

# <ReactNative性能优化 & 企业级应用>

 × 赵宏罡

# 自我介绍

```
{  
    "姓名": "赵宏罡",  
    "所在团队": "腾讯-QQ浏览器成都研发团队",  
    "职位": "资深前端开发",  
    "方向": "前终端一体化",  
    "学校": "电子科技大学",  
    "工作经历": [{  
        "之前的团队": "阿里飞猪前端团队",  
        "之前的职位": "高级前端开发",  
        "之前的方向": "hybrid-离线包"  
    }],  
    "联系方式": "1152534558@qq.com"  
}
```

# 背景



工具应用



拓展业务场景



发现平台

# 背景



# 背景



## 前端实现

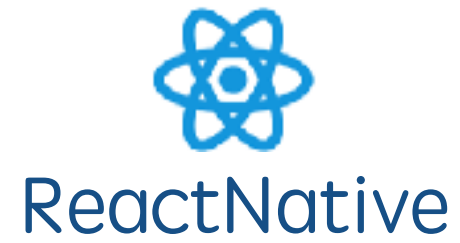
- ✓ 开发效率高
- ✓ 双平台支持
- ✓ 可动态发布
- ✗ 体验相对差
- ✗ 调原生能力弱

## 终端实现

- ✓ 体验好
- ✓ 调原生能力强
- ✗ 开发效率低
- ✗ 无法跨平台
- ✗ 发版本很麻烦

# 背景

前端实现



终端实现

✓ 开发效率高    ✓ 双平台支持    ✓ 可动态发布    ✓ 体验好

背 景

QQ浏览器的

之路

ReactNative

# 背景

## QQ浏览器的ReactNative之路

体育、资讯、视频浮层、小视频、全景广告、音频等**23个**业务





# RN 性能优化经验分享 (前端篇)

# 重构手势事件

RN:



持续发送手势事件



```
<TouchableWithoutFeedback>  
  <View>...</View>  
</TouchableWithoutFeedback>
```

```
let pan = PanResponder.create({  
  ...  
});
```

✗ 前终端通信的通道被大量占用

# 重构手势事件

优化:



终端



前端

```
<View onTouchMove={handleTouch()}>
  ...
  <Text onPress={handlePress()}>...</Text>
  ...
</View>
```

- ✓ 不用再用<Touchable>组件包裹
- ✓ 通信通道负担大大减小

# 重构手势事件

实现：



终端

createNode



前端

```
<View onTouchMove={handleTouch()}>
  ...
  <Text onPress={handlePress()}>...</Text>
  ...
</View>
```

# 重构手势事件

实现:

```
<View onTouchMove={handleTouch()}>
  ...
  <Text onPress={handlePress()}>...</Text>
  ...
</View>
```

createNode

```
{
  ...
  "id": 5
  "name": "Text"
  "props": {
    "onPress": true
  }
}
```



终端

\_\_fbBatchedBridge.flushedQueue  
UIManager.createView

EventModule.eventDispatch(5, e)

\_\_fbBatchedBridge.callFunctionReturnFlushedQueue

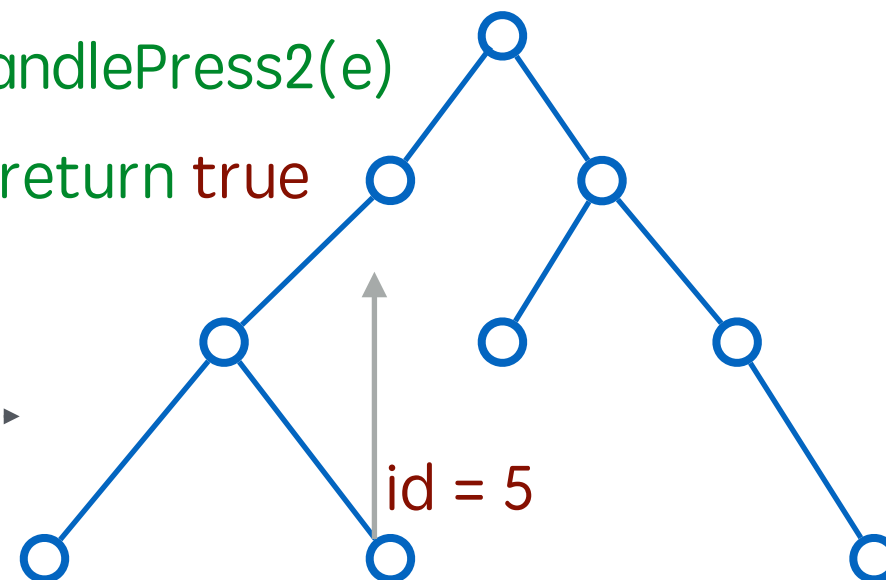
handlePress2(e)

return true

id = 5

handlePress(e)

return false



# 重构手势事件

实现:

```
<View onTouchMove={handleTouch()}>
  ...
  <Text onPress={handlePress()}>...</Text>
  ...
</View>
```

createNode

```
{
  ...
  "id": 5
  "name": "Text"
  "props": {
    "onPress": true
  }
}
```

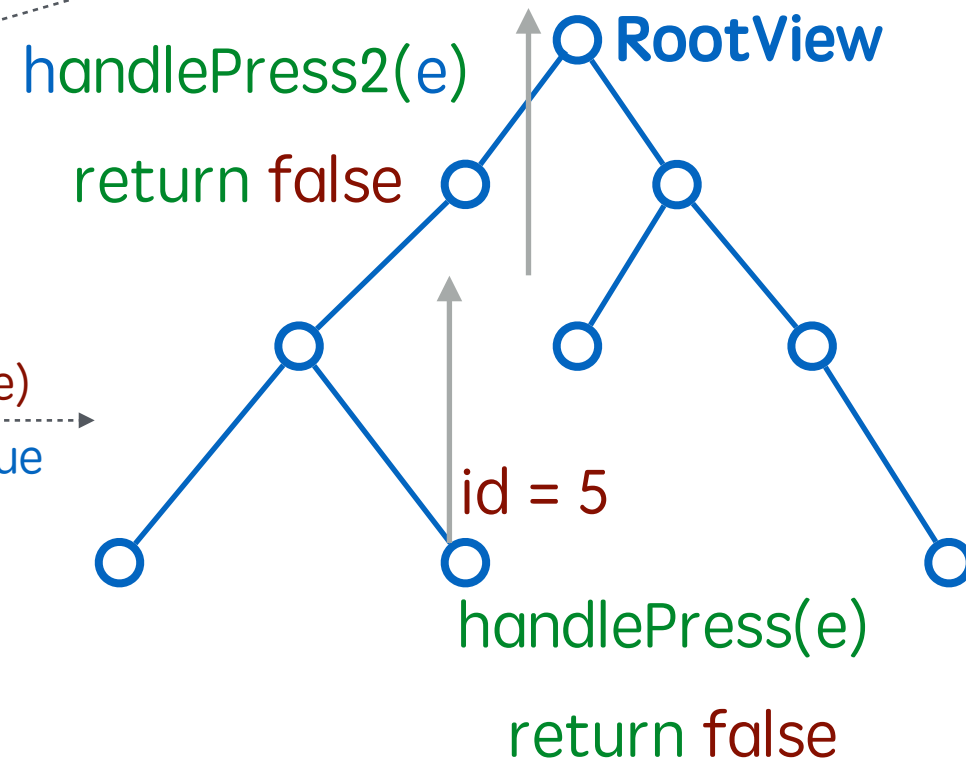


终端

\_\_fbBatchedBridge.flushedQueue  
UIManager.createView

EventModule.eventDispatch(5, e)

