

Android ReactNative

1RCT

2RCT58

3RCT

4

RCT

React NativeJavaScriptReactAPP

Learn once, write anywhere

ReactAndroid/iOSApp RCTApp RCTRCT > Write once, run anywhere

Android/iOSFE

RN

- WebViewWebView + NativeJsNative
- NativeX
-
- RN...

Native App

- JsApp

RN

- Android RNAndroid
-

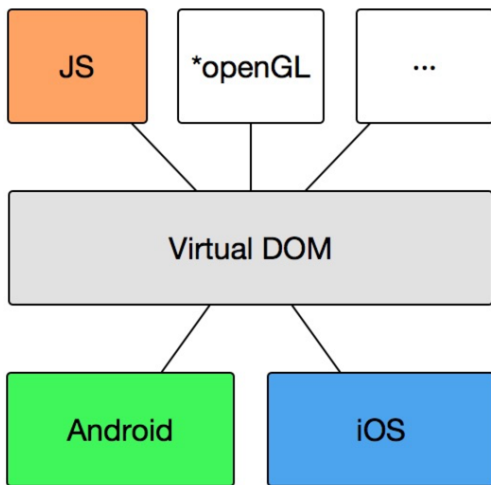
RNJS IOSjavascriptcore Android webkit.orgjsc.so frescookhttp RN

```

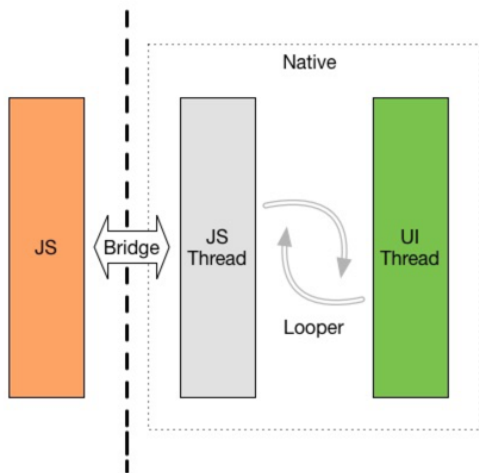
_releaseCompile - ## Internal use, do not manually configure ##
+--- com.android.support:appcompat-v7:23.0.0
|   \--- com.android.support:support-v4:23.0.0
|       \--- com.android.support:support-annotations:23.0.0
\--- com.facebook.react:react-native:0.11.+ -> 0.11.0
+--- com.google.code.findbugs:jsr305:3.0.0
+--- com.squareup.okhttp:okhttp:2.4.0
|   \--- com.squareup.okio:okio:1.4.0 -> 1.5.0
+--- org.webkit:android-jsc:r174650
+--- com.facebook.fresco:fresco:0.6.1
|   +--- com.facebook.fresco:drawee:0.6.1
|       +--- com.facebook.fresco:fbcore:0.6.1
|           \--- com.android.support:support-v4:21.0.3 -> 23.0.0 (*)
+--- com.facebook.fresco:imagepipeline:0.6.1
|   +--- com.nineoldandroids:library:2.4.0
|   +--- com.facebook.fresco:fbcore:0.6.1
|       +--- com.android.support:support-v4:21.0.3 -> 23.0.0 (*)
|       \--- com.parse.bolts:bolts-android:1.1.4
|   \--- com.facebook.fresco:fbcore:0.6.1
+--- com.facebook.fresco:imagepipeline-okhttp:0.6.1
|   +--- com.facebook.fresco:imagepipeline:0.6.1 (*)
|   +--- com.squareup.okhttp:okhttp:2.3.0 -> 2.4.0 (*)
|       \--- com.facebook.fresco:fbcore:0.6.1
+--- com.fasterxml.jackson.core:jackson-core:2.2.3
+--- com.android.support:appcompat-v7:22.2.0 -> 23.0.0 (*)
+--- com.squareup.okhttp:okhttp-ws:2.4.0
|   \--- com.squareup.okhttp:okhttp:2.4.0 (*)
\--- com.squareup.okio:okio:1.5.0

