

## 2.源码阅读方法

珠峰前端架构师技术分享课 <<https://ke.qq.com/course/272058>>

React源码在国内并没有完整的参考资料，大多数是某一个固定版本零散的原理介绍。

之前读过jquery,koa,redux等一些框架源码，由于源码不多且设计并不复杂，所以逐行扫描就可以了。我曾经也按照这种方式读react-15源码，结果很容易猜到以失败告终。

### 如何学习react源码？

对于react这种上万行的前端框架，逐行阅读无疑是行不通的。

分享几条笔者阅读框架源码技巧：

- 基线法：以低版本为基线开始阅读，逐渐了解源码的演进过程和思路
- 广度优先法：从项目的入口文件，广度优先遍历函数调用关系（主要看注释，不关注实现），了解依赖以及调用流程，最后在精度函数的具体实现，
- 调用栈调试法：从一些最基本的api入口入手，调试查看函数调用栈，依次了解主线流程。

### 广度优先法

看过basic中的示例之后，我们发现两个重要的调用过程：createClass和renderComponent。这两个接口都被定义在了src/core/React.js中：

```
"use strict";

var ReactCompositeComponent = require('ReactCompositeComponent');
var ReactComponent = require('ReactComponent');
var ReactDOM = require('ReactDOM');
var ReactMount = require('ReactMount');

var ReactDefaultInjection = require('ReactDefaultInjection');

ReactDefaultInjection.inject();

var React = {
  DOM: ReactDOM,
  initializeTouchEvents: function(shouldUseTouch) {
    ReactMount.useTouchEvents = shouldUseTouch;
  },
  autoBind: ReactCompositeComponent.autoBind,
  createClass: ReactCompositeComponent.createClass,
  createComponentRenderer: ReactMount.createComponentRenderer,
  constructAndRenderComponent: ReactMount.constructAndRenderComponent,
  constructAndRenderComponentByID: ReactMount.constructAndRenderComponentByID,
  renderComponent: ReactMount.renderComponent,
  unmountAndReleaseReactRootNode: ReactMount.unmountAndReleaseReactRootNode,
  isValidComponent: ReactComponent.isValidComponent
};
```

通过阅读src/core/React.js文件，就会发现，原来createClass和renderComponent都是其他模块提供的调用，React内部有包括ReactDOM、ReactComponent、ReactCompositeComponent、ReactMount、ReactDefaultInjection等模块。

于此同时，我们还发现，React中定义的其他接口包括: `autoBind`、`createComponentRenderer`、`isValidComponent`等接口。顺着这些接口看下去，就能够逐步理清React框架中各个源文件的接口信息。

**在了解了这些模块的基本内容之后，画出模块关系图与时序图，它能帮助你理清框架的整体流程。这点非常重要，因为人的头脑短暂记忆非常有限，图对于降低问题复杂度至关重要。**

然后就可以开始逐个的深度阅读接口内部实现。当React中定义的函数都被全部阅读之后，基本上就完成了框架源码的学习过程。

## 调用栈调试法

第一步与广度优先法相同，看过了basic中的示例之后，我们发现两个重要的调用过程：`createClass`和`renderComponent`。

```
createClass: ReactCompositeComponent.createClass,  
renderComponent: ReactMount.renderComponent,
```

依次调试`ReactCompositeComponent.createClass`和`ReactMount.renderComponent`两个函数，根据调用的接口顺序依次阅读接口的主要流程实现。

这样了解完调用栈的主线，对react的组件运行机制就有大致的了解。再深度阅读每个接口的调用。

这种方法相对于广度优先法更适合新人，而且速度比广度优先法快得多。边阅读边调试同样也易于理解。

---