\_\_fbBatchedBridge.callFunctionReturnFlushedQueue



# 重构手势事件

## 小槽点:

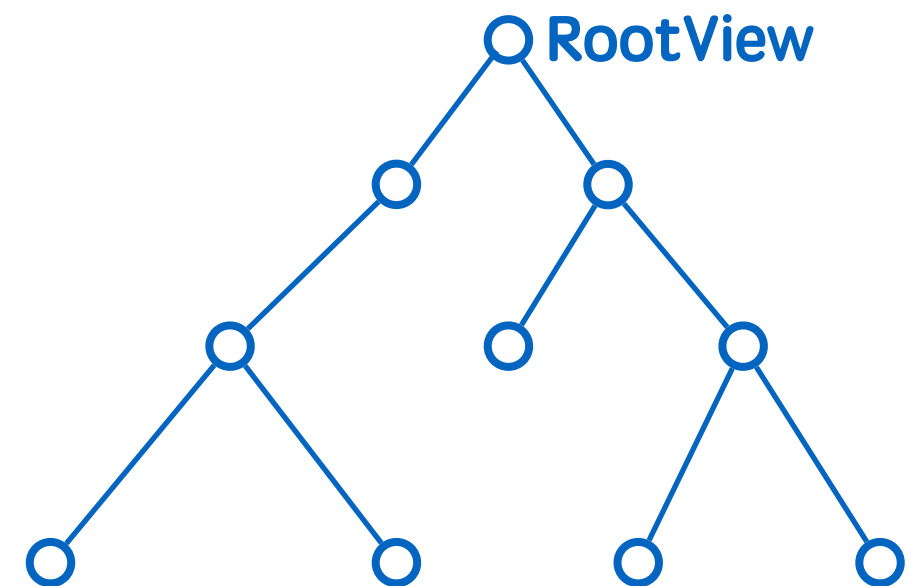
```
{  
  "id": 5,  
  "name": "Text",  
  "props": {  
    "onPress": true  
  },  
  "return": {...},  
  "child": {...},  
  "sibling": {...},  
  "index": 1  
}
```

origin

```
{  
  "id": 5,  
  "name": "Text",  
  "props": {  
    "onPress": true  
  },  
  "return": {...},  
  "child": [...],  
  "sibling": [...],  
}
```

better

```
{  
  ...  
  "id": 5,  
  "name": "Text",  
  "props": {  
    "onPress": true  
  }  
}
```



# 通信层的深度定制

RN:



尽量少用

`<TouchableWithoutFeedback />`

专属组件才可以绑手势事件

缓解通信通道的负担

前端消息队列

优化后:



业务写得更舒服

`<View onTouchmove={func} />`

所有组件均可绑事件



# 通信层的深度定制

RN

的实现

前端消息队列

# 通信层的深度定制

## RN 前端消息队列 の实现



UIManage.createView()

AsyncStorage.getItem()

UIManage.callUIFunction()

setTimeout()

class MessageQueue

enqueueNativeCall

push()

MessageQueue.\_queue

```
if (global.nativeFlushQueueImmediate &&  
    now - this._lastFlush >= 5) {  
    global.nativeFlushQueueImmediate(this._queue);  
    this._queue = [[], [], [], this._callID];  
    this._lastFlush = now;  
}
```

```
__fbBatchedBridge.  
invokeCallbackAndReturnFlushedQueue()
```

MQ.flushedQueue()



终端

# 通信层的深度定制

## RN 前端消息队列 的优化



UIManage.createView()

AsyncStorage.getItem()

UIManage.callUIFunction()

setTimeout()

class MessageQueue

enqueueNativeCall

push()

MessageQueue.\_queue

```
if (global.nativeFlushQueueImmediate &&  
now this._lastFlush >= 5) {  
  global.nativeFlushQueueImmediate(this._queue);  
  this._queue = [[], [], [], this._callID];  
  this._lastFlush = now;  
}
```

```
__fbBatchedBridge.  
invokeCallbackAndReturnFlushedQueue()
```

MQ.flushedQueue()



终端

# 通信层的深度定制

RN:



尽量少用

`<TouchableWithoutFeedback />`

专属组件才可以绑手势事件

缓解通信通道的负担

前端消息队列

优化后:



业务写得更舒服

`<View onTouchmove={func} />`

所有组件均可绑事件

通信通道无负担

前终端通信直接调用更高效

# 没有性能瓶颈的动画方案



因为有前终端通信，RN是否不适合做动画？



基于v8的高效，一般动画都很流畅：)



但动画伴随着大量render，就会卡：(

为什么？

能否规避？



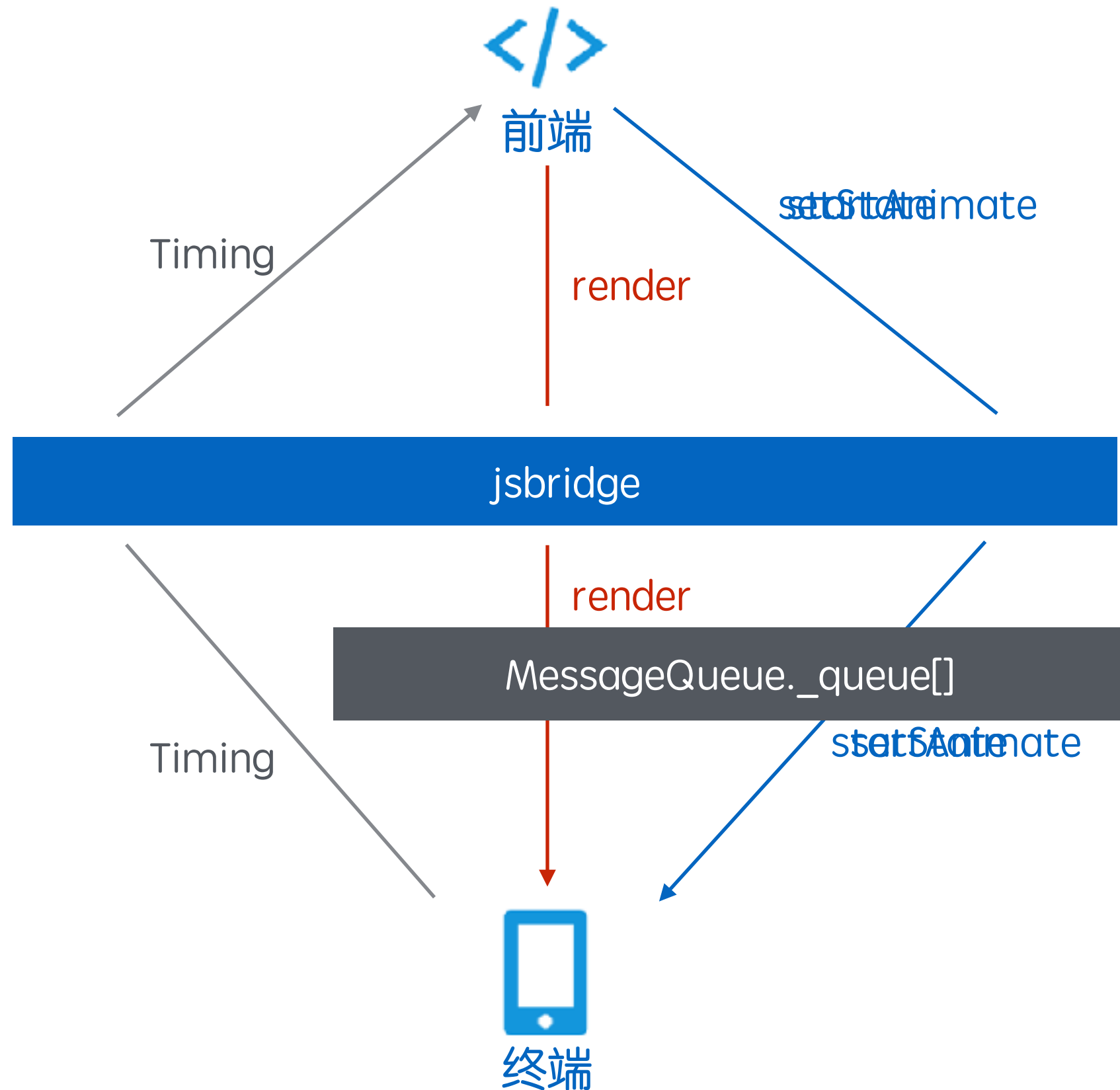
# 没有性能瓶颈的动画方案

RN:

```
<Animated.View
  style={{
    ...this.props.style,
    opacity: fadeAnim,
  }}
>
  {this.props.children}
</Animated.View>

...

componentDidMount() {
  Animated.timing(
    this.state.fadeAnim,
    {
      toValue: 1,
      duration: 10000,
    }
  ).start();
}
```



