

# 图形扩展

## 说明

smartbi中的图形对象是独立组件，在透视分析、组合分析、多维分析以及未来的电子表格中都有使用。

V6版本集成了Echarts图形，产品本身做了一些常用的图形化配置界面，如果用户对图形有更多要求，可以通过自定义属性或宏代码修改图形样式。

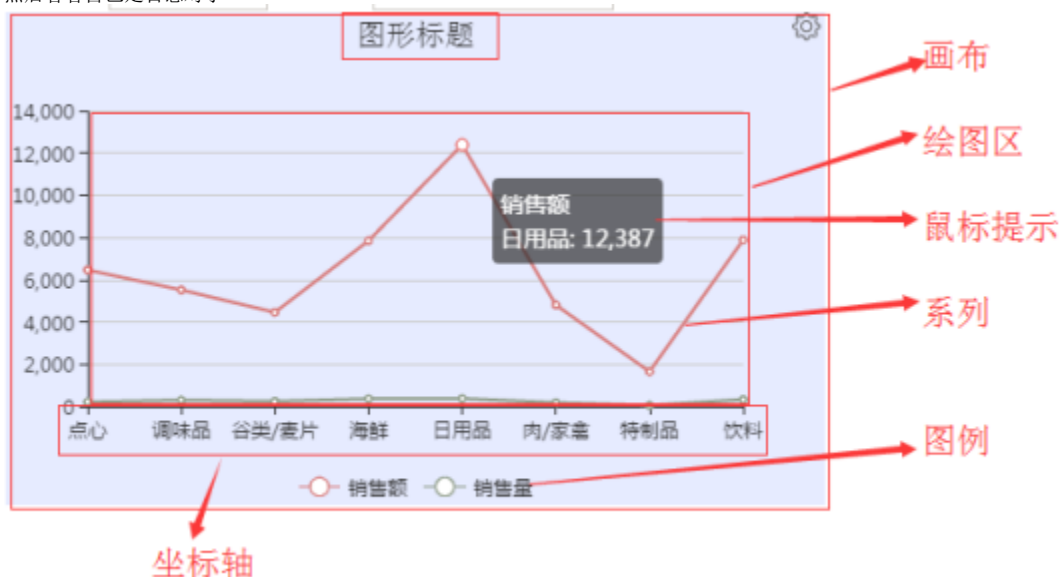
- 1、理解图形构成
- 2、扩展图形之：使用扩展属性
  - 何时使用及使用前提
  - 示例说明：
  - 使用技巧
- 3、扩展图形之：使用宏代码

## 1、理解图形构成

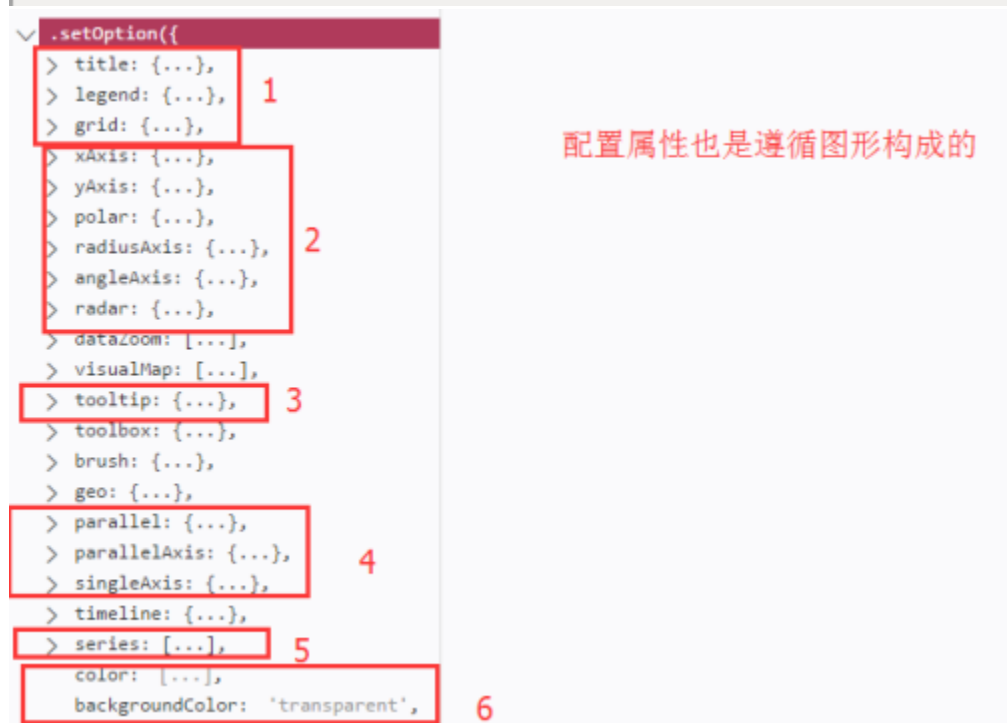
想象一下以下组成部分对应图形外观



然后看看自己是否想对了？



下面对比下Smartbi的图形配置界面和Echarts的图形配置项



## 2、扩展图形之：使用扩展属性

### 何时使用及使用前提

有了以上理解之后，什么情况下才会使用扩展属性扩展图形呢？



- 1、设置界面没有，echarts有
- 2、快速制作静态数据poc（使用覆盖模式）

使用扩展属性需要的前提条件，如果不了解，是无法做扩展的。

- 1、需要理解json对象、数组的概念及语法
- 2、掌握js基本数据类型语法（字符串、数字、逻辑值、null）

技巧:

### 使用技巧

使用扩展属性建议充分利用echarts在线实例（可编辑运行效果）

### 示例说明:

更多示例请参考: [ECharts 配置文档](#)及子目录下的示例

红色框中对应echarts实例的option = {...}对象的内容



想象一下POC需要下面图，您会怎么做？



#### 不满足点

- 多了个visualMap 是视觉映射组件
- Y轴刻度翻转
- X轴标线显示在0上，并且不显示轴线

所以，基本可以通过下面扩展属性调整

```
{
  "visualMap": {
    "calculable": true
    //.....
  },
  "xAxis": {
    "axisLine": {
      "onZero": true
    },
    "splitArea": {
      "show": false
    },
    "splitLine": {
      "show": false
    }
  },
  "yAxis": {
    "inverse": true
  }
}
```

## 使用技巧

对于初级使用者：

- 第一步，在echarts实例页面中找到类似效果的图。
- 第二步，去掉多余的属性，找关键属性。
- 第三步，查看属性相对option的位置。
- 第四步，将属性复制到smartbi中查看效果

对于高级人员，对Echarts本身比较了解的。

- 第一步，确定属性所属图形的元素。
- 第二步，在echarts配置属性中找可能的属性，确定该属性相对与option的位置。（要求对图形构成熟悉）
- 第三步，在echarts对于图形实例中测试属性，无效则重新进行第二步。
- 第四步，测试有效以后，拿到smartbi扩展属性中看效果

## 3、扩展图形之：使用宏代码

当用到动态效果的时候，使用扩展属性也难以达到效果，此时我们需要考虑使用“宏”  
宏的使用请参考：[图形对象\(chart\)及事件说明](#)