

React Native原理分析

2018 3.18

React Native背景

- React.js 是FB开源的前端框架，采用组件化的方式高效的开发前端页面。
- React Native 是基于React.js实现的可以开发移动端的框架。

React Native核心

React Native 核心是解决JS与OC通信的问题，包括互相调用和回调。

OC调用JS

使用JavaScriptCore

```
JSContext *context = [[JSContext alloc] init];
NSString *js = @"function add(a,b) {return a+b;}";
[context evaluateScript:js];
JSValue *result = [context[@"add"] callWithArguments:@[@2, @3]];
NSLog(@"result is %@", @(result.toInt32));
```

JS调用OC

1、使用block

```
JSContext *context = [[JSContext alloc] init];
context[@"add"] = ^(NSInteger a, NSInteger b) {
    NSLog(@"a + b = %@", @(a + b));
};
[context evaluateScript:@"add(2, 3)"];
```

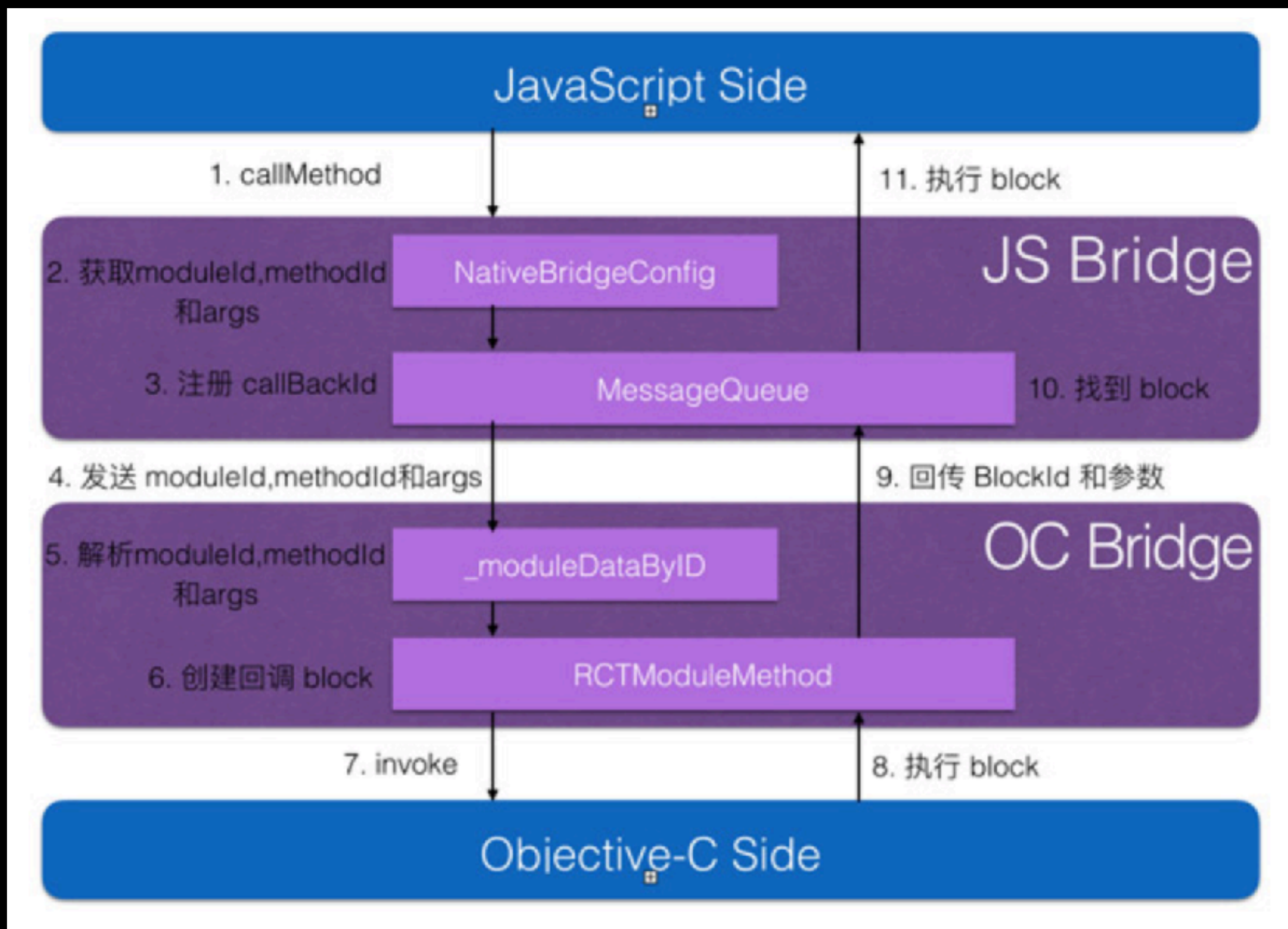
2、JSExport protocol

RN实现JS-OC通信的原理

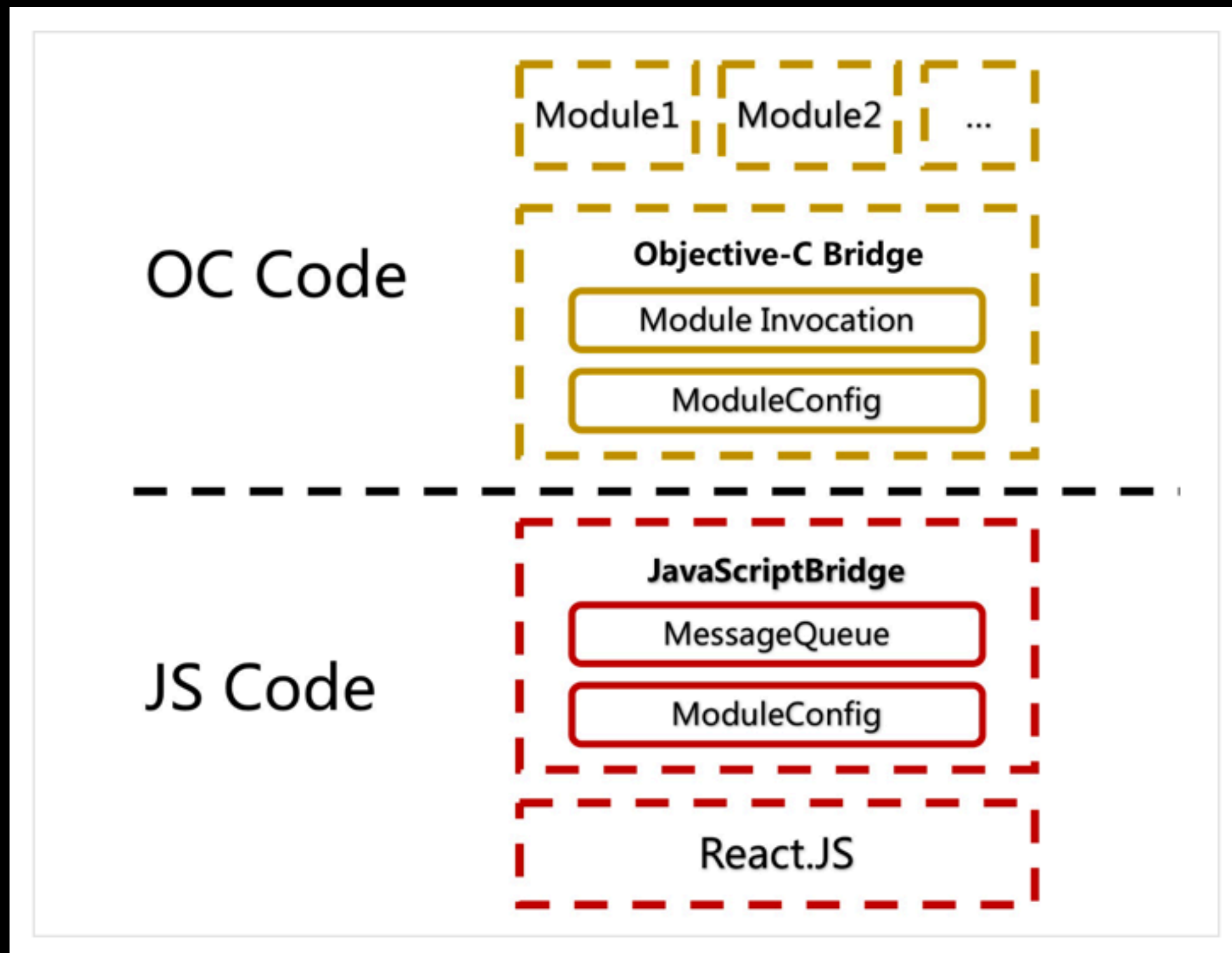
JS调用OC：RN在JS和OC两端都保存了一份配置表，记录了所有OC暴露给JS的模块（类）和方法，JS调用OC，都可以最终转换成传递ModuleID、MethodID和Arguments三个元素，分别表示类、方法和参数列表。OC接收到这三个值以后就可以通过runtime唯一确定调用哪一个方法。

JS调用OC之后，回调执行JS代码：在JS调用OC代码时，注册要回调的block，并且把BlockID作为参数传给OC，OC接收到参数之后会创建Block，在Block中执行JS的方法。

RN实现JS-OC通信流程图



RN架构图



RN架构图