```

RN



RN JSframeworkbundle,RNjs jsAPIIbridgenative;UIvirtual DOMJSbridgenativenativeView bridgeJS JAVABridgemodulemethod



RCT58

React Native

- react-native init xxxRN
- 58cloneandroid
- WubaRNLibRNmodule
- gale.propertiesandroid.useDeprecatedNdk=true
- android/appreact.gradleWubaRNLib;
- WubaRNLib.gradle

apply from: "react.gradle"

RNcloneRNgradleRNAar

RN

- gradle

mavenLocal()

- gradle

compile "com.facebook.react:react-native:+"
RNFbmaven0.24

RN

- gradle

compile "com.facebook.react:react-native:0.20.+"
RN

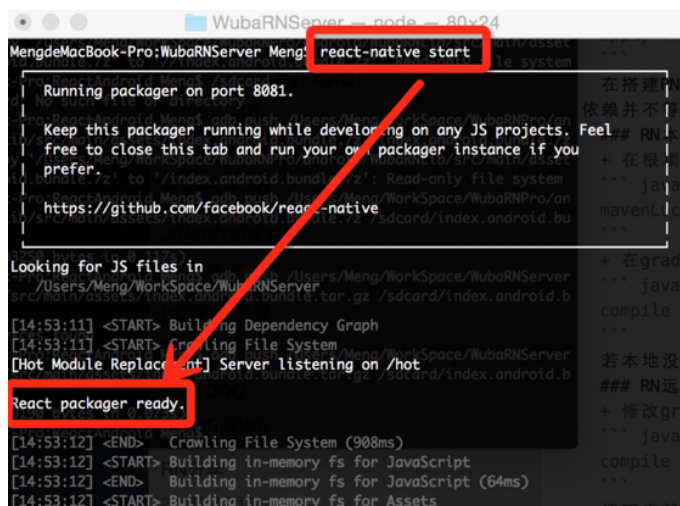
RCT

RNbundleRN react.gradlebundle

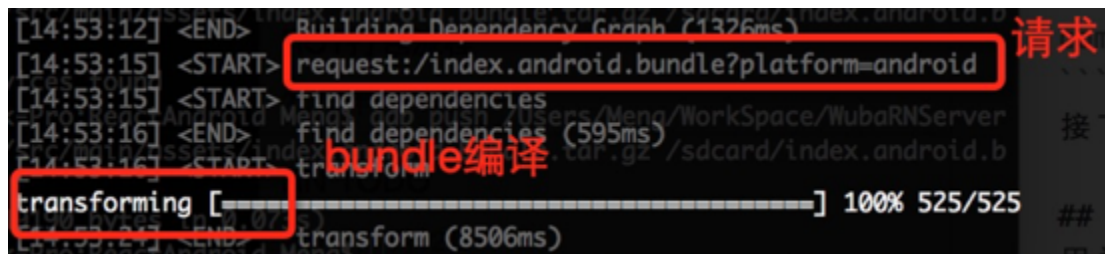
```
workingDir reactRoot

// Set up dev mode
def devEnabled = !targetName.toLowerCase().contains("release")
if (Os.isFamily(Os.FAMILY_WINDOWS)) {
    commandLine "cmd", "/c", "react-native", "bundle", "--platform", "android", "--dev", "${devEnabled}",
    "--entry-file", entryFile, "--bundle-output", jsBundleFile, "--assets-dest", resourcesDir
} else {
    commandLine "react-native", "bundle", "--platform", "android", "--dev", "${devEnabled}",
    "--entry-file", entryFile, "--bundle-output", jsBundleFile, "--assets-dest", resourcesDir
}
```

