

ALIBABA CLOUD

阿里云

DataV数据可视化
开发指南

文档版本：20220211

 阿里云

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
 危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 危险 重置操作将丢失用户配置数据。
 警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	 警告 重启操作将导致业务中断，恢复业务时间约十分钟。
 注意	用于警示信息、补充说明等，是用户必须了解的内容。	 注意 权重设置为0，该服务器不会再接受新请求。
 说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击设置>网络>设置网络类型。
粗体	表示按键、菜单、页面名称等UI元素。	在结果确认页面，单击确定。
Courier字体	命令或代码。	执行 <code>cd /d C:/window</code> 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	<code>bae log list --instanceid</code> <i>Instance_ID</i>
[] 或者 [a b]	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
{ } 或者 {a b}	表示必选项，至多选择一个。	<code>switch {active stand}</code>

目录

1. 组件开发快速入门	06
2. 组件开发包文件介绍	12
2.1. 组件开发包文件概览	12
2.2. index.js规范	12
2.3. package.json规范	13
2.4. 控件 (config) 配置	20
2.4.1. 控件概览	20
2.4.2. 控件配置规范	43
2.4.3. showInPanel配置规则	46
2.4.4. valuePath配置规则	50
2.4.5. text	53
2.4.6. number	54
2.4.7. select	56
2.4.8. search	57
2.4.9. color	59
2.4.10. multicolor	60
2.4.11. array	62
2.4.12. hidden	64
2.4.13. boolean	65
2.4.14. radio	65
2.4.15. checkbox	67
2.4.16. buttonRadio	68
2.4.17. iconRadio	69
2.4.18. percent	71
2.4.19. image	72
2.4.20. imageSelect	73

2.4.21. switch	75
2.4.22. stepper	76
2.4.23. slider	77
2.4.24. keyBoard	79
2.4.25. fill	79
2.4.26. suite	86
2.4.27. margin	89
2.4.28. padding	91
2.4.29. line	92
2.4.30. font	94
2.4.31. group	97
2.4.32. tabs	99
2.4.33. menu	103
3.ECharts组件封装指南	107
4.自定义组件开发常见问题	132
5.图形界面开发工具使用说明	134

1. 组件开发快速入门

本文介绍使用DataV开发工具完成组件开发的方法。

环境准备

1. 进入[NodeJS官网](#)，下载并安装NodeJS，推荐Node版本在 8.0.0 及以上，10.12.0以下。

 **说明** 如果您的系统已经安装过Node，并且还需要持续使用，推荐安装nvm来进行Node版本管理。使用nvm，您可以不用卸载之前的Node，安装现版本Node时不会造成冲突，安装地址：[GitHub - creationix/nvm: Node Version Manager - Simple bash script to manage multiple active node.js versions](#)。

2. 安装成功后，在命令行操作界面执行 `node -v` 和 `npm -v` 命令（Mac在terminal中执行，Windows在cmd中执行），查看Node和npm版本。

安装开发工具套件

1. 执行以下命令安装套件（Mac在terminal中执行，Windows在cmd中执行）。

```
npm install --registry=https://registry.npm.taobao.org datav-cli -g
```

2. 安装成功后，执行 `datav --version` 命令，查看开发工具版本。

设置语言环境

DataV开发工具默认的语言环境是 `English`，如果您需要使用其他语言环境，可以在命令行界面执行 `datav locale` 命令切换语言环境。

执行成功后，返回以下结果。

```
? What is your language? (Use arrow keys)
> English
  Chinese
  Japanese
```

按下键盘上的`↑`和`↓`键，并使用`Enter`键确认后，即可切换语言环境。

切换后，如果您想根据账户所在的阿里云区域，自动输入命令切换语言环境，可以在登录后执行 `datav locale-clear | datav lc` 命令，清空默认设置的语言环境。清空后，您也可以再次设置语言环境。

用户登录

通过 `datav login` 命令登录DataV开发者控制台（Mac在terminal中执行，Windows在cmd中执行），执行后，需要输入以下信息。

登录信息	说明
<code>用户名</code>	输入您在DataV控制台首页右上角的名字，如果是子账号，请注意使用主账号的名字。
<code>开发者识别码</code>	输入您在DataV控制台我的组件页面复制得到的开发者识别码。

登录信息	说明
需要设置别名吗?	输入 Y 或 n。
别名	输入您想要设置的别名。

当命令行中显示**设置成功**时，说明您已经登录成功。

说明 登录不是必须的，如果您只想使用创建组件和预览组件功能，可以不进行登录操作。只有发布功能需要登录。

生成组件包

通过 `datav init` 命令生成组件包（Mac在terminal中执行，Windows在cmd中执行），执行后，需要输入以下组件信息。

组件信息	说明
? 你要创建的组件名(字母, -, 数字)是:	组件名, 只能由字母、数字和短横线 (-) 组成。
? 你要创建的组件显示名是:	显示名是在可视化应用的组件列表中显示的组件名。
? 你的组件描述是:	输入对组件的描述。
? 请从组件模板中选择你要创建的组件 (Use arrow keys) 基本文字 基本文字 (国际版)	组件使用的模板, 包括 基本文字 和 基本文字 (国际版)。 按下键盘上的 ↑ 和 ↓ 键, 并使用 Enter 键确认后, 即可选择组件模板。

当命令行显示**组件创建完毕**时，说明您的组件包已经成功生成。



预览组件

您可以通过以下命令预览组件（Mac在terminal中执行，Windows在cmd中执行）。

```
cd 您的组件名
datav run
```

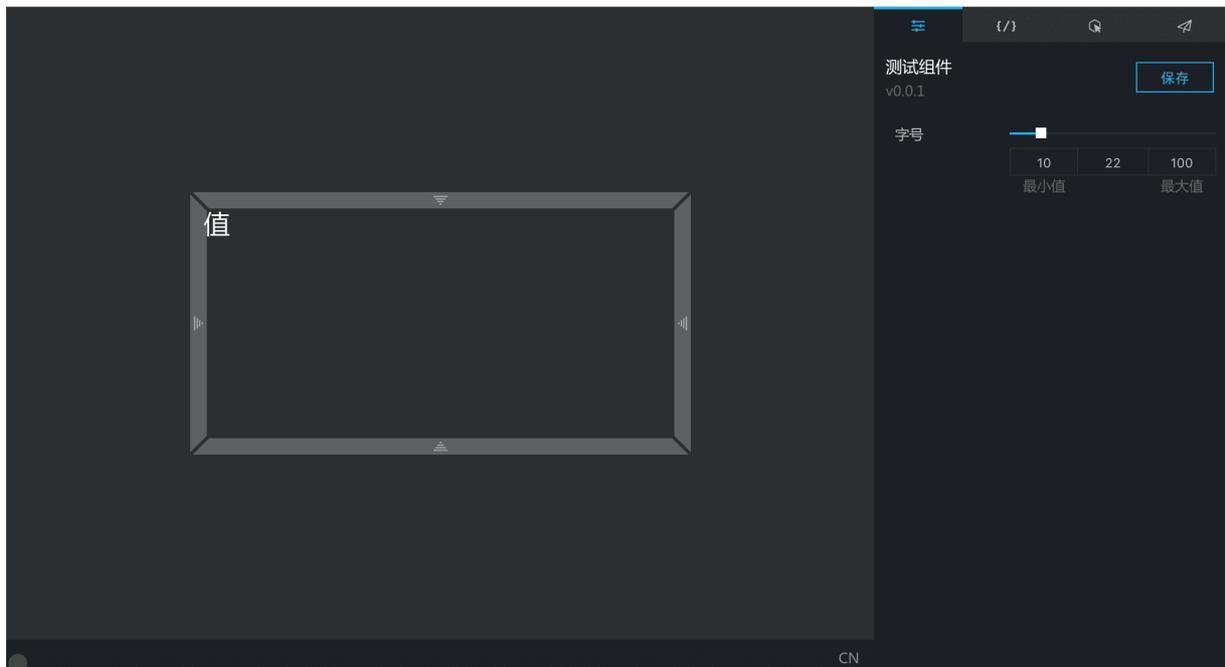
当命令行显示服务启动时，说明预览组件的服务已经启动，Chrome浏览器会被自动打开，并导航到组件预览页。



说明

- 如果浏览器没有自动打开，可能是您未安装Chrome浏览器。建议安装Chrome，完成后用Chrome浏览器手动打开localhost:1111/。
- 如果您看到端口占用冲突，可能是您的1111端口被别的应用程序占用了，可以使用 `datav run -p 1112` 命令来指定使用1112端口开启预览服务。
- 如果浏览器打开localhost:1111/，显示并没有此服务，可能是您的电脑并未配置hosts localhost指向127.0.0.1，可以在Chrome浏览器中访问127.0.0.1:1111/来预览组件。

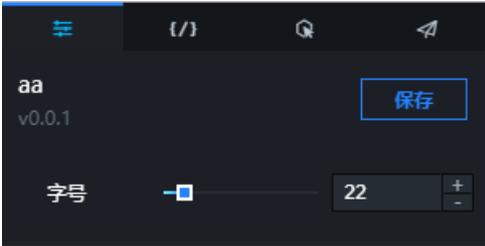
组件预览成功后的效果如下图所示。



预览页主要分为中心画布区，底部工具栏和右侧工具栏三部分，详细介绍如下：

- 中心画布区
 - 中心画布区是用来展现组件，实时观测组件变化的区域。

- 所有右侧工具栏的配置、数据修改都会实时展示在中心画布的组件上。
- 组件的白框代表了组件的容器范围大小，每个方向上的白框都可以缩放来测试组件在任意方向缩放的表现。
- 底部工具栏
可以切换组件的语言环境。
- 右侧工具栏
右侧工具栏分为配置、数据、交互和发布4个面板。

面板	说明
配置	<p>配置页面描述了组件可变动的一些配置项，如果在配置操作，改动会立即生效。例如在此页面拖拽字号滑动条，组件中的文字字号立即随之变化。单击右上的保存，可以保存当前修改的配置，并将当前配置作为此组件的默认配置。</p>  <p>The screenshot shows a configuration panel for a component. At the top, there are icons for menu, code, search, and zoom. Below that, the component name 'aa' and version 'v0.0.1' are displayed. A '保存' (Save) button is on the right. A '字号' (Font Size) slider is set to 22, with '+' and '-' buttons on either side.</p>
数据	<p>数据页面描述了组件的数据接口配置，数据页的数据一旦改动，组件都会进行相应的改动。单击右上的保存，可以保存当前修改的数据，并将当前数据作为此组件的默认数据。</p>  <p>The screenshot shows a data panel for a component. At the top, there are icons for menu, code, search, and zoom. Below that, the component name 'aa' and version 'v0.0.1' are displayed. A '保存' (Save) button is on the right. The panel is titled '接口描述' (Interface Description) and has a '配置完成' (Configuration Complete) indicator. Below this is a table with columns '字段' (Field), '描述' (Description), and '状态' (Status). The table has one row: 'value' '值说明' (Value Description) '匹配成功' (Match Success). Below the table is a code editor showing a JSON object: { "value": "值" }.</p>

面板	说明
交互	<p>交互页面描述了组件的交互说明。</p> 
发布	<p>发布页面描述了组件的类型、图标和发布版本等配置，单击右上角的发布，即可发布组件。</p> 

发布组件

您可以通过以下三种方式发布组件：

- 方式一（推荐）

进入组件的目录地址下，执行 `datav publish` 命令，组件将自动打包压缩发布至账号所在域的服务器。

- 方式二

进入组件的目录地址下，执行`dat av package`命令，在组件目录外会有一个以 `组件-版本号` 命名的 `tar.gz`压缩包，将此压缩包上传到`dat av.aliyun.com`的组件页，即可发布。

- 方法三

进入预览组件页面下的发布页面，单击发布，即可发布组件。

2. 组件开发包文件介绍

2.1. 组件开发包文件概览

本文介绍开发者在开发Dat aV组件时，需要遵循的文件结构。

在开发组件包之前，您需要新建一个文件夹，作为组件包。建议您使用组件名称，命名组件包。组件包中需要包含以下文件。

文档名	描述
index.js	组件主入口
package.json	组件配置
readme.md	组件描述
history.md	组件更新历史描述

2.2. index.js规范

index.js文件是组件的主入口文件。本文档提供了一个示例供您参考，并介绍了index.js文件中的常用函数。

查看模板案例[example组件](#)。

常用函数

函数	说明
<code>render(array: data)</code>	默认渲染方法。当组件被初始化后，组件渲染逻辑被调用，入参为数据。需要能够支持重渲染，即保证同样的数据输入，能够渲染出同样的效果。
<code>updateOptions(object: config)</code>	更新组件配置，接收的入参 <code>config</code> 是修改后的新配置。
<code>emit(string: event_name, object: value)</code>	抛出事件，抛出的 <code>value</code> 需要为object类型。此方法需要在package.json的events中声明后，节点编程中才会出现此事件。也可以在交互面板中使用。
<code>resize(int: width, int: height)</code>	缩放，当组件被拖拽、缩放时被调用。
<code>clear()</code>	清理，当组件被清理时被调用。
<code>destroy()</code>	销毁，当组件被销毁时调用。
<code>require(*)</code>	支持 js、html。

函数	说明
<code>getThemableConfig(themeConfig)</code>	设置主题配置，当配置组件的主题时被调用，返回主题与组件配置的映射关系，同时使用 <code>themeConfig</code> 中提供的主题配置项进行主题配置。调用前需要在组件配置中，使用 <code>supportTheme</code> 字段声明组件是否支持主题配置，设置为 <code>true</code> 表示支持，更多详情请参见 getThemableConfig(themeConfig)函数说明 。

getThemableConfig(themeConfig)函数说明

`getThemableConfig(themeConfig)` 函数的作用是使用 `themeConfig` 中的配置，配置组件主题，并返回主题与组件配置的映射关系。

`getThemableConfig(themeConfig)` 函数调用示例如下。

```
getThemableConfig: function(themeConfig) {
  if (!themeConfig) return null;
  const themeMap = {
    color: themeConfig.textColor,
    bgColor: palette[0]
  }
  return themeMap;
}
```

`themeConfig` 示例如下。

```
{
  "bgColor": "#202020",      // 背景颜色
  "textColor": "#FFFFFF",   // 文本颜色
  "axisColor": "#FFFFFF",   // 坐标轴颜色
  "assistColor": "#FFFFFF", // 辅助信息颜色
  "palette": [              // 系列颜色
    "#85A5FF",
    "#597EF7",
    "#2F54EB",
    "#1D39C4",
    "#10239E",
    "#061178",
    "#030852"
  ]
}
```

2.3. package.json规范

package.json文件是组件的配置文件。本文档介绍package.json文件的字段详情，您可以参考本文档的字段说明，根据自身需求，灵活修改package.json文件，自定义组件样式。

协议版本2规范示例

 说明 请着重查看下述案例中的备注信息。

```

{
  "name": "@leaf/pie-multi-radius-with-title", //命名空间/组件名（命名空间即组件包名，必须添加，否则无法发布）。
  "version": "0.1.5", //版本号。
  "dependencies": { //依赖，不包括datav:/com/依赖。
    "bcore": "0.0.9",
    "jquery": "2.1.4", //所有依赖的版本号必须锁死。
    "lodash": "4.6.1"
  },
  "datav": { //datav配置。
    "cn_name": "带标题的多维饼图", //组件中文名。
    "protocol": 2, //协议版本号。
    "type": ["regular_pie"], //组件类型。
    "supportTheme": true, //组件声明是否支持主题。
    "view": {
      "width": 300,
      "height": 200,
      "minWidth": 100,
      "minHeight": 50
    },
    "icon": "", //组件图标链接地址。
    "apis": { //组件接口，可以多个。
      "source": { //接口名。
        "handler": "render", //处理接口返回的组件方法名。
        "description": "多维度饼图接口", //接口描述。
        "fields": { //接口所需字段，可以多个。
          "x": { //字段名。
            "description": "类目", //字段描述。
            "type": "string", //字段类型。
            "optional": true //可选字段。
          },
          "y": {
            "description": "值",
            "type": "int"
          }
        }
      }
    },
    "config": { //组件配置，给编辑器识别用。
      "paxis": { //类型：组，详见<组件config配置说明>。
        "type": "group",
        "name": "标签",
        "children": {
          "dx": { //类型：文本，详见<组件config配置说明>。
            "type": "text",
            "name": "标签距中心",
            "default": 220
          }
        }
      }
    },
    "title": {
      "type": "group",
      "name": "标题",
      "children": {

```

```

    "value" : {
      "hasVisibility" : "true",
      "visible" : "true",
      "type" : "text",
      "name" : "标题名",
      "default" : "我是标题"
    },
    "font-size" : {
      "type" : "number", //类型：数字，详见<组件config配置说明>。
      "name" : "字体",
      "min" : 10,
      "default" : 32,
      "max" : 100
    },
    "text-align" : {
      "name" : "对齐方式",
      "type" : "select", //类型：下拉框，详见<组件config配置说明>。
      "options" : [
        {"name" : "左对齐", "value" : "left"},
        {"name" : "右对齐", "value" : "right"},
        {"name" : "居中对齐", "value" : "center"}
      ],
      "default" : "center"
    },
    "color" : {
      "name" : "字体颜色",
      "type" : "color", //类型：颜色，详见<组件config配置说明>。
      "default" : "#fff"
    },
    "background-color" : {
      "name" : "背景",
      "type" : "color",
      "default" : "#000"
    }
  }
}
},
"api_data":{ //接口数据，可以多个。
  "source" : [ //接口名，必须和apis中的接口名一致，限制6K。
    {"x": "普货", "y" : 5},
    {"x": "普货", "y" : 22},
    {"x": "泡货", "y" : 22},
    {"x": "设备", "y" : 14},
    {"x": "矿产", "y" : 15},
    {"x": "钢铁", "y" : 15},
    {"x": "建材", "y" : 12},
    {"x": "食品", "y" : 12},
    {"x": "粮食", "y" : 28}
  ],
},
"events": { //全局参数事件配置。
  "event-name": { //事件名。
    "description": "事件描述", //事件描述。
  }
}

```

```

    "fields": {
      "value": {
        "description": "值描述"
      }
    },
    "publicHandler": {
      "show": {
        "name": "显示",
        "description": "描述",
        "type": "object",
        "fields": {
          "data": {
            "name": "数据",
            "type": "array",
            "children": {
              ...
            }
          }
        }
      }
    }
  },
}

```

配置项config字段说明

`config` 配置用来说明组件有哪些配置。这些配置既用于传给组件，也用于说明编辑器的选项。

`config` 配置有以下两个要求：

- 包含组件默认配置。
- 描述清楚配置的输入规范，以便编辑器进行识别。

`config` 配置中的 `type` 字段用来配置组件的类型和样式。不同的 `type` 类型会展示不同的编辑器UI，合理使用 `type` 类型和配置项归类，可以极大提高组件的质量与用户体验，详情请参见[控件配置](#)。

apis字段与api_data字段

- `apis` 字段

该字段定义了组件接口名，以及该接口所需的字段。接口和字段均可为多个。

```

"apis": {
  "source": {
    "handler": "render",
    "description" : "多维度饼图接口",
    "fields" : {
      "x" : {
        "description" : "类目",
        "type" : "string",
        "optional": true
      },
      "y" : {
        "description" : "值",
        "type" : "int"
      }
    }
  },
  "source2":{
    ...
  }
}
    
```

字段名	含义	是否必选	备注
apis	组件接口	是	一个组件可以包含多个接口。
source	接口名	否	可自定义。
handler	处理接口返回的数据的方法名	是	该方法名必须在index.js源码中实现。
description	接口描述	否	无。
fields	接口所需字段	是	一个接口可以包含多个字段。
fields.x	字段名	是	可自定义。
fields.x.description	字段描述	否	无。
fields.x.type	字段类型	否	包括number、string、boolean、object和array。
fields.x.optional	是否为可选字段	否	为boolean类型。true表示对应字段为可选字段，false表示对应字段为必选字段。

• api_data 字段

该字段定义了接口数据，可以包含多个接口数据。接口名必须和 apis 中的接口名一致，限制为6KB（详情请参见注释）。

```

"api_data":{
  "source" : [
    {"x": "普货", "y" : 5},
    {"x": "普货", "y" : 22},
    {"x": "泡货", "y" : 22},
    {"x": "设备", "y" : 14},
    {"x": "矿产", "y" : 15},
    {"x": "钢铁", "y" : 15},
    {"x": "建材", "y" : 12},
    {"x": "食品", "y" : 12},
    {"x": "粮食", "y" : 28}
  ],
}

