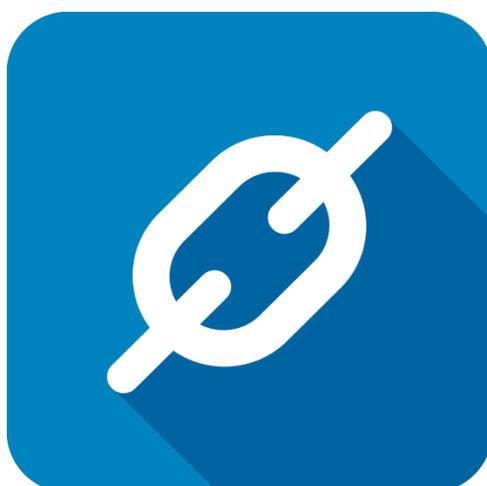


TotalLINK

产品手册



上海朝识智能科技有限公司

2017年5月

图表设计

目 录

图表设计.....	2
1 百度 ECharts 集成.....	3
2 启动图表设计.....	6
3 选择图表类型.....	7
4 南丁格尔玫瑰图实例.....	8
5 图表数据编辑.....	9
6 图表参数调整.....	10
7 完成设计.....	11
8 显示代码.....	12
9 复制代码.....	13
10 代码转义符号转换.....	14
11 代码脚本处理.....	15
12 图表配置手册.....	16
13 动态取值.....	17
14 模型脚本填写.....	18
15 展现结果.....	19
16 模型脚本修改.....	20
17 重新查看展现结果.....	22
18 查看图表.....	23
19 直接展示图表.....	24
20 高级图表设计.....	25

文档控制

■ 主要内容

TotalLINK 系统集成百度 Echarts 图表，开发者可根据本文配置图表，在 app 查看效果，也可以集成仅看板中。

■ 更改记录

日期	版本	作者	备注
2017-05	1.0	Randy	初始发布

■ 支持版本

非特殊说明的功能，默认前后版本都支持

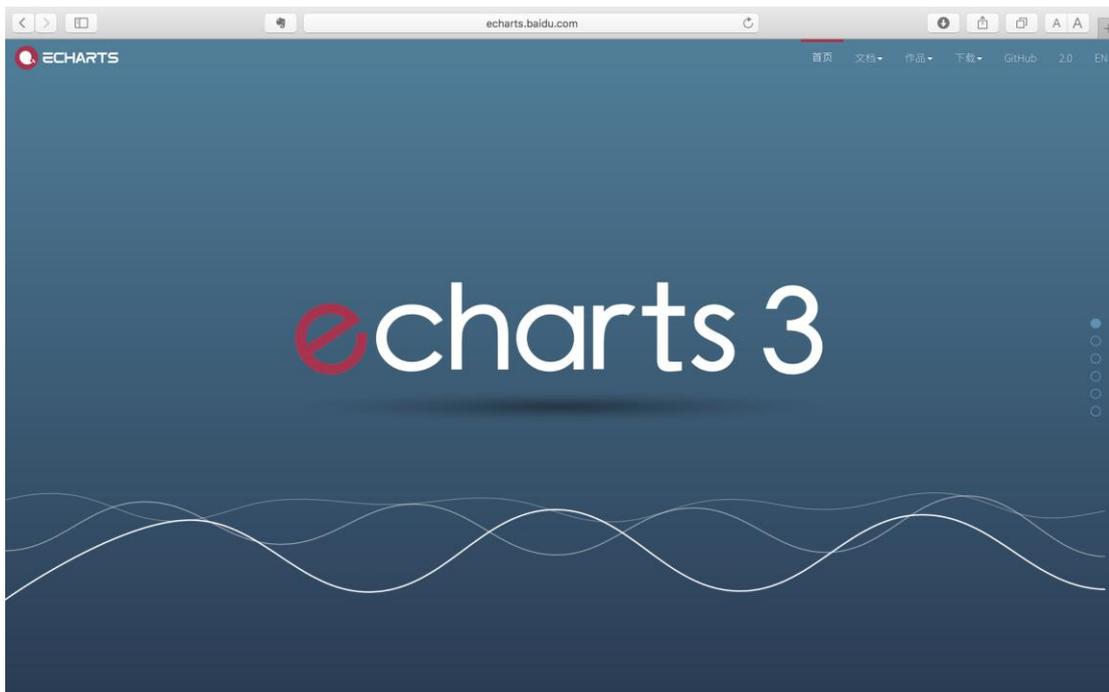
仅支持V20版本及以后版本的功能点

-

1 百度 ECharts 集成



- TotalLINK 系统与百度 ECharts 实现深度集成，从系统中启动图表设计向导
- 可以方便、快捷地利用百度 ECharts 的功能实现各种数据图表



- 系统已经设计了完善的数据传递模式，对于各类业务系统的数据，只需要在模型设计时，通过数据获取取得数据集，即可实现迅速实现使用百度 ECharts 在移动端的数据展现
- 因此，可以充分利用 TotalLINK 集成各种不同业务系统数据的特点，实现丰富的图表展现
- 有关百度 ECharts 的具体功能介绍，请参考相关网站
 - <http://echarts.baidu.com>

2 启动图表设计



- 百度 eCharts 具备丰富的工具，用于方便实现图表展现
- 其特点为：
 - 零编程玩转图表
 - 大数据彰显魅力
- 启动 ECharts 团队出品的“百度图说”功能，
- 打开下面的连接，可以直接使用图表设计功能
 - <http://tushuo.baidu.com>

