

7659 游戏中心 xSDK 接入帮助文档

(v2.0.3)

修订日期： 2015 年 07 月 10 日

7659 游戏运营中心

日期	说明	版本	负责人
2015年05月20日	正式发布	v2.0.0	MTang
2015年06月24日		v2.0.1	MTang
2015年07月10日		v2.0.3	MTang

1.综述

1.1.目的

本文档是提供给与 7659 游戏中心联运的网络游戏开发商，完成 7659SDK 接入使用的技术文档。文档描述网络游戏接入 7659SDK 时需要调用的接口、参数等详细技术信息。

1.2.预期的读者和阅读建议

网络游戏联运商的具有 iOS 端开发经验的工程师，熟悉网游支付流程。

1.3.产品适用范围

7659 游戏中心此版 xSDK 是一款集更新检测、账号系统、支付功能为一体的集合。适用于 iOS6.x 至 iOS8.x，目前 7659 客户端最低支持系统 iOS6，请开发商注意。

7659SDK 目前支持主流的游戏引擎 unity，coco2dx。

1.4.名词解释

1) 7659 通行证 (7659 通行证账号 , 7659 账号)

xSDK 用户系统的用户身份标识，用于游戏玩家登录 xSDK 用户系统。每一个 7659 通行证可以拥有多个子账号 (guid)，子账号对应了游戏中的游戏账号。

注：游戏账号应该对应 guid，而不是 7659 通行证账号。

2) guid、subguid、gameuid

在不同的文档中可能会见到这三种不同的称谓，但其实同指 7659 通行证下的子账号 id。游戏应用该子账号对应游戏账号使用。

3) tokenkey

在游戏注册、登录或切换账号后，出于安全考虑，xSDK 并不直接给出 guid，而是给出 tokenkey。CP 需要通过 xSDK 用户系统将该 tokenkey 转换为 guid (相关请求过程请参见用户系统服务器对接文档)。

4) 游戏老账号

在 7659 初期，曾兼容游戏自己的用户系统，游戏的用户系统的账号称为游戏老账号。随着 7659 平台的发展，以及 7659 自身账号系统的不断完善，游戏老账号已被 7659 账号系统取代，并通过账号绑定的方式连接 7659 账号与游戏老账号，游戏老账号正在逐步被淘汰。**大部分游戏是不需要游戏老账号的，请主动屏蔽掉，具体调用的接口，请看下面的（2.3 接口定义）**

没有游戏老账号的登陆界面：



The screenshot shows the login interface for 7659 mobile center. At the top, there is a logo for 7659 手游中心. Below the logo, the text "通行证快速登录" (Passport Quick Login) is displayed. There are two input fields: the first contains the phone number "13240409007" and the second contains a masked password ".....". A blue button labeled "立即登录" (Login Immediately) is positioned below the password field. At the bottom left, there is a link for "快速注册" (Quick Registration), and at the bottom right, there is a link for "忘记了?" (Forgot?).

有游戏老账号的登陆界面：



The screenshot shows the login interface for 7659 mobile center, which includes an option for game old accounts. It features the same logo and "通行证快速登录" header as the previous screenshot. The input fields and "立即登录" button are identical. However, there is an additional white button labeled "使用游戏老账号登录" (Use Game Old Account Login) located below the "立即登录" button. The "快速注册" and "忘记了?" links are also present at the bottom.

2.接入

2.1.接入前准备

2.1.1.获得接入参数

请先登录“7659 开放平台”创建或添加游戏，待审核完毕后，将会获得游戏接入所需的相关参数，包含“Game Key”；“Game Pay”；“APP Secret”；“Game ID”（请参照下面截图），其中“Game Pay”和“APP Secret”是在下面客户端接入中将会用到。



关于注册、登录、创建或添加游戏的相关问题，请参照“7659 开放平台”的相关文档。

2.2.2.获取接入资源包

在“7659 开放平台”获取接入所需的库和相关资源。



2.2.3.添加 7659 游戏启动图片



放置位置:游戏启动第一页;

