

HDWebViewHost 介绍

HDWebViewHost 是一套解决 H5 和 native 协作开发的整体框架和服务

原理

往 WKWebView 注入 JavaScript 脚本，使得 window 对象拥有了一些能力，与 web 端约定对应的触发以及接收出口、入口实现互通功能

职责介绍

HDWebViewHost

1		└─ Core	核心功能，可独立存在
2		└─ Preloader	预渲染控制器
3		└─ RemoteDebug	本地/远端 debug

Core

1		Core	
2		└─ Category 帮助方法	
3		└─ NSBundle+HDWebViewHost.h	包含获取 framework bundle 方法
4		└─ NSBundle+HDWebViewHost.m	
5		└─ NSObject+HDWebViewHost.h	
6		└─ NSObject+HDWebViewHost.m	
7		└─ HDWHJSCoreManager.h	暂无实现，将来可用于管理H5、小程序、Flutter 等容器
8		└─ HDWHJSCoreManager.m	
9		└─ HDWHSchemeTaskDelegate.h	
10		└─ HDWHSchemeTaskDelegate.m	
11		└─ HDWebViewHost.h	
12		└─ HDWebViewHostEnum.h	
13		└─ HDWebViewHostProtocol.h	
14		└─ HDWebViewHostResponse.h	
15		└─ HDWebViewHostResponse.m	
16		└─ HDWebViewHostViewController+Dispatch.h	h5调用native接口的分发器
17		└─ HDWebViewHostViewController+Dispatch.m	
18		└─ HDWebViewHostViewController+Extend.h	对 webview 什么周期有特殊需要的时候，可以继承 H
19		└─ HDWebViewHostViewController+Extend.m	

```

20 |— HDWebViewHostViewController+Progressor.h 处理顶部进度条
21 |— HDWebViewHostViewController+Progressor.m
22 |— HDWebViewHostViewController+Scripts.h 注入脚本相关
23 |— HDWebViewHostViewController+Scripts.m
24 |— HDWebViewHostViewController+Timing.h 性能测试相关, 暂时只记录时间
25 |— HDWebViewHostViewController+Timing.m
26 |— HDWebViewHostViewController+Utils.h 其他扩展函数
27 |— HDWebViewHostViewController+Utils.m
28 |— HDWebViewHostViewController.h 核心渲染
29 |— HDWebViewHostViewController.m
30 |— HTMLParser 加载本地 html 解析器
31 |   |— HTMLNode.h
32 |   |— HTMLNode.m
33 |   |— HTMLParser.h
34 |   |— HTMLParser.m
35 |   └— README.md
36 |— Helper 一些与业务无关的工具函数, 比如记录某页面的历史滚动位置
37 |   |— HDWHScriptMessageDelegate.h
38 |   |— HDWHScriptMessageDelegate.m
39 |   |— HDWHUtil.h
40 |   |— HDWHUtil.m
41 |   |— HDWHWebViewScrollPositionManager.h
42 |   |— HDWHWebViewScrollPositionManager.m
43 |   |— HDWebViewHostCookie.h
44 |   └— HDWebViewHostCookie.m
45 |— Intermediate 处理网页内容的调度器
46 |   |— HDWHRequestMediate.h
47 |   └— HDWHRequestMediate.m
48 |— Resources 资源文件家
49 |   |— app-access.txt 白名单, 包含 scheme 和 host 白名单, 如果都为空则不过滤
50 |   |— eval.js 为了替换 self.webView evaluateJavaScript:javaScriptString completionH
51 |   └— webViewHost_version_1.0.0.js JSSDK
52 |— Response 为 h5 提供的业务相关的实现
53 |   |— HDWHAppLoggerResponse.h
54 |   |— HDWHAppLoggerResponse.m
55 |   |— HDWHBuiltInResponse.h
56 |   |— HDWHBuiltInResponse.m
57 |   |— HDWHDebugResponse.h
58 |   |— HDWHDebugResponse.m
59 |   |— HDWHNavigationBarResponse.h
60 |   |— HDWHNavigationBarResponse.m
61 |   |— HDWHNavigationResponse.h
62 |   |— HDWHNavigationResponse.m
63 |   |— HDWHResponseManager.h
64 |   └— HDWHResponseManager.m
65 |— URLChecker 检查器, 判断白名单或验证地址有效性
66 |   |— HDWHAppWhiteListParser.h
67 |   └— HDWHAppWhiteListParser.m

```

```
68 |   └─ HDWHURLChecker.h
69 |   └─ HDWHURLChecker.m
```

Preloader

```
1 | Preloader  预渲染页面，需要时取出
2 |   └─ ResourcePreloader
3 |     └─ HDWHSimpleWebViewController.h
4 |     └─ HDWHSimpleWebViewController.m
5 |     └─ HDWHWebViewPreLoader.h
6 |     └─ HDWHWebViewPreLoader.m
7 |   └─ ViewControllerPreRender
8 |     └─ HDWHViewControllerPreRender.h
9 |     └─ HDWHViewControllerPreRender.m
10 |   └─ html
11 |     └─ preload_resources.html
```

