ssr是服务端渲染。即在node后台将组件渲染成HTML，发送给浏览器。

优点：

1.更好的SEO，搜索引擎可以直接查看渲染好的页面

2.更加快的内容到达时间，后台渲染好了，浏览器收的快

缺点：

1.后台的必须有node的服务

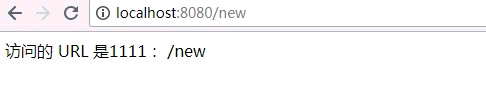
2.服务器压力大

过程：

· 1.简单实验

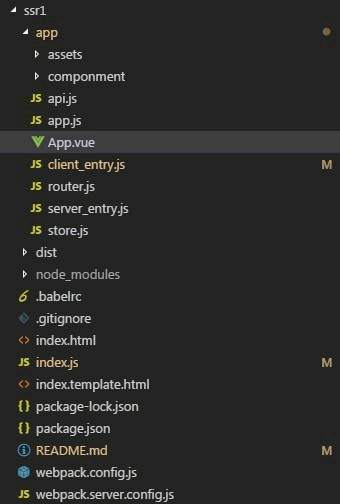


结果



2.起一个vue项目

我的项目结构



说明：

app的是vue项目

client\_entry.js是客户端入口

server\_entry.js是服务端入口

app.js是生成vue对象的工厂函数

index.js的node后台服务启动

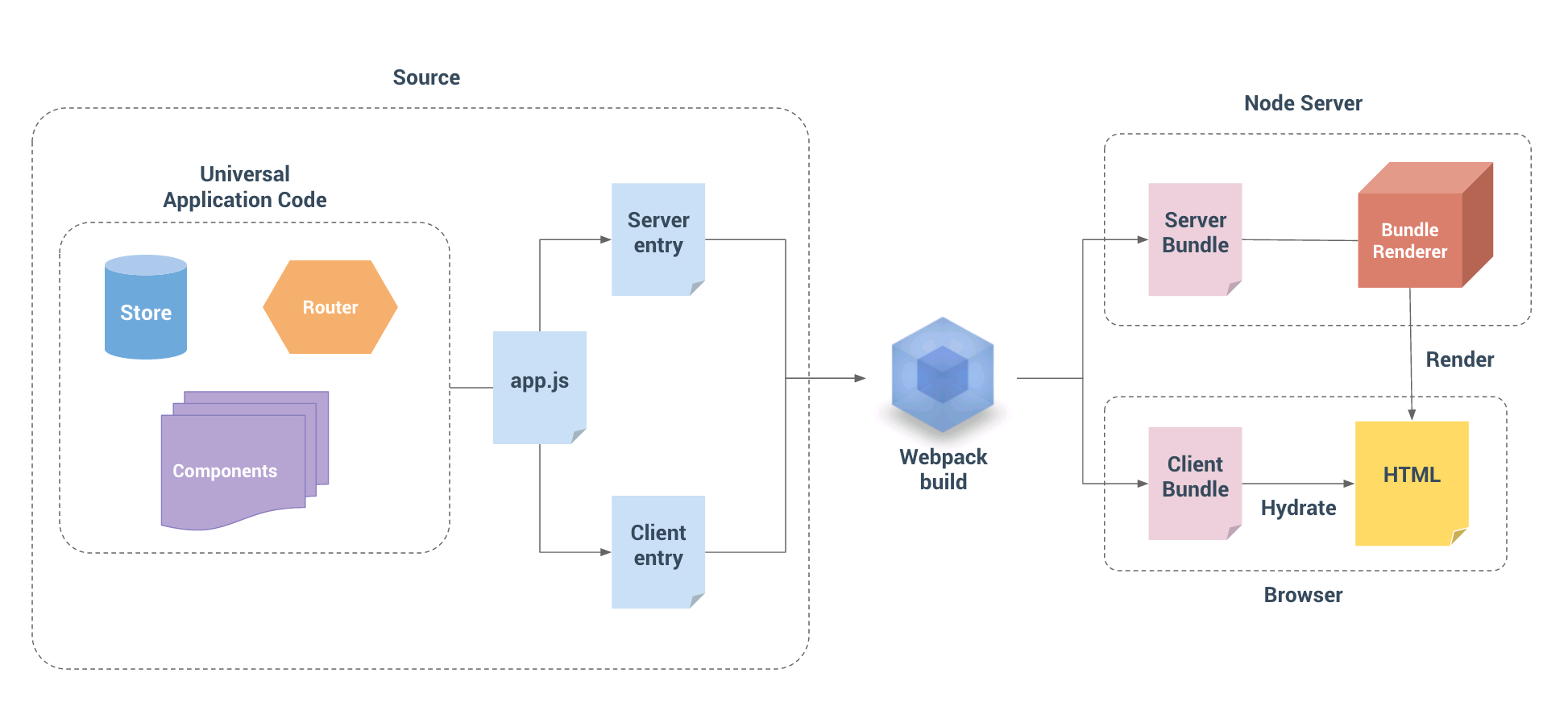
index.template.html是后台用到的模板

webpack.config.js是生成生成客户端构建清单(client build manifest) 的配置

webpack.server.config.js是生成server bundle 的配置

.babelrc是babel的配置

3.步骤



App.vue

包含一个id为app的div（用来挂载在vNode）

app.js

export default function createApp(){

return new Vue({

store,

router,

//js渲染模板 h是createElement函数

render:h => h(App)

})

