

Table of Contents

Dashboard介绍	1.1
Dashboard能做什么	1.2
自定义图表	1.3
自定义图表介绍	1.3.1
自定义图表能做什么	1.3.2
如何开发一个图表	1.3.3
案例：水球图	1.3.4
案例代码	1.3.5

- [1.1.1. Dashboard介绍](#)
- [1.1.2. Dashboard能做什么](#)
- [1.1.3. 自定义图表](#)
 - [1.1.3.1. 自定义图表介绍](#)
 - [1.1.3.2. 自定义图表能做什么](#)
 - [1.1.3.3. 如何开发一个图表](#)
 - [1.1.3.4. 案例：水球图](#)
 - [1.1.3.5. 案例代码](#)

1.1.1. Dashboard介绍

dashboard 是商业智能仪表盘的简称，它是一般商业智能都拥有的实现数据可视化的模块，是向企业展示度量信息和关键业务指标（KPI）的数据虚拟化工具。dashboard 以丰富的，可交互的可视化界面为数据提供更好的使用体验。

1.1.2. Dashboard能做什么

本产品是自由可拖拽式的 Dashboard，采用栅格系统的自由可视化布局，内置了几十种可视化图表组件，支持在线和离线的 GIS 地图应用，图标之间支持相互交互联动，支持图标的自由定制接入，数据的动态刷新，主题的自由切换，且支持 skd 的二次开发。

所以本产品具备统计数据和大数据可视化能力、大屏可视化能力、实时数据可视化能力。在阿坝项目、新疆项目的高危人群预警和画像、食药监、五粮液等项目中均有使用。

1.1.3. 自定义图表

1.1.3.1. 自定义图表介绍

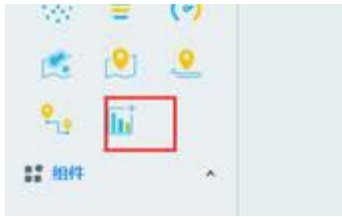
自定义图表为自己代码实现方式的组件，相当于 dashboard 的扩展接口组件，可实现各种图标，如水球图，折线图饼图的混合图等，并且提供接入数据功能，可与其他组件形成联动与被联动关系。

1.1.3.2. 自定义图表能做什么

当仪表盘内置图表组件或者内置组件的展示方式不符合需求的时候，可根据需要按照开发手册写出自己想要的图表，并且实现想要的图表逻辑和展示方式。

1.1.3.3. 如何开发一个图表

1、打开 dashboard 界面，拖拽左边自定义图表组件进入中间画布区域。



- 2、选中画布区域的自定义组件。
- 3、在组件左边根据需要配置数据源(可以不配置)
- 4、定义数据源，并且点击获取数据按钮，(ps:缓存数据查询)。如不需要绑定数据，可不用填写。

数据源

样式

来自于

自定义SQL查询

数据库连接

dw_market

SQL查询

```
SELECT '食品经营' 食品大类, 1000012 全
额,'349212,13423,129843,22541' 数据 UNION
ALL
      SELECT '餐饮服
务',1000012,'349212,13423,129843,22541'
UNION ALL
      SELECT '食品流
通',2000012,'349212,13423,129843,22541'
UNION ALL
      SELECT '食品生
```

参数查询

名称	值
qwe	参数值

获取数据

- 5、切换到样式选项，查看或者编写自定义代码。下面是一个自定义折线图的代码解析

```

//data 是 获取的数据
var data = this.cfg.chartDefinition.data;
if(!data || !data.resultset){
    return false;
}
//dashboard 内置echarts，用按如下方式使用；
var echarts = arguments[0];
//如需使用ds 请先加载，然后再回调中写逻辑
//注意： 绘图引擎切换 请先手动清除之前的数据，否则造成多个引擎渲染的图表存在
//清除方式，可以根据引擎的方式清除，如echarts，可以调用echartsInstance.dispose()
Dashboard.queryAction.getScriptOnce([
    name: 'ds',
    src: 'https://d3js.org/d3.v5.min.js'
]).then(function(){
    console.log('succ');
}).catch(function(e){console.log(e)});
var option={xAxis:{type:'category',data:['Mon','Tue','Wed','Thu','Fri','Sat','Sun']},yAxis:{type:'value'},series:[{data:[820,932,901,934,
//重置echarts实例
var echartsInstance = this.echartsInstance || echarts.init(this.htmlObj);
echartsInstance.dispose();
echartsInstance = echarts.init(this.htmlObj);

echartsInstance.setOption(option);
//echartsInstance.resize();
//刷新方法 在函数里面写自己的刷新逻辑 也就是重新渲染
this.resize = function(){
    echartsInstance.clear();
    echartsInstance.resize();
    echartsInstance.setOption(option, true);
}
}