# 没有性能瓶颈的动画方案

## 优化:

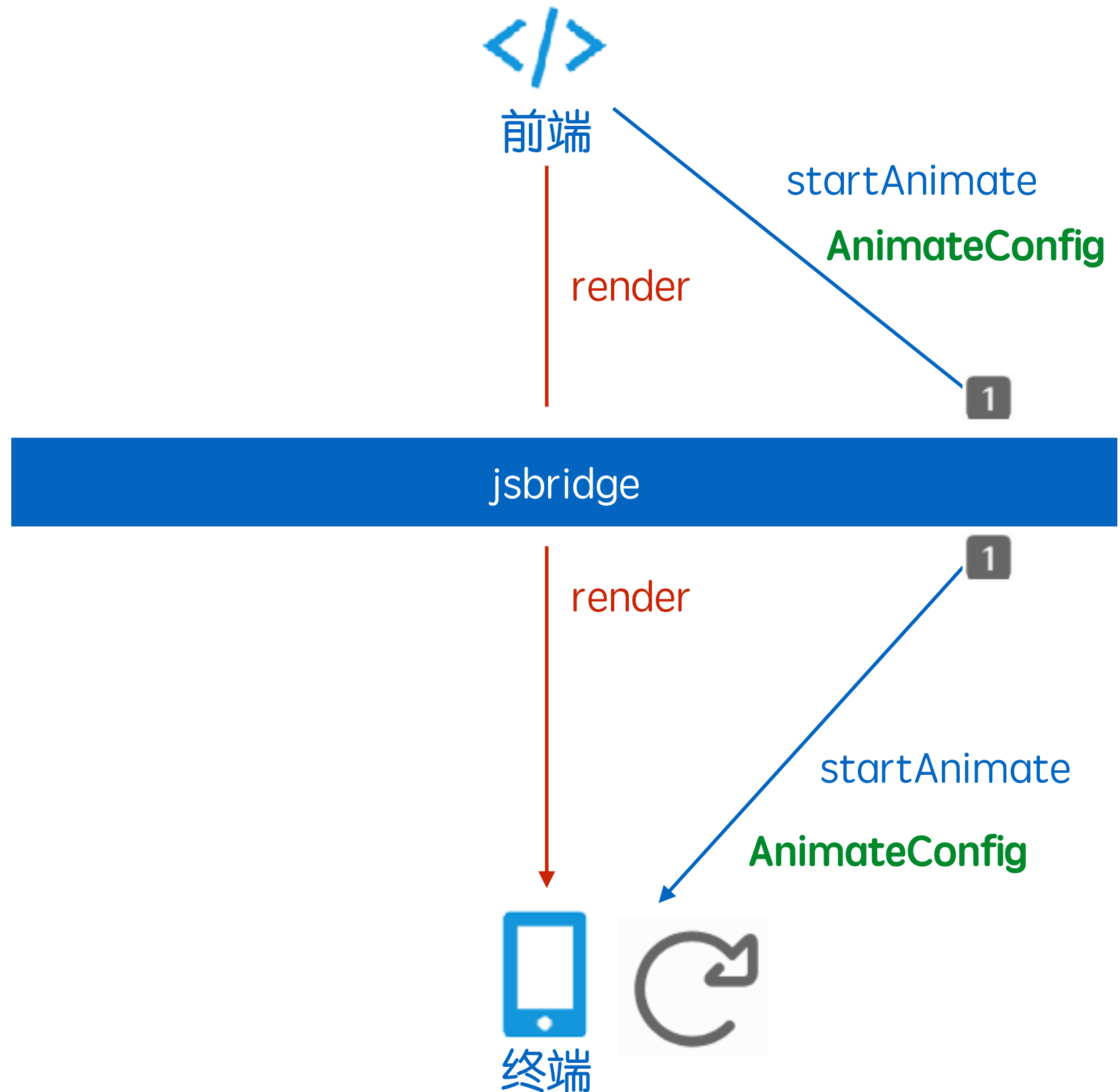
```
<View style={{ width: this.animate1}}>  
  <Text style={{color:"#ff0000"}}>  
    {'hello wordk'}  
  </Text>  
</View>
```

...

```
this.animate1 = new Animation({  
  mode: "timing",  
  delay: 0,  
  startValue: 100,  
  toValue: 300,  
  duration: 3,  
  timingFunction: "linear",  
  repeatCount: "loop"  
});
```

...

```
this.animate1.start();
```



# 没有性能瓶颈的动画方案

## 优化:

```
this.animate2 = new Animation({
  mode: "timing",
  delay: 0,
  startValue: this.animate1,
  toValue: 100,
  duration: 3,
  valueType: "color",
  timingFunction: "linear",
  repeatCount: 1
});
```

```
this.animateSet = new AnimationSet({
  repeatCount: 1,
  children: [
    {
      animation: this.animate1,
      follow: false
    },
    {
      animation: this.animate2,
      follow: false
    }
  ]
});
```



前端

startAnimate

AnimateConfig

render

1

jsbridge

render

startAnimate

AnimateConfig

1



终端





# 分包、裁包和js内存的故事

RN，sdk代码和业务代码一个jsbundle：

名称	修改日期	类型	大小	
feeds.index.jsbundle	2017/7/22 15:45	JSBUNDLE 文件	1,061 KB	业务A代码 + sdk代码
market.index.jsbundle	2017/7/22 15:46	JSBUNDLE 文件	644 KB	业务B代码 + sdk代码

改造rn的打包工具，jsbundle分包：

名称	修改日期	类型	大小	
feeds.index.jsbundle	2017/7/22 17:34	JSBUNDLE 文件	661 KB	新增业务内存增量降低414KB
market.index.jsbundle	2017/7/22 17:34	JSBUNDLE 文件	243 KB	
common.index.jsbundle	2017/7/22 17:33	JSBUNDLE 文件	414 KB	
index.android.jsbundle	2017/7/24 11:09	JSBUNDLE 文件	710 KB	RN提供的组件并非全部必要，裁剪了42%

