IOS SSO SDK文档

北京新潮讯捷信息技术有限公司

编号：WEIBO\_IOS\_SSO\_SDK

版本：WEIBO\_IOS\_SSO\_SDK V1.0.0

修订记录

|  |  |  |  |
| --- | --- | --- | --- |
| 时间 | 文档版本 | 修订人 | 备注 |
| 2012/7/19 | 1.0.0 | 陈行政 | 初稿 |
|  |  |  |  |

目录

[IOS SSO DSK文档 1](#_Toc330481107)

[一、 概述 3](#_Toc330481108)

[名词解释 3](#_Toc330481109)

[二、 流程 4](#_Toc330481110)

[三、 接口说明 5](#_Toc330481111)

[SinaWeibo 5](#_Toc330481112)

[SinaWeiboDelegate 7](#_Toc330481113)

[SinaWeiboRequestDelegate 8](#_Toc330481114)

[SinaWeiboAuthorizeView 8](#_Toc330481115)

[SinaWeiboRequest 9](#_Toc330481116)

[SinaWeiboConstants.h 9](#_Toc330481117)

[四、 集成步骤及示例分析 9](#_Toc330481118)

[1. 新建工程，引入相关文件 9](#_Toc330481119)

[2. 定义所需常量 9](#_Toc330481120)

[3. 实现SinaWeiboDelegate协议 10](#_Toc330481121)

[4. 实现SinaWeiboRequestDelegate协议 11](#_Toc330481122)

[5. AppDelegate中初始化SinaWeibo对象 12](#_Toc330481123)

[6. 实现登录操作 13](#_Toc330481124)

[7. 实现请求微博API操作 13](#_Toc330481125)

[8. 实现退出操作 13](#_Toc330481126)

[五、 操作过程解析 14](#_Toc330481127)

[1. SSO登录过程 14](#_Toc330481128)

[2. Oauth2.0认证登录过程 15](#_Toc330481129)

[3. 一般请求过程 16](#_Toc330481130)

## 概述

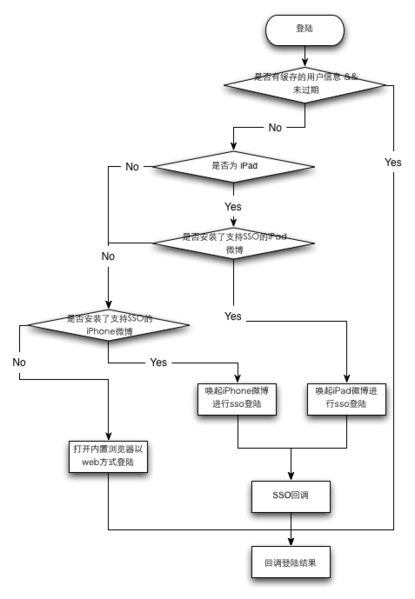
微博SDK为开发者提供访问微博api的集成接口，将身份验证以及api调用集成起来。并集成sso登录功能，使第三方应用可通过新浪微博官方客户端快速通过Oauth2.0授权，并完成用户登录操作。

本文档将对使用IOS SDK时所用的一些参数、接口进行说明，并分析一个简单示例，帮助第三方更方便的使用SDK（一些不使用的接口只做简单说明）。

### 名词解释

|  |  |
| --- | --- |
| AppKey | 分配给每个第三方应用的app key。用于鉴权身份，显示来源等功能。 |
| APPSecret | 生成请求request Token的secret，与app key一起分配。 |
| AccessToken | 表示用户身份的token，用于微博API的调用。 |
| ExpirationDate | 过期时间，用于判断登录是否过期。 |
| RedirectURI | 应用回调页面，可在新浪微博开放平台->我的应用->应用信息->高级应用->授权设置->应用回调页中找到。 |
| ssoCallbackScheme | sso回调地址，info.plist中定义的URL types，用于在微博客户端完成sso登录后进行回调。 |

