

# Mobile Org Tree

## Introduction

移动端树形数据选择方案，包括组织机构、角色以及人员等分类，支持单选、多选、关键字段自定义以及多种事件及插槽等，适用于大部分组织选人场景。

- 1. 支持懒加载回调，点击获取当前组织下的子组织及人员等
- 2. 支持单选、多选、必选等条件限制
- 3. 支持自定义子节点、key 值、头像等字段
- 4. 多事件绑定类型，实现全生命周期全监测
- 5. 多部位插槽，支持默认样式以及快速实现定制化

## Modules



## Instructions



Links

- [Online Demo](#)
- [Online Docs](#)
- [Download](#)

Tree Props

prop	type	options	default	description
data	Array	—	[]	渲染数据源
label	String	—	name	展示字段名称
nodeKey	String	—	id	数据 key(如: id)
children	String	—	children	子节点的字段名称
icon	String	—	avatar	头像的字段名称
isShowClearBtn	Boolean	—	true	是否显示搜索框右侧的清除按钮
isShowIcon	Boolean	—	false	是否显示图标
defaultIcon	String	—	—	默认图标，本地图片引用方式： <code>:defaultIcon="require('@assets/demo/icon.jpg')"</code>
isMultiple	Boolean	—	true	是否多选
isSelectRequired	Boolean	—	true	是否必选
selectedList	Array	—	[]	已经选择的数据，可用作回显
slideDistance	Number	—	100	滑动手时触发距离

prop	type	options	default	description
searchPlaceholder	String	—	搜索	搜索框提示占位符文字
defaultShowType	String	org/role	org	默认展示类型
orgText	String	—	组织	切换按钮 org 文字
roleText	String	—	角色	切换按钮 role 文字
submitText	String	—	提交	提交按钮文字
cancelText	String	—	取消	取消按钮文字

## Tree Events

name	description	parameter
on-select	点击选项时触发，返回当前点击的选项信息	item
on-submit	提交时触发，返回所有已选项	selectedItems
on-cancel	点击取消按钮时候触发，可用来关闭组件等	—
on-search	搜索框输入时候触发	searchKey
on-expand	点击下级时候触发	currentNode
on-nav	点击面包屑导航时触发	currentNode
on-clear	点击清楚搜索框时候触发	—
on-slide	滑动手势触发的事件	—
on-switch-show-type	点击切换显示类型按钮触发	showType

## Tree Slots

name	description	parameter
switch-show-type-btn	切换显示类型按钮插槽	showType
content-area	中间内容区	renderData
empty-tips	无数据时提示信息	—
selected-list	已经选择项的集合展示区	selectedItems
result-area	底部操作区	selectedItems

## Usage

1. 将下载的组件 `MobileTree.vue` 复制到项目组件目录中，如：

```
MobileOrgTree
├── src
│   ├── App.vue
│   └── components
│       └── MobileTree
│           └── MobileTree.vue
```

```
|   └─ main.js  
└─ vue.config.js
```

## 2. 在需要用到该组件处引入

```
import MobileTree from "@components/MobileTree/MobileTree";
```

## 3. 注册组件

```
<script>  
  export default {  
    components: {  
      MobileTree,  
    },  
  };  
</script>
```

## 4. 启用组件

```
<MobileTree  
  :data="data"  
  label="name"  
  children="children"  
  icon="avatar"  
  :isMultiple="true"  
  :selectedList="selectedList"  
  @on-submit="handleSubmit"  
  @on-search="handleSearch"  
  @on-expand="handleExpand"  
  @on-nav="handleNav"  
  @on-clear="clearSearchKey"  
  @on-switch-show-type="handleSwitchShowType"  
>  
  <!-- 此处示例插槽只做展示，非必需，如无特殊需求，不建议使用。 -->  
  <!-- 切换显示类型按钮插槽 -->  
  <template slot="switch-show-type-btn" slot-scope="scope">  
    通过 scope.showType 来获取切换的显示类型  
  </template>  
  
  <!-- 内容区插槽 -->  
  <template slot="content-area" slot-scope="scope">  
    通过 scope.renderData 来获取显示区数据  
  </template>  
  
  <!-- 无数据提示信息插槽 -->  
  <template slot="empty-tips">  
    暂无数据  
  </template>  
  
  <!-- 底部已选项展示区插槽 -->  
  <template slot="selected-list" slot-scope="scope">  
    通过 scope.selectedItems 来获取已选项数据
```

```
</template>

<!-- 底部操作区插槽 -->
<template slot="result-area" slot-scope="scope">
  通过 scope.selectedItems 来获取已选项数据
</template>
</MobileTree>
```

## Function realization

### 数据渲染

将 `data` 绑定对应数据即可

```
// 需将数据转换为以下结构, name 和 id 可通过 prop 属性进行自定义
[
  {
    name: "赵启",
    id: "8ef2eo26w5181i51qd5",
    // ...
  },
  {
    name: "王刚",
    id: "k215tg1eqwh56yil15w",
    // ...
  },
  // 其他
];
```

### 获取下级数据

1. 通过 `on-expand` 事件来获取当前级信息
2. 通过当前级 `id` 或其他标识来请求下级信息并将返回值赋值给 `data`

### 单选/多选

通过给 `isMultiple` 绑定 `true` 或 `false` 来实现单选/多选

### 提交数据

通过 `on-submit` 绑定的事件来获取已选项信息

### 搜索数据

通过 `on-search` 事件来获取搜索关键字并作出对应操作

### 清空搜索关键字

1. 通过 `on-clear` 事件来清空搜索关键字
2. 重新获取数据并赋值给 `data`

### 切换显示类型

1. 通过 `on-switch-show-type` 事件来切换显示类型

2. 根据显示类型获取对应的数据
3. 将获取的数据赋值给 `data`

## 点击导航

1. 通过 `on-nav` 事件获取当前点击的导航信息
2. 根据导航信息来获取对应的数据
3. 将获取的数据赋值给 `data`
4. 接收参数为 `-1` 时，表示当前点击的为 `全部`