为什么说：前端篇？

# 在终端层还有大量核心优化


启动：慢2s → 预加载引擎 / 懒加载前终端模块 → 持平

list性能：老大难 → RecyclerView  
终端虚拟node树  
高效diff算法 → 帧率和内存持平

UI适配：机型问题太多 → MIUI文字截断  
base64本地图  
Emoji表情文字 → 在1亿DAU场景下  
无反馈异常

RN再企来业一应波用开实货践

# 开发调试



```
> feeds@1.0.0 start /Users/huskyzhyu/saizhao/hippy-qb/feeds  
> node node_modules/hippy-pack/cli/start.js index.js
```

```
8082端口被占用了, hippy开始自动清理端口...  
8082端口已经清理完成, 开始启动hippy调试...
```

```
开始首次编译代码...
```

```
2018-05-31 21:57:58 编译结束...
```

```
开始启动调试服务器...
```

```
调试服务已启动, 监听端口8082
```

```
2018-05-31 21:58:18 Fetching... /index.bundle?platform=ios&dev=true&minify=false
```

```
/Users/huskyzhyu/saizhao/hippy-qb/feeds/node_modules/hippy-pack/pack/server/temp/index.bundle
```

```
2018-05-31 21:58:18 Fetch complete...
```

```
2018-05-31 21:58:19 Fetching... /index.bundle?platform=ios&dev=true&minify=false
```

```
/Users/huskyzhyu/saizhao/hippy-qb/feeds/node_modules/hippy-pack/pack/server/temp/index.bundle
```

```
2018-05-31 21:58:19 Fetch complete...
```

```
Launching Dev Tools...
```

```
2018-05-31 21:58:21 Fetching... /index.bundle?platform=ios&dev=true&minify=false
```

```
/Users/huskyzhyu/saizhao/hippy-qb/feeds/node_modules/hippy-pack/pack/server/temp/index.bundle
```

```
2018-05-31 21:58:21 Fetch complete...
```

```
2018-05-31 21:58:21 Fetching... /index.bundle?platform=ios&dev=true&minify=false
```

```
/Users/huskyzhyu/saizhao/hippy-qb/feeds/node_modules/hippy-pack/pack/server/temp/index.bundle
```

```
2018-05-31 21:58:21 Fetch complete...
```

# 打包工具

→ feeds git:(master) npm run pack

> feeds@1.0.0 pack /Users/huskyzhyu/saizhao/hippy-qb/feeds

> node node\_modules/hippy-pack/cli/pack.js index.js

发布目录重新创建完毕: /Users/huskyzhyu/saizhao/hippy-qb/feeds/dist

开始android打包, 请稍等, 结束时会有提示

切换到目录/Users/huskyzhyu/saizhao/hippy-qb/feeds/dist/android, 开始准备打包

android端zip包生成完成

开始上传 android包到cdn

===== android =====

下载地址: <http://soft.imtt.qq.com/browser/browser/rn/tmp/feeds.1527775529490.zip>

QB地址: qb://hippy?

module=feeds&url=http%3A%2F%2Fsoft.imtt.qq.com%2Fbrowser%2Fbrowser%2Frn%2Ftmp%2Ffeeds.1527775529490.zip

安装二维码:



开始ios打包, 请稍等, 结束时会有提示

切换到目录/Users/huskyzhyu/saizhao/hippy-qb/feeds/dist/ios, 开始准备打包

ios端zip包生成完成...

# 打包工具

→ feeds git:(master) npm run pack

> feeds@1.0.0 pack /Users/huskyzhyu/saizhao/hippy-qb/feeds

> node node\_modules/hippy-pack/cli/pack.js index.js

发布目录重新创建完毕: /Users/huskyzhyu/saizhao/hippy-qb/feeds/dist

开始android打包, 请稍等, 结束时会有提示

切换到目录/Users/huskyzhyu/saizhao/hippy-qb/feeds/dist/android, 开始准备打包

android端zip包生成完成

开始上传 android包到cdn

===== android =====

下载地址: <http://soft.imtt.qq.com/browser/browser/rn/tmp/feeds.1527775529490.zip>

QB地址: qb://hippy?

module=feeds&url=http%3A%2F%2Fsoft.imtt.qq.com%2Fbrowser%2Fbrowser%2Frn%2Ftmp%2Ffeeds.1527775529490.zip

安装二维码:

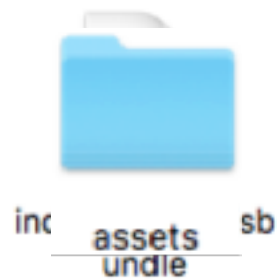


开始ios打包, 请稍等, 结束时会有提示

切换到目录/Users/huskyzhyu/saizhao/hippy-qb/feeds/dist/ios, 开始准备打包

ios端zip包生成完成...

# 打包工具

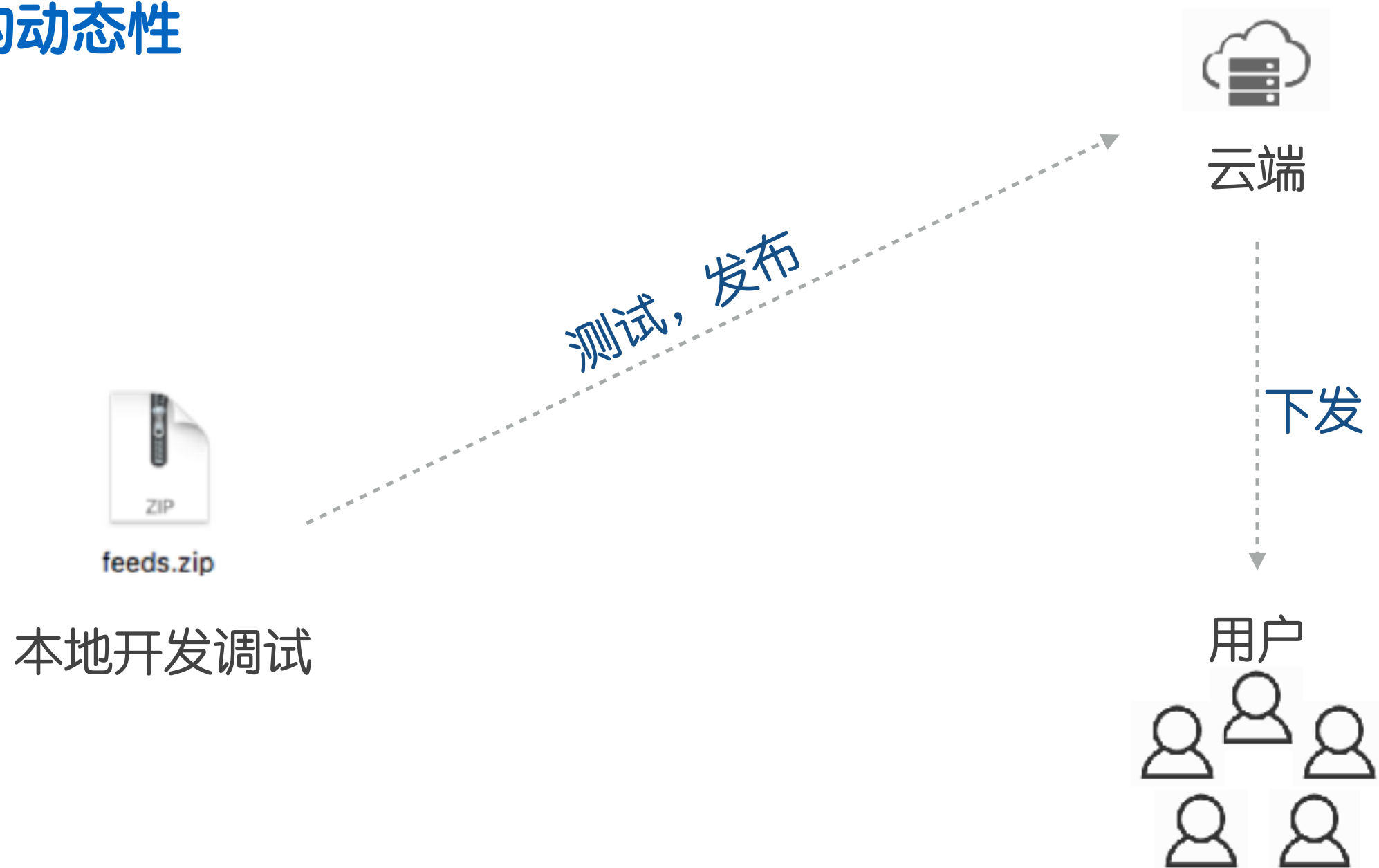


```
{  
  "packType": "rnfqb",  
  "module": "feeds",  
  "version": "1304",  
  "versionname": "1754",  
  "minQBVersion": "8.1.0.0",  
  "maxQBVersion": "",  
  "minSdkVersion": "",  
  "maxSdkVersion": "",  
  "backurl": "http://qbrnweb.html5.qq.com",  
  "windowtitle": "",  
  "favicon": "",  
  "systembarcolor": "",  
  "statusbarstyle": "0",  
  "enablenoimage": "true"  
}
```