持续时间:3 秒

注意事项:不得涂改、拼接。若没有,在 SDK 下载包(闪屏图片里.zip)面有,或者请问商务、运营、技术支持人员索要

2.2.项目设置

2.2.1.需引入的库

序号	库
1	libsqlite3.dylib
2	AdSupport.framework
3	SystemConfiguration.framework
4	CoreTelephony.framework
5	libXSDK.a



说明：由于目前大多数 CP 接入已处于游戏开发完毕的状态，很少有使用模拟器接入和调试的需求，因此资源包里的库文件仅支持真实设备，不支持模拟器。如果 CP 确有此需求，可联系技术支持索要支持模拟器的版本。但由于 xSDK 整合了其他 SDK，因此不保证其他 SDK 也能很好的支持模拟器。如遇到问题，请先在真实设备上运行，如果问题依旧，请联系技术支持。

2.2.2.头文件、源文件

序号	库
1	XSDK.h
2	GTMBase64.h
3	GTMBase64.m
4	GTMDefines.h
5	
6	

2.2.3.资源包

序号	库
1	XSDKResource.bundle
2	AlipaySDK.bundle
3	
4	
5	
6	

2.2.4.设置第三方 App 回调的 URL Schemes

为了区分游戏自身可能会用到其它 URL Schemes 设置，请将该设置的 Identifier 设置为

“com.ky.xSDK.alipay”，URL 设置建议设置为项目的反向域名(reverse-domain)形式。



2.3.接口定义

XSDK	
xSDK 接口类，游戏开发商通过该类调用 xSDK 功能	
属性	<code>@property(weak, nonatomic) id<XSDKDelegate> xsdkDelegate;</code>
说明	为 XSDK 设置委托
接口定义	<code>+(id) instanceXSDK;</code>
说明	获取 XSDK 的全局单例，所有接口方法均通过此实例调用
返回值	XSDK 的全局单例
接口定义	<code>-(void) setAppKey:(NSString*) appKey;</code>
说明	设置 AppKey
appKey	添加游戏后获得的 AppKey
接口定义	<code>-(void) setOnlySupportIPhone:(BOOL) isOnlySupportIPhone;</code>
说明	是否只支持 iPhone
isOnlySupportIPhone	如果游戏当前接入的版本并未对 iPad 进行适配（当在 iPad 上运行时，居中显示，四周是黑框的情况）请传 YES，否则为 NO
接口定义	<code>-(void) setISSupportGameNo:(BOOL) isSupportGame</code>

	No;
说明	是否只支持游戏账号登陆 (默认不支持), 该设置仅对 7659 平台有效, 其它平台均为不支持。
isSupportGameNo	是否只支持游戏账号登陆
接口定义	-(void) checkUpdate;
说明	初始化数据, 检查更新
接口定义	-(void) login;
说明	登陆, 如果之前使用通行证账号成功登录过, 则直接显示自动登录
接口定义	-(void) logout; (如果游戏账号登录, 并且该游戏账号没有绑定 7659 通行证, 不要调用该方法)
说明	注销, 回到登录界面
接口定义	-(void) setGameContact:(NSString *)contact;
说明	当玩家使用游戏账号时 (并未绑定 7659 通行证的情况), 如果忘记密码, 则通过此信息进行密码找回。需要游戏开发商设置相关的联系方式或找回方法。
contact	联系方式或找回方式, 字数限制为 40 以内
接口定义	-(void) payWithDealSeq:(NSString *)dealSeq andFee:(NSString *)fee andPayID:(NSString *)payID andGamesvr:(NSString *)gamesvr andSubject:(NSString *)subject;
说明	游戏支付接口, 仅支持 7659 通行证或绑定过 7659 通行证的游戏账号
dealSeq	游戏产生的订单号
fee	订单金额
payID	准备工作中的 PayID 字段
gamesvr	游戏区服, 如果没有特殊的需求, 可填@""
subject	商品名称
接口定义	-(BOOL) handleApplication:(UIApplication *)application openURL:(NSURL *)url sourceApplication:(NSString *)sourceApplication annotation:(id)annotation;
说明	过滤 AppDelegate 的 -(BOOL) application:(UIApplication *)application openURL:(NSURL *)url sourceApplication:(NSString *)sourceApplication annotation:(id)annotation; 方法。请置于该方法的第一行, 通过执行结果判断是否需要游戏自己进行处理。
application	
url	
sourceApplication	