}

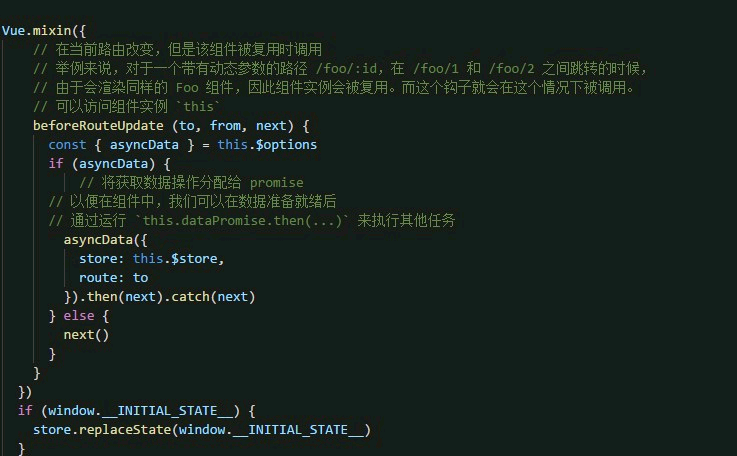
store.js是vuex的store初始化，就一个test方法

api.js是axios的接口api的

router.js是路由说明的

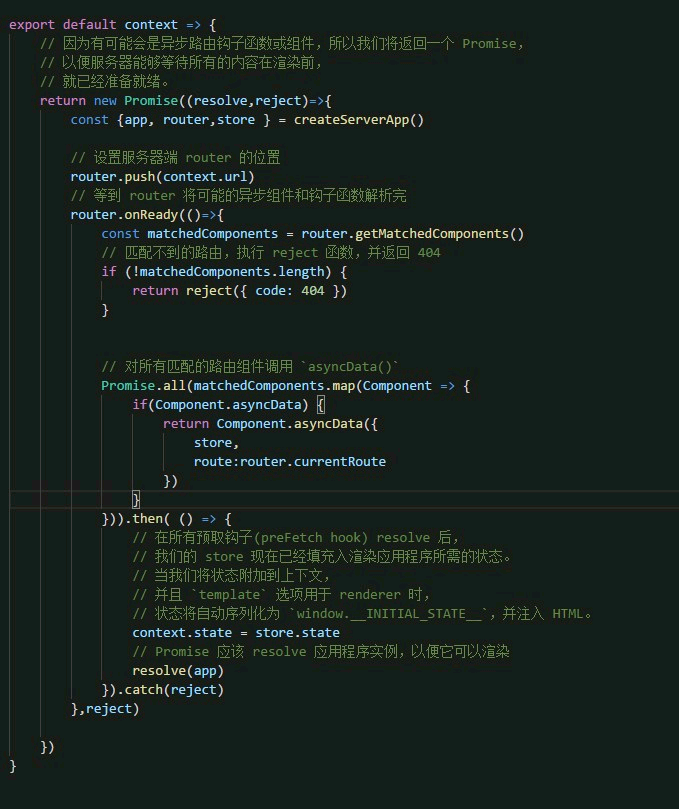
componment是组件

client\_entry.js





server\_entry.js



组件内调用：

asyncData ({ store, route }){ //触发函数

return store.dispatch("test",1)

},

css管理：

在webpack的配置中vue-loader加

options: {

// enable CSS extraction

extractCSS: true

}

添加css-loader

{

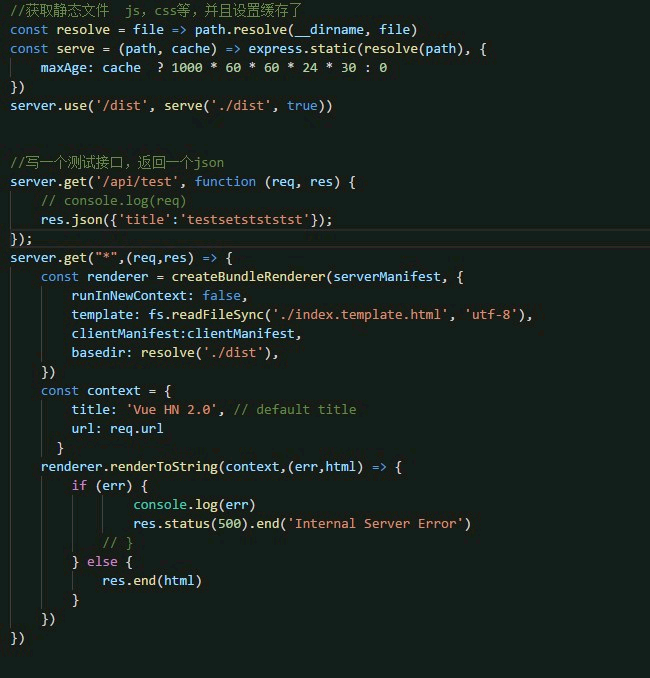
test: /\.css$/,

use: ['vue-style-loader', 'css-loader']

}

后台服务：

index.js

 后记：

只是简单入门项目，运用起来应该还有很多地方有坑。只是根据官网

<https://ssr.vuejs.org/zh/>[实现了基本](https://ssr.vuejs.org/zh/实现了基本Dom)Dom，路由，异步数据加载（基于axios，vuex）这 些功能。

参考：

<https://ssr.vuejs.org/zh/>

<https://github.com/vuejs/vue-hackernews-2.0/>

链接：

nuxt <https://github.com/vuejs/vue-hackernews-2.0/>

next <https://nextjs.org/>