```

字段名	含义	是否必选	备注
api_data	组件接口数据	是	一个组件可以包含多个接口数据。
source	接口名	否	必须和 apis 字段中的接口名一致，限制 6KB。

events交互事件规范

DataV的交互事件由 emit（组件代码中触发 emit）和 events 描述（package.json的事件描述）组成。

- events 字段描述

events 字段在 datav 字段下，与 config 字段同级，定义了交互事件的事件名，描述以及变量名等。

```

{
  "events": {
    "click-me": {      "description": "点击触发事件",
      "fields": {
        "value": {
          "description": "点击值"
        }
      }
    }
  }
}

```

字段名	含义	是否必选	备注
click-me	事件名	是	可自定义。
description	事件描述	否	用于描述该事件的具体作用。
fields	字段	是	可包含多个字段名及其描述信息。
fields.value	字段名	是	可自定义。

字段名	含义	是否必选	备注
fields.value.description	字段描述	否	无。

- emit 触发

emit 方法作为一个基础类方法，会将事件名及需要的数据作为参数抛出，其他组件即可通过变量名获取到该组件的参数值，需要在index.js源码中实现（详情请参见注释）。

```
render: function () {
  this.container.on('click', () => {
    this.emit('click-me', data) // data必须为一个对象，而不是一个简单值，属性名即为变量名。
  })
}
```

type 字段

type 字段定义了该组件的组件类型（详情请参见注释）。

```
type: ["regular_bar", ...] //一个组件可以属于多个组，多个组件类型以下划线分隔: "type1_type2"。
{
  // 常用组件类型。
  regular: "常规图表",
  map: "地图",
  media : "媒体",
  network: "关系网络",
  text: "文字",
  interact: "交互",
  material: "素材"
}
```

publicHandler 字段

DataV的行为事件定义是通过 publicHandler 注册而成，示例如下。

```
"publicHandler": {
  "select": {
    "name": "选择",
    "description": "选择某一项",
    "type": "object",
    "fields": {
      "data": {
        "name": "数据",
        "type": "array",
        "children": {
          ...
        }
      }
    }
  }
}
```

如上示例注册了一个名称为 `select` 的行为事件，并且组件 `index.js` 中需要实现 `select` 方法，否则方法将报错。

注意

- 当您还没有实现 `show`、`hide` 以及 `updateOptions` 方法时，Dat aV会自动帮您注册并实现这3个默认方法。
- Dat aV为您注册的 `updateOptions` 需要您自行实现，如果没有实现，Dat aV会调用组件的 `render` 方法。

字段名	含义	是否必选	备注
<code>name</code>	事件名	是	在节点编程配置中使用。
<code>description</code>	事件描述	否	无严格的字数限制。
<code>type</code>	事件第一入参类型	否	可选，固定为object。
<code>fields</code>	字段	否	无。
<code>fields.data.name</code>	字段名	是	无。
<code>fields.data.description</code>	字段描述	否	无。
<code>fields.data.type</code>	字段类型	是	可选，包括number、string、boolean、object和array。
<code>fields.data.children</code>	子节点	否	当 <code>fields.data.type</code> 为object或array时需要设置children 字段。
<code>fields.data.default</code>	默认值	否	当 <code>fields.data.type</code> 为number、string或boolean时，需要填写默认值。
<code>fields.data.optional</code>	是否可选	否	可选，为boolean类型。 <code>true</code> 表示对应字段为可选字段， <code>false</code> 表示对应字段为必选字段。

2.4. 控件（config）配置

2.4.1. 控件概览

本文介绍Dat aV支持的控件（组件配置项）。您可以通过package.json文件中定义的 `type` 字段，来定义控件的类型和配置。

类型概览

`type` 字段支持以下类型。

类别	类型	说明
基础控件	text	文本输入框，支持定制前后缀。
	number	数值输入框。
	select	选择器。支持开启过滤和自定义输入。当需要设置字体时，可开启渲染字体功能，预览字体样式。
	search	下拉搜索框。
	color	颜色选择器。
	multicolor	渐变颜色选择器。可选单色或渐变色。
	array	数据系列。适用于多个系列配置或多个颜色循环配色。
	hidden	隐藏配置项。Dat aV不会渲染此配置。
	boolean	布尔配置项。
	radio	单选框。
	checkbox	复选框。支持为选项设置宽度。
	buttonRadio	单选按钮。
	iconRadio	图标单选器。
	percent	百分比值。
	image	图片框。
	imageSelect	装饰元素。
	switch	开关。支持选择是否展示文字状态。
	stepper	步进器。支持定制步长、最大值和最小值以及前缀和后缀。
	slider	滑动条。支持单滑块和双滑块，同时支持定制步长、最大值和最小值、前缀和后缀、精度，可开启范围显示。
	keyBoard	热键定义器。
fill	填充框。支持三种填充类型以及它们的组合。	
高频套件	margin和padding	边距高频套件。由四个步进器配置项组成，可定制上下左右四个边距值。
	line	线样式高频套件。由线粗细步进器、平滑方式图标单选器、样式选择器和纯色填充框组成。
	font	字体高频套件。由字体选择器、字体粗细选择器、字体大小步进器和纯色填充框组成。

类别	类型	说明
组合控件	suite	组合套件。由一组控件组成。
	tabs	标签组。切换标签可展示当前标签下的内容，支持动态增减标签组。
	menu	菜单。用来组织配置项的结构，目前支持一级菜单和二级菜单。
	group	配置项组。可包含多个子配置项（建议多个同类元素配置项归于一个group），可用来组织内容，并支持折叠或展开组的内容。

 **注意** 请勿使用关键字作为配置的key值，例如 `default`、`show` 和 `range`。

配置示例

通过以下示例为组件添加配置项，并使用该配置，通过DataV GUI渲染得到配置面板。

图表 坐标轴 系列 其它

x轴 y轴

x轴可见

两边留白 0.6 + -

分割间隔 0.6 + -

轴标签 轴标签

文本 微软雅黑 字体 normal 文字粗细

12 px 字号 rgb(14) 颜色

轴标签展示 - / | 0 数量

轴单位

轴线

网格线

```
{  
  "options": {  
    "type": "menu",  
    "children": [  

```

```
    "children": {  
      "chart": {  
        "name": "图表",  
        "children": {  
          "margin": {  
            "type": "margin",  
            "name": "边距",  
            "default": {  
              "top": 20,  
              "bottom": 30,  
              "left": 35,  
              "right": 15  
            }  
          },  
          "loadAmount": {  
            "name": "最多加载",  
            "description": "为保证大屏展示效果此组件开启数据加载限制，最多加载前指定条数据记录进行计算、布局与绘制",  
            "type": "stepper",  
            "default": 2000,  
            "step": 1  
          },  
          "bar": {  
            "name": "柱图样式",  
            "type": "group",  
            "children": {  
              "bgColor": {  
                "name": "背景颜色",  
                "type": "fill",  
                "default": "RGBA(255,255,255,0.1)"  
              }  
            }  
          },  
          "numericalLabel": {  
            "name": "值标签",  
            "type": "group",  
            "children": {  
              "show": {  
                "name": "显示/隐藏",  
                "type": "boolean",  
                "default": false  
              },  
              "textStyle": {  
                "type": "font",  
                "name": "文本",  
                "default": {  
                  "fontFamily": "Microsoft Yahei",  
                  "fontWeight": "normal",  
                  "fontSize": 10,  
                  "color": "#000"  
                }  
              },  
              "pos": {  
                "name": "位置",  
                "type": "iconRadio".
```

```
      type: "button",
      "default": "center",
      "options": [
        {
          "label": "顶部",
          "value": "top",
          "src": "top-center-pos"
        },
        {
          "label": "中间",
          "value": "center",
          "src": "middle-center-pos"
        },
        {
          "label": "底部",
          "value": "bottom",
          "src": "bottom-center-pos"
        }
      ]
    },
    "emptyData": {
      "name": "空值数据",
      "type": "switch",
      "default": false
    }
  },
  "enableHide": true
}
},
"type": "menuChild",
"mode": "single"
},
"axis": {
  "name": "坐标轴",
  "children": {
    "xaxis": {
      "name": "x轴",
      "children": {
        "isShow": {
          "type": "switch",
          "name": "x轴可见",
          "default": true
        },
        "boundaryGap": {
          "type": "slider",
          "name": "两边留白",
          "default": 0.6,
          "step": 0.01,
          "min": 0,
          "max": 1,
          "precision": 2,
          "showInPanel": {
            "conditions": [
              [
                ".isShow",
```

```
        "$eq",
        true
      ]
    ],
    "logicalType": "$and"
  }
},
"interval": {
  "type": "slider",
  "name": "分割间隔",
  "default": 0.6,
  "step": 0.01,
  "min": 0,
  "max": 0.95,
  "precision": 2,
  "showInPanel": {
    "conditions": [
      [
        ".isShow",
        "$eq",
        true
      ]
    ],
    "logicalType": "$and"
  }
},
"label": {
  "name": "轴标签",
  "type": "group",
  "enableHide": true,
  "children": {
    "show": {
      "default": true
    },
    "format": {
      "name": "显示格式",
      "type": "select",
      "default": "%m月",
      "description": "时间请参照 %m/%d/%Y %H:%M:%S, 整数参照 d, 浮点参照 .1f",
      "options": [
        {
          "value": "d",
          "label": "11(整数)"
        },
        {
          "value": ".1f",
          "label": "11.1(浮点数)"
        },
        {
          "value": ".2f",
          "label": "11.11(浮点数)"
        },
        {
          "value": "%Y年",
```

```
    "label": "2016年"
  },
  {
    "value": "%Y",
    "label": "2016(年份)"
  },
  {
    "value": "%m月%d日",
    "label": "01月01日"
  },
  {
    "value": "%m/%d",
    "label": "01/01(月/日)"
  },
  {
    "value": "%H:%M:%S",
    "label": "02:30:00"
  },
  {
    "value": "%H:%M",
    "label": "02:30(时:分)"
  },
  {
    "value": "%Y/%m/%d %H:%M",
    "label": "2016/01/01 02:30"
  },
  {
    "value": "%Y/%m/%d",
    "label": "2016/01/01"
  },
  {
    "value": "%m/%d %H:%M",
    "label": "01/01 02:30"
  },
  {
    "value": "%m月",
    "label": "1月"
  },
  {
    "value": "%d日",
    "label": "2日"
  },
  {
    "value": "%Hh",
    "label": "02h"
  },
  {
    "value": "%H",
    "label": "02(时)"
  },
  {
    "value": "%m-%d",
    "label": "01-01"
  },
  },
```

```
        {
          "value": "%m.%d",
          "label": "01.01"
        }
      ],
      "filterable": true,
      "allowCustom": true,
      "showInPanel": {
        "conditions": [
          [
            "..type",
            "$ne",
            "category"
          ]
        ]
      }
    },
    "textarea": {
      "type": "font",
      "name": "文本",
      "default": {
        "fontFamily": "Microsoft Yahei",
        "fontWeight": "normal",
        "fontSize": 12,
        "color": "rgb(144, 160, 174)"
      }
    },
    "display": {
      "type": "suite",
      "name": "轴标签展示",
      "children": {
        "angle": {
          "name": "角度",
          "type": "iconRadio",
          "default": "0",
          "options": [
            {
              "value": "0",
              "label": "水平",
              "src": "horizontal"
            },
            {
              "value": "45",
              "label": "斜角",
              "src": "incline"
            },
            {
              "value": "90",
              "label": "垂直",
              "src": "vertical"
            }
          ]
        },
        "col": 12
      }
    },
    "-----": {
```

```
        "amount": {
          "name": "数量",
          "type": "stepper",
          "default": 0,
          "min": 0,
          "step": 1,
          "description": "坐标轴刻度标签最多显示个数",
          "col": 12
        },
        "unit": {
          "type": "text",
          "default": "",
          "name": "轴单位",
          "col": 12
        }
      }
    },
    "showInPanel": {
      "conditions": [
        [
          ".isShow",
          "$eq",
          true
        ]
      ]
    },
    "axisLine": {
      "name": "轴线",
      "type": "group",
      "children": {
        "show": {
          "default": true
        },
        "color": {
          "type": "fill",
          "name": "颜色",
          "default": "rgba(255, 255, 255, 0.1)"
        }
      },
      "enableHide": true,
      "showInPanel": {
        "conditions": [
          [
            ".isShow",
            "$eq",
            true
          ]
        ]
      }
    },
    "grid": {
      "name": "网格线",
      "type": "group".
```

```
    type: "group",
    "children": {
      "show": {
        "default": false
      },
      "color": {
        "type": "fill",
        "name": "颜色",
        "default": "#fff"
      }
    },
    "enableHide": true,
    "showInPanel": {
      "conditions": [
        [
          ".isShow",
          "$eq",
          true
        ]
      ]
    }
  },
  "enableHide": true
},
"yaxis": {
  "name": "y轴",
  "children": {
    "isShow": {
      "name": "y轴可见",
      "type": "switch",
      "default": true
    },
    "extent": {
      "type": "suite",
      "name": "范围",
      "children": {
        "min": {
          "type": "select",
          "name": "最小值",
          "default": "0",
          "description": "可输入数值来自定义标签最小值。",
          "options": [
            {
              "value": "auto",
              "label": "自动取整"
            },
            {
              "value": "dataMin",
              "label": "数据最小值"
            }
          ]
        },
        "filterable": true,
        "allowCustom": true,
        "col": 12
      }
    }
  }
}
```

```
    },
    "max": {
      "type": "select",
      "name": "最大值",
      "default": "dataMax",
      "description": "可输入数值来自定义标签最大值。",
      "options": [
        {
          "value": "auto",
          "label": "自动取整"
        },
        {
          "value": "dataMax",
          "label": "数据最大值"
        }
      ],
      "filterable": true,
      "allowCustom": true,
      "col": 12
    }
  },
  "showInPanel": {
    "conditions": [
      [
        ".isShow",
        "$eq",
        true
      ]
    ]
  }
},
"label": {
  "name": "轴标签",
  "type": "group",
  "enableHide": true,
  "children": {
    "format": {
      "name": "显示格式",
      "type": "select",
      "default": ".0f",
      "description": "整数参照 d, 浮点参照 .1f",
      "options": [
        {
          "value": null,
          "label": "默认"
        },
        {
          "value": ".0f",
          "label": "11(整数)"
        },
        {
          "value": ".1f",
          "label": "11.1(浮点数)"
        }
      ]
    }
  }
}
```

```
    {
      "value": ".2f",
      "label": "11.11(浮点数)"
    },
    {
      "value": "%",
      "label": "11%"
    },
    {
      "value": ".1%",
      "label": "11.1%"
    },
    {
      "value": ".2%",
      "label": "11.11%"
    }
  ],
  "filterable": true,
  "allowCustom": true
},
"textarea": {
  "type": "font",
  "name": "文本",
  "default": {
    "fontFamily": "Microsoft Yahei",
    "fontWeight": "normal",
    "fontSize": 12,
    "color": "rgb(144, 160, 174)"
  }
},
"display": {
  "type": "suite",
  "name": "轴标签展示",
  "children": {
    "amount": {
      "name": "数量",
      "type": "stepper",
      "default": 6,
      "min": 0,
      "step": 1,
      "description": "坐标轴刻度标签最多显示个数",
      "col": 12
    },
    "unit": {
      "type": "text",
      "name": "单位",
      "default": "",
      "col": 12,
      "enableHide": true
    }
  }
},
"showInPanel": {
```

```
        "conditions": [
          [
            ".isShow",
            "$eq",
            true
          ]
        ]
      }
    },
    "axisLine": {
      "name": "轴线",
      "type": "group",
      "children": {
        "show": {
          "default": true
        },
        "color": {
          "type": "fill",
          "name": "颜色",
          "default": "rgba(255, 255, 255, 0.1)"
        }
      },
      "enableHide": true,
      "showInPanel": {
        "conditions": [
          [
            ".isShow",
            "$eq",
            true
          ]
        ]
      }
    },
    "grid": {
      "name": "网格线",
      "type": "group",
      "children": {
        "show": {
          "default": false
        },
        "color": {
          "type": "fill",
          "name": "颜色",
          "default": "#434343"
        }
      },
      "enableHide": true,
      "showInPanel": {
        "conditions": [
          [
            ".isShow",
            "$eq",
            true
          ]
        ]
      }
    }
  }
}
```

```
    }
  }
},
"enableHide": true
}
},
"type": "menuChild",
"mode": "multiple"
},
"series": {
  "type": "menuChild",
  "name": "系列",
  "children": {
    "series2type": {
      "type": "switch",
      "name": "系列转类型",
      "default": false
    },
    "series": {
      "type": "tabs",
      "name": "数据系列",
      "default": [
        {
          "color": {
            "type": "flat",
            "value": "rgb(10, 115, 255)"
          }
        }
      ],
      "template": {
        "type": "object",
        "name": "系列<%= i + 1%>",
        "children": {
          "color": {
            "name": "颜色",
            "type": "fill",
            "default": {
              "type": "flat",
              "value": "rgba(131, 32, 220, 0.5)"
            },
            "components": [
              "flat",
              "linearGradient"
            ]
          }
        }
      }
    }
  }
},
"others": {
  "name": "其它",
  "children": {
    "animation": {
```

```
animation: {
  "name": "缓动动画",
  "type": "group",
  "fold": true,
  "children": {
    "show": {
      "default": true
    },
    "setting": {
      "type": "suite",
      "name": "动画设置",
      "children": {
        "animationEasing": {
          "name": "缓动效果",
          "type": "select",
          "options": [
            {
              "value": "linear",
              "label": "linear"
            },
            {
              "value": "quadraticIn",
              "label": "quadraticIn"
            },
            {
              "value": "quadraticOut",
              "label": "quadraticOut"
            },
            {
              "value": "quadraticInOut",
              "label": "quadraticInOut"
            },
            {
              "value": "cubicIn",
              "label": "cubicIn"
            },
            {
              "value": "cubicOut",
              "label": "cubicOut"
            },
            {
              "value": "cubicInOut",
              "label": "cubicInOut"
            },
            {
              "value": "quarticIn",
              "label": "quarticIn"
            },
            {
              "value": "quarticOut",
              "label": "quarticOut"
            },
            {
              "value": "quarticInOut",
              "label": "quarticInOut"
            }
          ]
        }
      }
    }
  }
}
```

```
    },
    {
      "value": "quinticIn",
      "label": "quinticIn"
    },
    {
      "value": "quinticOut",
      "label": "quinticOut"
    },
    {
      "value": "quinticInOut",
      "label": "quinticInOut"
    },
    {
      "value": "sinusoidalIn",
      "label": "sinusoidalIn"
    },
    {
      "value": "sinusoidalOut",
      "label": "sinusoidalOut"
    },
    {
      "value": "sinusoidalInOut",
      "label": "sinusoidalInOut"
    },
    {
      "value": "exponentialIn",
      "label": "exponentialIn"
    },
    {
      "value": "exponentialOut",
      "label": "exponentialOut"
    },
    {
      "value": "exponentialInOut",
      "label": "exponentialInOut"
    },
    {
      "value": "circularIn",
      "label": "circularIn"
    },
    {
      "value": "circularOut",
      "label": "circularOut"
    },
    {
      "value": "circularInOut",
      "label": "circularInOut"
    },
    {
      "value": "elasticIn",
      "label": "elasticIn"
    },
    {
```

```
        "value": "elasticOut",
        "label": "elasticOut"
    },
    {
        "value": "elasticInOut",
        "label": "elasticInOut"
    },
    {
        "value": "backIn",
        "label": "backIn"
    },
    {
        "value": "backOut",
        "label": "backOut"
    },
    {
        "value": "backInOut",
        "label": "backInOut"
    },
    {
        "value": "bounceIn",
        "label": "bounceIn"
    },
    {
        "value": "bounceOut",
        "label": "bounceOut"
    },
    {
        "value": "bounceInOut",
        "label": "bounceInOut"
    }
],
"default": "cubicInOut",
"filterable": true,
"allowCustom": true,
"col": 12
},
"animationAfterPreviousSeries": {
    "name": "各系列依次动画",
    "type": "switch",
    "default": false,
    "col": 12
}
},
"enter": {
    "type": "suite",
    "name": "入场动画",
    "children": {
        "animationDuration": {
            "name": "初始动画时长",
            "type": "stepper",
            "step": 1,
            "default": 1000,
```

```
        "col": 12,
        "suffix": "ms"
      }
    }
  },
  "update": {
    "type": "suite",
    "name": "更新动画",
    "children": {
      "animationDurationUpdate": {
        "name": "更新动画时长",
        "type": "stepper",
        "default": 300,
        "step": 1,
        "col": 12,
        "suffix": "ms"
      },
      "animationUpdateFromPrevious": {
        "name": "是否从之前位置开始",
        "type": "switch",
        "default": true,
        "col": 12
      }
    }
  },
  "enableHide": true
},
"tooltip": {
  "name": "提示框",
  "type": "group",
  "fold": true,
  "children": {
    "show": {
      "default": true
    },
    "hideDelay": {
      "name": "消失延迟时间",
      "type": "stepper",
      "default": 100,
      "step": 1,
      "suffix": "ms"
    },
    "trigger": {
      "type": "suite",
      "name": "触发方式",
      "children": {
        "type": {
          "name": "触发类型",
          "type": "iconRadio",
          "options": [
            {
              "value": "item",
              "label": "数据项",
              "icon": "data-item"
            }
          ]
        }
      }
    }
  }
}
```

```
        "src": "item"
      },
      {
        "value": "axis",
        "label": "坐标轴",
        "src": "axis"
      }
    ],
    "default": "item",
    "col": 12
  },
  "action": {
    "name": "触发动作",
    "type": "iconRadio",
    "options": [
      {
        "value": "hover",
        "label": "悬浮",
        "src": "hover"
      },
      {
        "value": "click",
        "label": "点击",
        "src": "click"
      }
    ],
    "default": "hover",
    "col": 12
  }
},
"textStyle": {
  "name": "文本样式",
  "type": "font",
  "default": {
    "fontFamily": "Microsoft Yahei",
    "fontWeight": "normal",
    "fontSize": 14,
    "color": "#fff"
  }
},
"axisPointer": {
  "name": "坐标轴指示器",
  "type": "group",
  "fold": true,
  "children": {
    "show": {
      "name": "显示/隐藏",
      "type": "boolean",
      "default": true
    }
  },
  "_type": {
    "name": "类型",
    "type": "select",
    "options": [
```

```
options : [
  {
    "name": "直线指示器",
    "value": "line"
  }
],
"default": "line",
"showInPanel": {
  "conditions": [
    [
      ".show",
      "$eq",
      true
    ]
  ]
}
},
"lineStyle": {
  "name": "指示线样式",
  "type": "suite",
  "fold": true,
  "showInPanel": {
    "conditions": [
      [
        ".show",
        "$eq",
        true
      ],
      [
        "_type",
        "$eq",
        "line"
      ]
    ]
  ]
},
"children": {
  "width": {
    "name": "宽度",
    "type": "stepper",
    "step": 1,
    "default": 1,
    "col": 12,
    "suffix": "px"
  },
  "_type": {
    "name": "类型",
    "type": "iconRadio",
    "options": [
      {
        "name": "实线",
        "value": "solid",
        "src": "solid"
      },
      {
        "name": "虚线",
```

```
        "value": "dashed",
        "src": "dashed-line"
    },
    {
        "name": "点线",
        "value": "dotted",
        "src": "dot-line"
    }
],
"default": "solid",
"col": 12
},
"color": {
    "name": "颜色",
    "type": "fill",
    "default": "#f00",
    "col": 24
}
}
},
"enableHide": true,
"showInPanel": {
    "conditions": [
        [
            ".trigger.type",
            "$eq",
            "axis"
        ]
    ]
}
},
"bgBox": {
    "name": "背景框样式",
    "type": "group",
    "children": {
        "backgroundColor": {
            "name": "背景色",
            "type": "fill",
            "default": "rgba(0, 0, 0, 0.65)"
        },
        "customSize": {
            "name": "自定义",
            "type": "suite",
            "children": {
                "show": {
                    "default": false
                },
                "width": {
                    "name": "宽度",
                    "type": "stepper",
                    "default": 150,
                    "step": 1,
                    "col": 12,
```

```
        "suffix": "px"
      },
      "height": {
        "name": "高度",
        "type": "stepper",
        "default": 50,
        "step": 1,
        "col": 12,
        "suffix": "px"
      }
    },
    "enableHide": true
  },
  "padding": {
    "name": "内边距",
    "type": "stepper",
    "default": 10,
    "step": 1,
    "suffix": "px"
  },
  "offset": {
    "type": "suite",
    "name": "偏移量",
    "children": {
      "xOffset": {
        "name": "水平偏移量",
        "type": "stepper",
        "default": 6,
        "step": 1,
        "col": 12,
        "suffix": "px"
      },
      "yOffset": {
        "name": "垂直偏移量",
        "type": "stepper",
        "default": 10,
        "step": 1,
        "col": 12,
        "suffix": "px"
      }
    }
  },
  "border": {
    "type": "suite",
    "name": "边框",
    "children": {
      "borderWidth": {
        "name": "边框粗细",
        "type": "stepper",
        "step": 1,
        "default": 0,
        "col": 12,
        "suffix": "px"
      }
    }
  },
}
```

```

        "borderColor": {
          "name": "边框颜色",
          "type": "fill",
          "default": "#333",
          "col": 12
        }
      }
    }
  },
  "enableHide": true
}
},
"type": "menuChild",
"mode": "single"
}
}
}
}
}