RNreact-native startRN
React packager ready

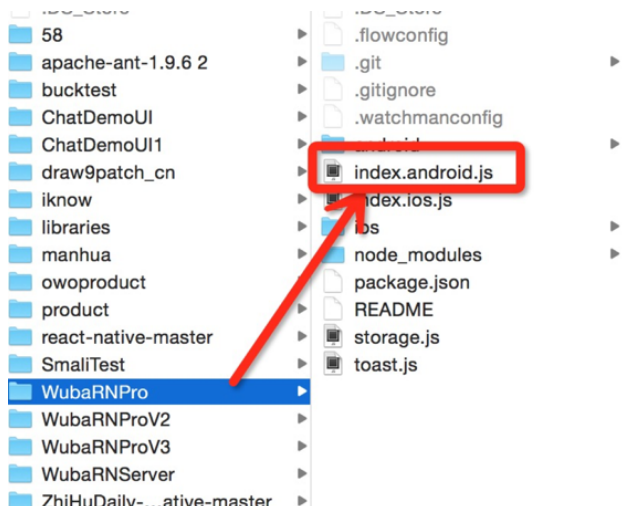


Native AppRNNative Appbundle



bundleNative AppRN
bundleReactJs

Js



index.android.jsAndroidRN
index.android.jsstorage.jstoast.js

```
'use strict';
import React, {
  AppRegistry,
  Component,
  StyleSheet,
  Text,
  View
} from 'react-native';

var Toast = require('./toast');
var Storage = require('./storage');

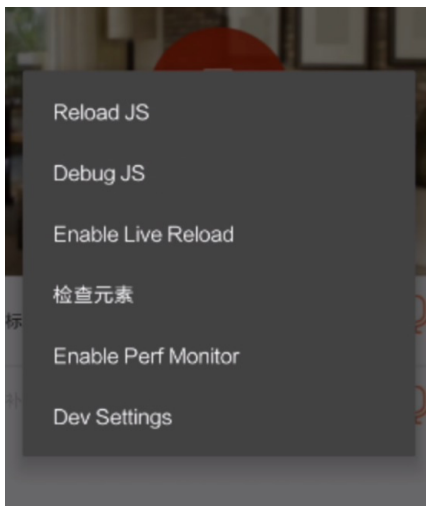
Storage.put("rntest","com from rn",(result)=>{
  Toast.show("result = "+result,Toast.SHORT);
});

class WubaRNPro extends Component {
  render() {
```

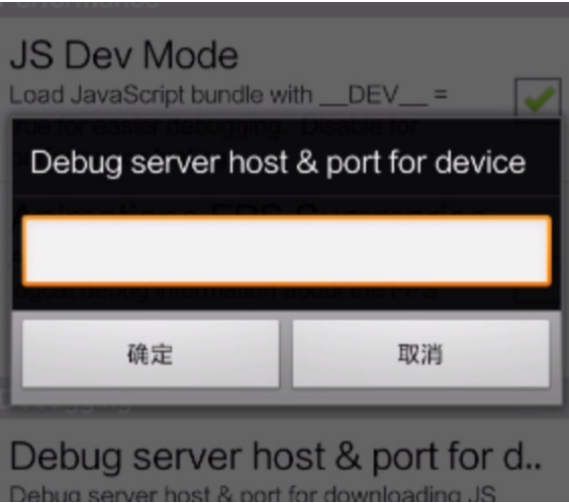
index.android.jsstorage.jstoast.jsindex.android.bundle
bundle

`curl "http://localhost:8081/index.android.bundle?platform=android" -o "android/app/src/main/assets/index.android.bundle"`

