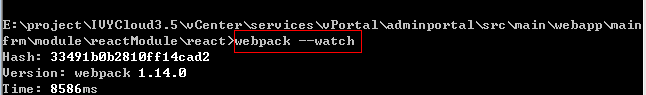
React+webpack模式和IVYCLoud框架融合

## 开发模式

按照webpack的开发模式，React模块的代码和IVYCloud的代码实现了分离，由IVYCloud框架加载经由webpack打包的bundle.js和vendors.js文件。

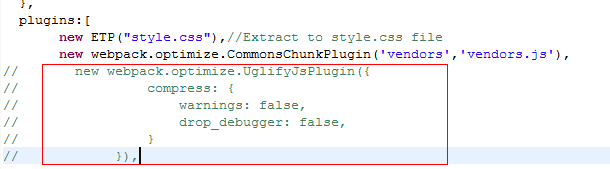
开发人员编写React部分的代码时使用node对代码进行编译打包。具体如下：



如上图，在命令行进入IVYCloud项目文件的react模块的根目录下，执行webpack –watch指令，这样webpack便会自动检测代码变化并进行编译打包。如果没有—watch指令，则每次修改代码都需要手工编译打包。

## 调试方式

在react代码中仍旧使用debugger或者console.log方式可以进行调试。在进行debugger调试的时候需要注意，webpack.config.js文件的设置不能设置为压缩模式，否则将无法进入debugger模式。



如上图，在开发过程中需要将红色框内的代码注释掉，上线的时候为了压缩文件提升加载速度，则需要开启该功能。具体压缩知识点参见《React-Router在单界面开发中的应用》。

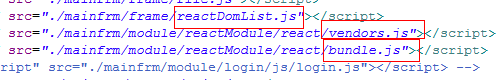
## 组件根节点

React的开发模式中，需要在入口文件（app.js/index.js）中设置渲染根节点。经过测试，当浏览器加载经由webpack打包的bundle.js和vendors.js并渲染时，该渲染根节点需要先于这两个文件之前便存在于DOM中。因此，考虑如下实现方式：在加载bundle.js和vendors.js时，先加载文件reactDomList.js文件，并在reactDomList.js文件中创建不可见的节点，以供react组件寻找渲染。在进入相应react模块时再将该根节点置为可见，并转移至相应的节点下。具体实现代码如下：

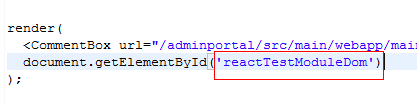
### 创建不可见节点并添加进body：



### 加载reactDomList.js、vendors.js、bundle.js文件：



### 在React入口设置相应渲染根节点:



### 在相应模块处显示该节点（react组件）：

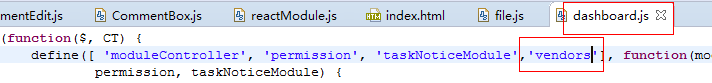


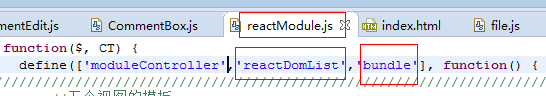
### 存在的问题：

该种方式要求在网页加载的时候就需要加载reactDomList.js、vendors.js、bundle.js三个文件，有可能导致该模块并没有被用户使用到，而空加载会拖慢加载速度。

#### 解决办法：

和webpack+React模块内部的按需加载不同（参见《React-Router在单界面开发中的应用》），考虑能否在IVYCloud现有框架内实现按需加载这三个文件。由于bundle.js需要在vendors.js加载并执行之后才能有效执行，因此对于vendors.js(相当于react的依赖库)放在dashboard.js的require模块引入，而reactDomList.js、bundle.js则在我们的reactModule模块的require模块引入。如下图：





不必在index.html中引入相关文件。

## 获取IVYCloud框架内的共有参数

在React模块内部获取IVYCloud框架内的共有参数，如\_.GetUrl()函数、\_.Lang()函数。经过测试，可以直接在React组件内部直接引用。

### **存在的问题**：

AJAX请求失败(POST);

解决办法：

由于jquery的版本问题，使用npm安装的jquery为最高版本V3.1.1,和IVYCloud的V1.12.1版本不匹配。因此在我们的react模块中无需使用”import $ from ‘jquery’”。由于该模块是在IVYCloud环境下运行，浏览器会匹配相应的jquery库去处理react中的$符号。