```

2.4.2. 控件配置规范

本文介绍控件（组件配置项）的配置规范，包括配置规则和通用字段。

配置规则

单个控件配置 = 通用配置 + 私有配置

例如，以下示例中 `name`、`type`、`default` 都是通用配置，`prefix` 是私有配置。

```

{
  "name": "文本框",
  "type": "text",
  "default": "",
  "description": "带描述-带标注",
  "prefix": "今日进度："
}

```

通用配置

字段名	含义	类型	是否必选	备注
<code>name</code>	显示名	string	是	示例值： <code>"文本框"</code> 。
<code>type</code>	类型	string	是	示例值： <code>"text"</code> 。当设置为 <code>hidden</code> 时，DataV将不对组件进行渲染。
<code>default</code>	配置项默认值	string、number、object、array	否	无。

字段名	含义	类型	是否必选	备注
<code>showInPanel</code>	配置项是否在面板显示	string	否	可定义配置项的显隐规则。不配置时，显示配置项；配置时，按照配置规则显示，配置规则请参见 showInPanel配置规则 。
<code>caption</code>	配置项标注	string	否	不配置时，将不显示标注。
<code>description</code>	配置项描述	string	否	无。
<code>handler</code>	处理函数名	string	否	无。
<code>col</code>	控件主体部分所占栅格数	number	否	采用24栅格系统。
<code>valuePath</code>	配置项对应值的路径	string	否	配置规则请参见 valuePath配置规则 。

字段名	含义	类型	是否必选	备注
show	自定义配置项的显示与隐藏	object	否	<p>配置示例如下。</p> <pre> { "xAxis": { "type": "group", "name": "x轴", "children": { "show": { "type": "boolean", "name": "显示", "default": true }, "color": { "type": "color", "name": "颜色", "default": "#ccc", "show": [["show", "\$eq", true]] } } } } </pre> <ul style="list-style-type: none"> <code>children.show</code> : 表示在x轴右侧显示一个点选框。该配置对兄弟属性无效。 <code>color.show</code> : 表示配置单个属性是否显示。 <code>["show", "\$eq", true]</code> : 兄弟属性 <code>children.show</code> 等于 <code>true</code> 时, <code>color</code> 属性才显示。

私有配置

控件不同，私有配置不同，详情请参见[控件配置](#)。

2.4.3. showInPanel配置规则

通过showInPanel字段，您可以定义对应控件（组件配置项）的显隐规则。

使用场景

当一些配置项在某些条件下为无效配置，但在其他条件下为有效配置时，可以使用 showInPanel 字段，控制自身是否在配置面板中显示。使用 showInPanel 字段，可以减少信息干扰。其中条件为配置中其他配置的值。例如，当数据类型从数值型切换为时间型时，数据格式配置项会显示出来，提供给您进行配置。



配置规则

```

{
  "conditions": [
    [ "path", "operator", "value" ],
    ...
  ],
  "logicalType": "logicalType"
}

```

参数	说明
conditions	配置条件，array类型。其中的每一项用长度为3的数组描述一个判断条件，数组中包含 path、operator 和 value。
path	配置条件依赖的配置项的路径，支持绝对路径和相对路径： <ul style="list-style-type: none"> 绝对路径：从顶层根节点开始，使用 . 来拼接路径，例如 chart.legend。 相对路径：从当前配置开始，用 . 来定位路径，例如 .type 或 ..legend.show。

参数	说明
operator	<p>操作符，支持的类型包括：</p> <ul style="list-style-type: none"> • eq : 相等 • \$ne : 不相等 • \$gt : 大于 • \$lt : 小于 • \$gte : 大于等于 • \$lte : 小于等于 • \$in : 在数组中 • \$nin : 不在数组中
value	<p>配置项的值。</p> <p>以使用场景为例，当 dataType 与 time 同级，且 dataType 值等于 time 时，显示数据格式。conditions 配置如下。</p> <ul style="list-style-type: none"> • 相对路径 <pre>[".type", "\$eq", "time"]</pre> • 绝对路径 <pre>["options.axis.xaxis.type", "\$eq", "time"]</pre>
logicalType	<p>conditions 定义了若干条件，logicalType 则定义了条件之间的逻辑关系。可选值包括 \$and (与) 和 \$or (或)，默认为 \$and。</p> <p>当 logicalType 为 \$and 时，可以只配置 conditions 数组。例如当 b.switch 等于 true 并且 a.switch 不等于 false 时，显示对应的配置项，showInPanel 配置如下。</p> <pre>[["b.switch", "\$eq", true], ["a.switch", "\$ne", false]]</pre>

示例一

例如配置简写如下。

```
{
  chart: {
    font: {},
    margin: {}
  },
  legend: {
    switch: {},
    color: {}
  }
}
```

想实现的效果：当switch等于true，或者font不等于pingfang时，margin出现在面板中，否则不出现。

此时可以按照以下方式配置 `showInPanel` 字段：

- 绝对路径

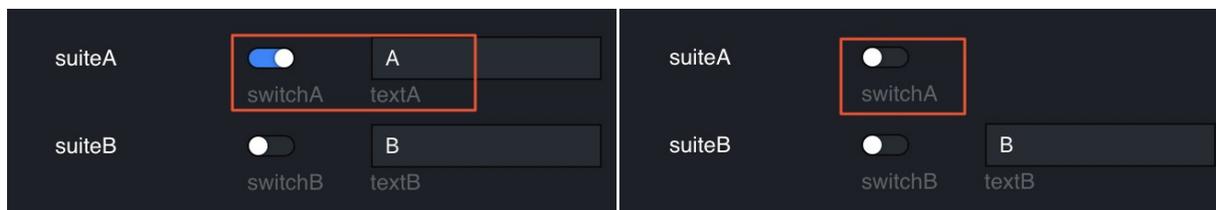
```
{
  conditions: [[ 'legend.switch', '$eq', true ], [ 'chart.font', '$ne', 'pingfang' ]],
  logicalType: '$or'
}
```

- 相对路径

```
{
  conditions: [[ '..legend.switch', '$eq', true ], [ '.font', '$ne', 'pingfang' ]],
  logicalType: '$or'
}
```

示例二

如下图所示，textA的显示条件为switchA为true或switchB为true。



```
{
  "a": {
    "type": "suite",
    "name": "suiteA",
    "children": {
      "switchA": {
        "type": "switch",
        "name": "switchA",
        "default": true,
        "col": 8
      },
      "textA": {
        "type": "text",
        "name": "textA",
        "default": "A",
        "col": 16,
        "showInPanel": {
          "conditions": [
            [
              "b.switchB",
              "$eq",
              true
            ],
            [
              "a.switchA",
              "$ne",
              false
            ]
          ]
        },
        "logicalType": "$or"
      }
    }
  },
  "b": {
    "type": "suite",
    "name": "suiteB",
    "children": {
      "switchB": {
        "type": "switch",
        "name": "switchB",
        "default": false,
        "col": 8
      },
      "textB": {
        "type": "text",
        "name": "textB",
        "default": "B",
        "col": 16
      }
    }
  }
}
```

2.4.4. valuePath配置规则

当图形用户界面GUI（Graphical User Interface）的渲染结构和数据结构不一致时，您可以通过valuePath字段处理对应的关系，使配置和数据一一对应。

使用场景

一般情况下，GUI渲染结构和数据结构是一致的，例如：

- GUI渲染结构

```
{
  "apply": {
    "type": "switch",
    "default": true
  },
  "snapTrd": {
    "type": "stepper",
    "default": 20
  }
}
```

- 数据结构

```
{
  "apply": true,
  "snapTrd": 20
}
```

以上示例中，在进行GUI渲染的时候，`snapTrd` 和 `apply` 处于同一层级。但实际数据结构中，`snapTrd` 处于 `space` 对象中，`space` 和 `apply` 处于同一层级，此时可以使用 `valuePath` 进行处理，将配置与数据对应起来。例如：

- 配置

```
{
  "apply": {
    "type": "switch",
    "default": true
  },
  "snapTrd": {
    "type": "stepper",
    "default": 20,
    "valuePath": "space.snapTrd"
  }
}
```

- 数据（加入 `valuePath` 后提取的默认值和抛出值的结构）

```
{
  "apply": true,
  "space": {
    "snapTrd": 20
  }
}
```

配置规则

- 当在基础控件中配置 `valuePath` 时，直接添加即可，详情请参见[示例一](#)。
- 当在高频套件、group和menu中配置 `valuePath` 时，它们的子元素会继承父元素的 `valuePath` ，详情请参见[示例二](#)。
- 当在tabs中配置 `valuePath` 时，tabs的子元素只能继承父元素的 `valuePath` ，而不能配置自己的 `valuePath` ，详情请参见[示例三](#)。

示例一

- 配置

```
{
  "paddingTop": {
    "type": "stepper",
    "name": "top",
    "default": 3,
    "valuePath": "chart.padding.top"
  },
  "paddingLeft": {
    "type": "hidden",
    "name": "left",
    "default": 10
  }
}
```

- 数据

```
{
  "chart": {
    "padding": {
      "top": 3
    }
  },
  "paddingLeft": 10
}
```

示例二

- 配置

```
{
  "padding": {
    "type": "group",
    "name": "组",
    "default": {
      "top": 3,
      "left": 10
    },
    "valuePath": "chart.padding",
    "children": {
      "top": {
        "type": "stepper",
        "valuePath": "abc.top"
      },
      "left": {
        "type": "stepper"
      }
    }
  }
}
```

- 数据

```
{
  "abc": {
    "top": 3.5
  },
  "chart": {
    "padding": {
      "left": 10
    }
  }
}
```

示例三

- 配置

```

{
  "padding": {
    "type": "tabs",
    "name": "组",
    "default": [
      {
        "top": 3,
        "left": 10
      }
    ],
    "valuePath": "chart.padding",
    "template": {
      "name": "系列<%= i + 1%>",
      "description": "这是系列<%= i + 1%>",
      "children": {
        "top": {
          "type": "stepper"
        },
        "left": {
          "type": "stepper"
        }
      }
    }
  }
}

```

● 数据

```

{
  "chart": {
    "padding": [
      {
        "top": 3,
        "left": 10
      }
    ]
  }
}

```

2.4.5. text

text表示组件的配置项类型为文本输入框，支持定制前后缀。

配置项说明

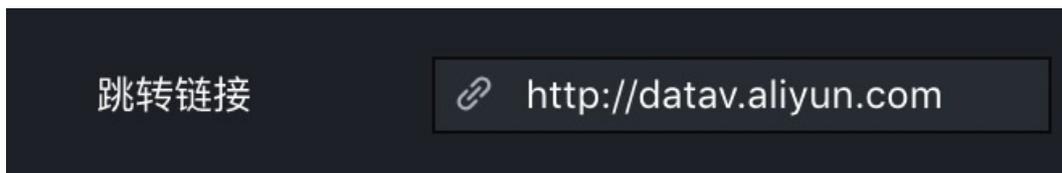
字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。

字段名	含义	类型	是否必选	备注
default	默认显示值	string	否	没填时为空。
prefix	前缀	string	否	无。
suffix	后缀	string	否	无。
prefixIcon	前缀图标名称	string	否	图标库中的名称，例如 link。
suffixIcon	后缀图标名称	string	否	图标库中的名称，例如 link。

值说明

条件	数据类型	示例	默认值
无	string	"系列一"	""

配置示例



```
"link": {
  "name": "跳转链接",
  "type": "text",
  "default": "http://datav.aliyun.com",
  "prefixIcon": "link"
}
```

2.4.6. number

number表示组件的配置项类型为数值输入框，已被stepper和slider代替，建议使用新配置项。

配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认显示值	number	否	没填时为 16。

字段名	含义	类型	是否必选	备注
min	最小值	number	否	min、range[0]、range.min 三者有其一就设置最小值成功，都没有时退化为text，不保证运行正确。
max	最大值	number	否	max、range[1]、range.max 三者有其一就设置最大值成功，都没有时配置项类型退化为text，不保证运行正确。
range	取值区间	array或object	否	配置 range 则显示滑动条。range 可以为数组 [10, 55]，也可以为对象 {min: 10, max: 55}，与 min 和 max 都没有时，配置项类型退化为text，不保证运行正确。

值说明

条件	数据类型	示例	默认值
无	number	12	0

配置示例

- 基础用法（不配置 range）

字号

22

+
-

```

"number": {
  "type": "number",
  "name": "字号",
  "default": 22,
  "min": 10,
  "max": 55
}
    
```

- 带滑动条（配置 range）

字号

22

+
-

55

> 文档版本：20220211

```

"number": {
  "type": "number",
  "name": "字号",
  "default": 22,
  "range": [
    10,
    55
  ]
}

```

2.4.7. select

select表示组件的配置项类型为选择器，支持开启过滤和自定义输入。当需要设置字体时，可开启字体渲染功能，预览字体样式。

配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认显示值	string	否	没填时为 options[0] 。
options	选项列表	array	否	每一项都包含 label 和 value 。 label 表示显示文本， value 表示文本的值。
filterable	是否支持过滤	boolean	否	默认: true。
allowCustom	是否支持自定义输入内容	boolean	否	filterable 为true时生效。默认: true。
useFont	是否渲染字体	boolean	否	无。

值说明

条件	数据类型	示例	默认值
无	string	"left"	""

配置示例



```

"font": {
  "name": "字体",
  "type": "select",
  "useFont": true,
  "default": "SimSun",
  "options": [
    {
      "value": "Microsoft Yahei",
      "label": "微软雅黑"
    },
    {
      "value": "SimSun",
      "label": "宋体"
    },
    {
      "value": "SimHei",
      "label": "黑体"
    }
  ]
}

```

2.4.8. search

search表示组件的配置项类型为下拉搜索框，已被select代替，建议使用新配置项。

配置项说明

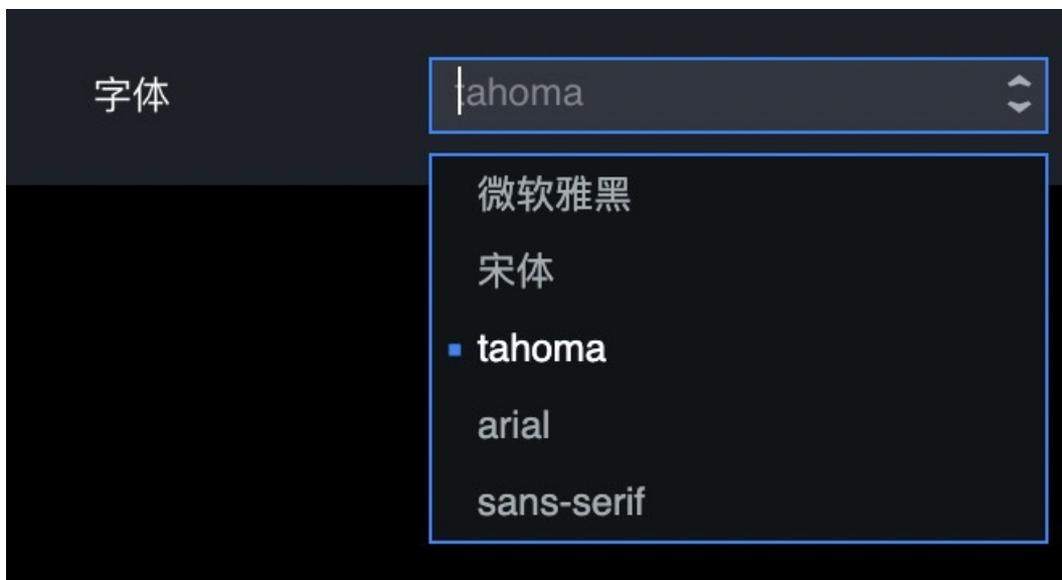
字段名	含义	类型	是否必选	备注
name	显示名	string	否	无。
type	类型	string	是	无。
default	默认显示值	string	否	没填时为 options[0] 。

字段名	含义	类型	是否必选	备注
range	取值区间	array	否	示例值如 <code>[[name: "",value: ""]]</code> ，也可以为 <code>[[key: value]]</code> 型，也可以只有 <code>[value]</code> 。没有时配置项类型退化为 <code>text</code> 。

值说明

条件	数据类型	示例	默认值
无	string	"Microsoft Yahei"	""