bundledown
RNiplocalhostRN
RNRN



ip



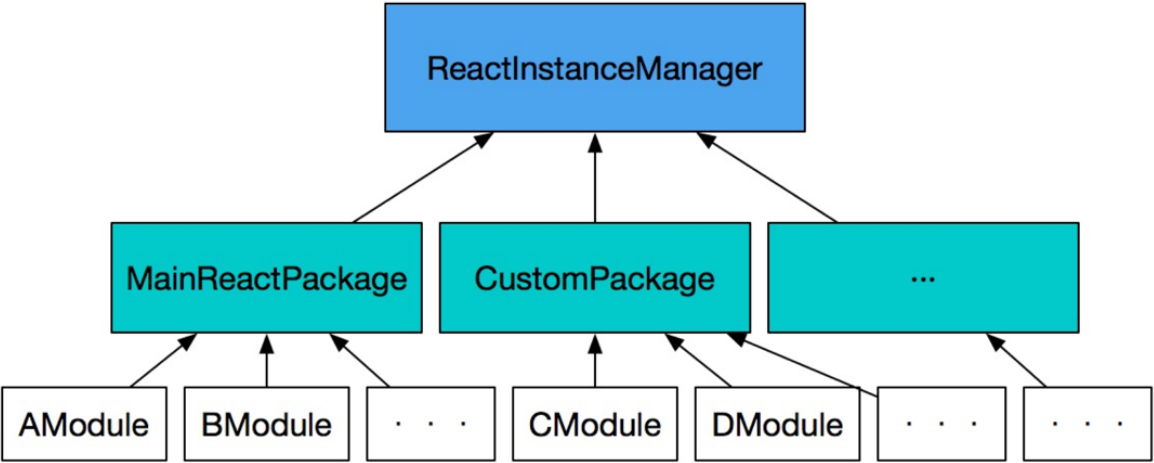
MIUI

RNdebugreleaseddebugreleasejsbundleassets react.gradle bundleRN release

ReactNativeCan't find variable: __fbBatchedBridge

1RN Native

RN Native



ReactInstanceManagerRN Native""RN

```

ReactInstanceManager.Builder build = ReactInstanceManager.builder()
    .setApplication(application)
    //js
    .setJSMainModuleName("index.android")
    //RNPackage
    .addPackage(new MainReactPackage())
    // Package
    .addPackage(new WubaRCTPackage())
    // Debug
    .setUseDeveloperSupport(BuildConfig.DEBUG)
    // React
    .setInitialLifecycleState(LifecycleState.BEFORE_RESUME);
if (RCTUpdateService.getInstance().isLocalBundleExsists()) {
    // bundle
    build.setJSBundleFile(RCTUpdateService.getInstance().getLocalBundlePath());
} else {
    // bundleasserts
    build.setBundleAssetName("index.android.bundle");
}
mRNManager = build.build();

```

Packagecom.facebook.react.ReactPackage

```

public interface ReactPackage {

    /**
     * @param reactContext react application context that can be used to create modules
     * @return list of native modules to register with the newly created catalyst instance
     */
    List<NativeModule> createNativeModules(ReactApplicationContext reactContext);

    /**
     * @return list of JS modules to register with the newly created catalyst instance.
     *
     * IMPORTANT: Note that only modules that needs to be accessible from the native code should be
     * listed here. Also listing a native module here doesn't imply that the JS implementation of it
     * will be automatically included in the JS bundle.
     */
    List<Class<? extends JavaScriptModule>> createJSMODULES();

    /**
     * @return a list of view managers that should be registered with {@link UIManagerModule}
     */
    List<ViewManager> createViewManagers(ReactApplicationContext reactContext);
}

```

- List<NativeModule> createNativeModules(ReactApplicationContext reactContext) modules
- List<ViewManager> createViewManagers(ReactApplicationContext reactContext) UImodules
- List<Class<? extends JavaScriptModule>> createJSMODULES() Native

module

ReactContextBaseJavaModule

```

/**
 * @return the name of this module. This will be the name used to {@code require()} this module
 * from javascript.
 */
String getName();

```

NativeJsNative“Wuba”React“Wuba”Native

```
AppRegistry.registerComponent('Wuba', () => AwesomeProject);
```

UI

SimpleViewManager

```

/**
 * @return the name of this view manager. This will be the name used to reference this view
 * manager from JavaScript in createReactNativeComponentClass.
 */
public abstract String getName();
/**
 * Subclasses should return a new View instance of the proper type.
 * @param reactContext
 */
protected abstract T createViewInstance(ThemedReactContext reactContext);

createViewInstanceThemedReactContextReactContextReactContextContextWrapperThemedReactContextContext

public class ReactContext extends ContextWrapper {

```

2RN

RNsoRN7.3MRN

RN

- armeabi-v7ax86so
-
-

1so

armeabiarmeabi-v7ax86socopyarmeabiWubaRNLlib/libs/armeabiRNsoWubaRNLlibbuild.gradleso

```

abi {
    reset()
    enable enableSeparateBuildPerCPUArchitecture
    universalApk false // If true, also generate a universal APK
    include "armeabi"
}

```

4.4M60.2%

2

RNokHttpfrescoRNRNRNRNRN

3

RN

3RN4.1

RN4.1ChoreographerChoreographerRN

4NativeJs

-

1. public
2. void;
- 3.

