

移动端静态资源缓存方案 - CandyWebCache

张建伟@网易前端技术部

通用、可扩展、开源

CandyWebCache

主要内容

一、由来与简介

二、功能与特点

三、集成与使用

四、设计与实现

五、项目集成还需要额外做的事儿

六、协议扩展

一、由来与简介

移动App开发需要的静态资源有哪些？

Game

图片

音频

视频

HotFix

JSPatch

ReactNative

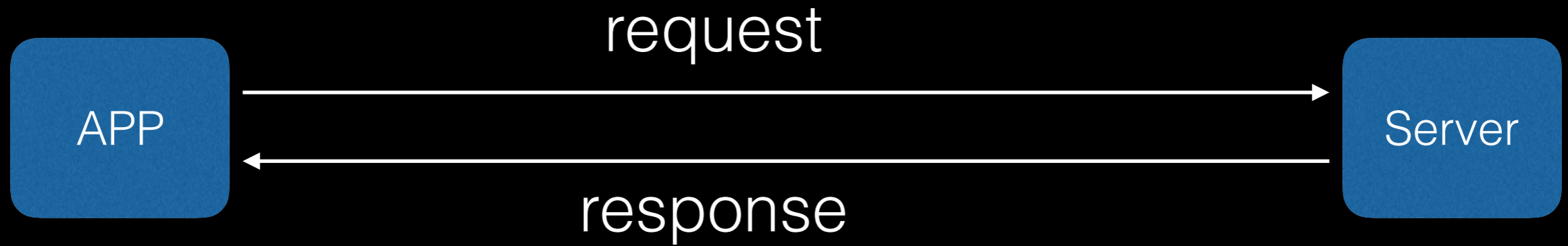
WebView

HTML

CSS

JS

静态资源更新的流程?



更新过程中面临的共同问题?

受网络状态影响 资源的缓存与使用 版本管理

流量节省 资源同步 资源安全

移动端WebView加载页面资源使用上的问题

- 加载阶段白屏、菊花
- 首次加载必须走网络
- 缓存机制受HTTP协议+WebView自身控制
- 由于缓存容量大小等原因，缓存资源可能会被清理掉
- HTML5离线应用Manifest只支持全量更新，单个html页面只能支持单个manifest文件



受网络状态影响 资源的缓存与使用 版本管理
流量节省 资源同步 资源安全

CandyWebCache

What's CandyWebCache



移动端web资源的离线缓存解决方案，能够拦截webview的请求，并优先使用本地缓存静态资源进行响应，以此来对webview加载页面性能进行优化。

Why CandyWebCache

- 解决WebView的资源缓存问题
- 解决离线资源缓存与更新的通用问题



二、功能与特点

CandyWebCache解决WebView的问题

- 本地预缓存Web资源
- 请求透明拦截，优先使用本地资源进行响应，提升页面响应速度，流量节省
- 资源缓存策略的配置与管理，对缓存资源具有更强的控制力
- 离线资源打包到安装包，可以做到首次请求不走网络

CandyWebCache解决WebView的问题

网易考拉活动页性能测试数据

	iOS	Android
2G	92.6%	92.3%
3G	68.5%	58.0%
4G	48.9%	30.8%
Wifi	46.7%	21.2%

测试说明

- 开始请求页面到页面DOM树解析完成(domInteractive)耗时
- iOS: iPhone6s,iOS9.3 Android: Galaxy Nexus,Android4.2.2

CandyWebCache解决离线资源缓存通用问题

- 资源的版本管理
- 资源的下载与更新
- 增量更新与全量更新
- 完整性校验
- 项目构建时自动打包

CandyWebCache其他特点

- 缓存命中加速
- 线程安全
- 配置灵活
- 协议可扩展
- 模块低耦合

三、集成与使用

- 工程引入CandyWebCache SDK
- 配置CandyWebCache：服务器地址、自动更新周期、缓存黑名单、内存缓存大小、是否增量更新、禁用CandyWebCache等
- 在合适的时间主动请求资源更新：app启动、前台切换、收到推送通知等
- 调试与测试：日志开关、日志级别、测试服务器测试