## 流程



使用带sso功能的sdk进行登录，只需调用登录接口，并完成回调方法对接收登录结果即可。SDK中自动完成对是否进行SSO登录的判断，若支持，则唤起微博客户端，用户确认后返回请求的应用，SDK对请求结果进行解析，最后交给第三方实现的回调方法进行处理；否则，SDK将通过内置浏览器请求登录，用户输入用户名密码提交后，仍是SDK解析请求结果，并由第三方应用实现相应的回调方法进行最后处理。

## 接口说明

### SinaWeibo

SinaWeibo作为提供给第三方应用的主要接口，提供登录，登出，以及访问微博api的功能。用户在使用sdk前需实现唯一的SinaWeibo对象，将在之后的一系列操作中用到。

|  |  |  |
| --- | --- | --- |
| 方法 | * (id)initWithAppKey:(NSString \*)\_appKey   appSecret:(NSString \*)\_appSecrect  appRedirectURI:(NSString \*)\_appRedirectURI  andDelegate:(id<SinaWeiboDelegate>)\_delegate | |
| 简介 | 初始化方法，构造一个SinaWeibo对象。此构造方法没有传入sso回调地址，适用于采用默认sso回调地址的第三方应用。 | |
| 参数&作用 | \_appKey | 分配给第三方应用的app key |
| \_appSecrect | 分配给第三方应用的app secret |
| \_appRedirectURI | 应用回调页 |
| \_delegate | 第三方应用实现的SinaWeiboDelegate协议对象，用来处理登录及登出结果 |
| 返回类型 | SinaWeibo:构造函数返回当前实现的对象，在第三方应用中调用此对象进行登入登出等操作 | |

|  |  |  |
| --- | --- | --- |
| 方法 | * (id)initWithAppKey:(NSString \*)\_appKey   appSecret:(NSString \*)\_appSecrect  appRedirectURI:(NSString \*)\_appRedirectURI  ssoCallbackScheme:(NSString \*)\_ssoCallbackScheme  andDelegate:(id<SinaWeiboDelegate>)\_delegate | |
| 简介 | 初始化方法，构造一个SinaWeibo对象。此构造方法传入自定义的sso回调地址。 | |
| 参数&作用 | \_appKey | 分配给第三方应用的app key |
| \_appSecrect | 分配给第三方应用的app secret |
| \_appRedirectURI | 应用回调页 |
| \_ssoCallbackScheme | 自定义的sso回调地址，在info.plist中的URL Types所定义的。 |
| \_delegate | 第三方应用实现的SinaWeiboDelegate协议对象，用来处理登录及登出结果 |
| 返回类型 | SinaWeibo:构造函数返回当前实现的对象，在第三方应用中调用此对象进行登入登出等操作 | |

|  |  |  |
| --- | --- | --- |
| 方法 | - (void)logIn | |
| 简介 | 登录操作，直接调用。SinaWeibo对象根据具体环境进行sso登录或是浏览器登录，接收到登录结果后解析，并通过构造时传入的SinaWeiboDelegate对象调用sinaweiboDidLogIn方法将登录结果返回给应用。 | |
| 参数&作用 | 无 | 无 |
| 返回类型 | 无 | |

|  |  |  |
| --- | --- | --- |
| 方法 | - (void)logOut | |
| 简介 | 登出操作，直接调用。在接收到登出结果后解析，并通过构造时传入的SinaWeiboDelegate对象调用sinaweiboDidLogOut方法将登出结果返回给应用。 | |
| 参数&作用 | 无 | 无 |
| 返回类型 | 无 | |

|  |  |  |
| --- | --- | --- |
| 方法 | * (BOOL)isLoggedIn | |
| 简介 | 判断是否已登录 | |
| 参数&作用 | 无 | 无 |
| 返回类型 | YES为已登录；NO为未登录 | |