annotation	
------------	--

XSDKDelegate	
XSDK 的委托，用户 XSDK 在玩家完成相关动作后回调游戏	
回调方法	<code>-(void)XSDKCheckUpdateCallBack;</code>
说明	游戏初始化结束后回调
回调方法	<code>-(void)XSDKLoginCallBack:(NSString *)tokenKey;</code>
说明	用户登录回调
tokenKey	登录成功后的 tokenkey，可据此通过服务器帐号验证接口，获取到用户名及 guid。
回调方法	<code>-(void)XSDKLogoutCallBack:(NSString *)guid type:(XSDKLOGOUTTYPE)type;</code>
说明	注销方法回调
guid	帐号的唯一标识
type	XSDKLOGOUT, //个人中心内，退出时回调 XSDKGAMELOGOUT//XSDKLogout 接口调用时回调
回调方法	<code>-(void)XSDKGameLoginCallBack:(NSString *)username password:(NSString *)password resultBlock:(void (^)(NSString * typeMessage))resultBlock;</code>
说明	游戏账号登陆回调
username	游戏账号名称
password	游戏账号登录密码
resultBlock	结果 block，结果如果判断正确，传入 resultBlock(@"suc")，否则错误信息显示传入字符
回调方法	<code>-(void)XSDKGameLoginSuc;</code>
说明	游戏账号登陆成功回调
回调方法	<code>-(void)XSDKChangeSubNOCallBack:(NSString *)oldGuid newTokenKey:(NSString *)newTokenKey;</code>
说明	切换子账号回调
oldGuid	当前账号唯一标示
newTokenKey	切换的新账号的 tokenkey
回调方法	<code>-(void)XSDKPayCallBack:(BOOL)isClientSuccess;</code>
说明	客户端获得的支付结果，CP 客户端应在收到此回调后发起服务器验证，获得最终结果
isClientSuccess	

2.4. 模块接入

2.4.1. 获取全局单例

xSDK 采用单例模式，请通过 `+(id) instanceXSDK;` 方法获取该单例。

2.4.2. 设置 AppKey

在使用 xSDK 的其它功能之前，请先通过 `-(void) setAppKey:(NSString*) appKey;` 方法设置在 7659 开放平台上获得的 appKey。

2.4.3. 设置委托

通过 `@property(weak, nonatomic) id<XSDKDelegate> xsdkDelegate;` 属性为 xSDK 实例设置委托。该委托应该实现了 XSDKDelegate 协议。

2.4.4. 初始化更新模块

xSDK 本地初始化，并从服务器获得基础参数及更新信息等。

游戏开发商通过调用 `-(void) checkUpdate` 接口打开初始化及更新界面，并通过 `-(void) XSDKCheckUpdateCallBack` 回调获知初始化及更新动作已完成。

2.4.5. 登录模块

显示 xSDK 登录或自动登录界面 若当前用户还未注册 则通过登录界面进入注册模块。

游戏开发商通过调用 `-(void) login` 接口打开登录界面，如果当前用户之前曾使用 7659 通行证账号成功登录过，则直接显示自动登录界面。

如果登录成功，游戏开发商通过 `-(void) XSDKLoginCallBack:(NSString*) tokenKey` 方法获知登录成功，并获得登录账号的 tokenKey，该 tokenKey 用于游戏服务器端向 7659 服务器请求账号信息。

2.4.6. 游戏老账号登录

注：游戏老账号是 7659 平台早期的产物，属于历史遗留问题（参照名词解释里的游戏老账号），如果 CP 从未接过 xSDK 以前的版本的 SDK，或在以前版本的 SDK 中从未处理过老账号相关动作，则请跳到下一节。如果您依然不确定是否需要处理老账号问题，请联系技术支持。