3 选择图表类型



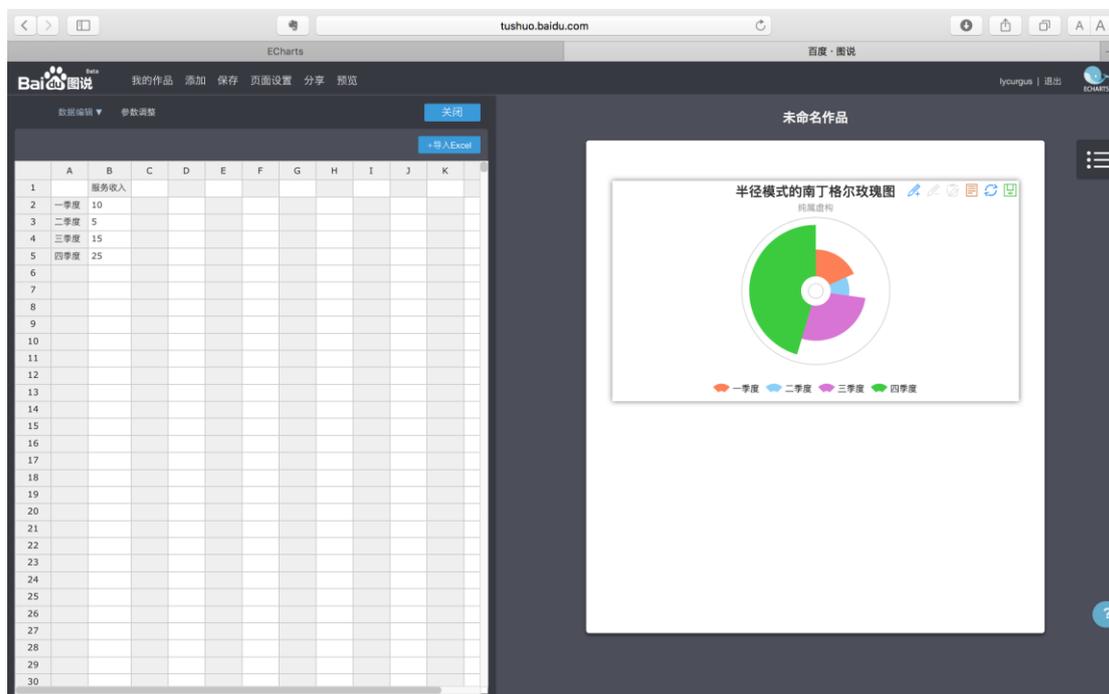
- 点击“百度图说”功能的“开始制作图表”，选择“创建图表”
- 根据需要选择“图表类型”，当前可选的图表类型包括：
 - 折线图
 - 柱状图
 - 饼图
 - 地图
 - 散点图
 - 气泡图
 - 雷达图
 - 漏斗图
 - 仪表盘
- 百度 ECharts 的功能还在不断丰富中，截止到 3.4 版本，已经发布了“象形柱图”和“主题河流图”等
- 具体内容可参考百度 ECharts 网站的相关文档
- 以下链接为 ECharts 实例
 - <http://echarts.baidu.com/examples.html>

