**SDK-总体机制**

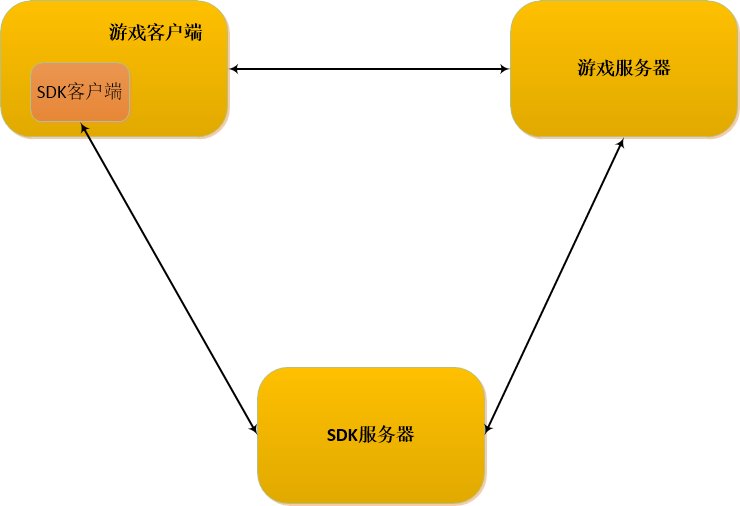
**术语表：**

| **名称** | **说明** |
| --- | --- |
| mem\_id | 用户在平台注册账号的唯一标识码 |
| user\_token | 登陆会话标示 |
| client\_id | 客户端ID,服务器生成 |
| client\_key | 客户端key，服务器生成，写入游戏包中 |
| app\_id | 游戏标识。由平台分配，用于区别不同的游戏合作厂商。 |
| app\_key | 签名密钥。由平台为每个游戏合作厂商分配的消息签名密钥，用于在双方系统数据传输过程进行数据签名时使用。 |
| SDK | 为游戏开发商提供基础服务（登录、购买等)的系统，包括“SDK客户端”和”SDK 服务器”两部分。 |
| 游戏客户端 | 游戏开发商开发的，运行在用户终端上的软件。 |
| 游戏服务器 | 游戏开发商开发的，配合”游戏客户端”为用户提供游戏服务的服务器端程序。 |

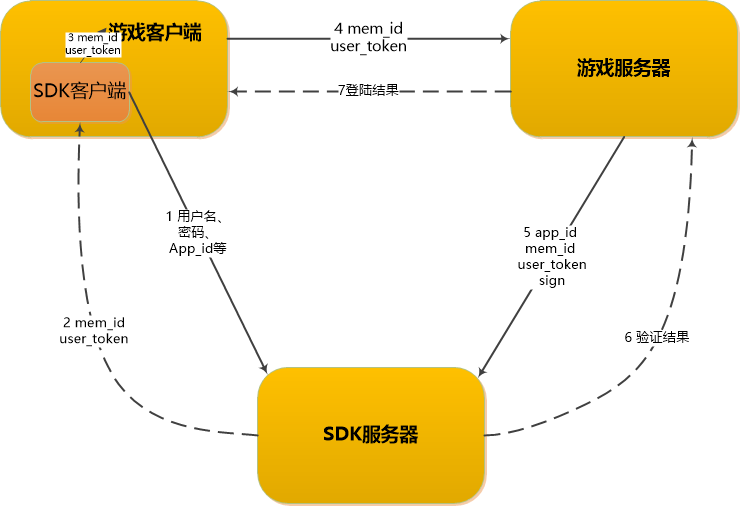
**总体架构：**

**“SDK”分为**

* “SDK 客户端”（集成于游戏中)、
* “SDK 服务器”两部分。

**“游戏客户端” (内含”SDK 客户端”)、”游戏服务器” “SDK 服务器”之间的关系如下图所示：**  


**1 登陆流程**



- 1 登陆："游戏客户端"调用"SDK 客户端"的登陆接口，传入账号、密码等参数登 陆"SDK 服务器"。

- 2 mem\_id、user\_token："SDK 服务器"验证用户信息成功之后，会返回user\_ token、mem\_id 等用户信息 给SDK 客户端"。

- 3 mem\_id user\_token："游戏客户端"通过"SDK 客户端"的接口回调，获取到mem\_id, user\_token等相关信息。

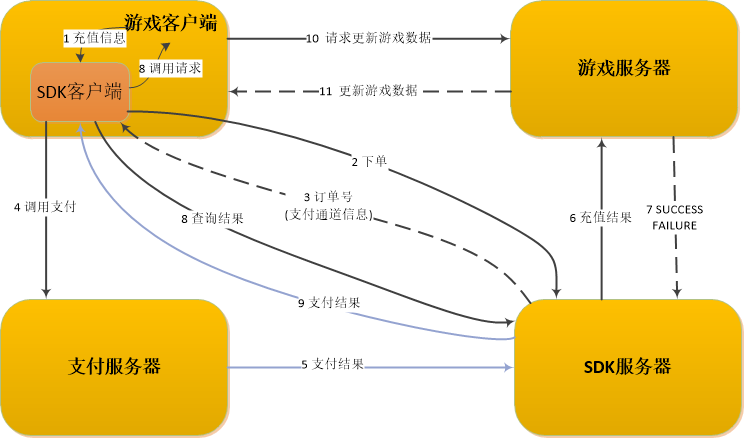
- 4 mem\_id user\_token："游戏客户端"把获取到的信息发送给"游戏服务器"。

- 5 app\_id mem\_id user\_token："游戏服务器"通过"SDK 服务器"的验证 user\_token 接口。

- 6 返回结果："SDK 服务器"返回user\_token是否登陆给"游戏服务器"。

- 7登陆结果："游戏服务器"返回结果给"游戏客户端"。

**2 充值流程**



1充值信息："游戏客户端"调用"SDK 客户端"的支付购买接口，传入商品名称、 商品金额、游戏区服、游戏角色给"SDK 客户端"。

2 下单mem\_id,app\_id等"SDK 客户端"把商品购买信息发送给"SDK 服务器"。

3 订单号(支付渠道信息)：SDK服务器返回“SDK订单号”

4 调用支付 SDK调用支付

5 支付结果 异步回调给SDK服务器

6 充值结果 充值成功后，SDK服务器调用支付回调通知游戏服务器，"游戏服务器"根据"SDK 服务器"发送过来的商品名称、价格、 厂家订单号等信息，跟自己的商品进行对比，如果商品跟价格都一致，则返回成功， 否则返回失败。

7 游戏服务器放回success/**fail**给SDK服务器

8 SDK客户端在支付结束后，向SDK服务器请求支付结果，

9 SDK服务器放回支付结果

10 支付成功结果返回给游戏客户端

11 游戏客户端向游戏服务器请求更新

12 更新游戏数据

**备注**

若游戏服务器有自动推送功能，可以省略 8-11 步骤。 在游戏服务器无自动推送功能时，使用8-11步骤，游戏厂商根据需要对接