|  |  |  |
| --- | --- | --- |
| 方法 | * (BOOL) isAuthorizeExpired | |
| 简介 | 判断是否过期 | |
| 参数&作用 | 无 | 无 |
| 返回类型 | YES为已过期；NO为未过期 | |

|  |  |  |
| --- | --- | --- |
| 方法 | * (BOOL) isAuthValid | |
| 简介 | 判断登录状态是否有效，即已登录且未过期 | |
| 参数&作用 | 无 | 无 |
| 返回类型 | YES为登录状态有效；NO为登录状态无效 | |

|  |  |  |
| --- | --- | --- |
| 方法 | * (void)applicationDidBecomeActive | |
| 简介 | 应用从后台唤醒到前台后，调用此方法，方法中判断若在登录过程中将退出当前登录，并把退出登录的操作通过SinaWeiboDelegate对象调用sinaweiboLogInDidCancel方法交给应用进行处理 | |
| 参数&作用 | 无 | 无 |
| 返回类型 | 无 | |

|  |  |  |
| --- | --- | --- |
| 方法 | - (BOOL)handleOpenURL:(NSURL \*)url | |
| 简介 | 接收SSO登录回调信息，并解析，属于logIn的中间一步。应用需要在openURL时调用此方法以完成sso登录的回调过程。 | |
| 参数&作用 | url | sso登录的回调信息 |
| 返回类型 | YES | |

|  |  |  |
| --- | --- | --- |
| 方法 | * (SinaWeiboRequest \*)requestWithURL:(NSString \*)url   params:(NSMutableDictionary \*)params  httpMethod:(NSString \*)httpMethod  delegate:(id<SinaWeiboRequestDelegate>)\_delegate | |
| 简介 | 微博API的请求接口，发送请求时附带token值。 | |
| 参数&作用 | url | 请求的api地址，如“users/show.json”(请求用户资料) |
| params | 请求的参数，如“[NSMutableDictionary dictionaryWithObject:sinaweibo.userID forKey:@"uid"]”(用户的uid) |
| httpMethod | 请求类型，如“@”GET””(GET类型) |
| \_delegate | SinaWeiboRequestDelegate对象，用于处理请求结果的回调处理。应用在请求之前需实现此协议并新建一个对应的对象，请求时作为\_delegate参数传入。 |
| 返回类型 | SinaWeiboRequest对象 | |

### SinaWeiboDelegate

应用要完成登录的功能，需实现此协议，用来处理登入登出的回调结果。用户登录之前，初始化

|  |  |  |
| --- | --- | --- |
| 方法 | * (void)sinaweiboDidLogIn:(SinaWeibo \*)sinaweibo | |
| 简介 | SinaWeibo对象登录操作完成后，调用此方法。应用根据登录结果可进行自己的相关处理。 | |
| 参数&作用 | sinaweibo | SinaWeibo对象，应用可在方法中通过此对象判断登录结果，以进行不同的处理。 |
| 返回类型 | 无 | |

|  |  |  |
| --- | --- | --- |
| 方法 | - (void)sinaweiboDidLogOut:(SinaWeibo \*)sinaweibo | |
| 简介 | SinaWeibo对象登出操作完成后，调用此方法。应用根据登出结果可进行自己的相关处理。 | |
| 参数&作用 | sinaweibo | SinaWeibo对象，应用可在方法中通过此对象判断登出结果，以进行不同的处理。 |
| 返回类型 | 无 | |

|  |  |  |
| --- | --- | --- |
| 方法 | - (void)sinaweiboLogInDidCancel:(SinaWeibo \*)sinaweibo | |
| 简介 | 取消登录操作，用户在sso登录过程中将应用重新唤醒到前台后，应调用[SinaWeibo applicationDidBecomeActive]方法，此方法将取消当前登录，并回调sinaweiboLogInDidCancel。 | |
| 参数&作用 | sinaweibo | SinaWeibo对象。 |
| 返回类型 | 无 | |