4 南丁格尔玫瑰图实例



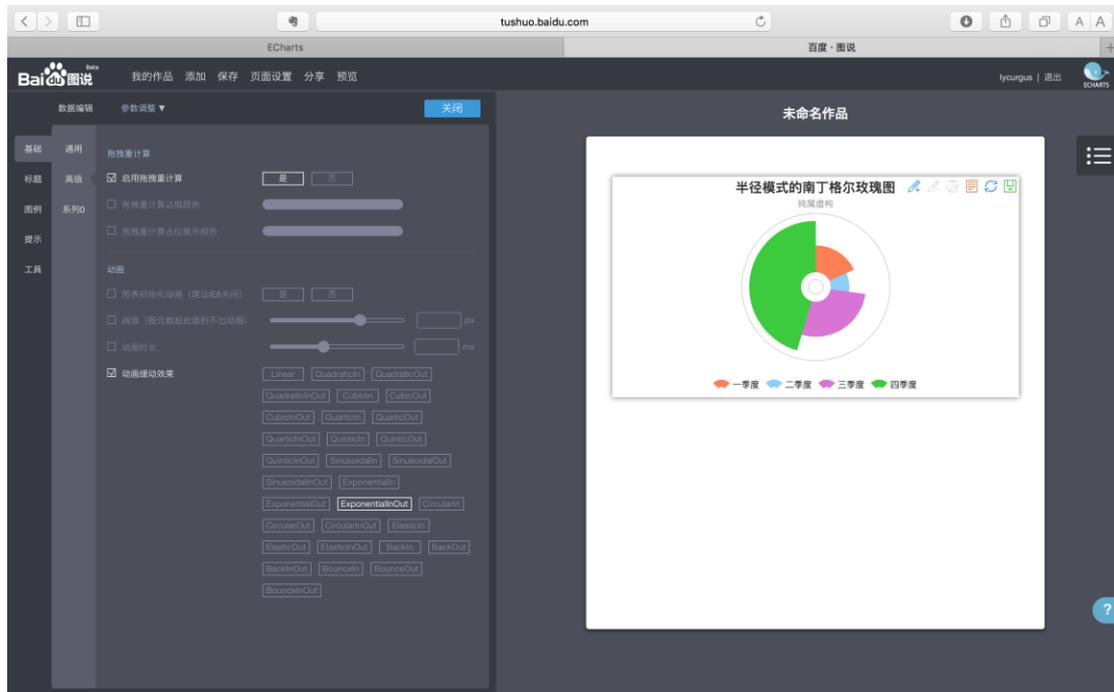
- 根据需要，本实例选择“南丁格尔玫瑰图”
- 接下来开始具体的图表设计

5 图表数据编辑



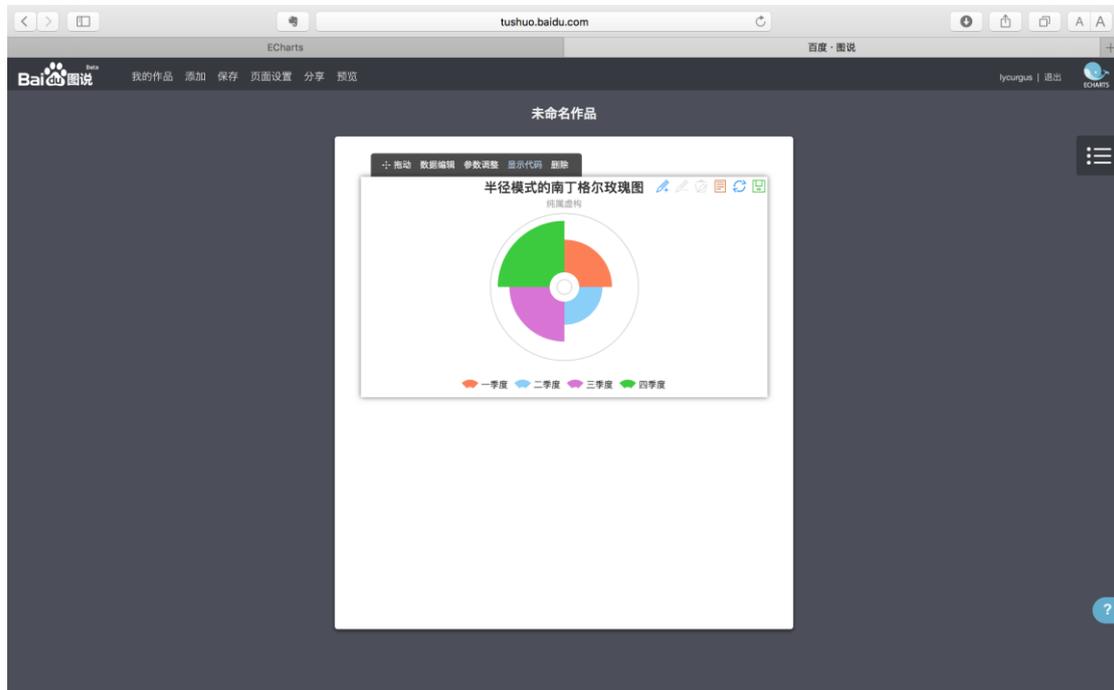
- 选择数据编辑功能
- 在左侧填入示例数据
- 可以观察到右侧的图表在跟随变化
- 此处只需要填入一些用于图表设计的示例数据即可，后面我们会修改图表脚本，以方便动态使用 TotalLINK 系统获取的各个业务系统的数据

6 图表参数调整