# 打包工具

## RN的动态性



# 发布系统

首页



QQ浏览器RN管理平台 | 正式环境

管理员入口

首页

AudioApp

+

circle

+

comment

+

Demo

+

dobby

+

eexplore

+

explorecamera

+

favnew

+

feeds

+

file

+

## 公告栏

owner的职责

《ReactNative业务owner的职责清单》

常见问题

《RN业务发布方法&规范》 《如何跟进RN业务线上异常》 《如何配置白名单》

# 发布系统

首页



QQ浏览器RN管理平台 | 正式环境

管理员入口

首页

AudioApp

+

circle

+

comment

+

Demo

+

dobby

+

eexplore

+

explorecamera

+

favnew

+

feeds

+

file

+

## 公告栏

owner的职责

《ReactNative业务owner的职责清单》

常见问题

《RN业务发布方法&规范》 《如何跟进RN业务线上异常》 《如何配置白名单》

# 发布系统

首页



QQ浏览器RN管理平台 | 正式环境

管理员入口

首页

AudioApp

—

打包发布

发布记录

数据报表

circle

+

comment

+

Demo

+

dobby

+

eexplore

+

explorecamera

+

## 公告栏

owner的职责

[《ReactNative业务owner的职责清单》](#)

常见问题

[《RN业务发布方法&规范》](#) [《如何跟进RN业务线上异常》](#) [《如何配置白名单》](#)

# 发布系统

## 打包发布

QQ浏览器RN管理平台 | 正式环境

管理员入口

首页

AudioApp

打包发布

发布记录

数据报表

circle

comment

Demo

dobby

eexplre

新建构建线

构建线用于对打包记录做分组，一条打包线对应一个git分支或tag

新建

搜索bundle

输入bundle的打包id或RDM异常上报id

搜索

## 构建记录

发布线

以下构建线打出的bundle均可用于正式发布，请谨慎操作

构建线名称	创建人	创建时间	备注	操作	
音频专区8.6发布线	saizhao	2018-05-18 11:18:03	--	编辑	删除
音频专区8.6.1发布线	saizhao	2018-03-23 21:35:20	--	编辑	删除
音频专区master	saizhao	2018-03-23 21:26:13	--	编辑	删除

< 1 >

# 发布系统

## 打包发布



QQ浏览器RN管理平台 | 正式环境

管理员入口

首页

AudioApp

打包发布

发布记录

数据报表

circle

comment

Demo

dobby

eexplore

explorecamera

### 新建构建线

#### 基本信息

构建线名称：音频专区831发布线

#### 源码库配置

git仓库地址：http://git.ccde.oa.com/mig\_audio/AudioReactNative.git

代码分支或Tag：Branch\_audio\_8.3.1

#### 额外信息

备注：for浏览器8.3.1新开专区的需求

创建

# 发布系统

## 打包发布

QQ浏览器RN管理平台 | 正式环境

管理员入口

首页

AudioApp

打包发布

发布记录

数据报表

circle

comment

Demo

dobby

eexplre

新建构建线

构建线用于对打包记录做分组，一条打包线对应一个git分支或tag

新建

搜索bundle

输入bundle的打包id或RDM异常上报id

搜索

## 构建记录

发布线

以下构建线打出的bundle均可用于正式发布，请谨慎操作

构建线名称	创建人	创建时间	备注	操作	
音频专区8.6发布线	saizhao	2018-05-18 11:18:03	--	编辑	删除
音频专区8.6.1发布线	saizhao	2018-03-23 21:35:20	--	编辑	删除
音频专区master	saizhao	2018-03-23 21:26:13	--	编辑	删除

< 1 >

# 发布系统

## 服务器端打包



QQ浏览器RN管理平台 | 正式环境

管理员入口

首页

AudioApp

打包发布

发布记录

数据报表

circle

comment

Demo

dobby

### 音频专区831发布线 (打包类型: release)

创建人: saizhao

代码地址: [http://git.code.oa.com/mig\\_audio/AudioReactNative.git#Branch\\_audio\\_8.3.1\\_release](http://git.code.oa.com/mig_audio/AudioReactNative.git#Branch_audio_8.3.1_release)

创建时间:

2018-03-23 21:35:20

选择打包平台:

☒ android ☐ ios

一键打包

打包id	平台 ▾	创建人	创建时间	备注 ▾	操作		
3	android	saizhao	2018-03-27 19:26:06	设置备注	下载bundle	生成二维码	转发布
2	android	saizhao	2018-03-26 20:31:29	设置备注	下载bundle	生成二维码	转发布
1	android	saizhao	2018-03-23 21:38:17	设置备注	下载bundle	生成二维码	转发布

<

1

>



# 发布系统

## 服务器端打包



QQ浏览器RN管理平台 正式环境

管理员入口

首页

AudioApp

打包发布

发布记录

数据报表

circle

comment

Demo

dobby

音频专区master (打包类型: release)

创建人: saizhao

代码地址: [http://git.codeoos.com/mig\\_audio/AudioResctNative.git#master](http://git.codeoos.com/mig_audio/AudioResctNative.git#master)

创建时间: 2016-05-31 21:25:13

依赖安装完成, 执行打包脚本..

打包id	平台	创建人	创建时间	备注	操作
36	android	saizhao	2016-05-31 22:26:03	设置备注	下载bundle 生成二维码 转发布
8	ios	saizhao	2016-05-31 21:00:25	设置备注	下载bundle 生成二维码 转发布
68	android	saizhao	2016-05-31 11:50:00	设置备注	下载bundle 生成二维码 转发布
7	ios	saizhao	2016-05-31 11:11:41	设置备注	下载bundle 生成二维码 转发布

# 发布系统

## 服务器端打包



QQ浏览器RN管理平台 | 正式环境

管理员入口

首页

AudioApp

打包发布

发布记录

数据报表

circle

comment

Demo

doboy

音频专区master (打包类型: release)

创建人: szzhan

代码地址: [http://git.code.oa.com/mig\\_audio/AudioReactNative.git#master](http://git.code.oa.com/mig_audio/AudioReactNative.git#master)

创建时间: 2018-03-23 21:28:13

打包成功

<https://softimtt.myapp.com/browser/browser/m/tmp/AudioApp.1527913564539.zip>

转发布

生成二维码

打包id	平台	创建人	创建时间	备注	操作
55	android	szzhan	2018-05-31 22:28:03	设置备注	下载bundle 生成二维码 转发布
8	ios	szzhan	2018-05-31 21:03:25	设置备注	下载bundle 生成二维码 转发布
53	android	szzhan	2018-05-31 11:50:09	设置备注	下载bundle 生成二维码 转发布
7	ios	szzhan	2018-05-31 11:11:41	设置备注	下载bundle 生成二维码 转发布

# 发布系统

## 服务器端打包

QQ浏览器RN管理平台 | 正式环境

管理员入口

首页

AudioApp

打包发布

发布记录

数据报表

circle

comment

Demo

dobby

音频专区master (打包)

创建人: saizhao 代码地址

打包id	平台	创建人	创建时间	备注	操作
6	ios	saizhao	2018-05-31 21:03:25	设置备注	下载bundle 生成二维码 转发布
68	android	saizhao	2018-05-31 11:50:09	设置备注	下载bundle 生成二维码 转发布
7	ios	saizhao	2018-05-31 11:11:41	设置备注	下载bundle 生成二维码 转发布
61	android	saizhao	2018-05-29 17:13:33	设置备注	下载bundle 生成二维码 转发布
60	android	saizhao	2018-05-29 16:25:54	设置备注	下载bundle 生成二维码 转发布