RemoteDebug

```
1 | RemoteDebug  调试、联调用
2 |   └─ GCDWebServer  修改的第三方的 webServer库，加入日志记录功能
3 |     └─ Core
4 |       └─ GCDWebServer.h
5 |       └─ GCDWebServer.m
6 |       └─ GCDWebServerConnection.h
7 |       └─ GCDWebServerConnection.m
8 |       └─ GCDWebServerFunctions.h
9 |       └─ GCDWebServerFunctions.m
10 |      └─ GCDWebServerHTTPStatusCodes.h
11 |      └─ GCDWebServerPrivate.h
12 |      └─ GCDWebServerRequest.h
13 |      └─ GCDWebServerRequest.m
14 |      └─ GCDWebServerResponse.h
15 |      └─ GCDWebServerResponse.m
16 |     └─ Requests
17 |       └─ GCDWebServerDataRequest.h
18 |       └─ GCDWebServerDataRequest.m
19 |       └─ GCDWebServerFileRequest.h
20 |       └─ GCDWebServerFileRequest.m
21 |       └─ GCDWebServerMultiPartFormRequest.h
22 |       └─ GCDWebServerMultiPartFormRequest.m
23 |       └─ GCDWebServerURLEncodedFormRequest.h
24 |       └─ GCDWebServerURLEncodedFormRequest.m
```

```

25 |   └─ Responses
26 |       └─ GCDWebServerDataResponse.h
27 |       └─ GCDWebServerDataResponse.m
28 |       └─ GCDWebServerErrorResponse.h
29 |       └─ GCDWebServerErrorResponse.m
30 |       └─ GCDWebServerFileResponse.h
31 |       └─ GCDWebServerFileResponse.m
32 |       └─ GCDWebServerStreamedResponse.h
33 |       └─ GCDWebServerStreamedResponse.m
34 | └─ HDWHDebugServerManager.h  下面几个文件是悬浮按钮和日志页的实现
35 | └─ HDWHDebugServerManager.m
36 | └─ HDWHDebugViewController.h
37 | └─ HDWHDebugViewController.m
38 | └─ HDWebViewHostAuxiliaryEntryWindow.h
39 | └─ HDWebViewHostAuxiliaryEntryWindow.m
40 | └─ HDWebViewHostAuxiliaryMainWindow.h
41 | └─ HDWebViewHostAuxiliaryMainWindow.m
42 | └─ src    启动本地 server 加载的本地网页目录，会被 webServer 拦截处理响应操作
43 |     └─ components
44 |         └─ tool-panel.js
45 |     └─ favicon.ico
46 |     └─ images
47 |         └─ mobile\ to\ pc.png
48 |         └─ pc\ to\ mobile.png
49 |     └─ logo.png
50 |     └─ profile
51 |         └─ pageTiming.js
52 |         └─ pageTiming_for_mac.js
53 |         └─ profiler.js
54 |         └─ profiler_for_mac.js
55 |     └─ renderjson.css
56 |     └─ renderjson.js
57 |     └─ server.css
58 |     └─ server.html
59 |     └─ server.js
60 |     └─ testcase.tpl
61 |     └─ thirdParty
62 |         └─ weinreSupport.js
63 |     └─ vue.js

```

开发新业务

新建业务类

继承自 `HDWebViewHostResponse`，需要导入头文件

#import <HDSERVICEKIT/HDWebViewHostResponse.h> , 实现对应协议

```
1 + (NSDictionary<NSString *, NSString *>)supportActionList {
2     return @{
3         @"method1_$": kHDWHResponseMethodOn,
4         @"method2": kHDWHResponseMethodOn,
5         @"method3_": kHDWHResponseMethodOn,
6     };
7 }
```

kHDWHResponseMethodOn 作用为开关该方法，也可后做他用，不设计为数组是此目的

实现对应函数

```
1 - (void) method2 {
2     HDWHLog(@"%@", NSStringFromSelector(_cmd));
3 }
4
5 - (void)method1:(NSDictionary *)dict callback:(NSString *)callback {
6
7     NSString *text = [dict objectForKey:@"text"];
8     HDWHLog(@"%@ --- %@", NSStringFromSelector(_cmd), text);
9
10    dispatch_after(dispatch_time(DISPATCH_TIME_NOW, (int64_t)(2.0 * NSEC_PER_SEC)), di
11        // 触发回调
12        [self fireCallback:callback param:@{@"cbParam": @"你是什么品种的蛤蟆? "});
13    });
14 }
15
16 - (void)method3:(NSDictionary *)dict {
17     NSString *text = [dict objectForKey:@"text"];
18     HDWHLog(@"%@ -- %@", NSStringFromSelector(_cmd), text);
19 }
```

对应函数名和参数以及回调请查看 HDWebViewHostResponse 的

handleAction:withParam:callbackKey: 方法

编写测试用例

比如为 method1 添加测试文档，具体使用方法查看 HDWebViewHostEnum 文件或者模仿内置的 response 写法

```
1 // clang-format off
2 wh_doc_begin(method1_$, "测试 h5 调用原生并且收到回调")
3 wh_doc_param(text, "字符串, 亲之前你要说什么话? ")
4 wh_doc_code(window.webViewHost.invoke('method1', { "text": "送你花花" }, function (p)
5 wh_doc_code_expect("原生收到调用后2秒回调被触发, h5 收到消息")
6 wh_doc_end;
7 // clang-format on
```

启动 webServer 之后, 浏览器打开调试页面点击"用例"或者执行命令 `:testcase` 可在客户端查看测试用例界面, 一个 App 生命周期内只生成一次测试用例页面, 调试界面的所有功能可在帮助下查看, 不在文档赘述