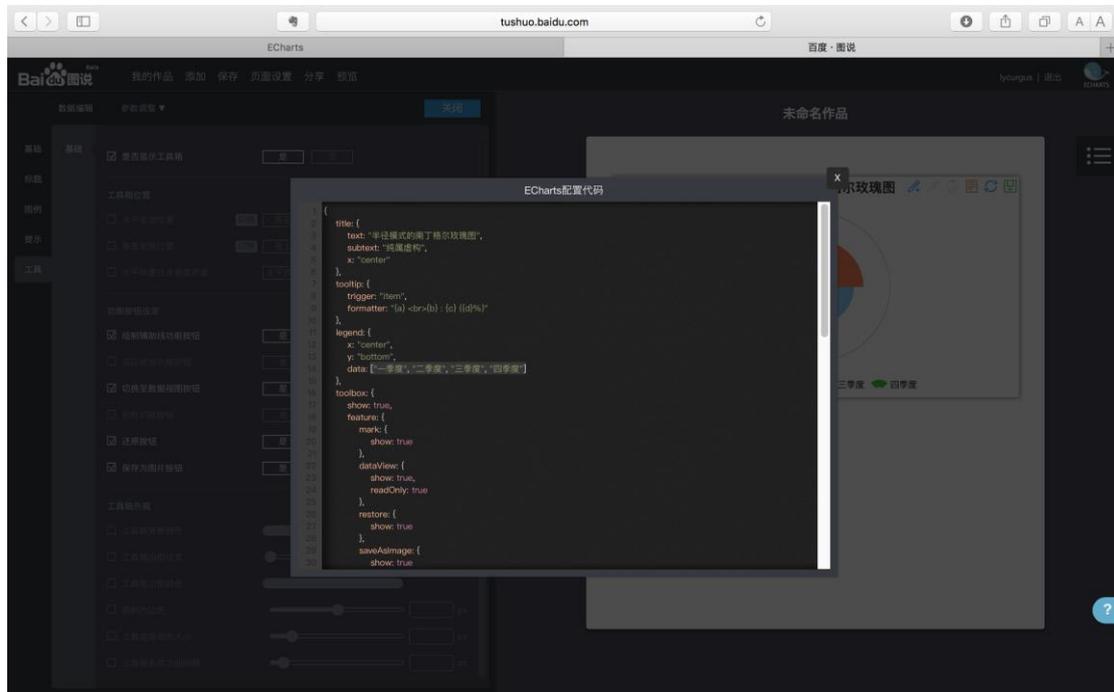
- 打开图表的参数调整功能，可以对图表的细节进行“设计”
- 可以设计的内容跟具体的图表有关，一般包括：
 - 基础，包括图表尺寸、系列格式、动画显示效果等
 - 标题，包括标题文本及副标题文本等
 - 图例，图例的显示格式及位置等
 - 提示，提示板的外观及格式等
 - 工具，工具箱的具体设置等

7 完成设计



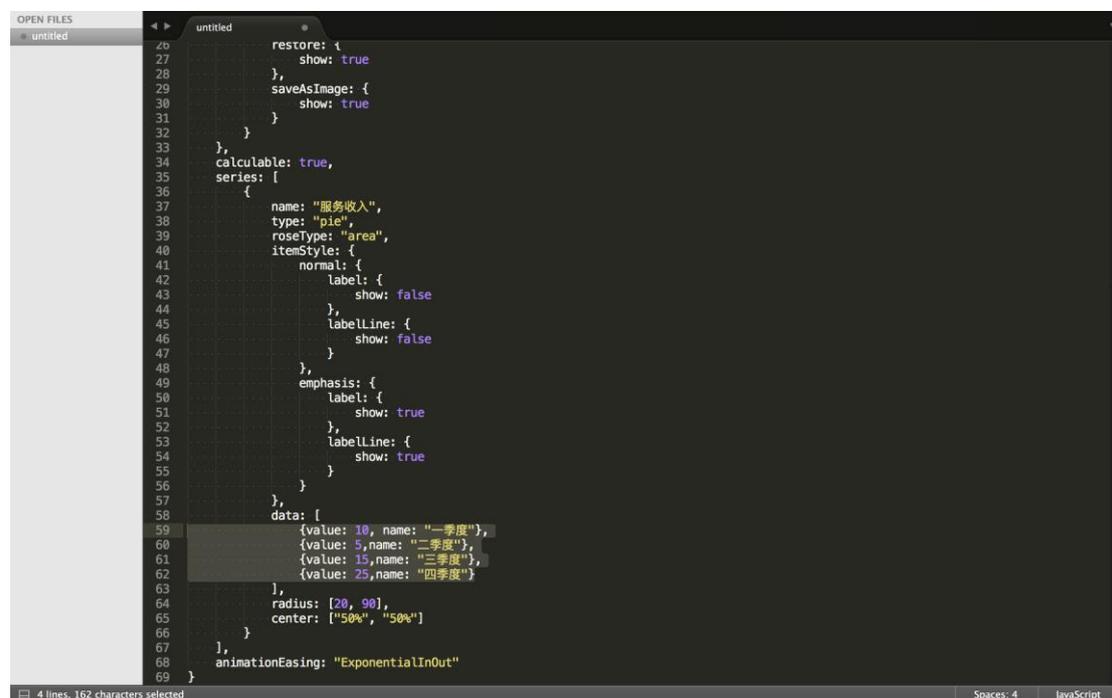
- 图表数据示例及格式设计完成后，系统已经为用户生成了图表的“代码”