配置示例



```

"search": {
  "name": "字体",
  "type": "search",
  "default": "Microsoft Yahei",
  "range": [
    {
      "微软雅黑": "Microsoft Yahei"
    },
    {
      "宋体": "SimSun"
    },
    "tahoma",
    "arial",
    "sans-serif"
  ]
}

```

2.4.9. color

color表示组件的配置项类型为颜色选择器，已被fill代替，建议使用新配置项。

配置项说明

字段名	含义	类型	是否必选	备注
type	类型	string	是	无。
name	显示名	string	是	无。
default	默认显示值	string	否	没填时为 #000 。

值说明

条件	数据类型	示例	默认值
无	string	12	"#000"

配置示例



```
"color": {
  "name": "字体颜色",
  "type": "color",
  "default": "#fff"
}
```

2.4.10. multicolor

multicolor表示组件的配置项类型为渐变颜色选择器，可选单色或渐变色。

配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。

字段名	含义	类型	是否必选	备注
type	类型	string	是	无。
default	默认显示值	object	是	需要包含以下参数： <ul style="list-style-type: none"> style : 可选值包括 single 和 double 。 value : 当 style 为 single 时的颜色值。 from : 当 style 为 double 时渐变的起始颜色值。 to : 当 style 为 double 时渐变的终点颜色值。 angle : 当 style 为 double 时, 渐变颜色的角度。

值说明

条件	数据类型	示例	默认值
style 为 single	object	<pre>{ "style": "single", "value": "#ccc" }</pre>	<pre>{ "style": "single", "value": "#000" }</pre>
style 为 double	object	<pre>{ "style": "double", "from": "#fff", "to": "#ccc", "angle": 0 }</pre>	<pre>{ "style": "double", "from": "#000", "to": "rgba(0,0,0,.5)", "angle": 0 }</pre>

配置示例



```
"color": {  
  "name": "渐变颜色",  
  "type": "multicolor",  
  "default": {  
    "style": "double",  
    "from": "#000",  
    "to": "#fff",  
    "angle": 0  
  }  
}
```

2.4.11. array

array表示组件的配置项类型为数据系列，适用于多个系列配置或多个颜色循环配色，已被tabs代替，建议使用新配置项。

配置项说明

字段名	含义	类型	是否必选	备注
type	类型	string	是	无。
name	显示名	string	是	无。
default	默认值	array	否	对象数组，对象的属性应与 child 字段中定义的字段名一致。
child	对象名	object	否	请将该字段内的 type 设置为 object 。
child.child	对象属性	object	否	定义对象属性及其默认值。

注意 数据系列的 default 值必须为一个对象数组，且数组内对象中的属性必须在 child.child 字段中逐一进行定义，并设置属性的显示名、类型及默认值。

值说明

条件	数据类型	示例	默认值
无	array	<pre>[{"name": "系列一", value: "进港"}, {"name": "系列二", value: "出港"}]</pre>	[]

配置示例



```

"array": {
  "name": "数据系列",
  "type": "array",
  "default": [
    {
      "name": "系列一",
      "value": "进港"
    },
    {
      "name": "系列二",
      "value": "出港"
    }
  ],
  "child": {
    "name": "系列<%=i+1%>",
    "type": "object",
    "child": {
      "name": {
        "name": "系列值",
        "type": "text",
        "default": "系列"
      },
      "value": {
        "name": "系列名",
        "type": "text",
        "default": ""
      }
    }
  }
}

```

2.4.12. hidden

hidden表示组件的配置项类型为隐藏配置项，DataV不会渲染此配置。

配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	string、 number、 array、 object	否	设置默认不显示的值，但是会作为config传给方法。

配置示例

```
"hiddenconfig": {
  "name": "值",
  "type" : "hidden",
  "default": 22
}
```

2.4.13. boolean

boolean表示组件的配置项类型为布尔配置项，已被switch代替，建议使用新配置项。

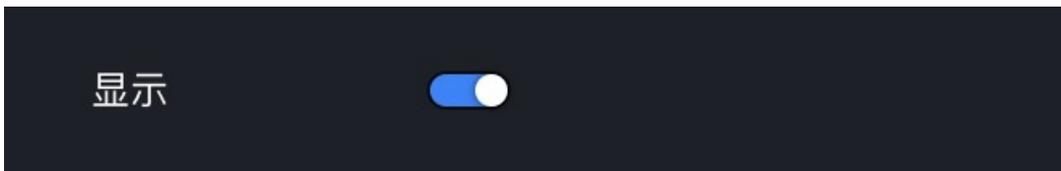
配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	boolean	否	没填时为 true 。

值说明

条件	数据类型	示例	默认值
无	boolean	true	true

配置示例



```
"boolean": {
  "type": "boolean",
  "name": "显示"
}
```

2.4.14. radio

radio表示组件的配置项类型为单选框。

配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。

字段名	含义	类型	是否必选	备注
type	类型	string	是	无。
default	默认值	string	否	不填时默认没有选中项。
options	选择列表	array	否	对象数组，包含 label 和 value 字段。label 为显示文本，value 为文本值。
optionCol	选项定宽	number	否	用来为选项设定固定宽度。

值说明

条件	数据类型	示例	默认值
无	string	"left"	""

配置示例



```

"align": {
  "name": "对齐方式",
  "type": "radio",
  "optionCol": 12,
  "options": [
    {
      "label": "左对齐",
      "value": "left"
    },
    {
      "label": "居中对齐",
      "value": "center"
    },
    {
      "label": "右对齐",
      "value": "right"
    }
  ]
}

```

2.4.15. checkbox

checkbox表示组件的配置项类型为复选框，支持为选项设置宽度。

配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	array	否	不填时值为空。
options	选项列表	array	否	对象数组，包含 label 和 value 字段。label 为显示文本，value 为文本值。
optionCo l	选项定宽	number	否	用来为选项设置固定宽度。

值说明

条件	数据类型	示例	默认值
无	array	["left", "right"]	[]

配置示例



```

"checkbox": {
  "name": "地图内容",
  "type": "checkbox",
  "optionCol": 8,
  "options": [
    {
      "value": "bg",
      "label": "背景"
    },
    {
      "value": "building",
      "label": "建筑物"
    },
    {
      "value": "road",
      "label": "道路"
    },
    {
      "value": "label",
      "label": "标注"
    }
  ]
}

```

2.4.16. buttonRadio

buttonRadio表示组件的配置项类型为单选按钮。

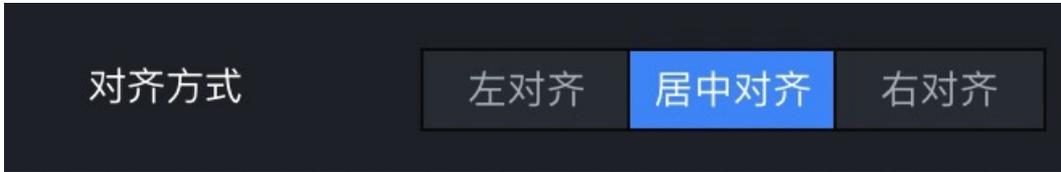
配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	string	否	不填时值为空。
options	选项列表	array	否	对象数组，包含 label 和 value 字段。label 为显示文本，value 为文本值。
evenlySplit	均匀等分	boolean	否	使图标在一行中均匀分布。当和 optionCol 同时出现时，优先使用 evenlySplit 的值。

值说明

条件	数据类型	示例	默认值
无	object	"left"	""

配置示例



```

"algin": {
  "name": "对齐方式",
  "type": "buttonRadio",
  "evenlySplit": true,
  "options": [
    {
      "value": "left",
      "label": "左对齐"
    },
    {
      "value": "right",
      "label": "居中对齐"
    },
    {
      "value": "bottom",
      "label": "右对齐"
    }
  ]
}

```

2.4.17. iconRadio

iconRadio表示组件的配置项类型为图标单选器。

配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	string	否	不填时值为空。
options	选项列表	array	是	对象数组，包含 label、value 和 src 字段。label 为显示文本，value 为文本值，src 为图片地址或图标名。

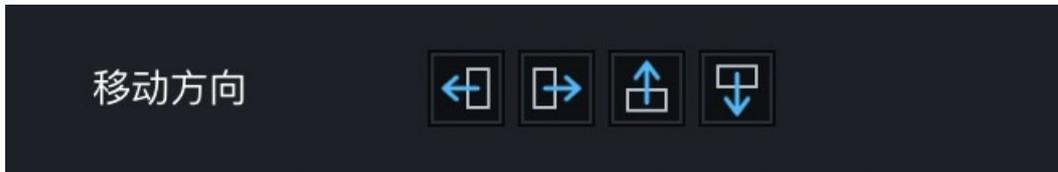
字段名	含义	类型	是否必选	备注
optionCol	选项定宽	number	否	用来为选项设定固定宽度。
evenlySplit	均匀等分	boolean	否	使图标在一行中均匀分布。当和 optionCol 同时出现时，优先使用 evenlySplit 的值。

值说明

条件	数据类型	示例	默认值
无	string	"left"	""

配置示例

- 使用图片

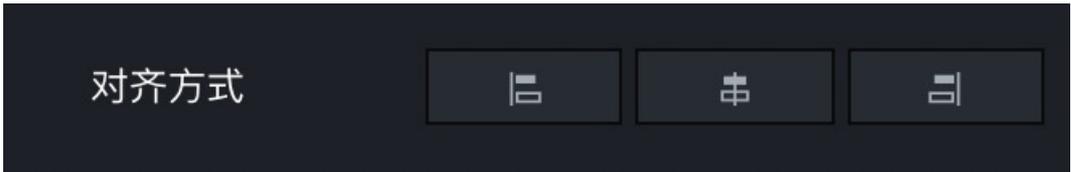


```

"direction": {
  "name": "移动方向",
  "type": "iconRadio",
  "options": [
    {
      "value": "left",
      "label": "左侧",
      "src": "https://img.alicdn.com/tfs/TB1.77AgxYaK1RjSZFnXXa80pXa-48-48.png"
    },
    {
      "value": "right",
      "label": "右侧",
      "src": "https://img.alicdn.com/tfs/TB1sWoggwDqK1RjSZSyXXaxEVXa-48-48.png"
    },
    {
      "value": "top",
      "label": "上侧",
      "src": "https://img.alicdn.com/tfs/TB1t0wjgCzqK1RjSZFjXXblCFXa-48-48.png"
    },
    {
      "value": "bottom",
      "label": "下侧",
      "src": "https://img.alicdn.com/tfs/TB1UAAjgwHqK1RjSZFkXXX.WFXa-48-48.png"
    }
  ]
}

```

- 使用图标



```

"algin": {
  "name": "对齐方式",
  "type": "iconRadio",
  "evenlySplit": true,
  "options": [
    {
      "value": "left",
      "label": "左侧",
      "src": "align-left"
    },
    {
      "value": "right",
      "label": "居中",
      "src": "align-center"
    },
    {
      "value": "bottom",
      "label": "下侧",
      "src": "align-right"
    }
  ]
}

```

2.4.18. percent

percent表示组件的配置项类型为百分比值，已被slider代替，建议使用新配置项。

配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	number	否	不填时值为 0 。最大值为 100 ，最小值为 0 ，步长为 1 。

值说明

条件	数据类型	示例	默认值
无	number	20	0

配置示例



```
"percent": {
  "name": "百分比值",
  "type": "percent",
  "default": 20
}
```

2.4.19. image

image表示组件的配置项类型为图片框。

配置项说明

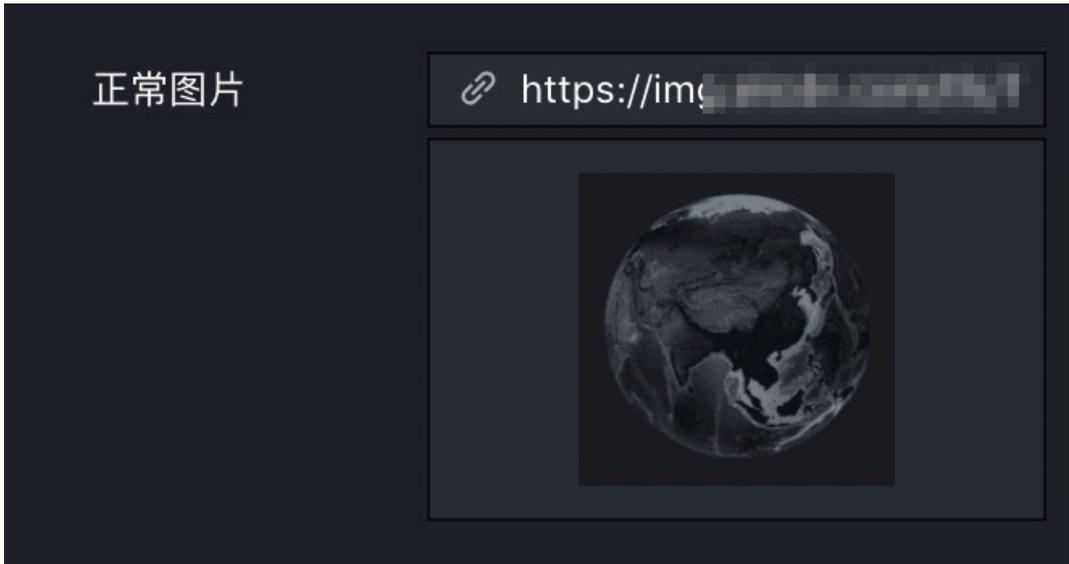
字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认显示值	string	否	没填时为空，无图片。

 **注意** 若图片链接为本地静态文件，则需要将图片放至当前组件目录的resources文件夹下，链接示例为./resources/xxxx.png。

值说明

条件	数据类型	示例	默认值
无	string	"https://img.alicdn.com/tfs/TB1cjY3ANTpK1RjSZFMXXbG_VXa-180-180.png"	""

配置示例



```
{
  "name" : "背景图",
  "type" : "image",
  "default": "http://datav.oss-cn-hangzhou.aliyuncs.com/uploads/images/c4ba3c6518c1997f4baa612a600c3fbe.png"
}
```

2.4.20. imageSelect

imageSelect表示组件的配置项类型为图片选择器。

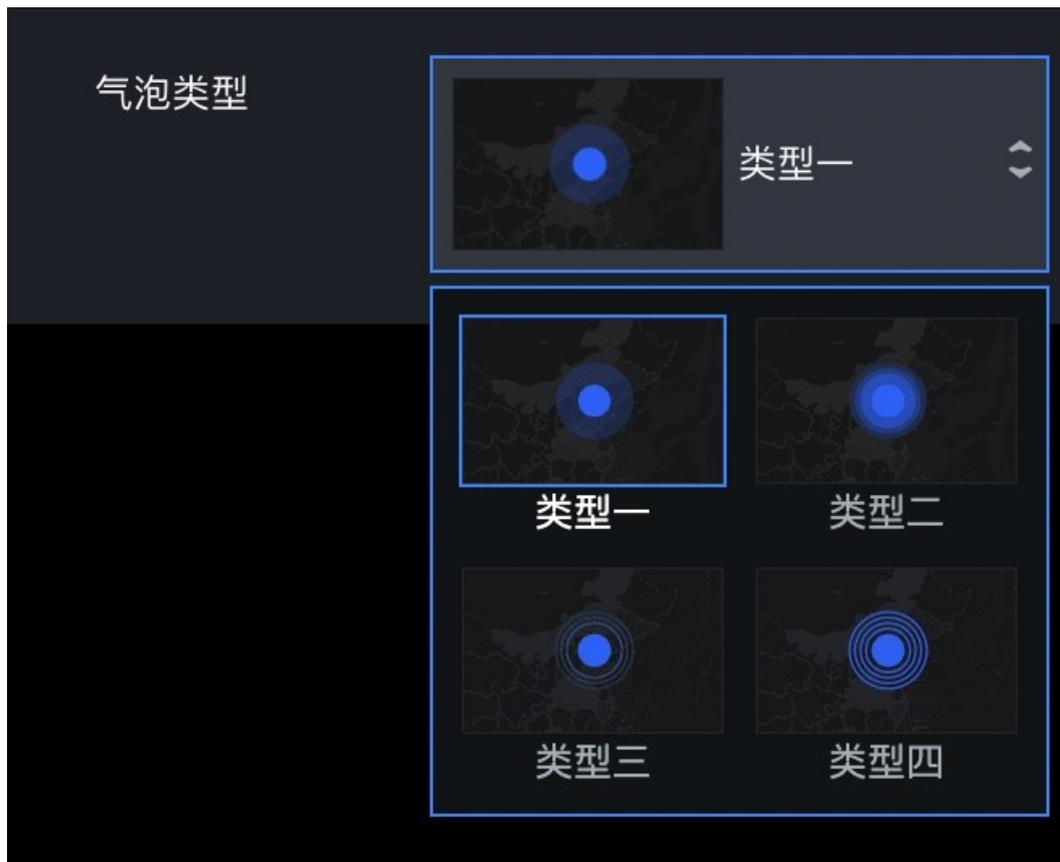
配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	不填时使用key名作为显示名，例如 margin。
type	类型	string	是	无。
default	默认值	array	否	不填时值为 options[0]。
options	选项列表	array	否	对象数组，包含 label、value 和 src 字段。label 为显示文本，value 为文本值，src 为图片地址或图标名。

值说明

条件	数据类型	示例	默认值
无	string	"left"	""

配置示例



```

"bubbleKey": {
  "name": "气泡类型",
  "type": "imageSelect",
  "default": "type1",
  "options": [
    {
      "label": "类型一",
      "value": "type1",
      "src": "https://img.alicdn.com/tfs/TB1BCgwoAT2gK0jSZFkXXcIQFXa-162-104.png"
    },
    {
      "label": "类型二",
      "value": "type2",
      "src": "https://img.alicdn.com/tfs/TB1EmgwoAT2gK0jSZFkXXcIQFXa-162-104.png"
    },
    {
      "label": "类型三",
      "value": "type3",
      "src": "https://img.alicdn.com/tfs/TB13x3soAP2gK0jSZFkXXcIQFXa-162-104.png"
    },
    {
      "label": "类型四",
      "value": "type4",
      "src": "https://img.alicdn.com/tfs/TB1pKUuoAL0gK0jSZFkXXcIQFXa-162-104.png"
    }
  ]
}

```

2.4.21. switch

switch表示组件的配置项类型为开关，支持选择是否展示文字状态。

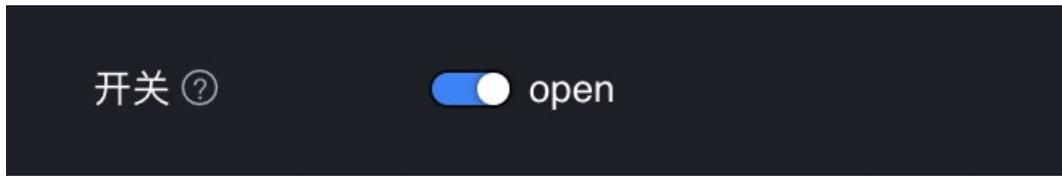
配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	boolean	否	无。
statusText	是否启用文字显示状态	boolean	否	默认为 false 。

值说明

条件	数据类型	示例	默认值
无	boolean	true	false

配置示例



```
"open": {
  "name": "开关",
  "type": "switch",
  "default": true,
  "statusText": true
}
```

2.4.22. stepper

stepper表示组件的配置项类型为步进器，支持定制步长、最大值和最小值以及前缀和后缀。

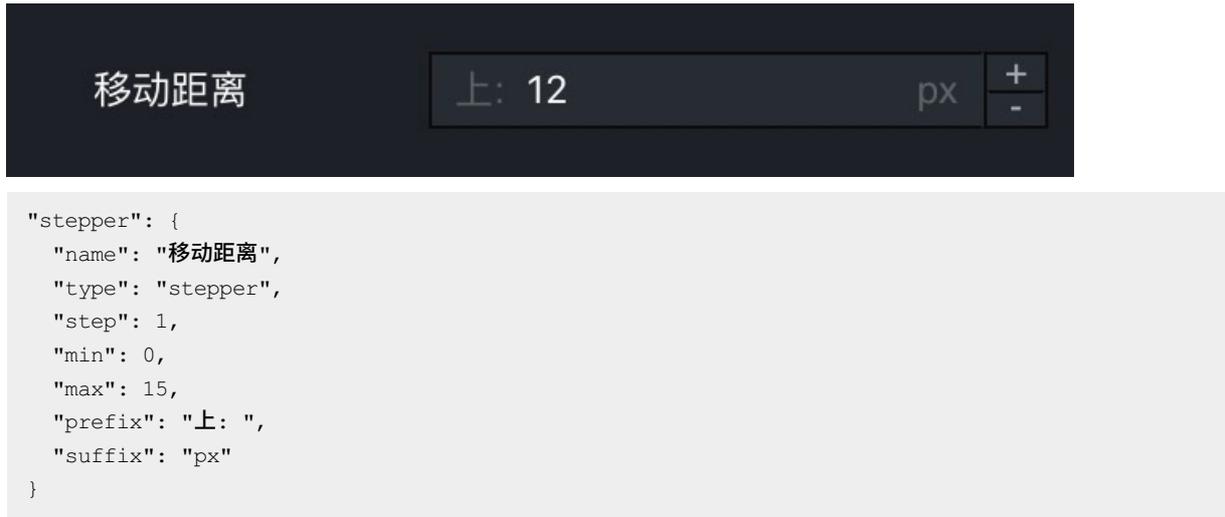
配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	number	否	不填时值为空。
step	步长	number	是	number类型。
min	最小值	number	是	number类型。
max	最大值	number	是	number类型。
prefix	前缀	string	否	string类型。
suffix	后缀	string	否	string类型。
precision	精度（小数位数）	number	否	显示精度，仅对小数值生效。

