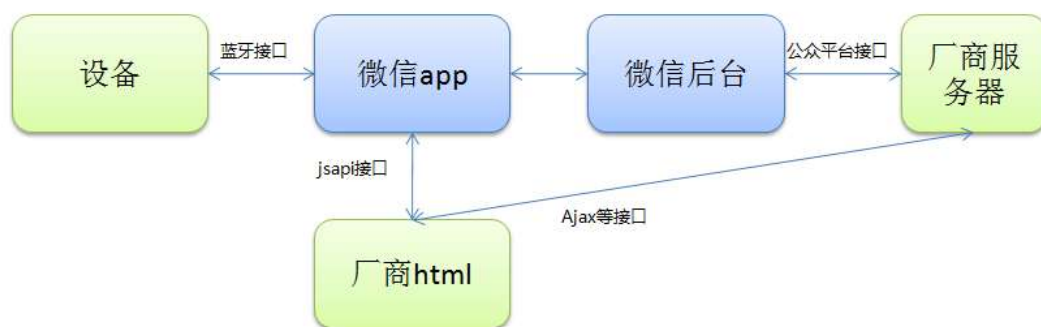


概述

时间	版本	变更	变更人
2014 年 12 月 12 日	1.0.5	修改 startScanWXDevice 接口，扫描时需指定蓝牙类型	Hakonzhao
2014 年 11 月 17 日	1.0.4	修改 configWXDeviceWiFi 接口	harl
2014 年 11 月 13 日	1.0.3	加入 onWXDeviceBluetoothStateChange，修改 onScanWXDeviceResult 名称	harl
2014 年 10 月 31 日	1.0.2	加入 configWXDeviceWiFi	clintonliu
2014 年 10 月 28 日	1.0.2	更改 onScanWXDevicesResult 参数说明	yuetong
2014 年 10 月 23 日	1.0.1	更改 getWXDeviceTicket 参数说明	yuetong
2014 年 10 月 22 日	1.0	创建文档	harl、ryan

微信 jsapi 提供给厂商，在网页通过 javascript，对设备操作的接口。

例如扫描设备，连接设备，收发数据，绑定设备等。



Html 通过 Jsapi 可以和设备本地收发数据（即 Html 发送给微信客户端，微信客户端发给设备），无需通过服务器中转，所以速度较快。

实时性要求高的蓝牙设备（如遥控汽车）可采用 jsapi 收发数据。

用户场景

用户打开微信，进入硬件公众号，点击菜单，进入厂家自定义的 html 界面。

用户点击 html 界面上的扫描按钮，开始扫描周围设备。当扫到多个设备的时候，html 界

面显示一个设备列表。

用户点击某个设备(该设备未绑定),进行绑定。html 界面显示绑定成功 ,并开始连接设备。

Html 界面显示连接设备成功。

用户点击 Html 界面上的开始同步按钮。界面显示进度条，再显示同步成功。

接口

Jsapi 包括函数和事件两部分。

接口版本 1.0(内测)

函数

函数名	openWXDeviceLib
描叙	初始化设备库
参数	无
返回值	err_msg：openWXDeviceLib:ok 成功，openWXDeviceLib:fail 失败 minVersion：当前微信客户端支持的最小 jsapi 版本 maxVersion：当前微信客户端支持的最大 jsapi 版本 bluetoothState：值有 on、off、restting、unauthorized、unknow。 unauthorized 表示用户没有授权微信使用蓝牙功能。 isSupportBLE：值有 yes、no、unkown

函数名	closeWXDeviceLib
描叙	关闭设备库
参数	无
返回值	err_msg : closeWXDeviceLib:ok 成功 , closeWXDeviceLib:fail 失败。

函数名	getWXDeviceInfos
描叙	获取设备信息
参数	无
返回值	err_msg : getWXDeviceInfos:ok 成功 , getWXDeviceInfos:fail 失败。 deviceInfos : {deviceInfos : [{"deviceId":"myDevice1", "state":"connected"}, {"deviceId":" myDevice2", "state":"connected"}]} deviceInfos 包含多个 deviceInfo , 每个 deviceInfo 形如 : {"deviceId":"myDevice1", "state":"connected"}。

函数名	sendDataToWXDevice
描叙	发送数据给设备
参数	deviceId (必填) : 设备 id base64Data (必填) : 数据 , 经过 base64 编码后的字符串。
返回值	err_msg :sendDataToWXDevice:ok 成功 , sendDataToWXDevice:fail 失败。

函数名	startScanWXDevice
-----	-------------------

描述	扫描设备
参数	btVersion: 需要扫描的设备蓝牙版本（IOS 可不填此参数，直接扫描 BC 和 BLE 蓝牙设备。Android 只能指定扫描一种蓝牙版本的设备，所以需要指定此参数）。参数取值为”ble”或”bc”
返回值	err_msg : startScanWXDevice:ok 成功， startScanWXDevice:fail 失败。

函数名	stopScanWXDevice
描述	停止扫描设备
参数	无
返回值	err_msg : stopScanWXDevice:ok 成功， stopScanWXDevice:fail 失败。

函数名	connectWXDevice
描述	连接设备
参数	deviceId（必填）：设备 id
返回值	err_msg : connectWXDevice:ok 成功， connectWXDevice:fail 失败。