8 显示代码



- 完成图表设计后，打开“显示代码”功能
- 可以看到系统已经自动生成了图表展示的代码
- 此时，代码中包含了一些“静态的示例数据”

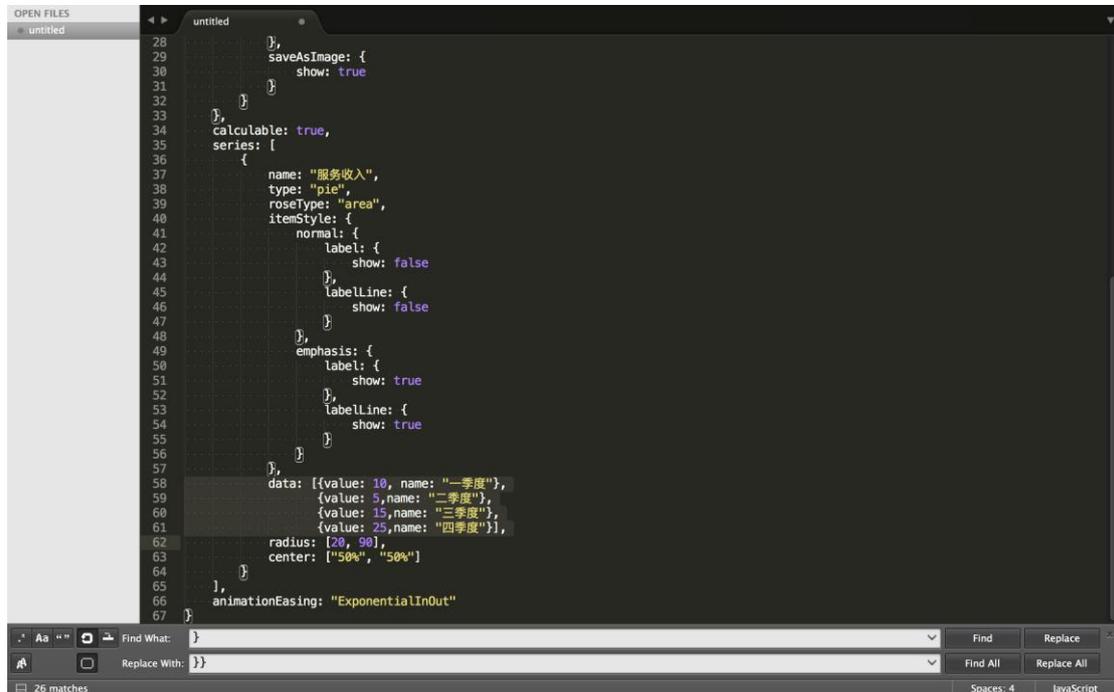
9 复制代码



```
26     restore: {
27       show: true
28     },
29     saveAsImage: {
30       show: true
31     }
32   },
33   calculable: true,
34   series: [
35     {
36       name: "服务收入",
37       type: "pie",
38       roseType: "area",
39       itemStyle: {
40         normal: {
41           label: {
42             show: false
43           },
44           labelLine: {
45             show: false
46           }
47         },
48         emphasis: {
49           label: {
50             show: true
51           },
52           labelLine: {
53             show: true
54           }
55         }
56       },
57       data: [
58         {value: 10, name: "一季度"},
59         {value: 5, name: "二季度"},
60         {value: 15, name: "三季度"},
61         {value: 25, name: "四季度"}
62       ],
63       radius: [20, 90],
64       center: ["50%", "50%"]
65     }
66   ],
67   animationEasing: "ExponentialInOut"
68 }
69 }
```

- 将系统自动生成的代码，复制出来
- 在 sublime 等文本编辑器中，可以看到代码的具体内容
- 如上一节描述，代码中包含了一些“静态的示例数据”，后文将描述如何使用“TotalLINK”系统获取的动态业务数据