如果游戏含有游戏老账号，并需要支持游戏老账号登录，请参照如下设置：

1) `-(void) setISSupportGameNo:(BOOL) isSupportGameNo` 接口，并设置参数为 YES。此接口仅对 7659 平台有效，其它平台则会被忽略。

2) 提供 `-(void)XSDKGameLoginCallback:(NSString *)username password:(NSString *)password resultBlock:(void (^)(NSString * typeMessage))resultBlock` 回调函数，并在该函数中自行检查账号登录，在检测完毕后，通过调用 `resultBlock` 将检测结果通知 xSDK。如果登录成功，则将 `block` 的 `typeMessage` 参数设置为：`@“suc”`；否则设置为错误提示。

3) 游戏账号登录成功后，xSDK 通过 `-(void)XSDKGameLoginSuc` 回调方法通知游戏。

4) xSDK 不再为游戏老账号提供密码找回的入口，请游戏通过 `-(void)setGameContact:(NSString *)contact` 接口显示此种情况下的处理方式，或提供联系方式。

2.4.7.注销模块

用户可以通过用户中心进行注销，如果游戏需要额外提供注销功能，需使用 `-(void)logout` 方法进行注销操作，在注销完成后，通过 `-(void)XSDKLogoutCallBack:(NSString *)guid type:(XSDKLOGOUTTYPE)type` 回调获得注销结果，并通过 `XSDKLOGOUTTYPE` 枚举进行区分。

注：回调处理时需检查 `XSDKLOGOUTTYPE` 枚举，如为：`XSDKGAMELOGOUT`，则 xSDK 会在注销后显示游戏界面，游戏可在适当的时候自行调用 `login` 方法登录；如为：`XSDKLOGOUT`，则 xSDK 会自动在注销后显示登录界面，游戏无需自行调用登录。

2.4.8.支付模块

显示支付界面进行支付。

游戏开发商通过调用 `-(void)payWithDealSeq:(NSString *)dealSeq andFee:(NSString *)fee andPayID:(NSString *)payID andGamesvr:(NSString *)gamesvr andSubject:(NSString *)subject`；方法打开支付界面进行支付操作，并通过 `-(void)XSDKPayCallback:(BOOL)isClientSuccess` 回调获得支付结果。

注意：获得支付结果时，如果 `isClientSuccess` 返回 YES，游戏还要通过自己的服务器验证一下，得到最终结果。

2.4.9.切换小号

个人中心的小号切换通过 `-(void)XSDKChangeSubNOCallBack:(NSString *)oldGuid newTokenKey:(NSString *)newTokenKey`；回调通知游戏。该回调回传了切换前的 `subguid` 及新的账号的 `tokenKey`，该 `tokenKey` 用于游戏服务器端向 7659 服务器请求账号信息。

注：由于此回调已给出切换前的 `subguid` 和切换后的新的 `tokenKey`，因此游戏可直接依次将旧的账号 (`subguid`) 注销，并通过 `tokenKey` 获取切换后的账号信息 (包含新的 `subguid`，此获取流程类似登录流程的相关步骤)，然后加载新账号进行游戏。

2.4.10.其它

1) 如果游戏未选择支持 iPad, 则 iOS 会在 iPad 上自动将游戏显示为画面居中, 四周有个黑框的样子。这将给 xSDK 界面显示造成混乱。所以, 如果接入的游戏如果是这种情况, 则须通过 `-(void)setOnlySupportIphone:(BOOL)isOnlySupportIphone` 接口告知 xSDK。

2) 由于 xSDK 会调用第三方 App 进行例如支付宝充值等动作, 所以需要在 appDelegate 的 `-(BOOL)application:(UIApplication *)application openURL:(NSURL *)url sourceApplication:(NSString *)sourceApplication annotation:(id)annotation` 回调函数中首先调用 `-(BOOL)handleApplication:(UIApplication *)application openURL:(NSURL *)url sourceApplication:(NSString *)sourceApplication annotation:(id)annotation` 接口进行接收, 并通过该接口的返回值来获得处理结果——YES 代表该消息是 xSDK 所需消息, 游戏无需处理; 反之表示并非 xSDK 相关的消息。