取消

确定

# 发布系统

## 服务器端打包 の 好处

- ✓ 打过的包都有记录
- ✓ 打过的包都可以直接转线上发布
- ✓ 避免了多人开发本地环境不一致对release打包的影响
- ✓ 测试同学可以自主打包测试

# 发布系统

## 服务器端打包



QQ浏览器RN管理平台 | 正式环境

管理员入口

首页

AudioApp

打包发布

发布记录

数据报表

circle

comment

Demo

doboy

音频专区master (打包类型: release)

创建人: sszhan

代码地址: [http://git.code.oa.com/mig\\_audio/AudioReactNative.git#master](http://git.code.oa.com/mig_audio/AudioReactNative.git#master)

创建时间: 2018-03-23 21:28:13

打包成功

<https://softimtt.myapp.com/browser/browser/m/tmp/AudioApp.1527913564539.zip>

转发布

生成二维码

打包id	平台	创建人	创建时间	备注	操作
55	android	sszhan	2018-05-31 22:28:03	设置备注	下载bundle 生成二维码 转发布
8	ios	sszhan	2018-05-31 21:03:25	设置备注	下载bundle 生成二维码 转发布
53	android	sszhan	2018-05-31 11:50:09	设置备注	下载bundle 生成二维码 转发布
7	ios	sszhan	2018-05-31 11:11:41	设置备注	下载bundle 生成二维码 转发布

# 发布系统

发布

QQ浏览器RN管理平台 | 正式环境

管理员入口

首页

AudioApp

打包发布

发布记录

数据报表

circle +

comment +

Demo +

dobby +

eexplore +

explorecamera +

favnew +

feeds +

音频专区master (打包id: 55)

创建人: saizhao 包地址: <https://softlimit.myapp.com/browser/browser/rn/tmp/AudioApp.1527776623465.zip> 创建时间: 2016-05-31 22:26:03 平台: android

## 发布配置

### 基本信息

业务VersionCode : 128 ⓘ

QB版本限制: 8.3.1.4075 最大QB版本限制-非必填 ⓘ  
如需限定特殊SDK版本发布点击此处

### 灰度规则

灰度比例: 0

### 额外信息

备注: 本次发布的备注信息

发布

# 发布系统

## 发布



QQ浏览器RN管理平台 | 正式环境

管理员入口

首页

AudioApp

打包发布

发布记录

数据报表

circle

comment

Demo

dobby

eexplore

explorecamera

favnew

feeds

音频专区master (打包id: 55)

创建人: saizheo

包地址: <https://softimtt.myapp.com/browser/browser/rn/tmp/AudioApp.1527776623465.zip>

创建时间: 2018-05-31 22:26:03

平台: android

## 发布配置

### 基本信息

业务VersionCode : 128 ⓘ

QB版本限制: 8.3.1.4075

- 最大QB版本限制-非必填 ⓘ

如需限定特殊SDK版本发布

[点击此处](#)

### 灰度规则

灰度比例 :