|  |  |  |
| --- | --- | --- |
| 方法 | - (void)sinaweibo:(SinaWeibo \*)sinaweibo  logInDidFailWithError:(NSError \*)error; | |
| 简介 | 登录失败后，将错误信息回调给此方法。 | |
| 参数&作用 | sinaweibo | SinaWeibo对象。 |
| error | 失败信息。 |
| 返回类型 | 无 | |

|  |  |  |
| --- | --- | --- |
| 方法 | - (void)sinaweibo:(SinaWeibo \*)sinaweibo  accessTokenInvalidOrExpired:(NSError \*)error; | |
| 简介 | 操作过程中出错时sinaweibo对象将调用此方法，如accessToken无效或者过期。用户接收到此错误信息后应重新登录。 | |
| 参数&作用 | sinaweibo | SinaWeibo对象。 |
| error | 失败信息。 |
| 返回类型 | 无 | |

### SinaWeiboRequestDelegate

应用进行请求前，需实现此协议，并将对象传入，用以监听请求结果。

|  |  |  |
| --- | --- | --- |
| 方法 | - (void)request:(SinaWeiboRequest \*)request  didFailWithError:(NSError \*)error | |
| 简介 | 请求失败时回调此方法。 | |
| 参数&作用 | request | SinaWeiboRequest对象，SDK中具体用来实现请求的类，应用可根据request.url来判断具体哪个请求失败。 |
| error | 失败信息。 |
| 返回类型 | 无 | |

|  |  |  |
| --- | --- | --- |
| 方法 | - (void)request:(SinaWeiboRequest \*)request  didFinishLoadingWithResult:(id)result | |
| 简介 | 请求完成后回调此方法。 | |
| 参数&作用 | request | SinaWeiboRequest对象，SDK中具体用来实现请求的类，应用可根据request.url来判断具体哪个请求。 |
| result | 返回的结果，一般为请求的内容。 |
| 返回类型 | 无 | |

|  |  |  |
| --- | --- | --- |
| 方法 | - (void)request:(SinaWeiboRequest \*)request  didReceiveResponse:(NSURLResponse \*)response; | |
| 简介 | 请求接收到响应（应用不需要实现此方法）。 | |
| 参数&作用 | request | SinaWeiboRequest对象。 |
| response | 响应信息。 |
| 返回类型 | 无 | |

|  |  |  |
| --- | --- | --- |
| 方法 | * (void)request:(SinaWeiboRequest \*)request   didReceiveRawData:(NSData \*)data; | |
| 简介 | 请求接收到数据（应用不需要实现此方法）。 | |
| 参数&作用 | request | SinaWeiboRequest对象。 |
| data | 响应数据，最后需要对数据解析。 |
| 返回类型 | 无 | |

### SinaWeiboAuthorizeView

微博认证组件，应用进行登录时，若环境中无新浪微博官方客户端或是版本不支持，则无法进行sso登录，SinaWeibo对象中自动进行判断，并最终调用此类，进行Oauth2.0认证，此页面将打开内置浏览器，并请求用户授权，授权成功后将授权结果通过SinaWeiboDelegated对象(构造SinaWeibo对象时传入的delegate)返回给应用。第三方不需要使用此类，其功能已集成到SinaWeibo的登录方法中了。