值说明

条件	数据类型	示例	默认值
无	number	12	0

配置示例



2.4.23. slider

slider表示组件的配置项类型为滑动条。支持单滑块和双滑块，同时支持定制步长、最大值和最小值、前缀和后缀、精度，可开启范围显示。

配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	number或array	否	不填时值为空。
step	步长	number	否	无。
min	最小值	number	否	无。
max	最大值	number	否	无。
prefix	前缀	string	否	无。
suffix	后缀	string	否	无。
precision	精度（小数位数）	number	否	显示精度，仅对小数值生效。
mode	滑块模式	string	否	可选值： <ul style="list-style-type: none"> single：单滑块（默认）。 double：双滑块。
showRange	是否显示范围	boolean	否	当 mode 为 double 时生效，默认为 false。

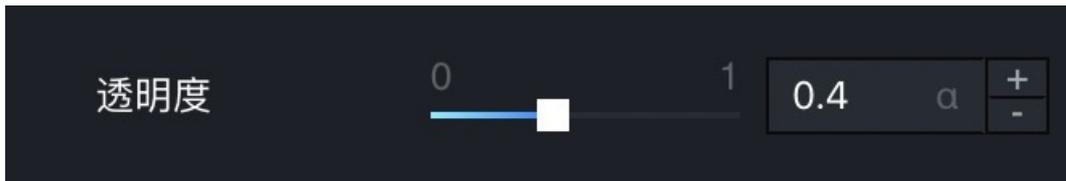
字段名	含义	类型	是否必选	备注
showCurrentRange	是否显示当前范围	boolean	否	当 mode 为 single 时生效，默认为 true。

值说明

条件	数据类型	示例	默认值
mode 为 single (滑块模式为单滑块)	number	22	0
mode 为 double (滑块模式为双滑块)	array	[10, 30]	[0, 0]

配置示例

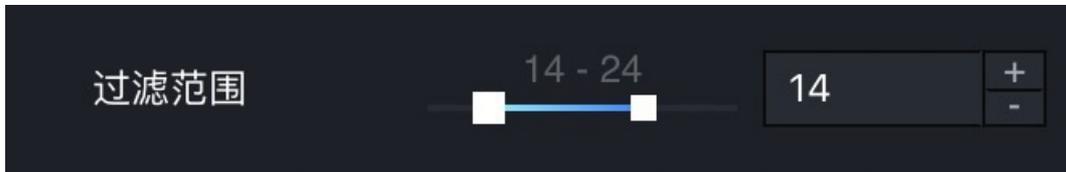
- mode 为 single (滑块模式为单滑块)



```

"slider": {
  "name": "透明度",
  "type": "slider",
  "step": 0.1,
  "min": 0,
  "max": 1,
  "suffix": "α",
  "showRange": true
}
    
```

- mode 为 double (滑块模式为双滑块)



```

"slider": {
  "name": "过滤范围",
  "type": "slider",
  "mode": "double",
  "step": 1,
  "min": 10,
  "max": 30,
  "showCurrentRange": true
}

```

2.4.24. keyBoard

keyBoard表示组件的配置项类型为热键定义器。

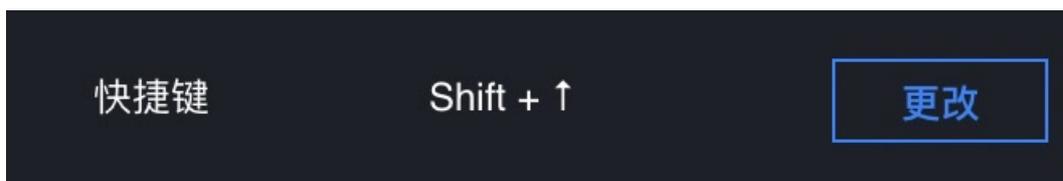
配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	array	否	不填时为 [16, 38] 。

值说明

条件	数据类型	示例	默认值	备注
无	array	[16, 38]	[16, 38]	快捷键的ASCII。

配置示例



```

"keyboard": {
  "name": "快捷键",
  "type": "keyboard",
  "default": [16, 38]
}

```

2.4.25. fill

fill表示组件的配置项类型为填充框，支持三种填充类型以及他们的组合。

配置项说明

字段名	含义	类型	是否必选	备注
<code>name</code>	显示名	string	是	无。
<code>type</code>	类型	string	是	无。
<code>default</code>	默认值	string或object	否	不填时值为空。取值请参见下文的 值说明 。
<code>components</code>	填充类型	array	否	支持以下三种填充类型，以及它们的组合： <ul style="list-style-type: none"> <code>flat</code>：纯色填充（默认）。 <code>linearGradient</code>：渐变填充。 <code>image</code>：图片填充。 <div style="border: 1px solid #add8e6; padding: 5px; margin-top: 10px;"> ? 说明 填充类型不同，对应的值也不同，详情请参见值说明。 </div>

值说明

条件	数据类型	示例	默认值	备注
当仅有纯色填充时	string	<code>"#333"</code>	<code>"#000"</code>	无。

条件	数据类型	示例	默认值	备注
当仅有线性渐变填充时	object	<pre> { "angle": 0, "stops": [{ "offset": 0, "color": "#000" }, { "offset": 100, "color": "red" }] } </pre>	<pre> { "angle": 0, "stops": [{ "offset": 0, "color": "#000" }, { "offset": 100, "color": "rgba(0,0,0,0.5)" }] } </pre>	无。

条件	数据类型	示例	默认值	备注
当仅有图片填充时	object	<pre>{ "fillType": "cover", "url": "https://img.alicdn.com/tfs/TB16ag8v4z1gK0jSZSgXXavwpXa-96-96.png", "tileSize": 100 }</pre>	<pre>{ "fillType": "stretch", "url": "https://img.alicdn.com/tfs/TB19njRJ7zoK1RjSZFlXXai4VXa-16-16.png", "tileSize": 100 }</pre>	无。
当支持多种填充类型时	object	<pre>{ "type": "linearGradient", "value": { "angle": 90, "stops": [{ "color": "#333", "offset": 0 }, { "color": "red", "offset": 100 }] } }</pre>	<pre>{ "type": "flat", "value": "#000" }</pre>	<ul style="list-style-type: none"> • <code>type</code> : 当前应用的填充类型。 • <code>value</code> : 填充值。

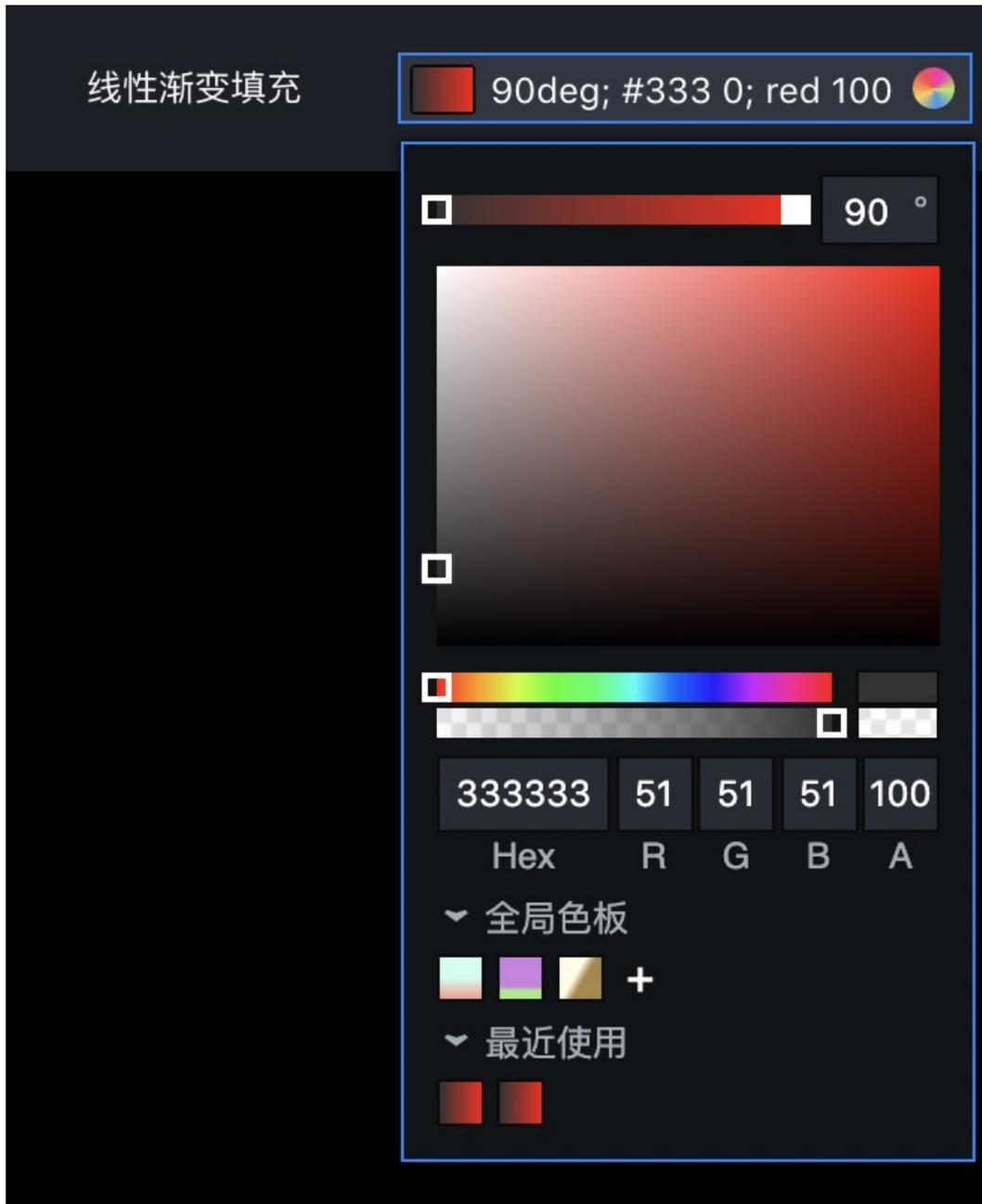
配置示例

- 填充类型为 `flat`



```
"flat": {  
  "name": "纯色填充",  
  "type": "fill",  
  "default": "#333"  
}
```

- 填充类型为 `linearGradient`



```
"linearGradient": {
  "name": "线性渐变填充",
  "type": "fill",
  "components": [
    "linearGradient"
  ],
  "default": {
    "angle": 90,
    "stops": [
      {
        "color": "#333",
        "offset": 0
      },
      {
        "color": "red",
        "offset": 100
      }
    ]
  }
}
```

- 填充类型为 `image`



```
"image": {
  "name": "图片填充",
  "type": "fill",
  "components": [
    "image"
  ],
  "default": {
    "fillType": "cover",
    "url": "https://img.alicdn.com/tfs/TB16ag8v4z1gK0jSZSgXXavwpXa-96-96.png",
    "tileSize": 100
  }
}
```

2.4.26. suite

suite表示组件的配置项类型为通用套件，由一组控件组成。

配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	object	否	不填时值为空。取值请参见值说明。
children	控件模块	object	否	默认为 {}。取值请参见值说明。
enableHide	配置项对应元素是否能够在画布隐藏	boolean	否	默认为 false。设置为 true 时，显示显示图标按钮，并在当前配置项中增加 show 字段。

值说明

条件	数据类型	示例	默认值
不配置 enableHide，或将 enableHide 配置为 false	object	<pre>{ "number": 12, "open": true }</pre>	<pre>{}</pre>
将 enableHide 配置为 true	object	<pre>{ "number": 12, "open": true, "show": true }</pre>	<pre>{ "show": true }</pre>

配置示例

- 基础用法（不配置 enableHide）



```
{
  "style": {
    "name": "套件",
    "type": "suite",
    "enableHide": true,
    "default": { "show": false, "open": true, "size": 7 },
    "children": {
      "open": {
        "name": "开关",
        "type": "switch",
        "col": 12
      },
      "size": {
        "type": "stepper",
        "caption": "大小",
        "min": 0,
        "max": 10,
        "step": 1,
        "col": 12
      }
    }
  }
}
```

- 允许配置项对应元素隐藏（配置 `enableHide` 为 `true`）



```
{
  "style": {
    "name": "套件",
    "type": "suite",
    "default": { "show": false, "open": true, "size": 7 },
    "children": {
      "open": {
        "name": "开关",
        "type": "switch",
        "col": 12
      },
      "size": {
        "type": "stepper",
        "caption": "大小",
        "min": 0,
        "max": 10,
        "step": 1,
        "col": 12
      }
    }
  }
}
```

2.4.27. margin

margin表示组件的配置项类型为边距套件，支持定义上下左右四个边的边距数值。

配置项说明

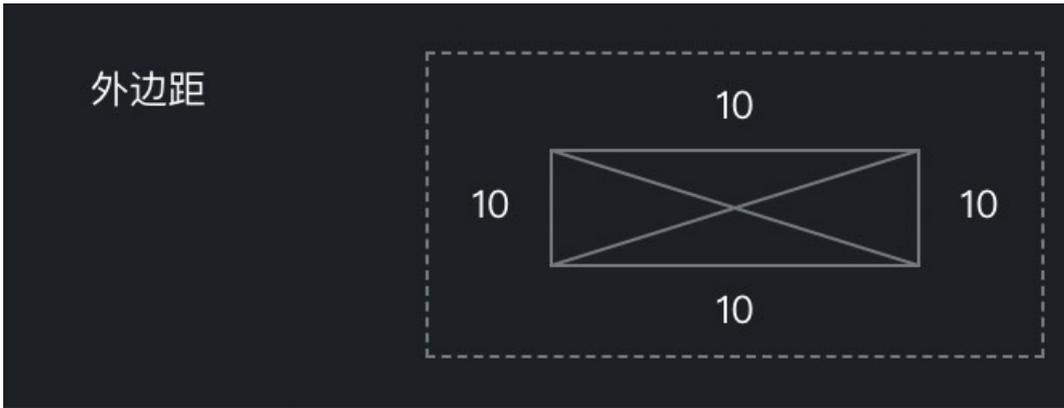
字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	string	否	不填时值为空。
components	配置项模块	array	否	默认值为 ["top", "bottom", "left", "right"]。支持的模块包括 "top"、"bottom"、"left" 和 "right"。
enableHide	配置项对应元素是否能够在画布隐藏	boolean	否	默认为 false。设置为 true 时，显示显示图标按钮，并在当前配置项中增加 show 字段。

值说明

条件	数据类型	示例	默认值
不配置 enableHide，或将 enableHide 配置为 false	object	<pre>{ "top": 12, "right": 24, "bottom": 8, "left": 24 }</pre>	<pre>{ "top": 10, "right": 10, "bottom": 10, "left": 10 }</pre>
将 enableHide 配置为 true	object	<pre>{ "top": 12, "right": 24, "bottom": 8, "left": 24, "show": true }</pre>	<pre>{ "top": 10, "right": 10, "bottom": 10, "left": 10, "show": true }</pre>

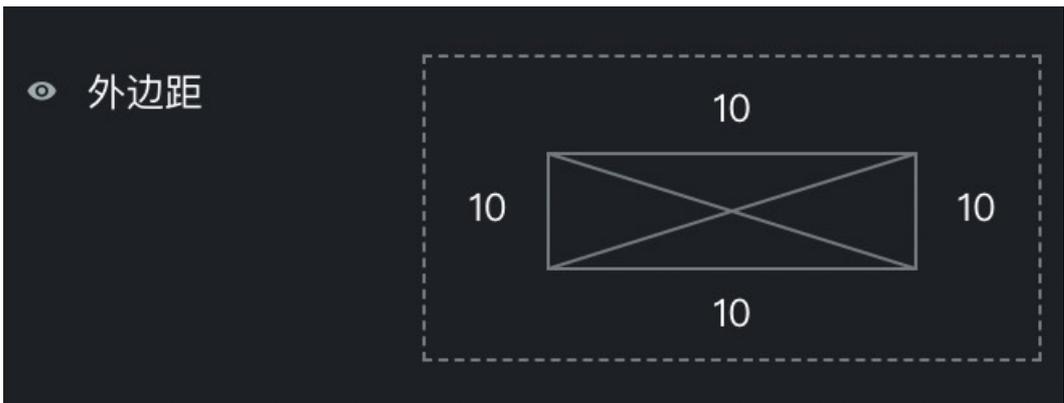
配置示例

- 基础用法（不配置 enableHide）



```
{
  "margin": {
    "name": "外边距",
    "type": "margin",
    "default": { "top": 10, "left": 10, "right": 10, "bottom": 10 }
  }
}
```

- 允许配置项对应元素隐藏（配置 enableHide 为 true ）



```
{
  "margin": {
    "name": "外边距",
    "type": "margin",
    "enableHide": true,
    "default": { "top": 10, "left": 10, "right": 10, "bottom": 10 }
  }
}
```

- 配置项模块插拔（设置 components ）



```

{
  "marginComponents": {
    "name": "外边距",
    "type": "margin",
    "enableHide": true,
    "components": ["top", "bottom"]
  }
}

```

2.4.28. padding

padding表示组件的配置项类型为边距套件，支持定义上下左右四个边的内边距数值。

配置项说明

字段	含义	数据类型	默认值	备注
min	最小值	number	0	无。
max	最大值	number	Infinity	无。

值说明

条件	数据类型	示例	默认值
无	object	<pre> { "top": 5, "bottom": 10, "left": 10, "right": 5 } </pre>	<pre> { "top": 5, "bottom": 10, "left": 10, "right": 5 } </pre>

配置示例



```

{
  "padding": {
    "type": "padding",
    "name": "内边距",
    "default": {
      "top": 5,
      "bottom": 10,
      "left": 10,
      "right": 5
    },
    "min": 0,
    "max": 1000
  }
}

```

2.4.29. line

line表示组件的配置项类型为线样式套件，由线粗细步进器、平滑方式图标单选器、样式选择器和纯色填充框组成。

配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	string	否	不填时值为空。
components	配置项模块	array	是	默认值为： <code>["width", "curve", "style", "color"]</code> 。支持的模块包括 <code>"width"</code> 、 <code>"curve"</code> 、 <code>"style"</code> 和 <code>"color"</code> 。
enableHide	配置项对应元素是否能够在画布隐藏	boolean	否	默认为 <code>false</code> 。设置为 <code>true</code> 时，显示显隐图标按钮，并在当前配置项中增加 <code>show</code> 字段。

值说明

条件	数据类型	示例	默认值
----	------	----	-----

条件	数据类型	示例	默认值
不配置 <code>enableHide</code> ，或 将 <code>enableHide</code> 配置为 <code>false</code>	object	<pre>{ "width": 4, "curve": "smooth", "style": "solid", "color": "#00f" }</pre>	<pre>{ "width": 1, "curve": "polyline", "style": "solid", "color": "#fff" }</pre>
将 <code>enableHide</code> 配置为 <code>true</code>	object	<pre>{ "width": 4, "curve": "smooth", "style": "solid", "color": "#00f", "show": true }</pre>	<pre>{ "width": 1, "curve": "polyline", "style": "solid", "color": "#fff", "show": true }</pre>

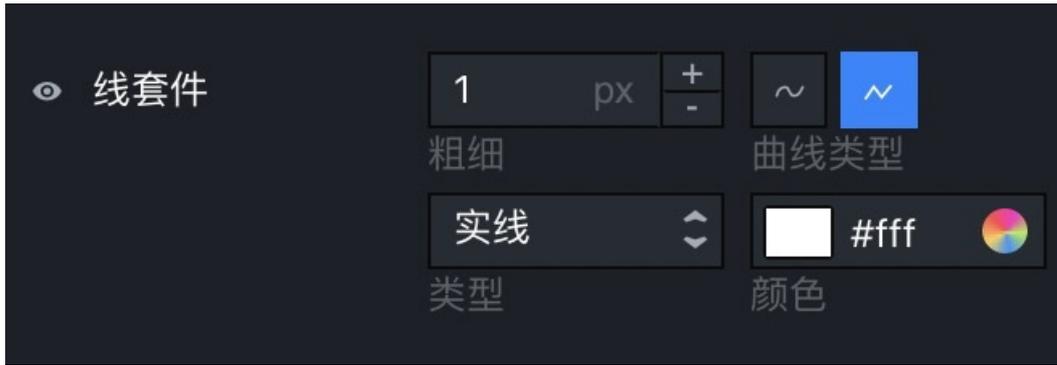
配置示例

- 基础用法（不配置 `enableHide`）



```
"line": {
  "name": "线套件",
  "type": "line"
}
```

