链滴

vue-bus: 一个 Vue.js 事件中心插件

作者：mingyu
原文链接：https://ld246.com/article/1478691804893
来源网站：链滴
许可协议：署名-相同方式共享 4.0 国际 (CC BY-SA 4.0)

vue-bus

一个 Vue.js 事件中心插件，同时支持 Vue 1.0 和 2.0

原因

Vue 2.0 重新梳理了事件系统，因为基于组件树结构的事件流方式实在是让人难以理解，并且在组件构扩展的过程中会变得越来越脆弱。虽然依然保留了父子组件间的事件流，但有诸多限制，比如不支跨多层父子组件通信，也没有解决兄弟组件间的通信问题。

Vue 推荐使用一个全局事件中心来分发和管理应用内的所有事件，详见文档。这是一个最佳实践，同适用于 Vue 1.0 和 2.0。你当然可以声明一个全局变量来使用事件中心，但你如果在使用 webpack 类的模块系统，这显然不合适。每次使用都手动 import 进来也很不方便，所以就有了这个插件：vue bus

vue-bus 提供了一个全局事件中心，并将其注入每一个组件，你可以像使用内置事件流一样方便的使全局事件。

安装

```
$ npm install vue-bus
```

如果在一个模块化工程中使用它，必须要通过 Vue.use() 明确地安装 vue-bus：

```
import Vue from 'vue';
import VueBus from 'vue-bus';

Vue.use(VueBus);
```

如果使用全局的 script 标签，则无须如此（手动安装）。

使用

监听事件和清除监听

```
// ...
created() {
  this.$bus.on('add-todo', this.addTodo);
  this.$bus.once('once', () => console.log('这个监听器只会触发一次'));
},
beforeDestroy() {
  this.$bus.off('add-todo', this.addTodo);
},
methods: {
  addTodo(newTodo) {
    this.todos.push(newTodo);
  }
}
```

触发事件

```
// ...
methods: {
  addTodo() {
    this.$bus.emit('add-todo', { text: this.newTodoText });
    this.$bus.emit('once');
    this.newTodoText = '';
  }
}
```

注意：$bus.on $bus.once $bus.off $bus.emit 只是 $bus.$on $bus.$once $bus.$off $bus.$emit 的别。 详见 API。

项目地址

GitHub，喜欢的话给个 Star 吧 :P