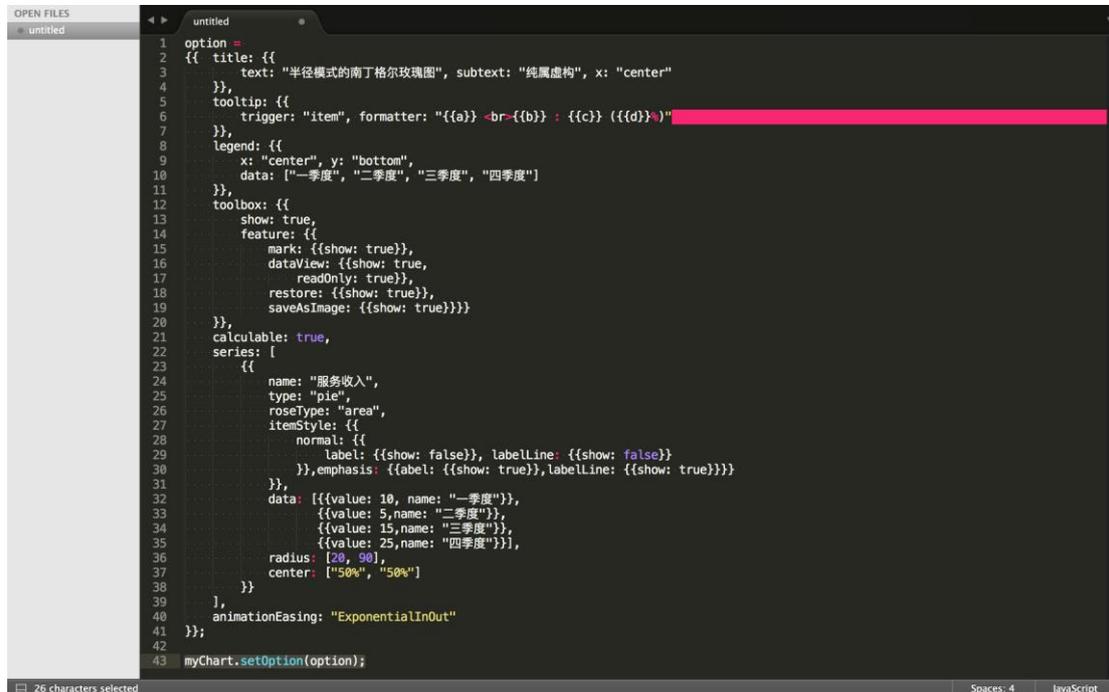
10 代码转义符号转换



```
28     },
29     saveAsImage: {
30       show: true
31     }
32   },
33   calculable: true,
34   series: [
35     {
36       name: "服务收入",
37       type: "pie",
38       roseType: "area",
39       itemStyle: {
40         normal: {
41           label: {
42             show: false
43           },
44           labelLine: {
45             show: false
46           }
47         },
48         emphasis: {
49           label: {
50             show: true
51           },
52           labelLine: {
53             show: true
54           }
55         }
56       },
57       data: [{value: 10, name: "一季度"},
58             {value: 5, name: "二季度"},
59             {value: 15, name: "三季度"},
60             {value: 25, name: "四季度"}],
61       radius: [20, 90],
62       center: ["50%", "50%"]
63     },
64     animationEasing: "ExponentialInOut"
65   ]
66 }
67 }
```

- 根据“TotalLINK”系统的要求，在进行脚本的动态处理时，可能会用到类似“{0}”等表示参数的方法
- 在图表代码中，用于描述图表的JSON数据中也包括了大量的“{”和“}”，为了系统正常理解“参数”和“文本中的{}”，需要对文本中的“{}”进行转义转换
- 转换的方法是：
 - 用“{{”代替“{”
 - 用“}}"代替“}”
- 注意
 - TotalLINK脚本中允许使用“{0}”等代替真实的数据参数，这里的“{”不要转义符号替换
 - 如果脚本中未使用到“{0}”等参数，可以跳过此步骤，不需要进行花括号的转义替换

11 代码脚本处理



```
1 option =
2 {{ title: {{
3   text: "半径模式的南丁格尔玫瑰图", subtext: "纯属虚构", x: "center"
4   }},
5   tooltip: {{
6     trigger: "item", formatter: "{{a}} -br-{{b}} : {{c}} ({{d}}%)"
7   }},
8   legend: {{
9     x: "center", y: "bottom",
10    data: ["一季度", "二季度", "三季度", "四季度"]
11  }},
12  toolbox: {{
13    show: true,
14    feature: {{
15      mark: {{show: true}},
16      dataView: {{show: true,
17        readOnly: true}},
18      restore: {{show: true}},
19      saveAsImage: {{show: true}}}
20  }},
21  calculable: true,
22  series: [
23    {{
24      name: "服务收入",
25      type: "pie",
26      roseType: "area",
27      itemStyle: {{
28        normal: {{
29          label: {{show: false}}, labelLine: {{show: false}}
30        }}, emphasis: {{abel: {{show: true}}, labelLine: {{show: true}}}
31      }},
32      data: [{{value: 10, name: "一季度"}},
33        {{value: 5, name: "二季度"}},
34        {{value: 15, name: "三季度"}},
35        {{value: 25, name: "四季度"}},
36      ],
37      radius: [20, 90],
38      center: ["50%", "50%"]
39    }}
40  ],
41  animationEasing: "ExponentialInOut"
42 }};
43 myChart.setOption(option);
```

- 参考上图，对数据脚本进行几个简单的转换
 - 在脚本的开头，添加“option=”
 - ◆ 表示，后面的数据是用户图表展示的 option 数据
 - ◆ 如果需要理解图表配置的具体内容，可以参考 ECharts 配置手册
 - ◆ <http://echarts.baidu.com/option.html#title>
 - 在脚本的最后，添加一行内容
 - ◆ myChart.setOption(option);
 - 因为这里使用的是 JavaScript 语法，对大小写是敏感的，需要注意脚本的大小写

12 图表配置手册

title

标题组件，包含主标题和副标题。

在 ECharts 2.x 中单个 ECharts 实例最多只能拥有一个标题组件。但是在 ECharts 3 中可以存在任意多个标题组件，这在需要标题进行排版，或者单个实例中的多个图表都需要标题时会比较有用。