### SinaWeiboRequest

请求微博API的类，其调用封装在SinaWeibo类中。

### SinaWeiboConstants.h

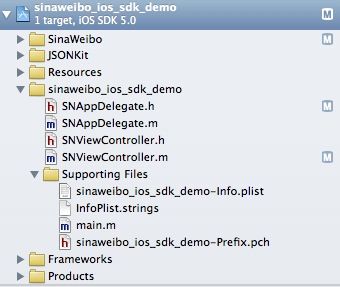
定义常量的头文件。

## 集成步骤及示例分析

以实际例子分析第三方应用引入WeiboSDK的过程。具体代码参考sinaweibo\_ios\_sdk\_demo工程。

### 新建工程，引入相关文件

创建新的工程，将SinaWeibo和JSONKit两个包拷贝并添加至工程中。



### 定义所需常量

定义应用SSO登录或者Oauth2.0认证所需的几个常量

* AppKey：第三方应用申请的appkey，用来身份鉴证、显示来源等；
* AppSecret：与appkey一起分配给第三方应用的appsecret，用来生成request token；
* AppRedirectURL：应用回调页，在进行Oauth2.0登录认证时所用。对于Mobile客户端应用来说，是不存在Server的，故此处的应用回调页地址只要与新浪微博开放平台->我的应用->应用信息->高级应用->授权设置->应用回调页中的url地址保持一致就可以了，如图所示：



* URL types：修改info.plist文件URL types项为自己的sso回调地址，默认格式为sinaweibosso.your\_app\_key（此处替换your\_app\_key为自己的app key）。若自定义为其他格式，比如my.app.custom.url，则初始化SinaWeibo对象时必须将此字符串作为ssoCallbackScheme参数传入（默认格式则不必传入此参数）。



### 实现SinaWeiboDelegate协议

第三方应用需要实现自己的SinaWeiboDelegate，来处理用户登录、退出以及取消登录的回调。方法详解见接口说明一节。需要实现如下方法：

|  |
| --- |
| #pragma mark - SinaWeibo Delegate    // 用户成功登陆  // 注意需要将登陆成功的用户信息缓存下来以便下次使用  - (void)sinaweiboDidLogIn:(SinaWeibo \*)sinaweibo  {  NSLog(@"sinaweiboDidLogIn userID = %@ accesstoken = %@ expirationDate = %@", sinaweibo.userID, sinaweibo.accessToken, sinaweibo.expirationDate);  [self storeAuthData];  }    // 用户退出登陆  // 注意需要清除已经缓存的用户信息  - (void)sinaweiboDidLogOut:(SinaWeibo \*)sinaweibo  {  NSLog(@"sinaweiboDidLogOut");  [self removeAuthData];  }    // 用户取消登陆过程  - (void)sinaweiboLogInDidCancel:(SinaWeibo \*)sinaweibo  {  NSLog(@"sinaweiboLogInDidCancel");  }    // 登陆失败  - (void)sinaweibo:(SinaWeibo \*)sinaweibo logInDidFailWithError:(NSError \*)error  {  NSLog(@"sinaweibo logInDidFailWithError %@", error);  }    // 缓存的token无效或者已经过期  // 注意需要清除缓存的用户信息  - (void)sinaweibo:(SinaWeibo \*)sinaweibo accessTokenInvalidOrExpired:(NSError \*)error  {  NSLog(@"sinaweiboAccessTokenInvalidOrExpired %@", error);  [self removeAuthData];  } |

### 实现SinaWeiboRequestDelegate协议

第三方应用实现自己的SinaWeiboRequestDelegate，来处理微博api请求的结果。方法详解见接口说明一节。需要实现以下方法：

|  |
| --- |
| #pragma mark - SinaWeiboRequest Delegate    // 请求失败  - (void)request:(SinaWeiboRequest \*)request didFailWithError:(NSError \*)error  {  // 当一个delegate需要处理多个请求回调时，可以通过url来判断当前的request  if ([request.url hasSuffix:@"users/show.json"])  {  }  else if ([request.url hasSuffix:@"statuses/upload.json"])  {  }      [self resetButtons];  }    // 请求成功  - (void)request:(SinaWeiboRequest \*)request didFinishLoadingWithResult:(id)result  {  // 当一个delegate需要处理多个请求回调时，可以通过url来判断当前的request  if ([request.url hasSuffix:@"users/show.json"])  {  }  else if ([request.url hasSuffix:@"statuses/upload.json"])  {  }    [self resetButtons];  } |