配置信息

```
CCCandyWebCacheConfig* config = [CCCandyWebCacheConfig new];  
config.serverAddress = @"https://m.kaola.com";  
config.appName = @"kaola";  
config.appVersion = @"1.0.1";  
config.blackListResourceTypes = @[@"html"];  
[CCCandyWebCache setDefaultConfiguration:config];
```

检查更新

```
[[CCCandyWebCache defaultWebCache] checkAndUpdateResource];
```

调试

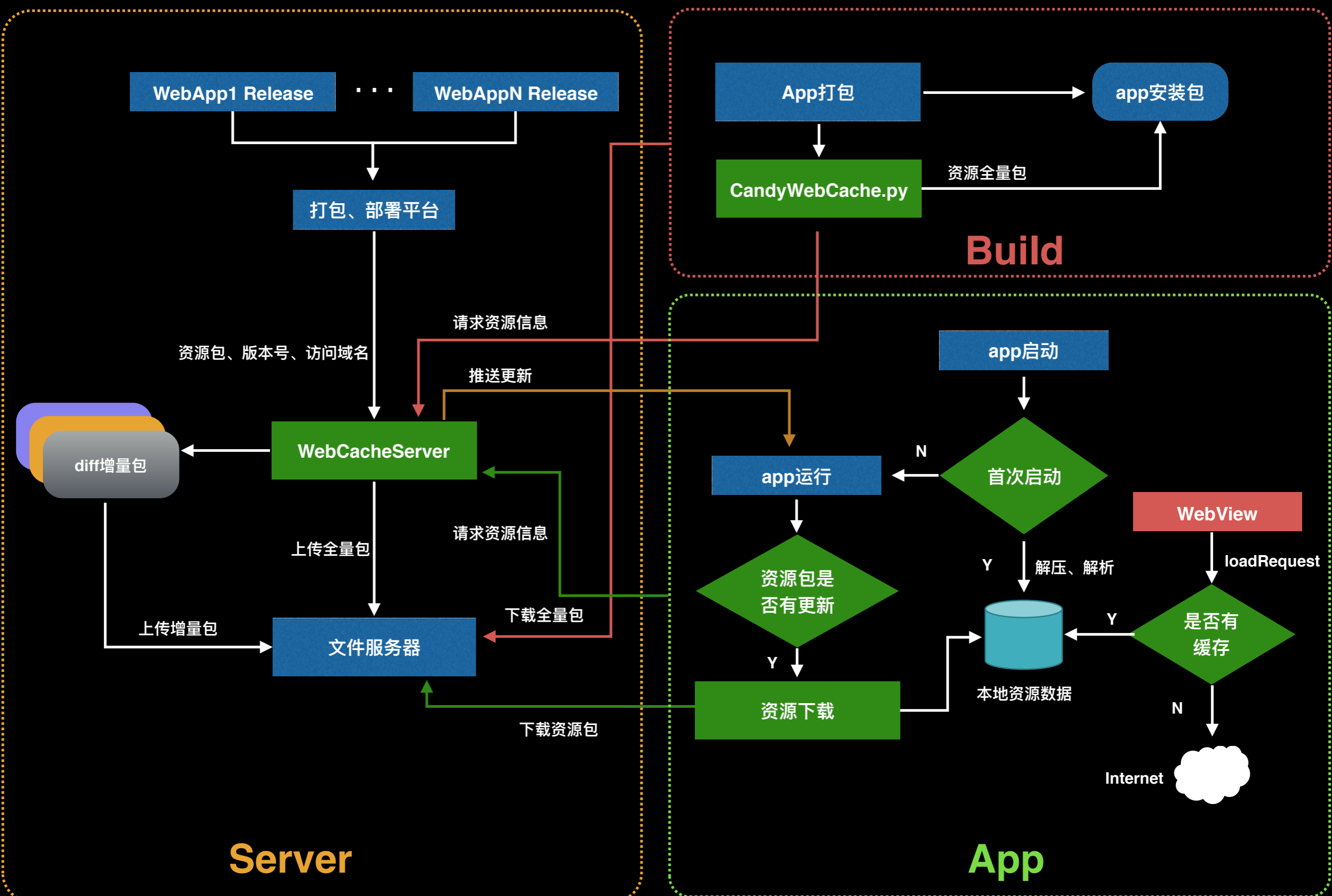
```
[CCandyWebCache setLogEnable:YES];  
[CCandyWebCache setLogLevel:CCLoggerLevelVerbose];
```