- 允许配置项对应元素隐藏（配置 `enableHide` 为 `true`）



```
"line": {
  "name": "线套件",
  "type": "line",
  "enableHide": true
}
```

- 配置项模块插拔（设置 `components`）



```
"line": {
  "name": "线套件",
  "type": "line",
  "enableHide": true,
  "components": [
    "style",
    "color"
  ]
}
```

2.4.30. font

font表示组件的配置项类型为字体套件，由字体选择器、字体粗细选择器、字体大小步进器和纯色填充框组成。

配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	array	否	不填时值为空。

字段名	含义	类型	是否必选	备注
<code>components</code>	配置项模块	array	否	默认值为 <code>["fontFamily","fontWeight","fontSize","color"]</code> 。支持的模块包括 <code>"fontFamily"</code> 、 <code>"fontWeight"</code> 、 <code>"fontSize"</code> 和 <code>"color"</code> 。
<code>enableHide</code>	配置项对应元素是否能够在画布隐藏	boolean	否	默认为 <code>false</code> 。设置为 <code>true</code> 时，显示显示图标按钮，并在当前配置项中增加 <code>show</code> 字段。

值说明

条件	数据类型	示例	默认值
不配置 <code>enableHide</code> ，或将 <code>enableHide</code> 配置为 <code>false</code>	object	<pre>{ "fontFamily": "simSun", "fontWeight": 400, "color": "#333", "fontSize": 12 }</pre>	<pre>{ "fontFamily": "Microsoft Yahei", "fontWeight": "normal", "color": "#fff", "fontSize": 12 }</pre>
将 <code>enableHide</code> 配置为 <code>true</code>	object	<pre>{ "fontFamily": "simSun", "fontWeight": 400, "color": "#333", "fontSize": 12, "show": true }</pre>	<pre>{ "fontFamily": "Microsoft Yahei", "fontWeight": "normal", "color": "#fff", "fontSize": 12, "show": true }</pre>

配置示例

- 基础用法（不配置 `enableHide`）



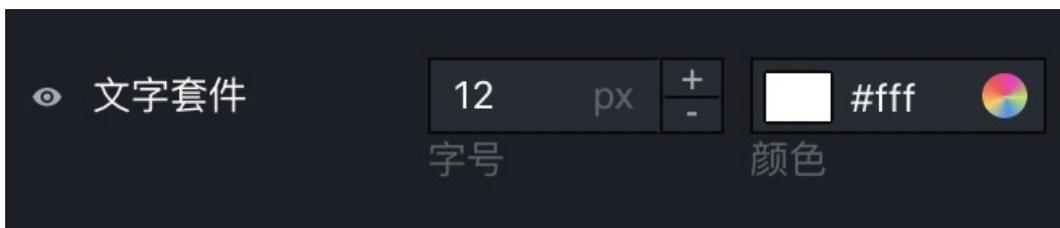
```
"font": {  
  "name": "文字套件",  
  "type": "font"  
}
```

- 允许配置项对应元素隐藏（配置 `enableHide` 为 `true`）



```
"font": {  
  "name": "文字套件",  
  "type": "font",  
  "enableHide": true  
}
```

- 配置项模块插拔（设置 `components`）



```
"font": {  
  "name": "文字套件",  
  "type": "font",  
  "enableHide": true,  
  "components": [  
    "fontSize",  
    "color"  
  ]  
}
```

2.4.31. group

group表示组件的配置项类型为配置项组，可包含多个子配置项（建议多个同类元素配置项归于一个group）。可用来组织内容，并支持折叠或展开组的内容。

配置项说明

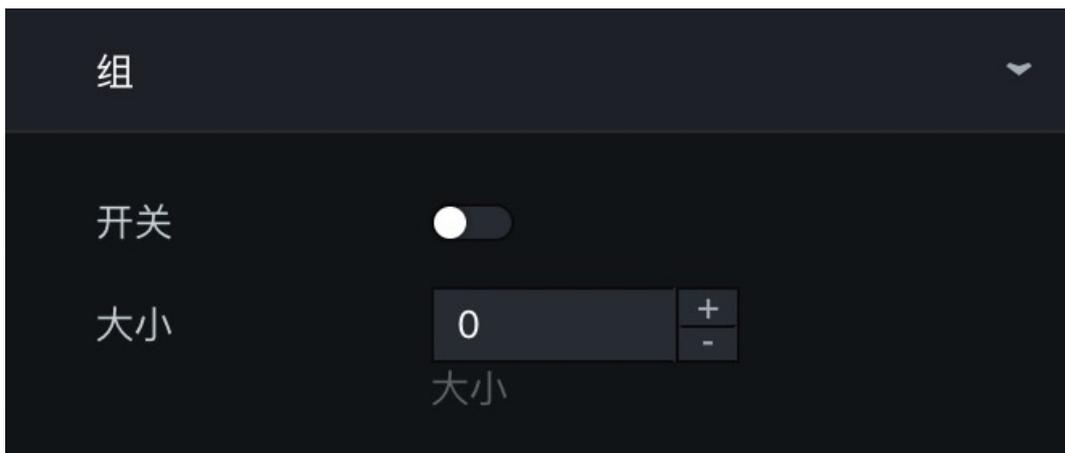
字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	object	否	没填时为空。
children	组内元素	object	否	没填时为空。
enableHide	配置项对应元素是否能够在画布隐藏	boolean	否	默认为 false。设置为 true 时，显示显示图标按钮，并在当前配置项值中增加 show 字段。
fold	是否默认展开	boolean	否	无。

值说明

条件	数据类型	示例	默认值
不配置 enableHide，或将 enableHide 配置为 false	object	<pre>{ "number": 12, "open": true }</pre>	<pre>{}</pre>
将 enableHide 配置为 true	object	<pre>{ "number": 12, "open": true, "show": true }</pre>	<pre>{ "show": true }</pre>

配置示例

- 基础用法（不配置 enableHide）



```
"group": {
  "name": "组",
  "type": "group",
  "children": {
    "open": {
      "name": "开关",
      "type": "switch",
      "col": 12
    },
    "size": {
      "type": "stepper",
      "caption": "大小",
      "min": 0,
      "max": 10,
      "step": 1,
      "col": 12
    }
  }
}
```

- 允许配置项对应元素隐藏（配置 `enableHide` 为 `true`）



```

"group": {
  "name": "组",
  "type": "group",
  "enableHide": true,
  "children": {
    "open": {
      "name": "开关",
      "type": "switch",
      "col": 12
    },
    "size": {
      "type": "stepper",
      "caption": "大小",
      "min": 0,
      "max": 10,
      "step": 1,
      "col": 12
    }
  }
}
    
```

2.4.32. tabs

tabs表示组件的配置项类型为标签组。切换标签可展示当前标签下的内容，支持动态增减标签组。

配置项说明

字段名	含义	类型	是否必选	备注
name	显示名	string	是	无。
type	类型	string	是	无。
default	默认值	object	否	不填时值为空。
addable	是否可以动态增减	boolean	否	默认为 true 。
template	动态增减的模板	object	否	默认为 {} 。
children	各标签组内容	array	否	默认为 [] 。子控件key字段不可使用关键字：_icon 、 id 和 _label 。 <div style="background-color: #e0f2f7; padding: 5px; margin-top: 10px;"> <p>说明 在每个子元素中，可以使用 _label 更改系列名（例如 系列一 改为 模板 ），_icon 更改图标。</p> </div>

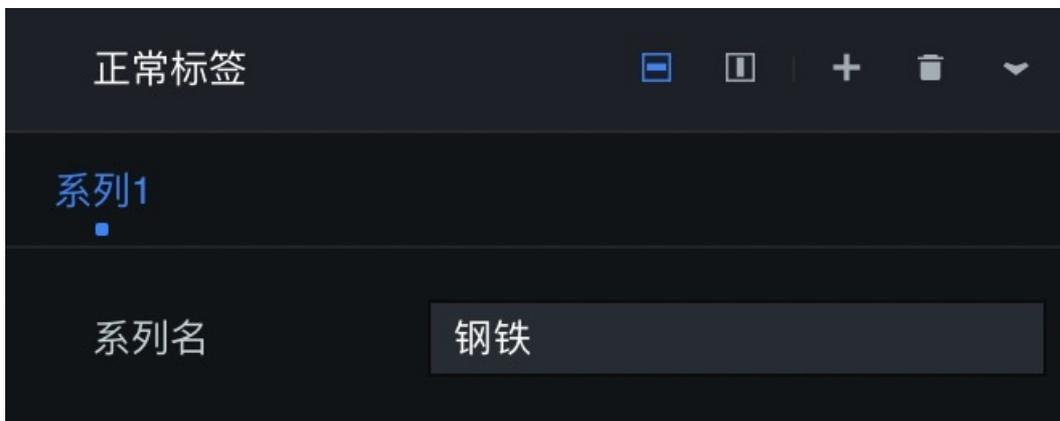
字段名	含义	类型	是否必选	备注
fold	是否默认展开	boolean	否	默认为 true 。

值说明

条件	数据类型	示例	默认值	备注
不涉及	array	<pre>[{ "name": "系列一" }, { "name": "系列二" }]</pre>	[]	在每个子元素中，可以使用 <code>_label</code> 设置独立的tab名，使用 <code>_icon</code> 增加图标。

配置示例

- 可增减（配置 `template` 动态增加模板内容）



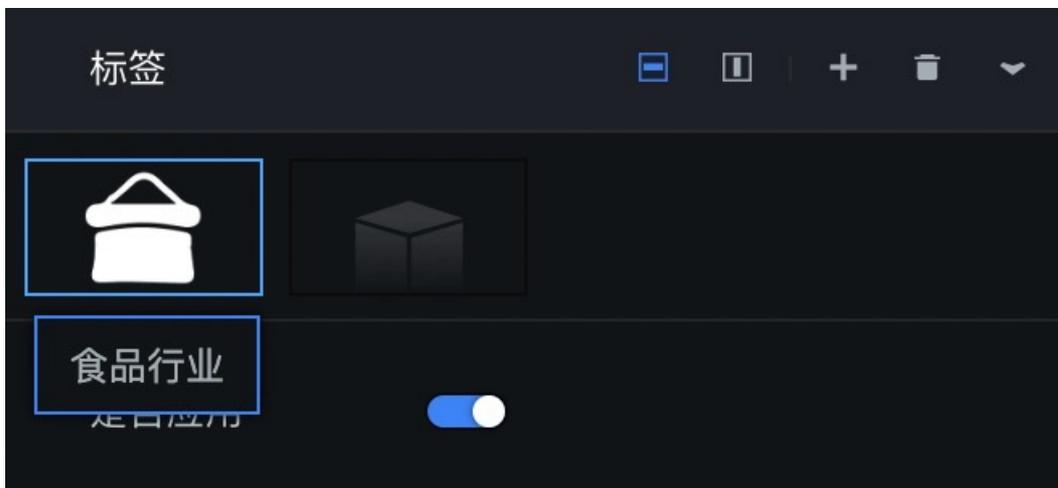
```
"tabs": {
  "name": "正常标签",
  "type": "tabs",
  "maxTabs": 10,
  "default": [
    {
      "seriesName": "钢铁"
    }
  ],
  "template": {
    "name": "系列<%= i + 1%>",
    "children": {
      "seriesName": {
        "type": "text",
        "name": "系列名"
      }
    }
  }
}
```

- 内容固定（不配置 `template`）



```
"tabs": {
  "name": "标签",
  "type": "tabs",
  "addable": false,
  "children": [
    {
      "name": "用户信息",
      "children": {
        "serieName": {
          "name": "姓名",
          "type": "text"
        }
      }
    },
    {
      "name": "是否应用",
      "children": {
        "switch": {
          "name": "开关",
          "type": "switch"
        }
      }
    }
  ]
}
```

- 使用独立的icon和label



此时需要通过值传入icon地址。

```
{
  "tabs": {
    "name": "标签",
    "type": "tabs",
    "default": [
      {
        "link": true,
        "_label": "食品行业",
        "_icon": "https://img.alicdn.com/tfs/TB1fck7voH1gK0jSZSyXXXt1pXa-201-200.png"
      }
    ],
    "template": {
      "name": "系列<%= i + 1%>",
      "children": {
        "link": {
          "type": "switch",
          "name": "Link"
        }
      }
    }
  }
}
```

2.4.33. menu

menu表示组件的配置项类型为菜单。菜单用来组织配置项的结构，目前支持一级菜单和二级菜单。

菜单样式

- 一级菜单



- 二级菜单



配置项说明

字段名	含义	类型	是否必选	备注
<code>name</code>	显示名	string	是	无。
<code>type</code>	类型	string	是	无。
<code>default</code>	默认值	object	否	不填时值为空。
<code>children</code>	菜单内容	object	否	默认为 <code>{}</code> 。必须包含 <code>mode</code> 和 <code>name</code> 字段。 <ul style="list-style-type: none"> <code>mode</code>：是否嵌套了二级菜单，<code>"single"</code> 表示仅有一级菜单，<code>"multiple"</code> 表示有二级菜单。 <code>name</code>：菜单名。

值说明

条件	数据类型	示例	默认值
----	------	----	-----

条件	数据类型	示例	默认值
不涉及	object	<pre> "options": { "menuA": { "switch": true }, "menuB": { "menuB1": { "text": "Hello" }, "menuB2": { "stepper": 2 } } } </pre>	{}

配置示例



```
"options": {
  "name": "菜单",
  "type": "menu",
  "children": {
    "menuA": {
      "name": "菜单A",
      "mode": "single",
      "children": {
        "switch": {
          "name": "开关",
          "type": "switch",
          "default": true
        }
      }
    },
    "menuB": {
      "name": "菜单B",
      "mode": "multiple",
      "children": {
        "menuB1": {
          "name": "菜单B1",
          "children": {
            "text": {
              "name": "文本",
              "type": "text",
              "default": "Hello"
            }
          }
        },
        "menuB2": {
          "name": "菜单B2",
          "children": {
            "stepper": {
              "name": "步进器",
              "type": "stepper",
              "default": 1
            }
          }
        }
      }
    }
  }
}
```

3.ECharts组件封装指南

您可在ECharts官网找到适合自己需求的图表，例如本教程的案例[柱状图动画延迟](#)。

抽离配置与数据

根据案例左侧代码栏中的配置，可以将一整个option抽离出配置与数据。案例的完整代码如下所示。

```
var xAxisData = [];  
var data1 = [];  
var data2 = [];  
for (var i = 0; i < 100; i++) {  
  xAxisData.push('类目' + i);  
  data1.push((Math.sin(i / 5) * (i / 5 - 10) + i / 6) * 5);  
  data2.push((Math.cos(i / 5) * (i / 5 - 10) + i / 6) * 5);  
}  
option = {  
  title: {  
    text: '柱状图动画延迟'  
  },  
  legend: {  
    data: ['bar', 'bar2'],  
    align: 'left'  
  },  
  toolbox: {  
    // y: 'bottom',  
    feature: {  
      magicType: {  
        type: ['stack', 'tiled']  
      },  
      dataView: {},  
      saveAsImage: {  
        pixelRatio: 2  
      }  
    }  
  },  
  tooltip: {},  
  xAxis: {  
    data: xAxisData,  
    silent: false,  
    splitLine: {  
      show: false  
    }  
  },  
  yAxis: {  
  },  
  series: [{  
    name: 'bar',  
    type: 'bar',  
    data: data1,  
    animationDelay: function (idx) {  
      return idx * 10;  
    }  
  }, {
```

```

    name: 'bar2',
    type: 'bar',
    data: data2,
    animationDelay: function (idx) {
      return idx * 10 + 100;
    }
  }],
  animationEasing: 'elasticOut',
  animationDelayUpdate: function (idx) {
    return idx * 5;
  }
};

```

其中含有数据项的配置，如下所示。

- `option.xAxis.data`
- `option.series[x].data`

为了转换为DataV能识别的细化到数据元的数据，您可进行如下操作。

```

[
  {
    x: 'option.xAxis.data的某一项值',
    y: 'option.series[x].data的某一项值',
    s: 'option.series[x].name'
  }
]

```

除上述三项配置属于数据外，其他都属于配置。

填写package.json

将梳理的配置和数据按照 [package.json 规范](#) 写入 `package.json`，需要注意以下几点。

- 可以删除自己组件不需要的配置。
- 若需要案例代码中没有的配置，可从 [ECharts配置项手册](#) 中获取。
- ECharts配置项种类繁多，大部分都已支持，但是不支持：
 - 配置项是函数。
 - ECharts自己的类型，例如 `echarts.dataTool.xxx`。
 - 目前也暂不支持一个配置项有多种类型，例如既可以是 `text` 类型，也可以是 `number` 类型。
- 配置结构与ECharts一致，若确实不能一致，需要在 `index.js` 中自己实现转换（所以也可以支持 `echarts.dataTool` 等 `echarts` 类型）。

可单击 [此处](#) 下载 `package.json` 完整示例文件。

示例文件中的ECharts转换后的部分 `package.json`，如下所示：

```

{
  "datav": {
    "cn_name": "柱状图",
    "icon": "",
    "protocol": 2,
    "type": [
      "column"
    ]
  }
}