@ReactMethod


```

@ReactMethod
public void show(String text,int duration){
    Toast.makeText(getReactApplicationContext(),text,duration).show();
}

```

Reacttoast.js

```

'use strict';
var{ NativeModules } = require('react-native');
var RCTToast = NativeModules.Toast;
var Toast = {
    SHORT : 500,
    LONG : 1000,
    show : function(message : string , duration : number) : void {
        RCTToast.show(message,duration);
    }
}
module.exports = Toast;

```

index.android.js

```

var Toast = require('toast')
Taost.show("Hello word",Toast.SHORT);

```

- ToastNativeReactToast.SHORTToast.LONG ReactContextBaseJavaModulepublic Map<String, Object> getConstants()

```

@Override
public Map<String, Object> getConstants() {
    Map<String,Object> contents = new HashMap<>();
    contents.put(OutputParamKeys.TOAST_LONG.key(), Toast.LENGTH_LONG);
    contents.put(OutputParamKeys.TOAST_SHORT.key(),Toast.LENGTH_SHORT);
    return contents;
}

```

React

```

'use strict';
var{ NativeModules } = require('react-native');
var RCTToast = NativeModules.Toast;
var Toast = {
    SHORT : RCTToast.SHORT,
    LONG : RCTToast.LONG,
    show : function(message : string , duration : number) : void {
        RCTToast.show(message,duration);
    }
}
module.exports = Toast;

```

-

ReactNativeNativeReactReactNativeNativeAPIReact callback

```

import com.facebook.react.bridge.Callback

@ReactMethod
public void getResult(String url,final Callback callback){
    new Thread(new Runnable() {
        @Override
        public void run() {
            try {
                ...
                callback.invoke(true,result);
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    }).start();
}

```

- Js

NativeReact

```

/**
 * Js
 */
private void pushPhotoPathsToJs(ArrayList<PicItem> picItems) {
    if(picItems == null){
        return;
    }
    WritableArray array = Arguments.createArray();
    for (PicItem item : picItems) {
        array.pushString(item.serverPath);
    }
    WritableMap param = Arguments.createMap();
    param.putString("coverLocalPath", picItems.size() == 0 ? "0" : picItems.get(0).path);
    param.putArray("imgsArray", array);
    ReactContext rctCtx = mRCTManager.getCurrentReactContext();
    if (rctCtx != null) {
        rctCtx.getJSModule(DeviceEventManagerModule.RCTDeviceEventEmitter.class)
            .emit("RCTPhotoPaths", param);
    }
}

• Js;

```

1

RNJsreleaseRNbundleRNRNManifestActivity/FragmentbundlebundleRN

RNbundle""hello worldbundle1.2MRNbundlebundleapkRNbundleapk(1.2± x N)M bundlebundlebundle

