

前端路由原理

稍微复杂一点的 SPA，都需要路由。Vue-router 是 Vue 全家桶标配之一。

Vue-router 路由模式有：**hash** 和 **H5 history**。

1. hash

- hash 变化会触发网页跳转，即浏览器的前进后退，但不会使浏览器刷新。
- 浏览器的前进后退也会改变 hash。
- hash 永远不会提交到 server 端。

// 监听 hash 变化。hash 变化：可以通过 js 改变 hash，可以手动输入地址改变，可以通过浏览器前进后退改变

```
window.onhashchange = function (event) {  
  console.log("event: ", event);  
  console.log("old url: ", event.oldURL); // 获取原地址  
  console.log("new url: ", event.newURL); // 获取新地址  
};
```

2. History

H5 history 需要服务端配合，因为单页应用只有一个页面，服务端要实现客户端无论访问什么地址都要返回同一个页面（如：index.html）。

H5 history 是使用 url 规范的路由，但跳转时不刷新页面。通过 **history.pushState** 和 **window.onpopstate** 来实现。

pushState() 方法不会触发页面刷新，只是导致 **History** 对象发生变化，地址栏会有反应。

History.pushState() 接受三个参数：

- **state**：一个与添加的记录相关联的状态对象，主要用于 **popstate** 事件。该事件触发时，该对象会传入回调函数。也就是说，浏览器会将这个对象序列化以后保留在本地，重新载入这个页面的时候，可以拿到这个对象。如果不需要这个对象，此处可以填 **null**。
- **title**：新页面的标题。但是，现在所有浏览器都忽视这个参数，所以这里可以填空字符串。
- **url**：新的网址，必须与当前页面处在同一个域。

每当同一个文档的浏览历史（即 **history** 对象）出现变化时，就会触发 **popstate** 事件。

仅仅调用 **pushState()** 方法或 **replaceState()** 方法，并不会触发该事件，只有用户点击浏览器倒退按钮和前进按钮，或者使用 JavaScript 调用 **History.back()**、**History.forward()**、**History.go()** 方法时才会触发。

```
history.pushState(state, '', 'page1');
```

// 点击浏览器前进后退按钮，或调用 back、forward、go 方法时才触发 popstate 事件

```
window.onpopstate = function(event) {  
    console.log("onpopstate", event.state, location.pathname);  
}
```

3. 总结

hash 模式使用 onhashchange 监听路由变化；history 模式使用 pushState 推入新页面，使用 popstate 监听路由变化。

history 适合用于需要搜索引擎爬虫搜索到的情况，例如企业官网。不需要搜索引擎搜索的使用 hash 模式，更简单。