```

```
    "regular_bar":
  ],
  "view": {
    "width": "600",
    "height": "200",
    "minWidth": "40",
    "minHeight": "20"
  },
  "apis": {
    "source": {
      "handler": "render",
      "description": "echarts动画延迟柱状图接口描述",
      "fields": {
        "x": {
          "description": "x坐标轴值"
        },
        "y": {
          "description": "y坐标轴值"
        },
        "s": {
          "description": "系列值",
          "optional": true
        }
      }
    }
  },
  "config": {
    "legend": {
      //...
    },
    "grid": {
      //...
    },
    "xAxis": {
      "name": "x轴",
      "type": "group",
      "children": {
        "show": {
          "name": "显示",
          "type": "boolean",
          "default": false
        },
        "offset": {
          "name": "位移",
          "type": "number",
          "default": 0
        },
        "type": {
          "name": "类型",
          "type": "text",
          "default": "category"
        }
      },
      "name": {
        "name": "名称",
        "type": "text".
```

```
    type: "color",
    "default": ""
  },
  "nameLocation": {
    "name": "名称位置",
    "type": "text",
    "default": "end"
  },
  "nameTextStyle": {
    "name": "名称样式",
    "type": "group",
    "children": {
      "color": {
        "name": "颜色",
        "type": "color",
        "default": "rgba(0,0,0,0)"
      },
      "fontStyle": {
        "name": "字体样式",
        "type": "text",
        "default": "normal"
      },
      "fontWeight": {
        "name": "字体粗细",
        "type": "text",
        "default": "normal"
      },
      "fontFamily": {
        "name": "字体",
        "type": "text",
        "default": "sans-serif"
      },
      "fontSize": {
        "name": "字号",
        "type": "number",
        "default": 10
      }
    }
  },
  "fold": true
},
"nameGap": {
  "name": "名称间隔",
  "type": "number",
  "default": 15
},
"nameRotate": {
  "name": "名称旋转",
  "type": "number",
  "default": null
},
"inverse": {
  "name": "反向",
  "type": "boolean",
  "default": false
},
},
```

```
"boundaryGap": {
  "name": "留白",
  "type": "boolean",
  "default": true
},
"min": {
  "name": "最小值",
  "type": "text",
  "default": "dataMin"
},
"max": {
  "name": "最大值",
  "type": "text",
  "default": "dataMax"
},
"scale": {
  "name": "自动伸缩",
  "type": "boolean",
  "default": false
},
"splitNumber": {
  "name": "分割段数",
  "type": "number",
  "default": 5
},
"minInterval": {
  "name": "最小间隔大小",
  "type": "number",
  "default": 0
},
"logBase": {
  "name": "对数轴底数",
  "type": "number",
  "default": 10
},
"silent": {
  "name": "静态",
  "type": "boolean",
  "default": false
},
"triggerEvent": {
  "name": "触发事件",
  "type": "boolean",
  "default": false
},
"axisLine": {
  "name": "轴线",
  "type": "group",
  "children": {
    "show": {
      "name": "显示",
      "type": "boolean",
      "default": false
    }
  }
},
```

```
    "onZero": {
      "name": "在零上",
      "type": "boolean",
      "default": true
    },
    "lineStyle": {
      "name": "轴线样式",
      "type": "group",
      "children": {
        "color": {
          "name": "颜色",
          "type": "multicolor",
          "default": {
            "style": "single",
            "value": "rgba(255,255,255,.8)"
          }
        },
        "width": {
          "name": "宽度",
          "type": "number",
          "default": 1
        },
        "type": {
          "name": "类型",
          "type": "text",
          "default": "solid"
        },
        "opacity": {
          "name": "透明度",
          "type": "number",
          "default": 1
        }
      },
      "fold": true
    },
    "fold": true
  },
  "axisTick": {
    "name": "刻度",
    "type": "group",
    "children": {
      "show": {
        "name": "显示",
        "type": "boolean",
        "default": false
      },
      "alignWithLabel": {
        "name": "和标签对齐",
        "type": "boolean",
        "default": false
      },
      "interval": {
        "name": "分割间隔",
```

```
        "type": "number",
        "default": 0
    },
    "inside": {
        "name": "内部",
        "type": "boolean",
        "default": false
    },
    "length": {
        "name": "长度",
        "type": "number",
        "default": 5
    },
    "lineStyle": {
        "name": "轴线样式",
        "type": "group",
        "children": {
            "color": {
                "name": "颜色",
                "type": "multicolor",
                "default": {
                    "style": "single",
                    "value": "rgba(255,255,255,.8)"
                }
            },
            "width": {
                "name": "宽度",
                "type": "number",
                "default": 1
            },
            "type": {
                "name": "类型",
                "type": "text",
                "default": "solid"
            },
            "opacity": {
                "name": "透明度",
                "type": "number",
                "default": 1
            }
        },
        "fold": true
    },
    "axisLabel": {
        "name": "刻度标签",
        "type": "group",
        "children": {
            "show": {
                "name": "显示",
                "type": "boolean",
                "default": true
            }
        },
        "fold": true
    }
}
```

```
    },
    "interval": {
      "name": "分割间隔",
      "type": "number",
      "default": 13
    },
    "inside": {
      "name": "内部",
      "type": "boolean",
      "default": false
    },
    "rotate": {
      "name": "旋转",
      "type": "number",
      "default": 0
    },
    "margin": {
      "name": "外间距",
      "type": "number",
      "default": 8
    },
    "showMinLabel": {
      "name": "显示最小标签",
      "type": "boolean",
      "default": true
    },
    "showMaxLabel": {
      "name": "显示最大标签",
      "type": "boolean",
      "default": true
    },
    "textStyle": {
      "name": "文本样式",
      "type": "group",
      "children": {
        "color": {
          "name": "颜色",
          "type": "color",
          "default": "rgba(255,255,255,.8)"
        },
        "fontStyle": {
          "name": "字体样式",
          "type": "text",
          "default": "normal"
        },
        "fontWeight": {
          "name": "字体粗细",
          "type": "text",
          "default": "normal"
        },
        "fontFamily": {
          "name": "字体",
          "type": "text",
          "default": "sans-serif"
        }
      }
    }
  }
}
```

```
    },
    "fontSize": {
      "name": "字号",
      "type": "number",
      "default": 10
    },
    "align": {
      "name": "对齐",
      "type": "select",
      "range": [
        {
          "name": "自动",
          "value": "auto"
        },
        {
          "name": "左对齐",
          "value": "left"
        },
        {
          "name": "居中对齐",
          "value": "center"
        },
        {
          "name": "右对齐",
          "value": "right"
        }
      ],
      "default": ""
    },
    "baseline": {
      "name": "基线",
      "type": "text",
      "default": ""
    }
  },
  "fold": true
},
"fold": true
},
"splitLine": {
  "name": "分割线",
  "type": "group",
  "children": {
    "show": {
      "name": "显示",
      "type": "boolean",
      "default": false
    },
    "interval": {
      "name": "分割间隔",
      "type": "number",
      "default": 0
    }
  },
  "lineStyle": {
```

```
    "name": "轴线样式",
    "type": "group",
    "children": {
      "width": {
        "name": "宽度",
        "type": "number",
        "default": 1
      },
      "type": {
        "name": "类型",
        "type": "text",
        "default": "solid"
      },
      "opacity": {
        "name": "透明度",
        "type": "number",
        "default": 1
      }
    },
    "fold": true
  },
  "fold": true
},
"splitArea": {
  "name": "分割区域",
  "type": "group",
  "children": {
    "interval": {
      "name": "分割间隔",
      "type": "number",
      "default": 0
    },
    "show": {
      "name": "显示",
      "type": "boolean",
      "default": false
    },
    "areaStyle": {
      "name": "区域样式",
      "type": "group",
      "children": {
        "opacity": {
          "name": "透明度",
          "type": "number",
          "default": 1
        }
      }
    }
  },
  "fold": true
}
},
"fold": true
},
"axisPointer": {
```

```
"name": "坐标轴指示器",
"type": "group",
"children": {
  "show": {
    "name": "显示",
    "type": "boolean",
    "default": true
  },
  "type": {
    "name": "类型",
    "type": "select",
    "default": "line",
    "range": [
      {
        "name": "直线指示器",
        "value": "line"
      },
      {
        "name": "阴影指示器",
        "value": "shadow"
      }
    ]
  },
  "snap": {
    "name": "自动吸附",
    "type": "boolean",
    "default": false
  },
  "value": {
    "name": "初始值",
    "type": "number",
    "default": null
  },
  "status": {
    "name": "状态",
    "type": "boolean",
    "default": false
  },
  "label": {
    "name": "标签",
    "type": "group",
    "children": {
      "show": {
        "name": "显示",
        "type": "boolean",
        "default": false
      },
      "precision": {
        "name": "小数精度",
        "type": "number",
        "default": "'auto'"
      },
      "margin": {
        "name": "外间距",
```

```
    "type": "boolean",
    "default": 3
  },
  "textStyle": {
    "name": "文本样式",
    "type": "group",
    "children": {
      "color": {
        "name": "颜色",
        "type": "color",
        "default": "#ffffff"
      },
      "fontStyle": {
        "name": "字体样式",
        "type": "text",
        "default": "normal"
      },
      "fontWeight": {
        "name": "字体粗细",
        "type": "text",
        "default": "normal"
      },
      "fontFamily": {
        "name": "字体",
        "type": "text",
        "default": "sans-serif"
      },
      "fontSize": {
        "name": "字号",
        "type": "number",
        "default": 10
      }
    }
  },
  "fold": true
},
"backgroundColor": {
  "name": "背景色",
  "type": "text",
  "default": "auto"
},
"borderColor": {
  "name": "边框色",
  "type": "text",
  "default": ""
},
"borderWidth": {
  "name": "边框粗细",
  "type": "text",
  "default": ""
}
},
"fold": true
},
"lineStyle": {
  "name": "线条样式",
  "type": "group",
  "children": {
    "color": {
      "name": "颜色",
      "type": "color",
      "default": "#000000"
    },
    "dash": {
      "name": "虚线",
      "type": "text",
      "default": ""
    },
    "width": {
      "name": "宽度",
      "type": "text",
      "default": ""
    }
  }
}
```

```
"name": "细线样式",
"type": "group",
"show": [
  [
    "type",
    "$eq",
    "line"
  ]
],
"children": {
  "color": {
    "name": "颜色",
    "type": "multicolor",
    "default": {
      "style": "single",
      "value": "rgba(0,0,0,0)"
    }
  },
  "width": {
    "name": "宽度",
    "type": "number",
    "default": 1
  },
  "type": {
    "name": "类型",
    "type": "text",
    "default": "solid"
  },
  "opacity": {
    "name": "透明度",
    "type": "number",
    "default": 1
  }
},
"fold": true
},
"shadowStyle": {
  "name": "阴影样式",
  "type": "group",
  "show": [
    [
      "type",
      "$eq",
      "shadow"
    ]
  ],
  "children": {
    "color": {
      "name": "颜色",
      "type": "multicolor",
      "default": {
        "style": "single",
        "value": "rgba(150,150,150,0.3)"
      }
    }
  }
}
```

```
    },
    "opacity": {
      "name": "透明度",
      "type": "number",
      "default": 1
    }
  },
  "fold": true
},
"handle": {
  "name": "拖拽手柄",
  "type": "group",
  "children": {
    "show": {
      "name": "显示",
      "type": "boolean",
      "default": false
    },
    "size": {
      "name": "大小",
      "type": "number",
      "default": 45
    },
    "margin": {
      "name": "外间距",
      "type": "number",
      "default": 50
    },
    "color": {
      "name": "颜色",
      "type": "text",
      "default": "#333"
    },
    "throttle": {
      "name": "刷新频率",
      "type": "number",
      "default": 40
    }
  }
},
"fold": true
}
},
"fold": true
},
"yAxis": {
  //...
},
"tooltip": {
  //...
},
"series": {
  "name": "系列",
```

```
"type": "array",
"fold": true,
"default": [
  {
    "name": "bar",
    "type": "bar",
    "legendHoverLink": true,
    "coordinateSystem": "cartesian2d",
    "label": {
      "normal": {
        "show": false,
        "textStyle": {
          "color": "#000",
          "fontStyle": "normal",
          "fontWeight": "normal",
          "fontFamily": "sans-serif",
          "fontSize": 10
        }
      },
      "emphasis": {
        "show": false,
        "textStyle": {
          "color": "#000",
          "fontStyle": "normal",
          "fontWeight": "normal",
          "fontFamily": "sans-serif",
          "fontSize": 10
        }
      }
    },
    "itemStyle": {
      "normal": {
        "color": {
          "style": "single",
          "value": "#00c2ff"
        },
        "borderColor": {
          "style": "single",
          "value": "#000"
        },
        "borderWidth": 0,
        "borderType": "solid",
        "barBorderRadius": 0,
        "opacity": 1
      },
      "emphasis": {
        "color": {
          "style": "single",
          "value": "#00c2ff"
        },
        "borderColor": {
          "style": "single",
          "value": "#000"
        }
      }
    }
  }
]
```

```
        "borderWidth": 0,
        "borderType": "solid",
        "opacity": 1
    }
},
"stack": "",
"barWidth": "50%",
"barMinHeight": 0,
"barGap": "30%",
"barCategoryGap": "20%",
"silent": false
},
{
    "name": "bar2",
    "type": "bar",
    "legendHoverLink": true,
    "coordinateSystem": "cartesian2d",
    "label": {
        "normal": {
            "show": false,
            "textStyle": {
                "color": "#000",
                "fontStyle": "normal",
                "fontWeight": "normal",
                "fontFamily": "sans-serif",
                "fontSize": 10
            }
        },
        "emphasis": {
            "show": false,
            "textStyle": {
                "color": "#000",
                "fontStyle": "normal",
                "fontWeight": "normal",
                "fontFamily": "sans-serif",
                "fontSize": 10
            }
        }
    },
    "itemStyle": {
        "normal": {
            "color": {
                "style": "single",
                "value": "#5bffb0"
            },
            "borderColor": {
                "style": "single",
                "value": "#000"
            },
            "borderWidth": 0,
            "borderType": "solid",
            "barBorderRadius": 0,
            "opacity": 1
        },
    },
}
```

```
    "emphasis": {
      "color": {
        "style": "single",
        "value": "#5bffb0"
      },
      "borderColor": {
        "style": "single",
        "value": "#000"
      },
      "borderWidth": 0,
      "borderType": "solid",
      "opacity": 1
    }
  },
  "stack": "",
  "barWidth": "50%",
  "barMinHeight": 0,
  "barGap": "30%",
  "barCategoryGap": "20%",
  "silent": false
}
],
"child": {
  "type": "object",
  "name": "系列<%= i+1 %>",
  "child": {
    "name": {
      "name": "名称",
      "type": "text",
      "default": ""
    },
    "legendHoverLink": {
      "name": "图例联动高亮",
      "type": "boolean",
      "default": true
    },
    "coordinateSystem": {
      "name": "坐标系",
      "type": "text",
      "default": "cartesian2d"
    },
    "label": {
      "name": "标签",
      "type": "group",
      "children": {
        "normal": {
          "name": "普通项",
          "type": "group",
          "children": {
            "show": {
              "name": "显示",
              "type": "boolean",
              "default": false
            }
          }
        }
      }
    }
  }
}
```

```
    "textStyle": {
      "name": "文本样式",
      "type": "group",
      "children": {
        "color": {
          "name": "颜色",
          "type": "color",
          "default": "#000"
        },
        "fontStyle": {
          "name": "字体样式",
          "type": "text",
          "default": "normal"
        },
        "fontWeight": {
          "name": "字体粗细",
          "type": "text",
          "default": "normal"
        },
        "fontFamily": {
          "name": "字体",
          "type": "text",
          "default": "sans-serif"
        },
        "fontSize": {
          "name": "字号",
          "type": "number",
          "default": 10
        }
      },
      "fold": true
    },
    "fold": true
  },
  "emphasis": {
    "name": "重点项",
    "type": "group",
    "children": {
      "show": {
        "name": "显示",
        "type": "boolean",
        "default": false
      },
      "textStyle": {
        "name": "文本样式",
        "type": "group",
        "children": {
          "color": {
            "name": "颜色",
            "type": "color",
            "default": "#000"
          },
          "fontStyle": {
            "name": "字体样式"
```

```
        "name": "字体样式",
        "type": "text",
        "default": "normal"
    },
    "fontWeight": {
        "name": "字体粗细",
        "type": "text",
        "default": "normal"
    },
    "fontFamily": {
        "name": "字体",
        "type": "text",
        "default": "sans-serif"
    },
    "fontSize": {
        "name": "字号",
        "type": "number",
        "default": 10
    }
},
"fold": true
}
},
"fold": true
},
"itemStyle": {
    "name": "元素样式",
    "type": "group",
    "children": {
        "normal": {
            "name": "普通项",
            "type": "group",
            "children": {
                "color": {
                    "name": "颜色",
                    "type": "multicolor",
                    "default": {
                        "style": "single",
                        "value": "rgba(0,0,0,0)"
                    }
                },
                "borderColor": {
                    "name": "边框色",
                    "type": "multicolor",
                    "default": {
                        "style": "single",
                        "value": "#000"
                    }
                },
                "borderWidth": {
                    "name": "边框粗细",
                    "type": "number"
                }
            }
        }
    }
}
```

```
      "type": "number",  
      "default": 0  
    },  
    },  
    "borderType": {  
      "name": "边框样式",  
      "type": "text",  
      "default": "solid"  
    },  
    },  
    "barBorderRadius": {  
      "name": "柱状图圆角",  
      "type": "number",  
      "default": 0  
    },  
    },  
    "opacity": {  
      "name": "透明度",  
      "type": "number",  
      "default": 1  
    }  
  },  
  "fold": true  
},  
"emphasis": {  
  "name": "重点项",  
  "type": "group",  
  "children": {  
    "color": {  
      "name": "颜色",  
      "type": "multicolor",  
      "default": {  
        "style": "single",  
        "value": "rgba(0,0,0,0)"  
      }  
    },  
    },  
    "borderColor": {  
      "name": "边框色",  
      "type": "multicolor",  
      "default": {  
        "style": "single",  
        "value": "#000"  
      }  
    },  
    },  
    "borderWidth": {  
      "name": "边框粗细",  
      "type": "number",  
      "default": 0  
    },  
    },  
    "borderType": {  
      "name": "边框样式",  
      "type": "text",  
      "default": "solid"  
    },  
    },  
    "opacity": {  
      "name": "透明度",  
      "type": "number",  
      "default": 1  
    }  
  }  
}
```

```
        }
      },
      "fold": true
    }
  },
  "fold": true
},
"stack": {
  "name": "堆叠图",
  "type": "text",
  "default": ""
},
"barWidth": {
  "name": "柱子宽度",
  "type": "text",
  "default": "50%"
},
"barMinHeight": {
  "name": "柱状图最小高度",
  "type": "number",
  "default": 0
},
"barGap": {
  "name": "柱状图间隔",
  "type": "text",
  "default": "30%"
},
"barCategoryGap": {
  "name": "类目间柱子间距",
  "type": "text",
  "default": "20%"
},
"silent": {
  "name": "静态",
  "type": "boolean",
  "default": false
}
}
}
},
"animation": {
  "name": "动画",
  "type": "boolean",
  "default": true
},
"animationThreshold": {
  "name": "动画阈值",
  "type": "number",
  "default": 2000
},
"animationDuration": {
  "name": "动画时长",
  "type": "number",
  "default": 1000
}
```

```
    },
    "animationEasing": {
      "name": "缓动效果",
      "type": "text",
      "default": "elasticOut"
    }
  },
  "api_data": {
    "source": [
      {
        "x": "类目0",
        "y": 0,
        "s": "bar"
      },
      {
        "x": "类目0",
        "y": -50,
        "s": "bar2"
      },
      {
        "x": "类目1",
        "y": -8.901463875624668,
        "s": "bar"
      },
      {
        "x": "类目1",
        "y": -47.18992898088751,
        "s": "bar2"
      },
      {
        "x": "类目2",
        "y": -17.025413764148556,
        "s": "bar"
      },
      {
        "x": "类目2",
        "y": -42.54426104547181,
        "s": "bar2"
      },
      {
        "x": "类目3",
        "y": -24.038196249566663,
        "s": "bar"
      },
      {
        "x": "类目3",
        "y": -36.290773900754886,
        "s": "bar2"
      },
      {
        "x": "类目4",
        "y": -29.66504684804471,
        "s": "bar"
      }
    ],
  },
}
```

```
{
  "x": "类目4",
  "y": -28.71517529663627,
  "s": "bar2"
},
{
  "x": "类目5",
  "y": -33.699527649688676,
  "s": "bar"
},
{
  "x": "类目5",
  "y": -20.146937097399626,
  "s": "bar2"
},
{
  "x": "类目6",
  "y": -36.00971978255796,
  "s": "bar"
},
{
  "x": "类目6",
  "y": -10.94374119697364,
  "s": "bar2"
},
{
  "x": "类目7",
  "y": -36.541005056170455,
  "s": "bar"
},
{
  "x": "类目7",
  "y": -1.4752538113770308,
  "s": "bar2"
},
{
  "x": "类目8",
  "y": -35.31542466107655,
  "s": "bar"
},
{
  "x": "类目8",
  "y": 7.893046603320797,
  "s": "bar2"
},
{
  "x": "类目9",
  "y": -32.427752866005996,
  "s": "bar"
},
{
  "x": "类目9",
  "y": 16.81528588241657,
  "s": "bar2"
},
}
```

```
},
{
  "x": "类目10",
  "y": -28.038563739693934,
  "s": "bar"
},
{
  "x": "类目10",
  "y": 24.979206795219028,
  "s": "bar2"
},
{
  "x": "类目11",
  "y": -22.364693082297347,
  "s": "bar"
},
{
  "x": "类目11",
  "y": 32.11821023962515,
  "s": "bar2"
},
{
  "x": "类目12",
  "y": -15.667600860943732,
  "s": "bar"
},
{
  "x": "类目12",
  "y": 38.02096119056733,
  "s": "bar2"
},
{
  "x": "类目13",
  "y": -8.240217424060843,
  "s": "bar"
},
{
  "x": "类目13",
  "y": 42.53821720798438,
  "s": "bar2"
},
{
  "x": "类目14",
  "y": -0.3929067389459173,
  "s": "bar"
},
{
  "x": "类目14",
  "y": 45.58667093073836,
  "s": "bar2"
},
{
  "x": "类目15",
  "y": 7.560799717904647,
  "s": "bar"
}
```

```
    },
    {
      "x": "类目15",
      "y": 47.14973738101559,
      "s": "bar2"
    },
    {
      "x": "类目16",
      "y": 15.318054209871054,
      "s": "bar"
    },
    {
      "x": "类目16",
      "y": 47.275355710354944,
      "s": "bar2"
    },
    ...
  ]
}
```

编写index.js

参考[index.js规范文档](#)，按照以下方法编写 *index.js* 文件：

1. 在初始化方法中，执行 *EChart.init*。
2. 在渲染方法中，执行 *chart.setOption*。
3. 在缩放方法中，执行 *chart.resize*。
4. 在清空方法中，执行 *chart.clear*。
5. 在销毁方法中，执行 *chart.dispose*。

可单击[此处](#)下载 *index.js* 完整示例文件。

4. 自定义组件开发常见问题

哪里有组件的案例？

命令行安装datav-cli，使用datav init命令下载安装模版组件，详情请参见[安装Datav开发工具套件](#)。

为什么我的组件发布不成功？

组件发布不成功有以下两种原因，您可针对自身情况进行处理：

- 组件没有发布权限。

组件名（`package.json`中的`name`字段）前未加组件包名，无法发布组件，正确组件名格式为 `@namespace/xxx`（`@namespace` 为组件包名）。