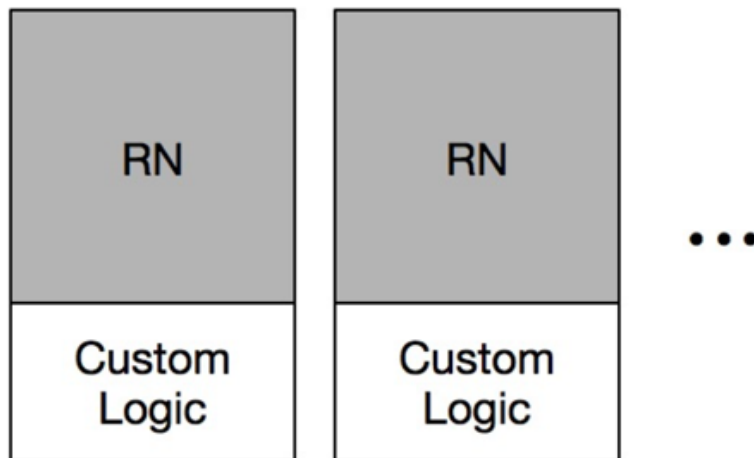
```

var ReactNative=babelHelpers.extends({

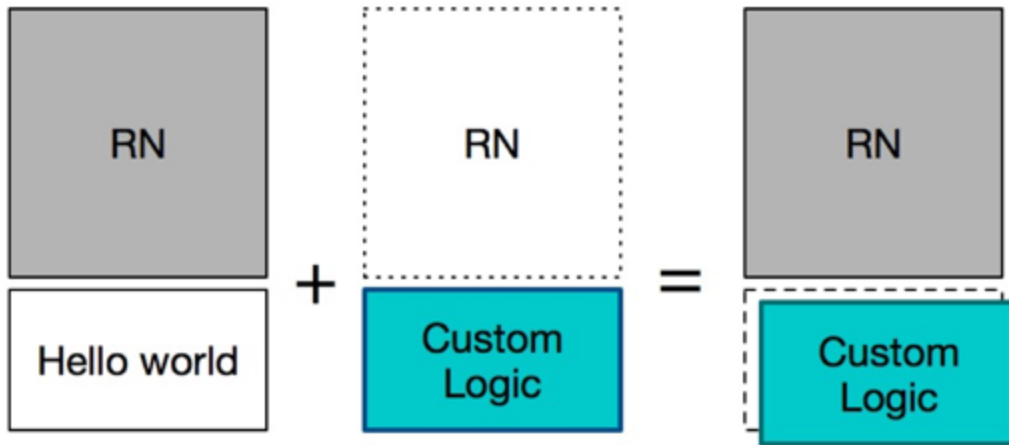
get ActivityIndicatorIOS(){return require(12 /* ActivityIndicatorIOS */);},
get ART(){return require(343 /* ReactNativeART */);},
get DatePickerIOS(){return require(349 /* DatePickerIOS */);},
get DrawerLayoutAndroid(){return require(356 /* DrawerLayoutAndroid */);},
get Image(){return require(358 /* Image */);},
get ImageEditor(){return require(359 /* ImageEditor */);},
get ImageStore(){return require(360 /* ImageStore */);},
get ListView(){return require(361 /* ListView */);},
get MapView(){return require(371 /* MapView */);},
get Modal(){return require(372 /* Modal */);},
get Navigator(){return require(373 /* Navigator */);},
get NavigatorIOS(){return require(392 /* NavigatorIOS */);},
get Picker(){return require(393 /* Picker */);},
get PickerIOS(){return require(394 /* PickerIOS */);},
get ProgressBarAndroid(){return require(396 /* ProgressBarAndroid */);},
get ProgressViewIOS(){return require(397 /* ProgressViewIOS */);},
get ScrollView(){return require(364 /* ScrollView */);},
get SegmentedControlIOS(){return require(398 /* SegmentedControlIOS */);},
get SliderIOS(){return require(399 /* SliderIOS */);},
get SnapshotViewIOS(){return require(400 /* SnapshotViewIOS */);},
get Switch(){return require(401 /* Switch */);},
get PullToRefreshViewAndroid(){return require(402 /* PullToRefreshViewAndroid */);},
get RecyclerViewBackedScrollView(){return require(403 /* RecyclerViewBackedScrollView */);},
get RefreshControl(){return require(404 /* RefreshControl */);},
get StatusBar(){return require(405 /* StatusBar */);},
get SwitchAndroid(){return require(406 /* SwitchAndroid */);},
get SwitchIOS(){return require(407 /* SwitchIOS */);},
get TabBarIOS(){return require(408 /* TabBarIOS */);},
get Text(){return require(350 /* Text */);},
get TextInput(){return require(409 /* TextInput */);},
get ToastAndroid(){return require(54 /* ToastAndroid */);},
get ToolbarAndroid(){return require(418 /* ToolbarAndroid */);},
get Touchable(){return require(351 /* Touchable */);},
get TouchableHighlight(){return require(419 /* TouchableHighlight */);},
get TouchableNativeFeedback(){return require(421 /* TouchableNativeFeedback */);},
get TouchableOpacity(){return require(422 /* TouchableOpacity */);},
get TouchableWithoutFeedback(){return require(416 /* TouchableWithoutFeedback */);},
get View(){return require(289 /* View */);},
get ViewPagerAndroid(){return require(431 /* ViewPagerAndroid */);},
get WebView(){return require(432 /* WebView */);},