函数名	disconnectWXDevice
描述	断开设备连接
参数	deviceId（必填）：设备 id
返回值	err_msg : disconnectWXDevice:ok 成功， disconnectWXDevice:fail

	失败。
--	-----

函数名	getWXDeviceTicket
描叙	获取操作凭证
参数	deviceId (必填) : 设备 id type (必填) : 获取的操作凭证类型, 1:绑定设备 2:解绑设备
返回值	err_msg : getWXDeviceTicket:ok 成功, getWXDeviceTicket:fail 失败。 ticket:操作凭证

函数名	configWXDeviceWiFi
描叙	调起原生 AirKiss 界面, 不需要先调用 openWXDeviceLib
参数	key (可选) : base64 编码的 AirKiss 的密钥
返回值	configWXDeviceWiFi:ok //配置成功 configWXDeviceWiFi:fail //超时 configWXDeviceWiFi:cancel //用户取消 //返回 res.desc, 取值如下: wifi_not_connected //当 res.err_msg 为 config_wx_device_wifi:cancel 时 //其它情况为空

事件

事件名	onWXDeviceBindStateChange
描述	微信客户端设备绑定状态改变事件
参数	deviceId (必填) : 设备 id state (必填) : bind 绑定 , unbind 解绑

事件名	onWXDeviceStateChange
描述	接收到设备数据
参数	deviceId (必填) : 设备 id state(必填) : connecting 连接中 , connected 连接上 , disconnected 连接断开。

事件名	onReceiveDataFromWXDevice
描述	接收到设备数据
参数	deviceId (必填) : 设备 id base64Data (必填) : base64 编码过的设备发到 H5 的数据。

事件名	onScanWXDeviceResult
描述	扫描到某个设备
参数	devices(必填) : 发现的设备列表数组。数组元素字段有 deviceId , base64BroadcastData。

	<p><i>deviceId (必填) : 设备 id。</i></p> <p><i>base64BroadcastData (非必填) : 如果是普通包 (即以 mac 结尾) , 为空。如果是确认包 (即以 0xfe 0x01 0x01 + mac 结尾) , 为广播包里的整个 manufacture data 的 base64 编码过的内容。(普通包和确认包的具体规定见微信蓝牙外设协议)</i></p> <p><i>isCompleted (必填) : 是否扫描结束。</i></p>
--	---

函数名	onWXDeviceBluetoothStateChange
描叙	手机蓝牙状态改变事件
参数	无
返回值	state : 值有 on、off

API 说明

对设备的使用流程可分为初始化库，扫描，连接，收发数据，断开连接，关闭库。

前提

在硬件公众号里打开的网页才可以调用设备 jsapi。

每个公众号的网页只能操作属于该公众号的设备。

初始化库

Html 需先调用 `openWXDeviceLib`，进行库的初始化。微信 app 会启动 html5 的硬件功能，并会把相关的事件传递给 Html。

当 Html 不想使用硬件功能的时候，调用 `closeWXDeviceLib`。微信 app 释放资源,并不再发送事件给 Html。

当调用了 `openWXDeviceLib` 之后，一般接着调用 `getWXDeviceInfos`，获取该用户绑定了的设备列表，以及设备状态。然后监听 `onWXDeviceBindStateChange`，更新已绑定的设备状态。

扫描设备

正常情况无需扫描设备，直接连接设备即可。除非是想绑定设备。

当扫描到某个设备的时候，html 会收到事件回调（`onScanWXDevicesResult`）。对于同一种类型的广播包（目前有两种类型：普通的广播包，确认包），无论手机收到多少个广播包，html 也只会收到一次事件回调。

连接设备

当 html 发现设备状态为未连接，可直接连接设备，无需扫描。

收发数据

当连上设备之后，即可对设备进行收发数据。

收发数据前需进行 base64 的编解码。

断开连接

当不想用设备时，即可断开连接。

关闭库

当不想使用硬件 jsapi 功能时，即可关闭库。如果 html 忘记关闭库，微信会在浏览器关闭的时候关闭库，但不会断开设备的连接。

绑定/解绑设备

当用户要绑定设备的时候，html 需要进行扫描 (startScanWXDevice)，以显示周边未绑定的设备的列表 (注意，已绑定的设备列表可以通过 getWXDeviceInfos 得到)。

当用户选择某个设备进行绑定之后，html 开始绑定设备的流程。首先获取一个绑定设备的 ticket (getWXDeviceTicket)，并把 ticket 发送给厂商服务器 (通过 ajax 等)，厂商服务器调用公众平台的接口 (需要用到 ticket)，进行绑定。绑定成功后，厂商服务器通知 html 绑定成功 (通过 ajax 等)，html 显示绑定成功。当监听到 onWXDeviceBindStateChange，则表示微信客户端也同步了绑定关系。

解绑流程类似。

使用 AirKiss 配置 WiFi 设备联网

AirKiss 是微信用于配置 WiFi 设备上网的协议。AirKiss 可以在 WiFi 设备未连接任何网络，手机不切换 WiFi 的情况下，给设备传送 ssid 和密码。

例子

`http://203.195.191.112/device/`