例如下面不同缓动函数效果的示例，每一个缓动效果图都带有一个标题组件：

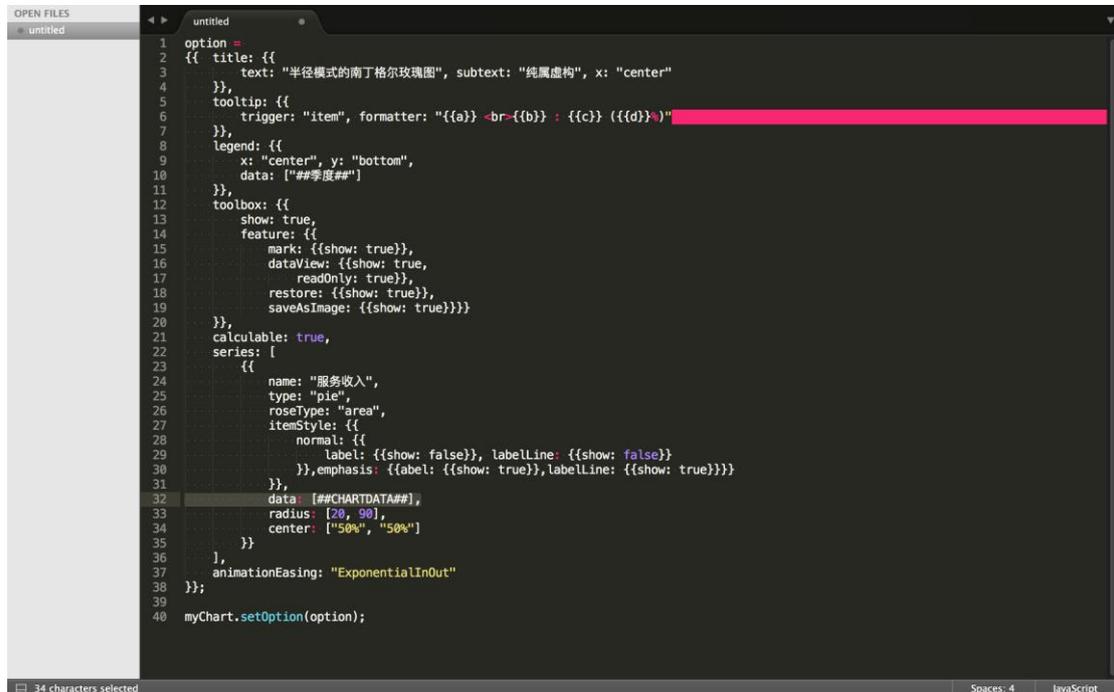
linear	quadraticIn	quadraticOut	quadraticInOut	cubicIn	cubicOut
cubicInOut	quarticIn	quarticOut	quarticInOut	quinticIn	quinticOut
quinticInOut	sinusoidalIn	sinusoidalOut	sinusoidalInOut	exponentialIn	exponentialOut
exponentialInOut	circularIn	circularOut	circularInOut	elasticIn	elasticOut
elasticInOut	backIn	backOut	backInOut	bounceIn	bounceOut
bounceInOut					

Different Easing Functions

title **target** | string
[default: 'blank']
指定窗口打开主标题超链接。
可选：
* 窗口打开前窗口打开

- 百度 ECharts 提供了详细的配置项手册
- 用户可以根据需要对照修改系统自动生成的图表配置项脚本
- 当然，根据这里手册的内容，通过代码编写图表展示格式也是完全可以的

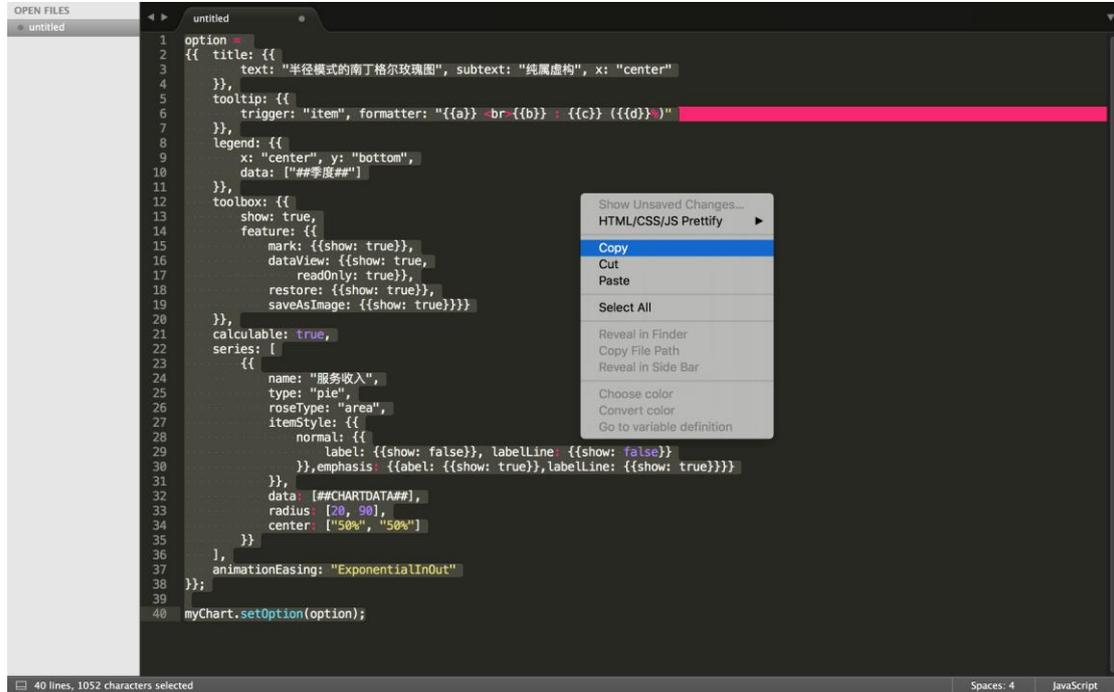
13 动态取值



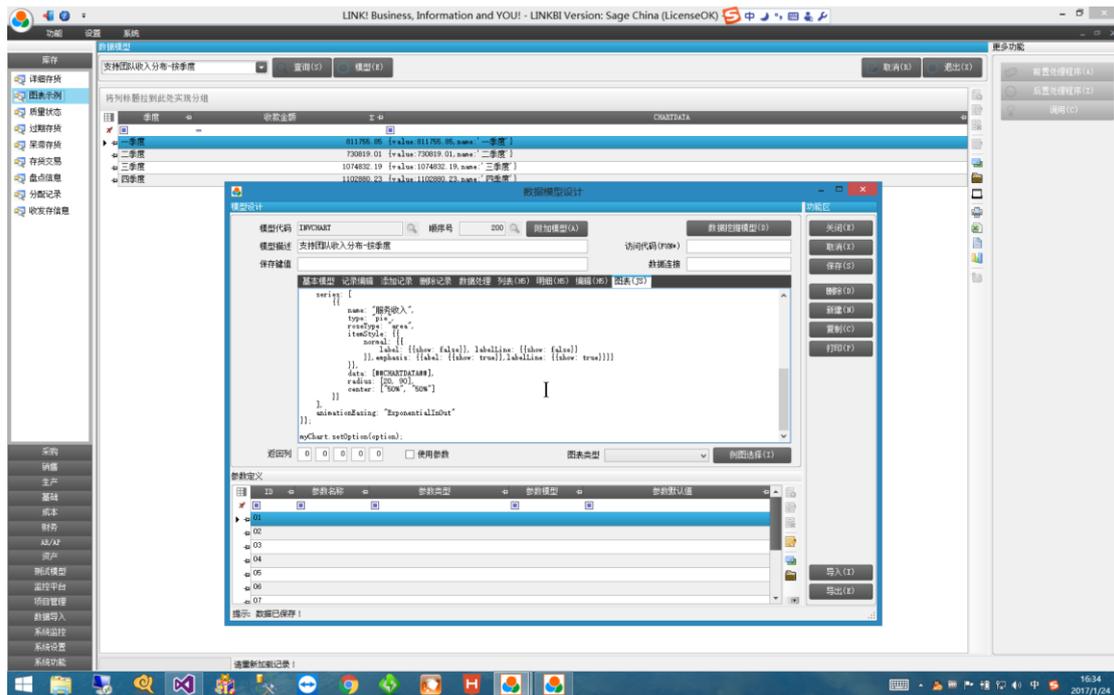
```
1 option =
2 {{ title: {{
3     text: "半径模式的南丁格尔玫瑰图", subtext: "纯属虚构", x: "center"
4     }},
5     tooltip: {{
6         trigger: "item", formatter: "{{a}} -br-{{b}} : {{c}} ({{d}}%)"
7     }},
8     legend: {{
9         x: "center", y: "bottom",
10        data: ["##季度##"]
11    }},
12    toolbox: {{
13        show: true,
14        feature: {{
15            mark: {{show: true}},
16            dataView: {{show: true,
17                readOnly: true}},
18            restore: {{show: true}},
19            saveAsImage: {{show: true}}}
20        }},
21    calculable: true,
22    series: [
23        {{
24            name: "服务收入",
25            type: "pie",
26            roseType: "area",
27            itemStyle: {{
28                normal: {{
29                    label: {{show: false}}, labelLine: {{show: false}}
30                }}, emphasis: {{abel: {{show: true}}, labelLine: {{show: true}}}
31            }},
32            data: ["##CHARTDATA##"],
33            radius: [20, 90],
34            center: ["50%", "50%"]
35        }}
36    ],
37    animationEasing: "ExponentialInOut"
38 }};
39
40 myChart.setOption(option);
```