全局开关

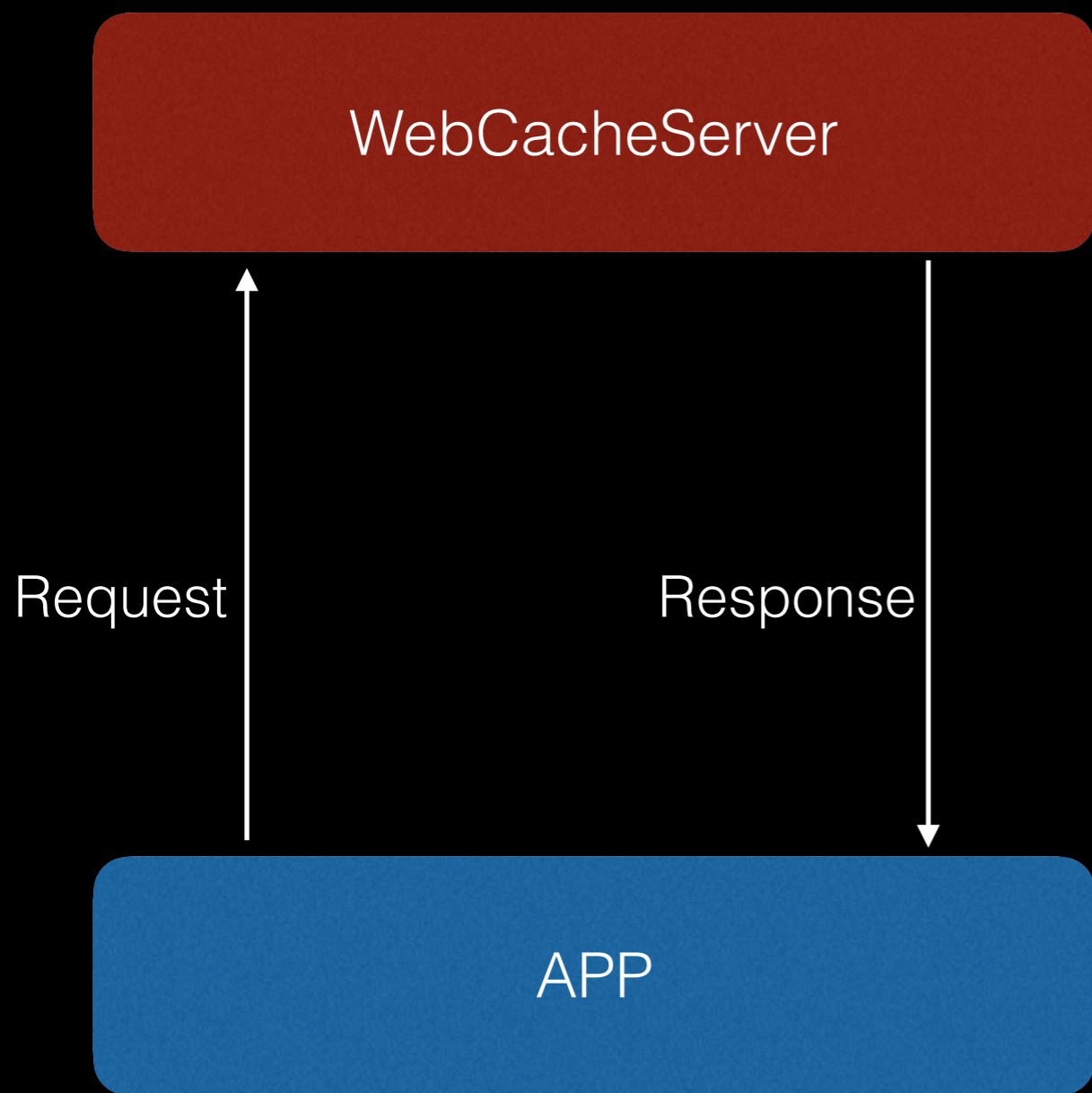
```
[CCandyWebCache defaultWebCache].enable = YES;  
[CCandyWebCache defaultWebCache].diffEnable = YES;
```