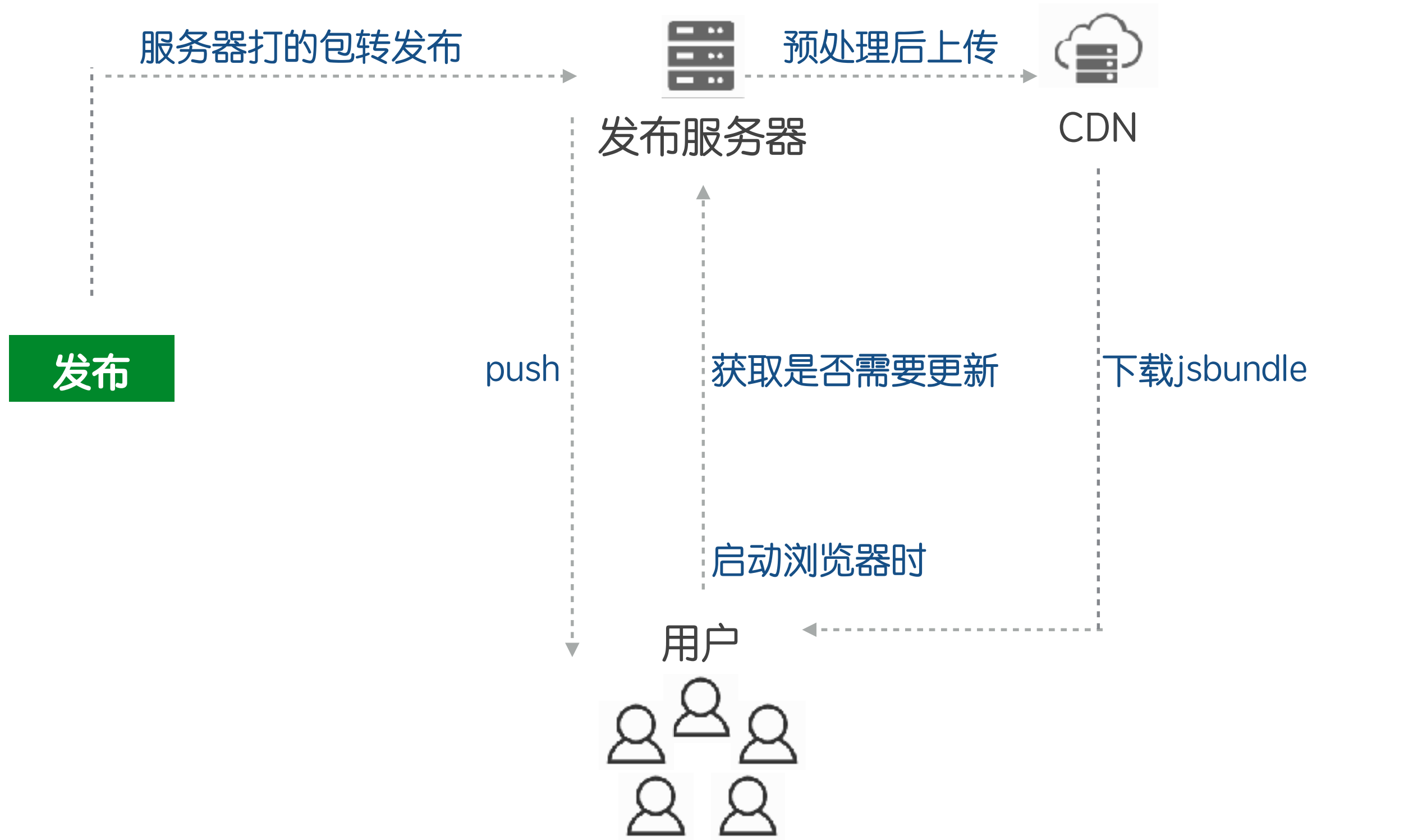
### 额外信息

备注: 本次发布的备注信息

发布

# 发布系统

发布





# 发布系统

## 发布 の 功能点

✓ 支持灰度和ABTest

✓ 默认走增量包发布

发布时在服务器端生成基于zip的增量包

默认先尝试以增量包更新

用户下载后在本地完成patch

若失败自动降级到走全量包继续更新

✓ 用MD5+RSA校验，确保包的一致性

✓ 具备push和流控的能力

✓ 支持2种强制更新模式

进入业务时，走进度条，拉取最新

使用业务时，直接弹窗更新

# 发布系统

## 发布记录

QQ浏览器RN管理平台 正式环境

管理员入口

首页

AudioApp

打包发布

发布记录

数据报表

circle

comment

Demo

dobby

eeexplore

explorecamera

fsview

feeds

## 发布记录

Android

IOS

### Android端发布记录

vc	vn	灰度规则 7	限制规则	创建人	最新操作时间	发布备注	运行状态	操作	
53	127	预发布   仅白名单生效	qb: 8.3.1.4075 - 无	saizhao	2018-05-31 11:53:40	—	run	下载发布包	修改发布配置
51	124	全量发布	qb: 8.3.1.4075 - 无	saizhao	2018-05-30 20:24:43	—	run	下载发布包	修改发布配置
48	121	全量发布	qb: 8.3.1.4075 - 无	damonruan	2018-05-10 22:35:18	10% 36% 100%	run	下载发布包	修改发布配置
41	120	预发布	qb: 8.3.1.0 - 无	saizhao	2018-05-10 11:12:03	—	run	下载发布包	修改发布配置
40	100	全量发布	qb: 8.4.0.0 - 无	saizhao	2018-04-28 20:27:10	—	run	下载发布包	修改发布配置
39	118	全量发布	qb: 8.4.0.0 - 无	saizhao	2018-04-26 09:46:47	—	run	下载发布包	修改发布配置
38	117	全量发布	qb: 8.3.1.4075 - 8.5.0.0	saizhao	2018-04-21 01:56:21	100%	run	下载发布包	修改发布配置
35	116	预发布	qb: 8.3.1.4075 - 无	damonruan	2018-04-20 18:17:51	50% 0%	run	下载发布包	修改发布配置
34	116	预发布	qb: 8.3.1.4075 - 无	damonruan	2018-04-20 16:06:51	—	run	下载发布包	修改发布配置
31	114	预发布	qb: 8.3.1.4075 - 无	saizhao	2018-04-18 17:25:35	—	run	下载发布包	修改发布配置

# 发布系统

## 发布配置修改（扩大灰度比例）



QQ浏览器RN管理平台 | 正式环境

管理员入口

首页

AudioApp

—

打包发布

发布记录

数据报表

circle

+

comment

+

Demo

+

dobby

+

eexplore

+

explorecamera

+

### 修改发布配置

#### 基本信息

业务version name :

127

QB版本限制:

8.3.1.4075

最大QB版本限制

如需限定特殊SDK版本

发布[点击此处](#)

#### 灰度规则

灰度比例 :



#### 额外信息

备注 :

本次发布的备注信息

提交修改

# 发布系统

## 发布数据报表

QQ浏览器RN管理平台 | 正式环境

管理员入口

首页

AudioApp

打包发布

发布记录

数据报表

circle

comment

Demo

doboy

eeexplore

explorecamera

favnew

feeds

打包发布

AudioApp拉取情况

选择版本: 51.0

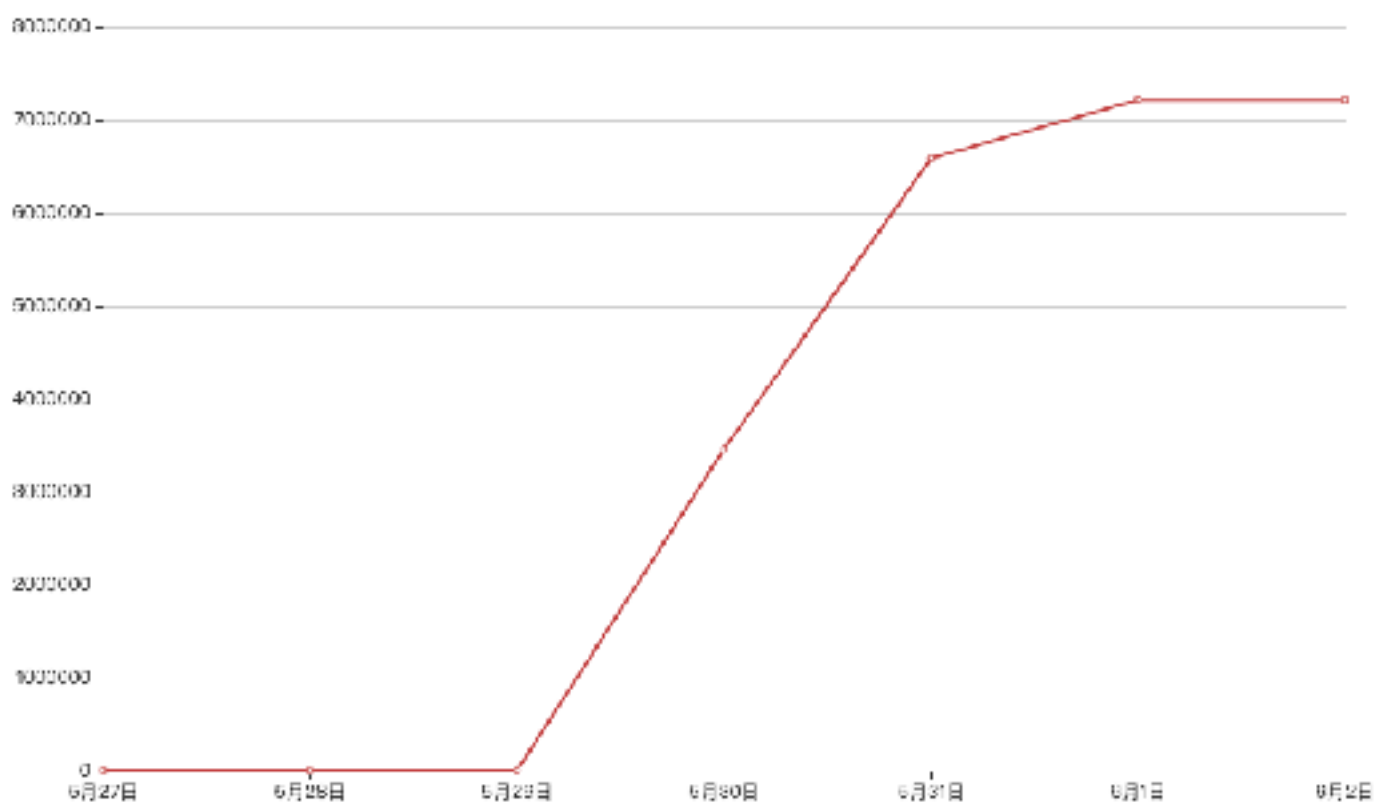
确定

选择时间跨度: 7天 | 30天

增量

总量

图 表 重绘 下载



# 发布系统

## 发布数据报表

QQ浏览器RN管理平台 | 正式环境

管理员入口

首页

AudioApp

打包发布

发布记录

数据报表

circle

comment

Demo

dobby

eeexplor

explorecamera

favnew

feetix

打包发布

发布记录

AudioApp拉取情况

选择版本: 51.0

确定

选择时间跨度:

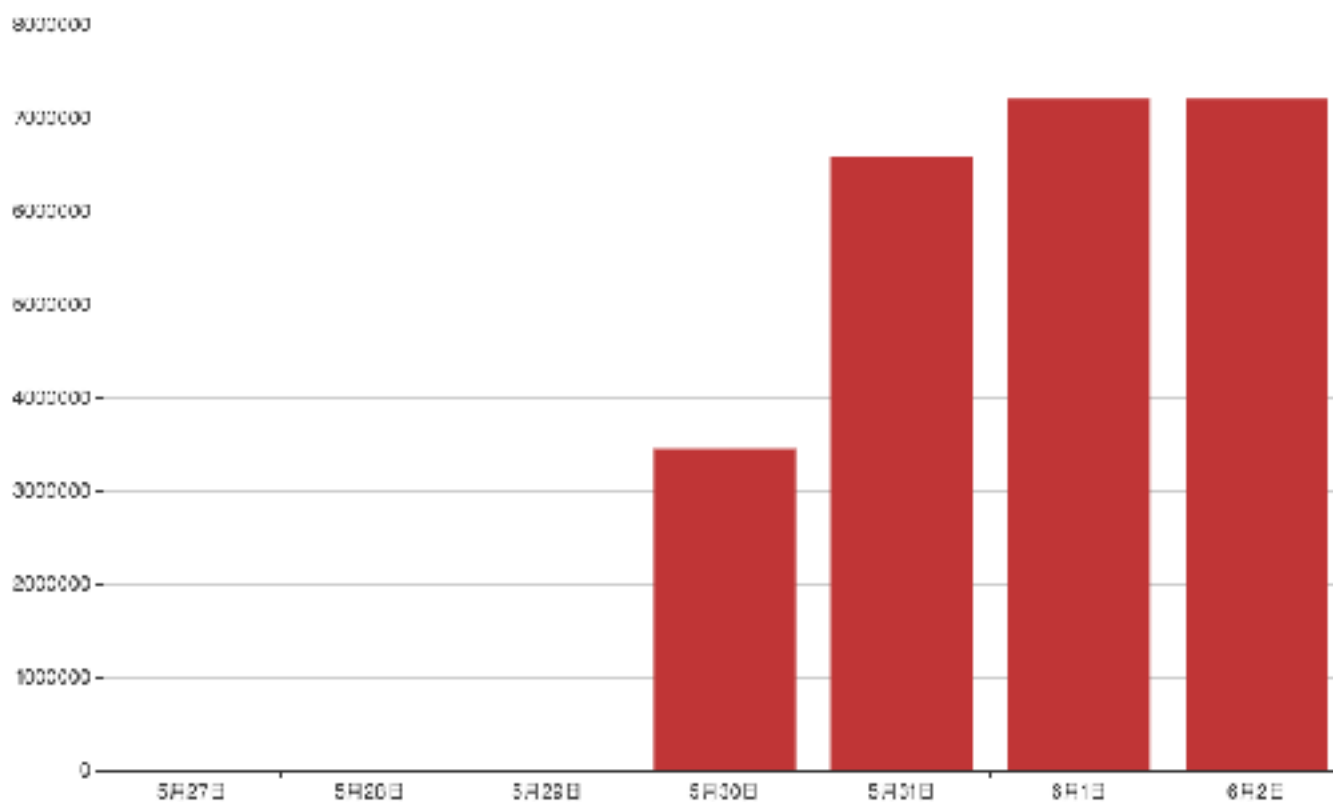
7天

30天

增量

总量

自定义 柱状图 折线图 饼图 下载



# 发布系统

## 发布数据报表

circle日活分布

日期: 20180601



Android

IOS

选择版本

☒ 全选

☒ 20 ☒ 24 ☒ 76 ☒ 15 ☒ 31 ☒ 77 ☒ 79

PV



UV



# 发布系统

## 找到对应jsbundle排查问题

QQ浏览器RN管理平台 正式环境

管理员入口

首页

AudioApp

打包发布

发布记录

数据报表

circle

comment

Demo

dobby

eeexplre

新构建线

构建线用于对打包记录做分组，一条打包线对应一个git分支或tag

新建

搜索bundle

输入bundle的打包id或RDM异常上报id

67

搜索

打包id	平台	创建时间	打包人	备注	操作
67	Android	2018-05-22 14:20:69	saizhao	--	下载bundle 访问构建线

## 构建记录

发布线

以下构建线打出的bundle均可用于正式发布，请谨慎操作

构建线名称	创建人	创建时间	备注	操作
资讯正文加载小程序	saizhao	2018-05-23 15:43:40	--	编辑 删除
资讯正文加载小程序	saizhao	2018-05-23 11:13:00	--	编辑 删除

# 转web方案



react-native-web

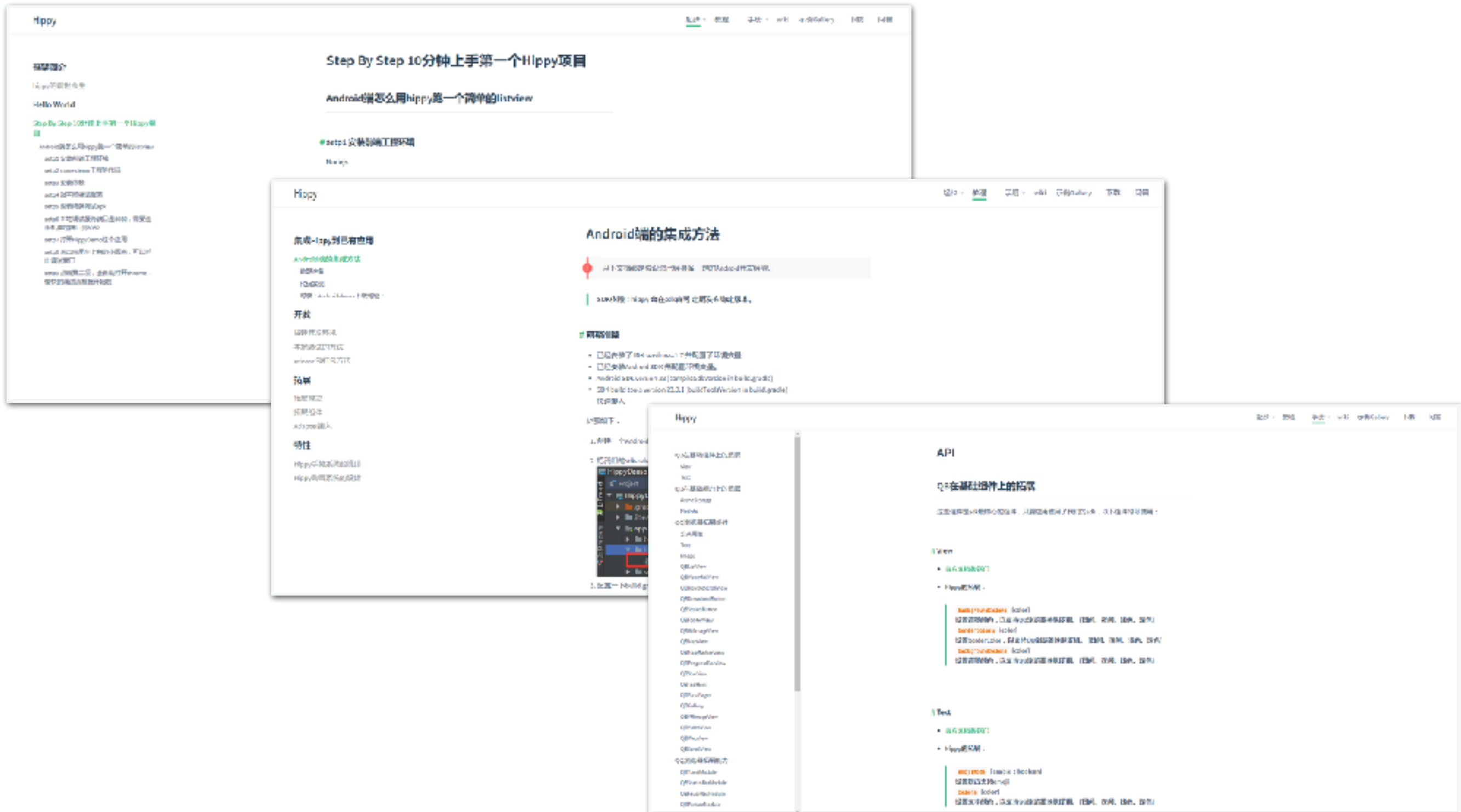
添加组件

优化手势实现

<http://qbrnweb.html5.qq.com>

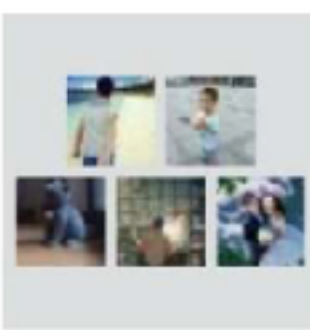


# 文档建设



</ReactNative性能优化 & 企业级应用>

谢 谢 大 家, 欢 迎 交 流



QQ浏览器RN交流群



该二维码7天内(6月9日前)有效，重新进入将更新