- 组件审核未通过或正在审核中。

若您的组件长时间未审核通过，请联系阿里云技术支持。

如何写数据事件，触发事件让别的组件监听？

1. `package.json`中 `datav` 字段下按正确格式定义好 `events` 事件名。
2. 在`index.js`中使用 `this.emit(event_name, value)` 方法发送事件名及参数。

 说明 参数value必须为一个对象，而不是基础类型值。

3. 其他的组件在数据页使用冒号加参数名，即可调用该参数对应的数据。

如何写一个ECharts组件？

详情请参见[ECharts组件封装指南](#)。

为什么本地的图片上传后不成功？

若需要将本地静态文件作为资源路径，需在组件根目录下新建`resources`文件夹，将图片等资源放至该目录下，调用路径，例如`./resources/xxx.png`。

为什么我的程序报找不到各种包？

如果执行`npm install`命令后仍有报错，可将`datav-cli`工具彻底删除后，重新安装。

为什么我的组件包还没审核通过？

如果您的组件长时间没有审核通过，请联系阿里云技术支持。

为什么我的组件突然没有发布权限？

如果您使用的是Datav开发者体验版，并且在两个月内未上传任何组件或教程，您的权限会被收回，请重新申请权限或联系阿里云技术支持。

为什么我的组件在本地使用正常，在产品中使用时报错？

可能有以下几方面原因：

- 组件不支持 `data` 为 `null`。
- 组件代码修改了jquery，例如重新赋值jquery。

- 组件代码修改了window对象。
- 当添加多个组件时修改配置不生效，需检查组件代码是否有CSS文件。
- 操作了容器外的DOM结构。
- 使用jquery操作DOM结构时，禁止 `$('.classname')`，必须从container容器中查找。

为什么在编辑页中找不到我的组件？

您可以在顶部导航栏的我的组件页面下查看您的组件。

为什么我的组件被购买后，购买者无法找到？

因为组件分类的设置不对，详情请参见[type字段](#)。

桌面版开发工具登录时，使用什么用户名？

Dat aV控制台右上角显示的阿里云账号用户名，详情请参见[图形界面开发工具使用说明](#)。

5. 图形界面开发工具使用说明

本文介绍桌面版开发工具的使用方法，包括下载和安装、登录、工作区管理和组件管理，帮助您快速使用桌面版开发工具开发组件，并进行发布。

下载并安装桌面版开发工具

1. 登录DataV控制台。
2. 切换至我的组件页签，并在左侧导航栏中单击我的组件包。
3. 在组件包管理页面，单击下载开发者工具。



4. 在选择您需要下载的版本对话框中，选择一个版本，单击下载。
系统提供MAC、WIN、Terminal三种版本的工具供您选择：
 - o **MAC**：适用于MacOS操作系统。
 - o **WIN**：适用于Windows操作系统。
 - o **Terminal**：适用于Linux操作系统，单击下载后会跳转至开发指南的[组件开发快速入门](#)文档，请参考文档安装开发工具套件。
5. (MAC和WIN版本) 下载完成后，双击应用进行安装。

登录桌面版开发工具

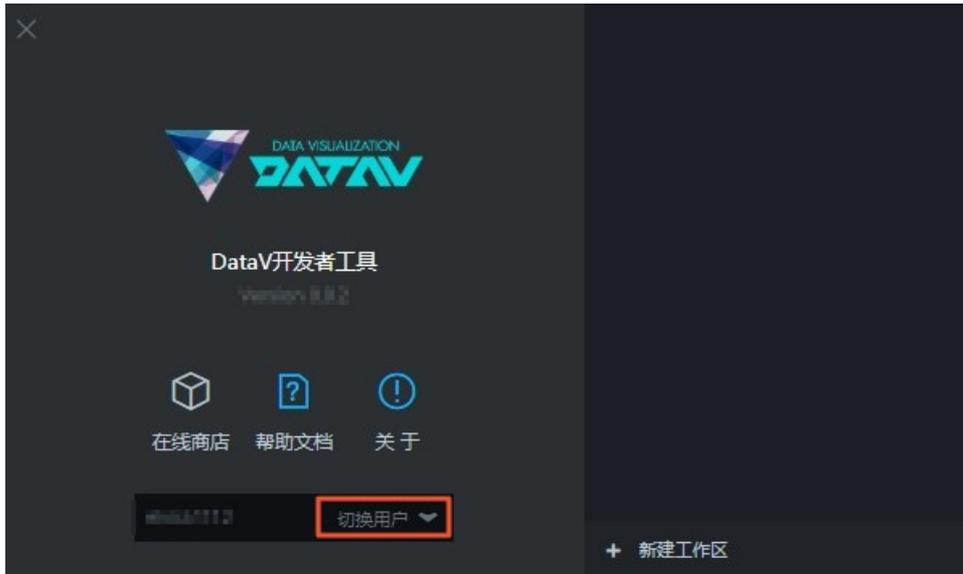
1. 安装完成后，双击打开工具，填写登录信息进行登录，也可以选择不登录离线开发。



 **注意** 离线开发将无法使用发布功能，请谨慎使用。

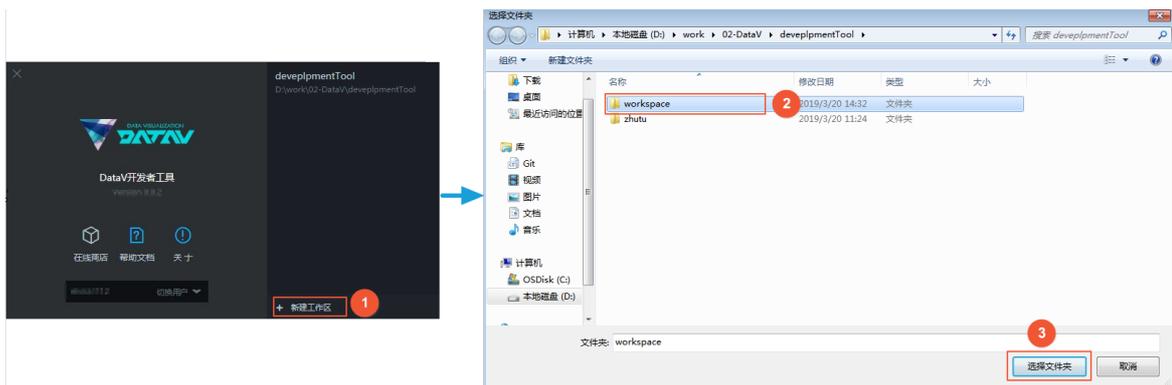
参数	描述
用户名	登录DataV所使用的阿里云账号（即主账号）。
令牌	开发者识别码。在DataV控制台我的组件页签，单击我的组件包，复制开发者识别码，粘贴到输入框中。

2. 登录成功后，可单击切换用户回到登录页面。



新建工作区

1. 在开发者工具首页，单击右下侧的+新建工作区。
2. 选择一个本地文件夹，作为组件包管理文件夹。
3. 单击选择文件夹，即可新建一个工作区。

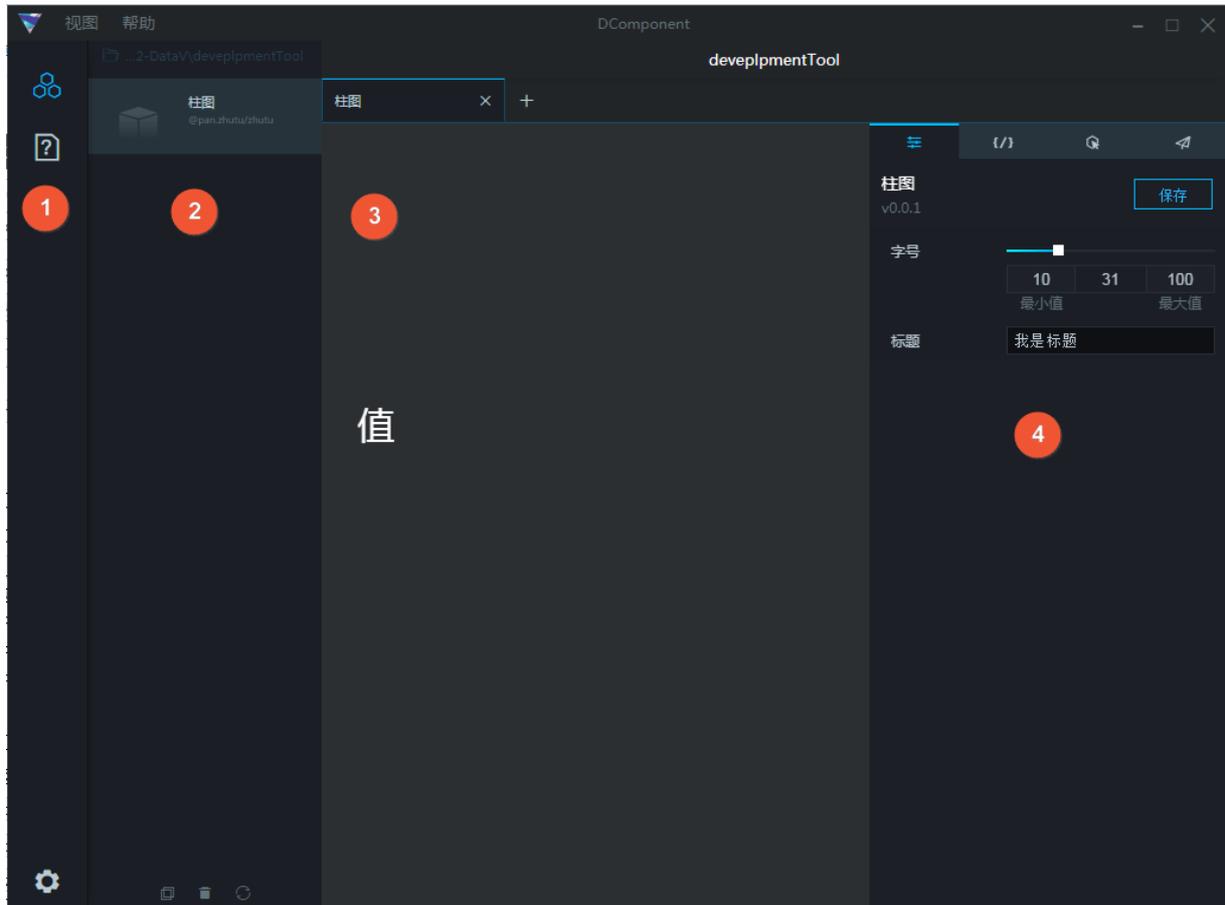


说明

- 工作区只是一个包含许多组件的文件夹，可以与线上的组件包同名，也可以不同名。
- 如果已存在包含组件的文件夹，可以选择打开该文件夹，工具会自动识别该文件夹中的DataV组件。

工作区概览

工作区创建成功后，会显示在页面右侧，单击工作区名称即可进入工作区页面。工作区页面包含以下四部分：



序号	区域	描述
1	侧边栏	包含了隐藏组件栏、查看帮助文档、退出登录等功能。
2	组件栏	显示组件列表、管理组件。 <ul style="list-style-type: none"> 显示组件列表。 包含了复制组件、重命名组件、删除组件、查看组件详细信息、刷新工作区等功能。
3	中心画布区	<ul style="list-style-type: none"> 中心画布区是用来展现组件，实时观测组件变化的区域。 所有右侧工具栏的配置、数据修改都会实时展示在中心画布的组件上。 组件的白框代表了组件的容器范围大小，每个方向上的白框都可以缩放来测试组件在任意方向缩放的表现。

序号	区域	描述
4	组件配置栏	分为配置、数据、交互和发布4个面板。 <ul style="list-style-type: none">● 配置：配置页面包含了组件可变动的一些配置项，进行配置操作时，改动会立即生效。例如在此页面拖拽字号滑动条，组件中的文字字号立即随之变化。单击右上角的保存，可以保存当前修改的配置，并将当前配置作为此组件的默认配置。● 数据：数据页面包含了组件的数据接口配置，数据页的数据一旦改动，组件都会进行相应的改动。单击右上的保存，可以保存当前修改的数据，并将当前数据作为此组件的默认数据。● 交互：交互页面包含了组件的交互说明。● 发布：发布页面包含了组件的基础属性、类型、图标和发布版本等配置，单击右上角的发布，即可发布组件。

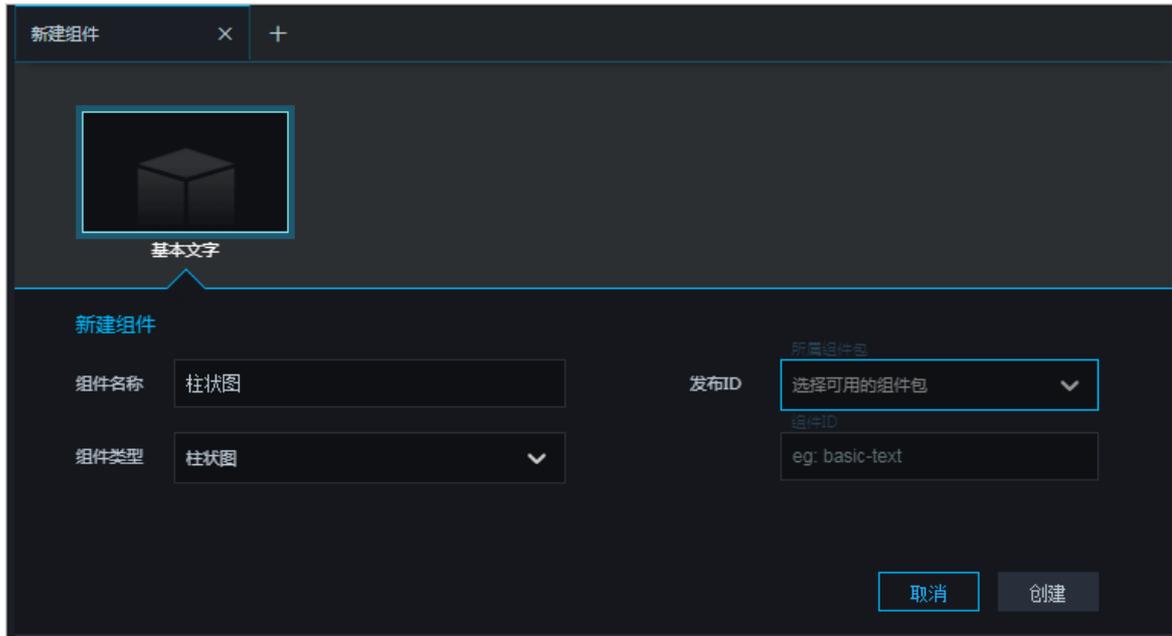
删除工作区

在工作区列表中，鼠标移至某个工作区上，单击右上角的X号，即可删除该工作区。



新建组件

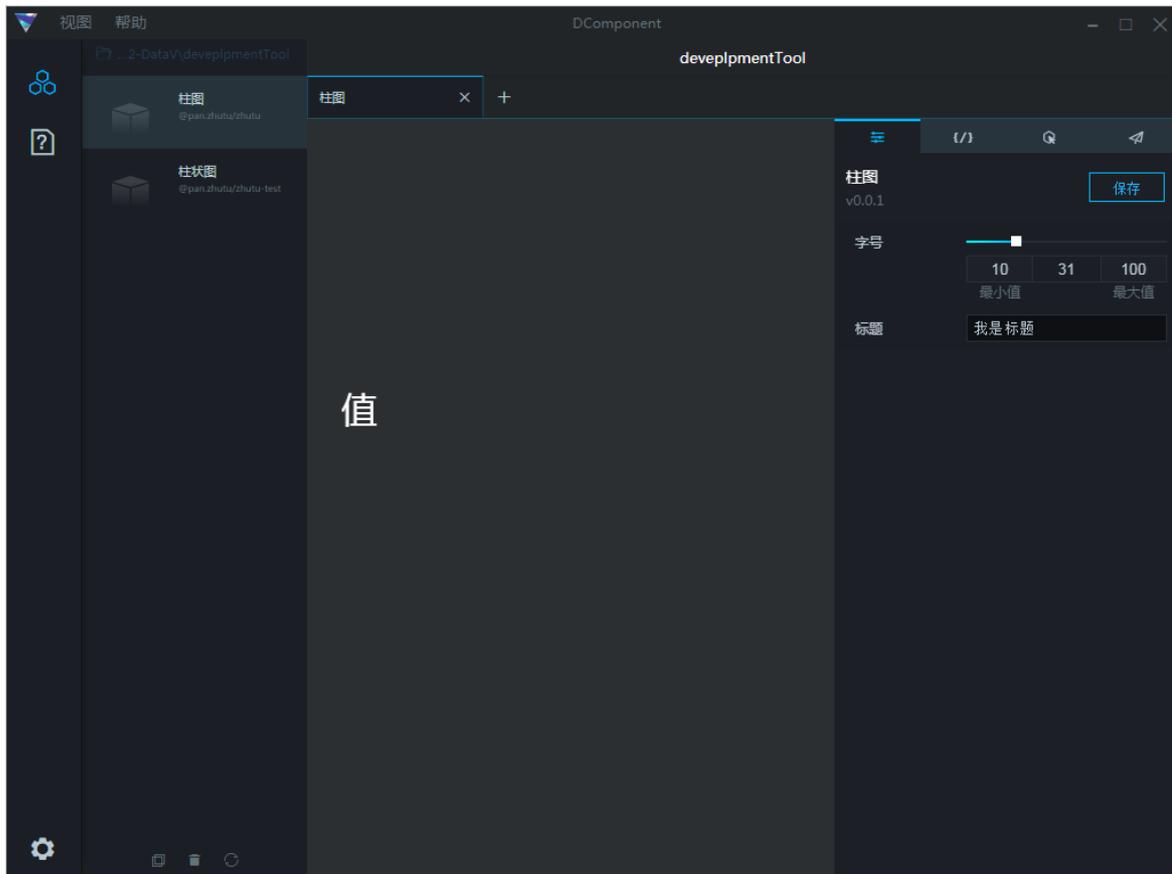
1. 单击工作区页面右侧的**+新建组件**，或者使用**Ctrl + N**快捷键，新建一个组件。
组件创建成功后，在**新建组件**页面出现系统默认的基本文字组件。
2. 单击**基本文字**组件，按照以下说明配置组件相关信息。



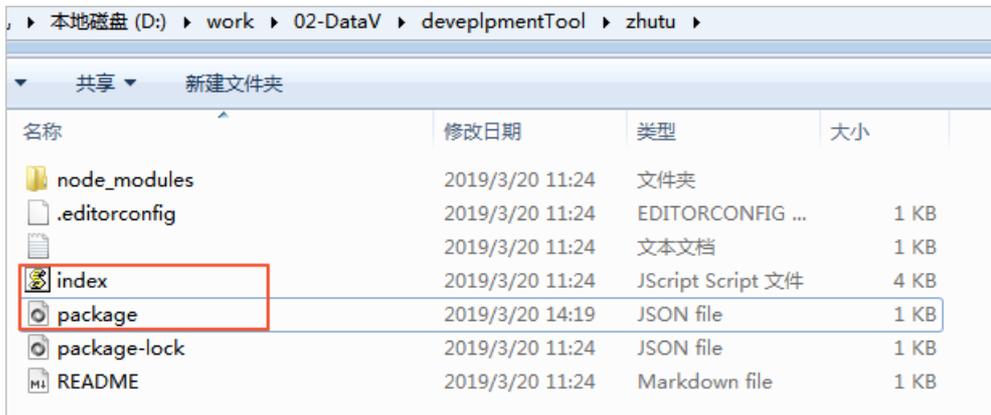
参数	描述
组件名称	输入组件名称。
组件类型	单击下拉框选择组件类型。
发布ID	单击下拉框选择可用的组件包，如果没有可用的组件包，可单击+ 自定义，跳转到DataV控制台上创建组件包，具体方法请参见 管理组件包 。
组件ID	输入组件ID，例如basic-text。

3. 信息填写完成后，单击**创建**即可完成组件的创建。

组件创建成功后，即可在中心画布区中查看组件，并在右侧的配置栏对组件进行配置。



组件创建成功后，在本地的组件包文件夹下会生成 *index.js* 和 *package.json* 文件，参考 [文档结构](#) 编辑这两个文件，可自定义组件的配置项。



预览组件

新建组件后，可单击页面左侧组件列表中的某个组件，在中心画布区查看该组件的样式。

复制/删除/重命名/查看组件详情

右键单击组件，选择制作副本/删除/重命名组件/显示属性，即可复制/删除/重命名/查看组件详情。

您也可以选择某个组件，单击组件栏底部的复制组件/删除组件图标，完成组件的复制/删除。



刷新工作区

单击组件栏底部的刷新工作区图标（），即可完成工作区的刷新。当您修改了组件的配置项后，可通过此功能显示最新的组件配置项。