四、设计与实现

三、设计与实现 - 整体架构



资源包版本检测

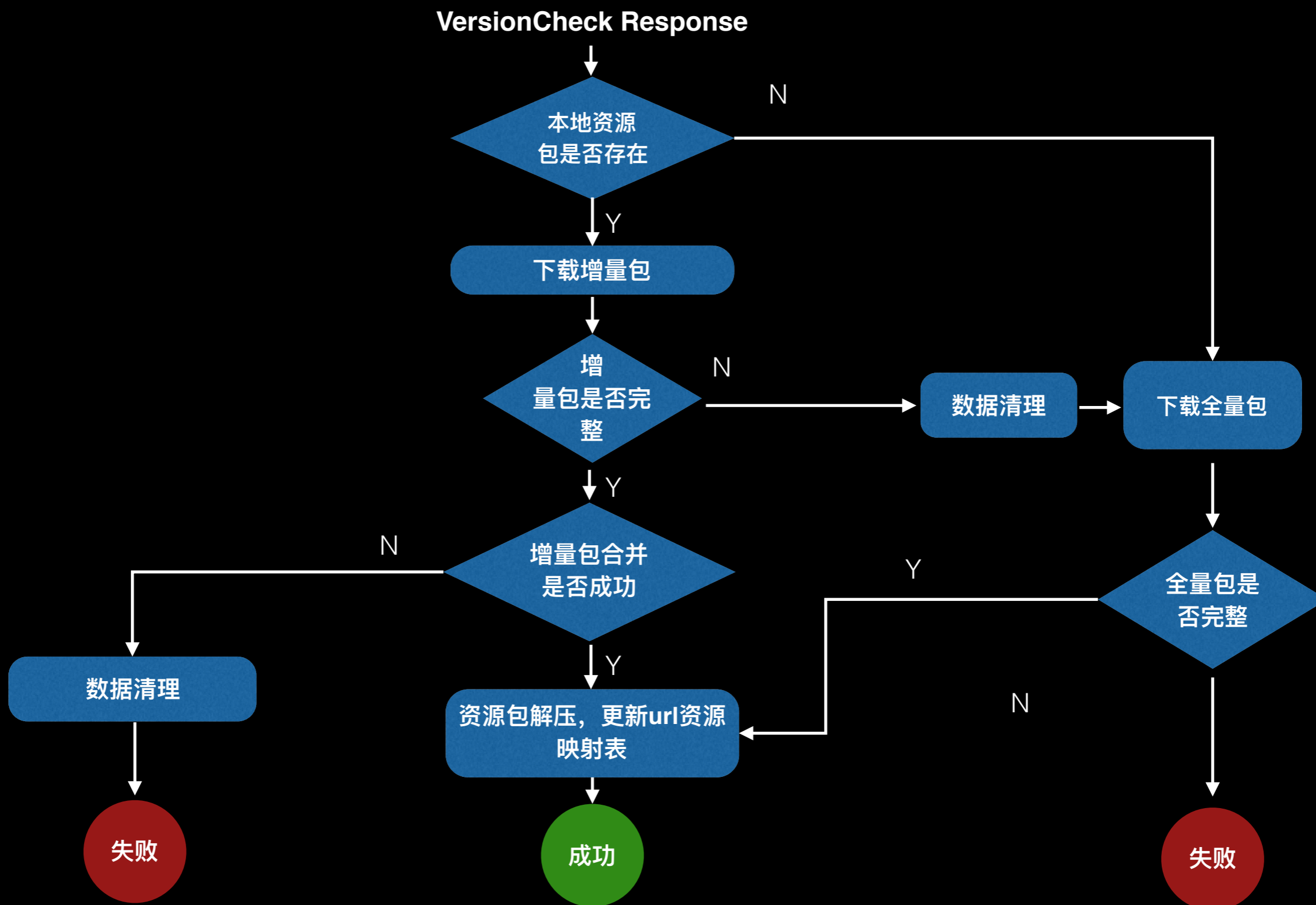


VersionCheck Response

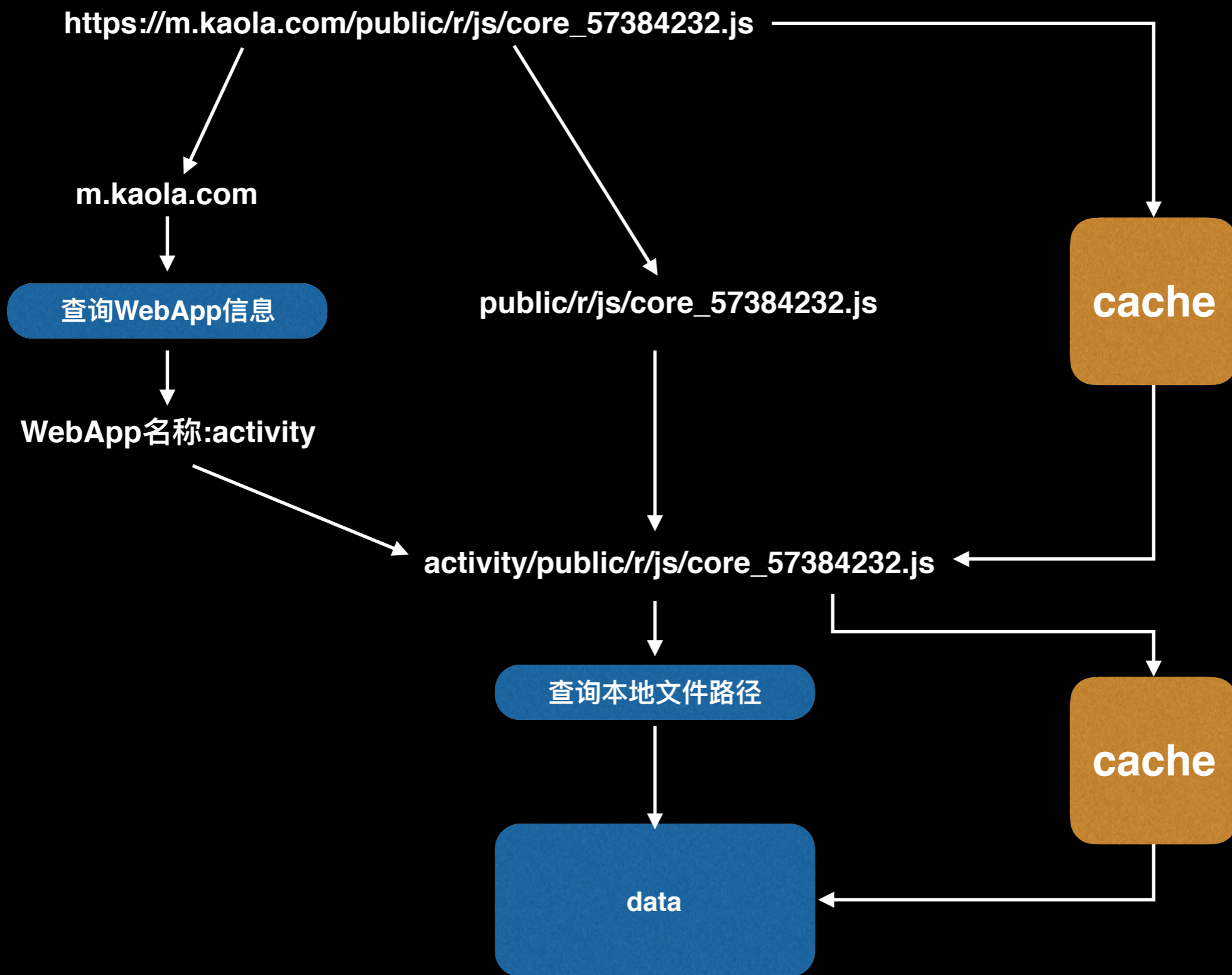
资源包信息：

- 资源名称
- 资源版本号
- 资源状态
- **VersionCheck Request**
- 增量包地址
- 增量包MD5
- 增量包版本
- 增量包MD5
- 资源包所在域名
- 手机系统
- 是否增量更新
- 是否添加新部署webapp
- 本地资源包信息：名称、版本号
- 协议扩展