```

bundlebundleRNbundle



bundleRNbundleRNbundle



[google-diff-match-patch](#)

patchpatch

patch

```

@@ -23486,21 +23486,17 @@
Native);
-var
%0A
+
%0A%0A%0A%0A%0A%0A%0A%0A
@@ -23507,16 +23507,443 @@
%0A%0A%0A%0A%0A%0A%0A%0A
+%0Avar _MainPage=require(531 /* ./component/MainPage */);
%0A%0AAwesom
@@ -24378,16 +24378,22 @@
tNative.
+Scroll
View,null
@@ -24389,24 +24389,216 @@
ollView,null
+,%0A_reactNative2.default.createElement(_SelectPhoto2.def
));%7D%7D%5D);retu
@@ -24644,16 +24644,17 @@
ent);%0A%0A%0A
+%0A
%0A%0A_react
@@ -1131130,25 +1131130,25 @@
e:require(53
-1
+8
/* ./assets
@@ -1131561,17 +1131561,17 @@
;%0A__d(53
-1
+8
/* reac
@@ -1196957,126 +1196957,26823 @@
%7D);%0A
-:require(16):%0A:require(25):%0A:require(0):%0A SSTOKENS

```

bundleblack.bundlepatch30kgz6k RN10k

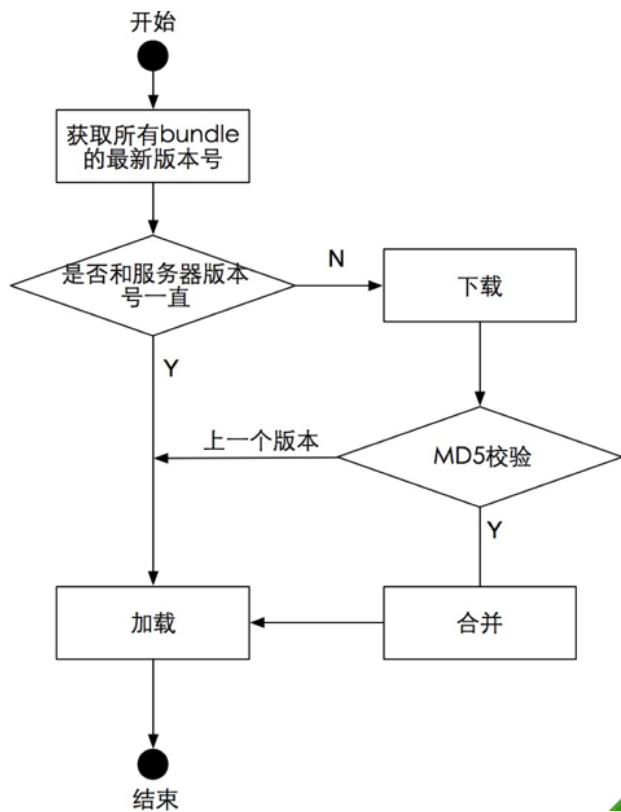
- bsdiff
- courgette

patch

MD5

补丁下载			
日期	下载链接	大小	MD5
2016/04/26	patch2.0.587.3w2补丁>>	90512KB	4e7c436e18acd1d878b258e748896403
2016/04/19	patch2.0.586.3w2补丁>>	67527KB	b4ba19d49a77a4d565eab37ebe6da447
2016/04/12	patch2.0.585.3w2补丁>>	78683KB	cf482030d7c03f1e0cda697902e3019d