- 实现数据的动态取值，是用图表展现业务数据的关键
- 在数据模型定义中，已经实现了数据从业务系统中的获取方法，在此处只需要将数据列用特殊的代换方式替代“静态的示例数据”即可
- 在这个例子中
 - 用“[“##季度##”]”，替代“[“一季度”、“二季度”、“三季度”、“四季度”]”
 - 用“[##CHARTDATA##]”，替代原来示例数据中的各行的静态数据
- 这样，在系统中使用到图表的时候，就自动加载了从业务系统中获取的数据
- 说明：
 - 如果模型中使用到模型的“参数”，只需要在脚本中保留类似“{0}”即可，系统执行时，可以实现自动参数“值”的替换

14 模型脚本填写



- 完成上述各个步骤后，在 TotalLINK 系统中实现图表的设计就非常简单了
- 只需要将上述模型脚本复制，粘帖到模型设计的“图表(JS)”一栏即可
- 完成后的模型如图所示



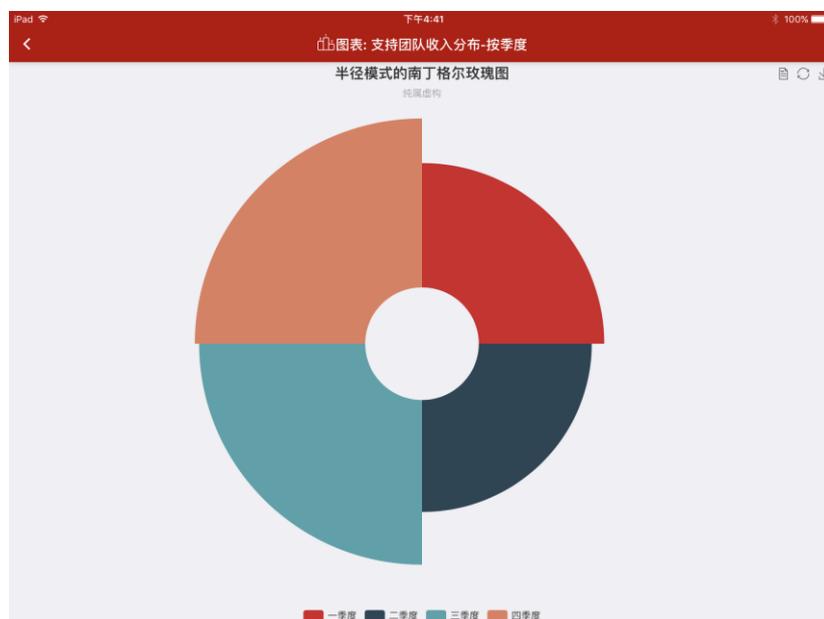
15 展现结果



- 如图所示，在移动设备查看模型，即可得到上面的结果
- 本例的模型，显示的尺寸比较小，下面我们再通过模型脚本修改使得显示效果更佳合理

- 实现的结果与手工更改脚本相同

17 重新查看展现结果

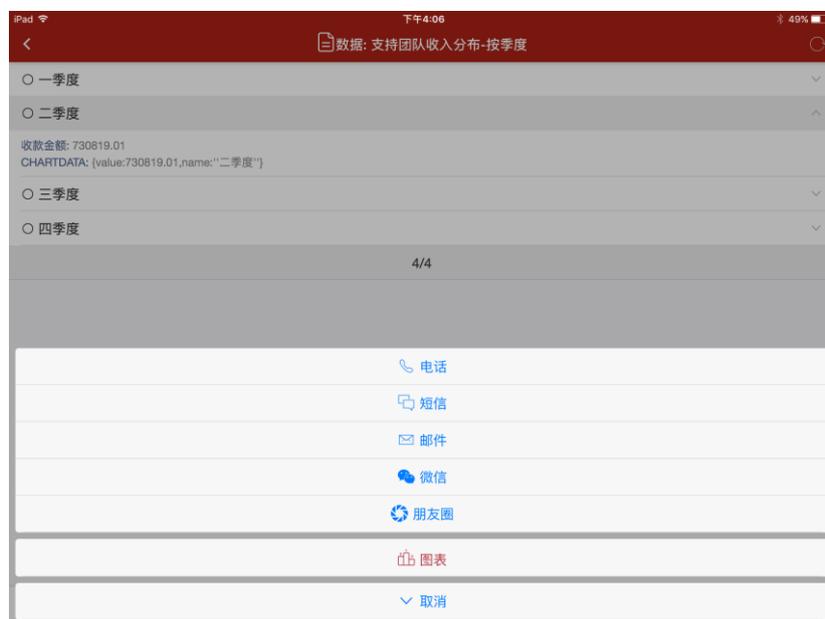


- 调整后，重新刷新图表，可以看到显示结果如图所示，符合设计要求

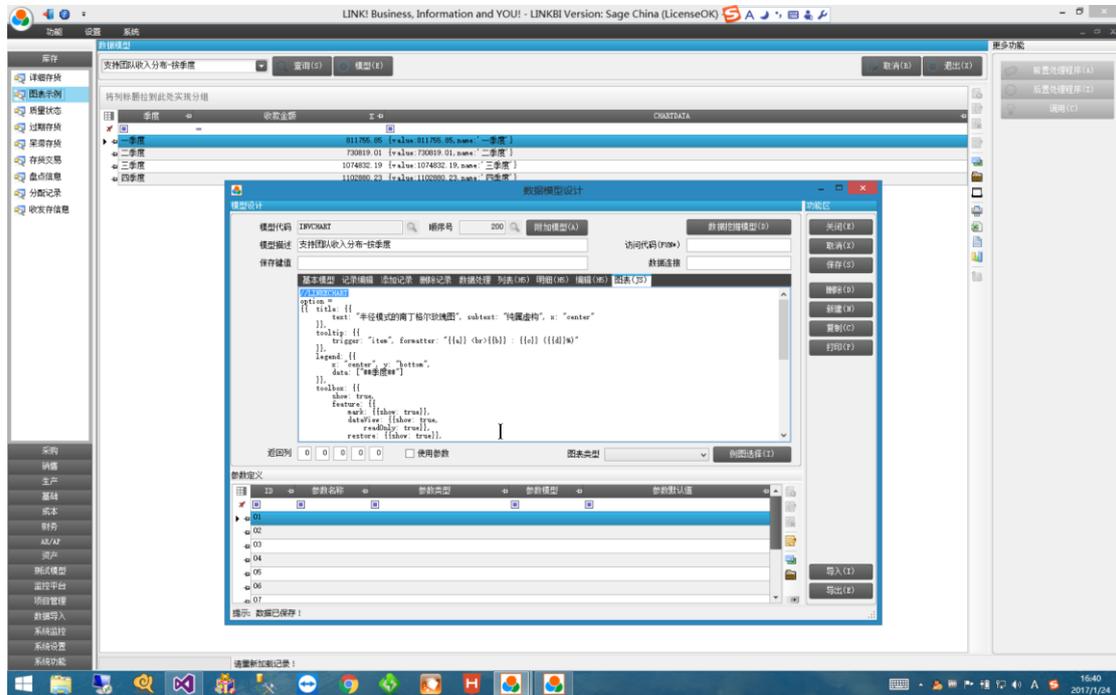
18 查看图表



- 通过上述设计，在移动端的菜单中，还是先加载数据，而后再根据数据加载图表的
- 如上图中的“支持团队收入分布-按季度”，这是图表展现的一种方式



19 直接展示图表

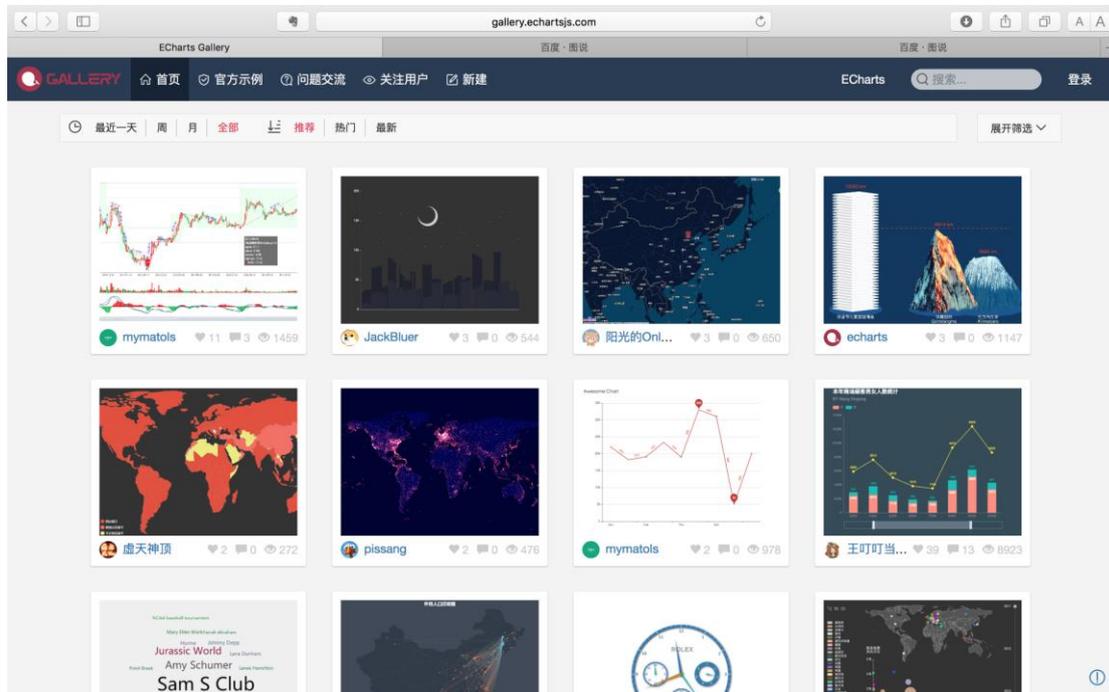


- 为了实现图表的直接展示，在图表的脚本的第一行加上“//LINKECHART”即可
- 移动端的显示效果如图所示



- 此时，直接通过点击菜单项，可以直接获得图表
- 如果该模型作为“数据挖掘”的子模型，也可以直接挖掘得到“图表”

20 高级图表设计



- 更多内容，请参考百度“echarts”实例及教程
 - <http://echarts.baidu.com/examples.html>
 - <http://gallery.echartsjs.com/explore.html#sort=rank~timeframe=all~author=all>
 - <http://echarts.baidu.com/tutorial.html#ECharts%20特性介绍>
 - <http://echarts.baidu.com/option.html#title>