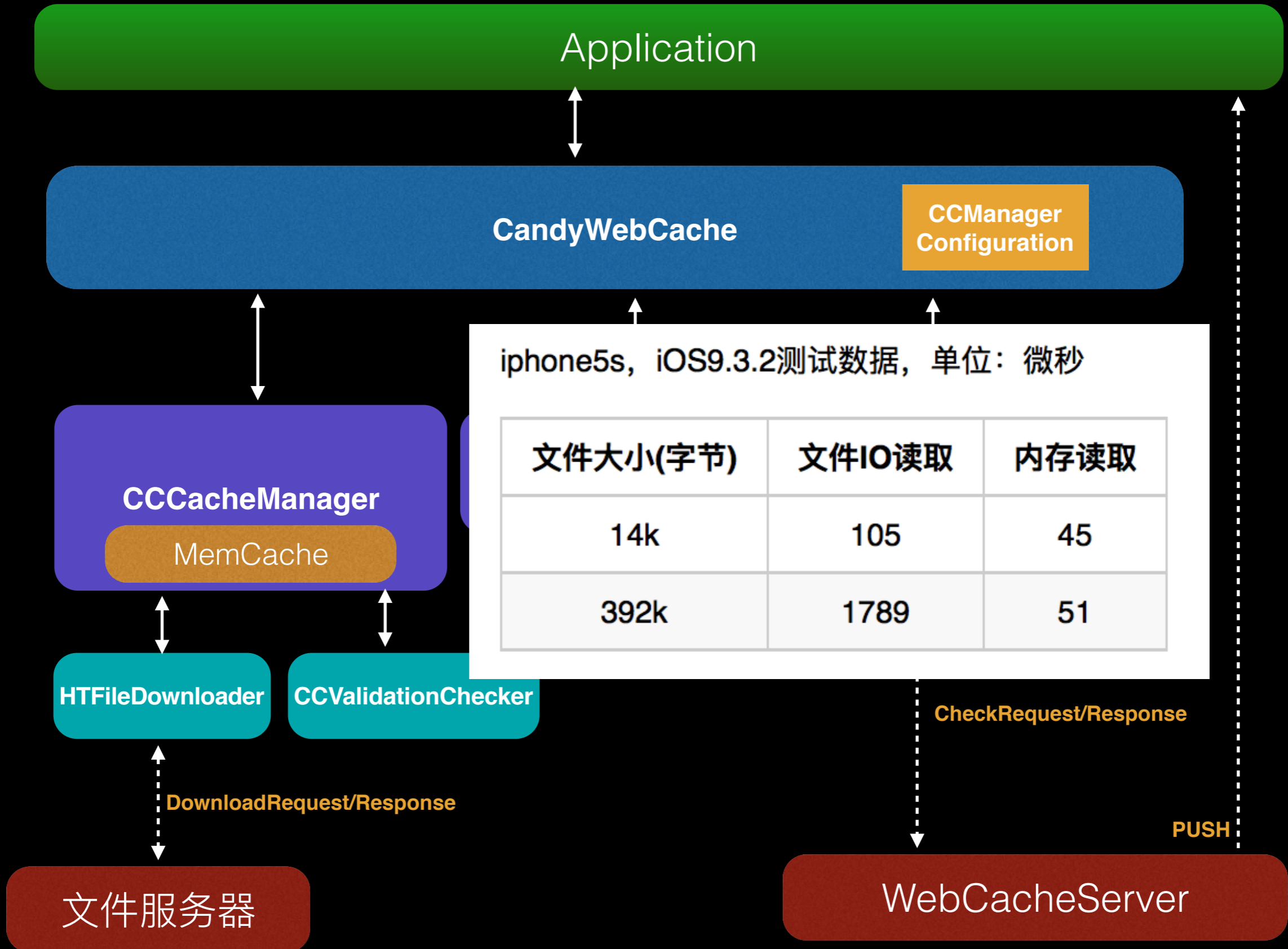
资源包更新



资源本地缓存命中规则



三、设计与实现 - 客户端SDK



五、项目集成还需要额外做的事儿

实现WebCacheServer的功能

- 资源包的上传与增量包的计算
- 资源包的MD5值的计算与加密
- 资源包域名信息的配置
- 资源包(全量包、增量包)上传至文件服务器
- 实现版本检测协议接口

CandyWebCache Demo Server:

<https://github.com/NEYouFan/ht-candywebcache-demo-server>

Web资源打包方式的要求

- 访问资源文件URL格式：需要将文件版本号打包到文件名上

`http://m.kaola.com/public/r/js/core_57384232.js`

支持

`http://m.kaola.com/public/r/js/core.js?v=57384232`

不支持

- 打包资源包目录相对路径要跟url能够对应

`http://m.kaola.com/public/r/js/core_57384232.js`

[domain]: m.kaola.com $\xrightarrow{\text{资源文件相对路径}}$ public/r/js/core_57384232.js

[domain]: m.kaola.com/public $\xrightarrow{\text{资源文件相对路径}}$ r/js/core_57384232.js

对于不满足要求的url，不会命中本地缓存资源，走网络请求。

力所能及的带来了一点甜头

CandyWebCache名字的由来



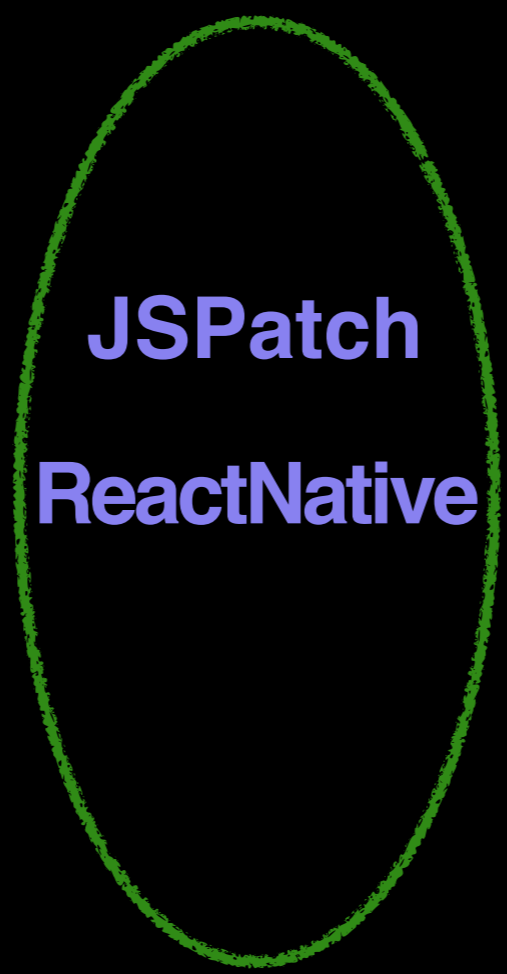
六、协议扩展

移动App开发需要的静态资源有哪些？

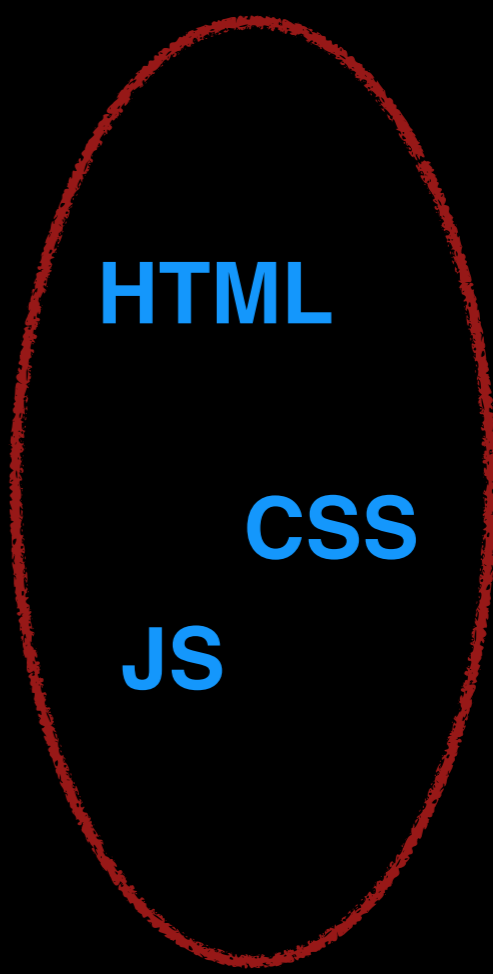
Game



HotFix



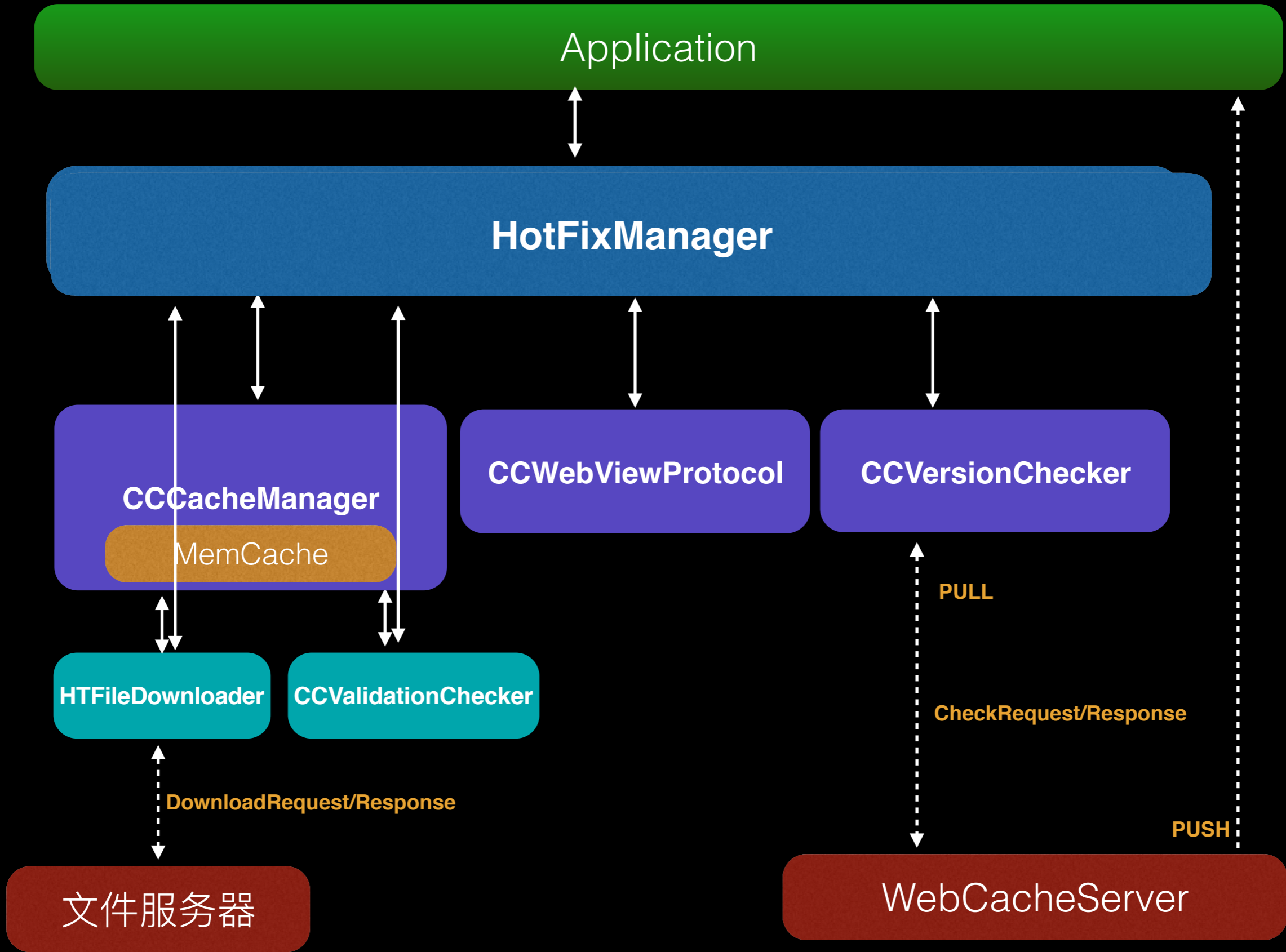
WebView



资源下载、资源更新、版本管理、资源安全

CandyWebCache通信协议支持扩展

- Web静态资源
- Hot Fix 代码资源
- 图片、音视频等其他需要做版本管理和更新的资源



项目开源地址

- [**candywebcache-ios**]

<https://github.com/NEYouFan/ht-candywebcache-ios>

- [**candywebcache-android**]

<https://github.com/NEYouFan/ht-candywebcache-android>

- [**candywebcache-demo-server**]

<https://github.com/NEYouFan/ht-candywebcache-demo-server>

Thanks~