ServerpatchNativepatchMD5NativepatchpatchMD5ServerpatchMD5



bundle

ReactInstanceManagerbundle

```
// bundle/sdcard/index.android.bundle
setJSBundleFile(String jsBundleFile)
```

```
// assertbundleassets://index.android.bundle
setBundleAssetName(String bundleAssetName)
```

```
ReactInstanceManager.Builder builder = ReactInstanceManager.builder()
    .setApplication(getApplication())
    .setJSMainModuleName(getJSMainModuleName())
    .setUseDeveloperSupport(getUseDeveloperSupport())
    .setInitialLifecycleState(mLifecycleState);
```

```
for (ReactPackage reactPackage : getPackages()) {
    builder.addPackage(reactPackage);
}
```

```
String jsBundleFile = getJSBundleFile();
```

```
if (jsBundleFile != null) {
    builder.setJSBundleFile(jsBundleFile);
} else {
    builder.setBundleAssetName(getBundleAssetName());
}
```

```
return builder.build();
```

ReactInstanceManagerbundle

ReactInstanceManagerBuilderReactInstanceManagerbundlebundleRNActivitybundlesetJSBundleFile

```

/**
 * Path to the JS bundle file to be loaded from the file system.
 *
 * Example: {@code "assets://index.android.js" or "/sdcard/main.jsbundle"}
 */
public Builder setJSBundleFile(String jsBundleFile) {
    mJSBundleFile = jsBundleFile;
    return this;
}

mJSBundlesetJSBundleFile

return new ReactInstanceManagerImpl(
    Assertions.assertNotNull(
        mApplication,
        "Application property has not been set with this builder"),
    mJSBundleFile,
    mJSMainModuleName,
    mPackages,
    mUseDeveloperSupport,
    mBridgeIdleDebugListener,
    Assertions.assertNotNull(mInitialLifecycleState, "Initial lifecycle state was not set"),
    mUIImplementationProvider,
    mNativeModuleCallExceptionHandler);
}

ReactInstanceManagerReactInstanceManagerImplmJSBundleFileReactInstanceMangerImpl

/* package */ ReactInstanceManagerImpl(
    Context applicationContext,
    @Nullable String jsBundleFile,
    @Nullable String jsMainModuleName,
    List<ReactPackage> packages,
    boolean useDeveloperSupport,
    @Nullable NotThreadSafeBridgeIdleDebugListener bridgeIdleDebugListener,
    LifecycleState initialLifecycleState,
    UIImplementationProvider uiImplementationProvider,
    NativeModuleCallExceptionHandler nativeModuleCallExceptionHandler,
    @Nullable JSCConfig jscConfig) {
    initializeSoLoaderIfNecessary(applicationContext);

    // TODO(9577825): remove this
    ApplicationHolder.setApplication((Application) applicationContext.getApplicationContext());
    setDisplayMetrics(applicationContext);

    mApplicationContext = applicationContext;
    mJSBundleFile = jsBundleFile;

    ReactInstanceManagerImplmJSBundleFileReactInstanceManagersetJSBundleFile
    mJSBundleFile

    private void recreateReactContextInBackgroundFromBundleFile() {
        recreateReactContextInBackground(
            new JSCJavaScriptExecutor.Factory(),
            JSBundleLoader.createFileLoader(mApplicationContext, mJSBundleFile));
    }

    mJSBundleFileJSBundleLoaderJSBundleLoaderrecreateReactContextInBackgroundrecreateReactContextInBackgroundmJ
    SBundleFile

```

```

private void resetBundleReflected(String bundleLocation) {
    if (mRNManager == null
        || TextUtils.isEmpty(bundleLocation)) {
        return;
    }
    Class<?> RCTManagerClazz = mRNManager.getClass();
    try {
        Method method = RCTManagerClazz.getDeclaredMethod("recreateReactContextInBackground",
JavaScriptExecutor.Factory.class, JSBundleLoader.class);
        method.setAccessible(true);
        method.invoke(mRNManager, new JSCJavaScriptExecutor.Factory(),
JSBundleLoader.createFileLoader(mApplicationCtx, bundleLocation));
    } catch (NoSuchMethodException e) {
        e.printStackTrace();
    } catch (InvocationTargetException e) {
        e.printStackTrace();
    } catch (IllegalAccessException e) {
        e.printStackTrace();
    }
}

```

2RN

Q

1RNbundle

2RN

3RNso