```

获取查询的数据

获取echarts插件

引入插件，当加载完成时调用then方法，加载的插件为全局插件

设置echarts所需的option数据

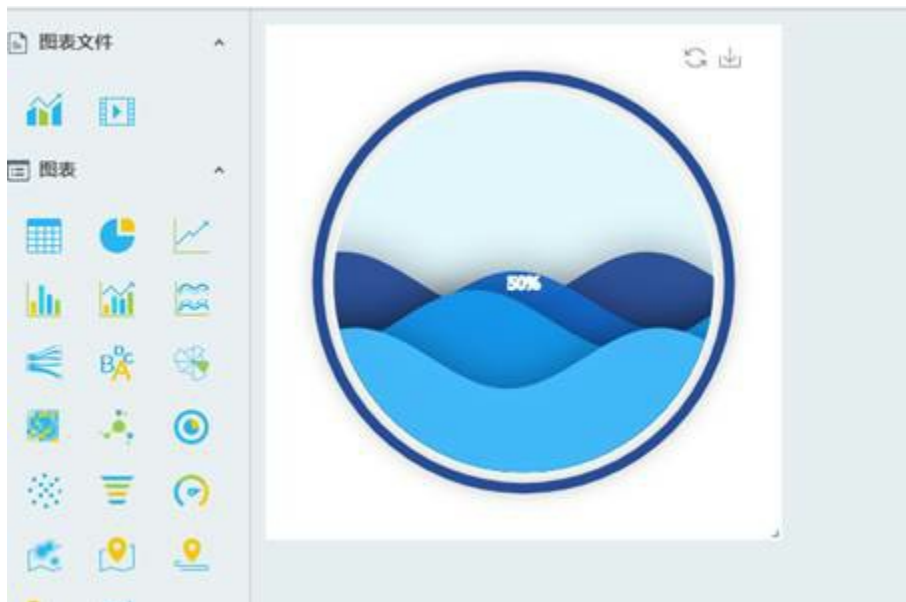
重置echarts实例

渲染echarts图表

当窗口变化或者移动的时候重新渲染图表

6、点击立即执行（重要），此时代码运行并且代码保存到数据中；

1.1.3.4. 案例：水球图



在当前选中的自定义图表的自定义代码中写入

a) 如需获取数据，则加入获取数据代码

```

//data 是获取的数据
var data this.cfg.chartDefinition.data

```

```
if data    data resultset
return false
```

b) 获取当前的 echarts 插件并且设置为全局属性(因为引入的水球图插件需要使用全局的 echarts 对象)

```
//dashboard内置echarts，用按如下方式使用：
var echarts    arguments 0
window echarts    echarts
```

c) 引入水球图的插件，并保存在变量 one 中

```
//引入水球图插件
var one    Dashboard queryAction getScriptOne
    name "liquidfill"
    src 'http://echarts.baidu.com/resource/echarts-liquidfill-1.0.4/dist/echarts-liquidfill.js'
```

d) 重置 echarts 实例

```
var echartsInstance    this echartsInstance    echarts init this htmlObj
echartsInstance dispose
echartsInstance    echarts init this htmlObj
```

e) 定义 echarts 所需 option

```
var option
    series
        type 'liquidFill'
        radius '80%'
        data 0.5 0.45 0.4 0.3
        label
            normal
                textStyle
                    color 'red'
                    insideColor 'yellow'
                    fontSize 50
```

f) 当插件加载完成时就渲染图表

```
one then function
    echartsInstance setOption option
```

g) 编写 resize 代码

```
this resize function
    echartsInstance clear
    echartsInstance resize
    echartsInstance setOption option
```

1.1.3.5. 案例代码

```
//data 是 获取的数据

var data  this cfg chartDefinition data

if data data resultset

    return false


//dashboard 内置echarts ， 用按如下方式使用：

var echarts arguments 0

window echarts echarts

//引入水球图插件

var one Dashboard queryAction getScriptOnce

    name 'liquidfill'

    src 'http://echarts.baidu.com/resource/echarts-liquidfill-1.0.4/dist/echarts-liquidfill.js'

var option

    series

        type 'liquidFill'

        radius '80%'
```

```

        data 0.5 0.45 0.4 0.3

        label

            normal

                textStyle

                    color 'red'

                    insideColor 'yellow'

                    fontSize 50

    })

    // 实例化echarts
    var echartsInstance = this.echartsInstance = echarts.init(this.htmlObj)

    echartsInstance.dispose()

    echartsInstance = echarts.init(this.htmlObj)

    this.resize = function() {

        echartsInstance.clear()

        echartsInstance.resize()

        echartsInstance.setOption(option)

    }

    one.then(function() {

        echartsInstance.setOption(option)
    })

```

2019-01-25 16:17:05