.on('click', function (params) {
  if (params.data && params.data.url) {
    window.open(params.data.url);
  }
});
```

});



最后效果：<https://an.rustfisher.com/an-tree.html>