### AppDelegate中初始化SinaWeibo对象

* 在AppDelegate中定义 sinaweibo 属性并使用自己APP的相关属性初始化 sinaweibo 对象
  + - AppKey / AppSecret / AppRedirectURI 是在新浪微博开放平台上定义的APP相关属性
    - delegate是你自己的程序中用于处理微博帐户登陆、退出相关的对象（SinaWeiboDelegate）
* 读取已缓存的相关用户信息，并传递给 sinaweibo 对象
* app回到前台或openUrl时通知sinaweibo对象，以实现sso功能

|  |
| --- |
| - (BOOL)application:(UIApplication \*)application didFinishLaunchingWithOptions:(NSDictionary \*)launchOptions  {  self.window = [[[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]] autorelease];  // Override point for customization after application launch.  self.viewController = [[[SNViewController alloc] initWithNibName:nil bundle:nil] autorelease];  self.window.rootViewController = self.viewController;  [self.window makeKeyAndVisible];    sinaweibo = [[SinaWeibo alloc] initWithAppKey:kAppKey appSecret:kAppSecret appRedirectURI:kAppRedirectURI andDelegate:your\_delegate];  NSUserDefaults \*defaults = [NSUserDefaults standardUserDefaults];  NSDictionary \*sinaweiboInfo = [defaults objectForKey:@"SinaWeiboAuthData"];  if ([sinaweiboInfo objectForKey:@"AccessTokenKey"] && [sinaweiboInfo objectForKey:@"ExpirationDateKey"] && [sinaweiboInfo objectForKey:@"UserIDKey"])  {  sinaweibo.accessToken = [sinaweiboInfo objectForKey:@"AccessTokenKey"];  sinaweibo.expirationDate = [sinaweiboInfo objectForKey:@"ExpirationDateKey"];  sinaweibo.userID = [sinaweiboInfo objectForKey:@"UserIDKey"];  }  return YES;  }    //当程序回到前台时通知sinaWeibo对象  - (void)applicationDidBecomeActive:(UIApplication \*)application  {  [self.sinaweibo applicationDidBecomeActive];  }    //当openUrl时通知SinaWeibo对象  - (BOOL)application:(UIApplication \*)application handleOpenURL:(NSURL \*)url  {  return [self.sinaweibo handleOpenURL:url];  }    - (BOOL)application:(UIApplication \*)application openURL:(NSURL \*)url sourceApplication:(NSString \*)sourceApplication annotation:(id)annotation  {  return [self.sinaweibo handleOpenURL:url];  } |

### 实现登录操作

应用创建SinaWeibo对象之后，只需调用[SinaWeibo logIn]方法，即可实现登录，之后的操作交给SDK完成，最后在自己实现的SinaWeiboDelegate类中接收登录结果即可。

### 实现请求微博API操作

应用登陆成功后，通过创建的SinaWeibo对象调用request方法实现微博API的请求。

|  |
| --- |
| - (void)userInfoButtonPressed  {  SinaWeibo \*sinaweibo = [self sinaweibo];  [sinaweibo requestWithURL:@"users/show.json"  params:[NSMutableDictionary dictionaryWithObject:sinaweibo.userID forKey:@"uid"]  httpMethod:@"GET"  delegate:self];  } |

### 实现退出操作

通过调用[SinaWeibo logOut]方法退出，并在SinaWeiboDelegate处理退出回调结果即可。

## 操作过程解析

### SSO登录过程

下图绘出了第三方应用通过IOS SDK进行SSO登录的基本流程。

注:

* 1. 图中黄色标注的文字描述的为微博SDK或微博客户端处理的流程，第三方应用需要实现的是白色文字描述的内容；
  2. 若用户在微博客户端登录时(未完成)，手动唤起应用到前台，应用需要在[AppDelegate applicationDidBecomeActive]中调用[SinaWeibo applicationDidBecomeActive]以取消当前登录。

### Oauth2.0认证登录过程

若没有SSO登录的环境(未安装微博官方客户端或版本不支持)，则进行Oauth2.0认证登录。

### 一般请求过程