

# 腾讯云 IM 开放-WebSDK-资料管理 REST 协议文档

文档编号		
文档名称	腾讯云 IM 开放-WebSDK-资料管理 REST 协议文档	
作者	peakerdong	
首次完成时间	2015-08-13	
当前版本	V1.0	
修改记录		
时间	修改人	修改说明

## 目 录

1 综述.....	- 3 -
2 REST 协议说明.....	- 3 -
2.1 设置资料.....	- 3 -
2.2 拉取资料.....	- 5 -
3 错误码说明.....	- 8 -
3.1 异常错误码说明.....	- 8 -

# 1 综述

腾讯云 IM 开放-Web SDK-资料管理 REST 包含以下功能：

- ✓ 设置资料；
- ✓ 拉取资料（搜索用户）；

## 2 REST 协议说明

目前 web sdk 提供以下几个跟用户资料有关的 api 接口：

API 名称	协议功能简述
setProfilePortrait	设置资料
getProfilePortrait	拉取资料

### 2.1 设置资料

Command	setProfilePortrait
请求参数示例	<pre>{   "From_Account": "id",   "ProfileItem": [     {       "Tag": "Tag_Profile_IM_Nick",       "Value": "MyNickName"     },     {       "Tag": "Tag_Profile_IM_Gender",       "Value": "Gender_Type_Female"     }   ] }</pre>

	<pre> {     "Tag": "Tag_Profile_IM_AllowType",     "Value": "AllowType_Type_NeedConfirm" } ] } </pre>
返回示例	<pre> {     "ActionStatus": "FAIL",     "ErrorCode": 10001,     "ErrorInfo": "Err_Profile_PortraitSet_Msg",     "ErrorDisplay": "" } </pre>
接口说明	<ul style="list-style-type: none"> <li>✓ 这个请求的语义是：设置帐号 id 的昵称和加好友验证方式。</li> <li>✓ <b>From_Account</b>：请求发起方的帐号，一般情况下为用户本人帐号，在管理员代替用户发起请求的情况下，这里应该填写被代替的用户的帐号。</li> <li>✓ 目前可以设置的字段包括： <ul style="list-style-type: none"> <li>昵称："Tag_Profile_IM_Nick"，限制长度为 64 字节，UTF8 格式</li> <li>性别："Tag_Profile_IM_Gender"，该字段有且仅有以下 2 种取值： <ul style="list-style-type: none"> <li>男="Gender_Type_Male"</li> <li>女="Gender_Type_Female"。</li> </ul> </li> <li>加好友验证方式："Tag_Profile_IM_AllowType"，该字段有且仅有以下三种取值： <ul style="list-style-type: none"> <li>需要确认="AllowType_Type_NeedConfirm"</li> <li>允许任何人="AllowType_Type_AllowAny"（默认）</li> <li>拒绝任何人="AllowType_Type_DenyAny"。</li> </ul> </li> </ul> </li> <li>✓ 设置资料协议不提供批量的命令，只能自己设置自己的资料，如果是后台导数据，需要循环发起调用。</li> <li>✓ <b>ErrorCode</b> 的意义请参考错误码描述章节部分。</li> </ul>

2.2 拉取资料

API 名称	getProfilePortrait
请求参数示例	<pre>{   "From_Account":"id",   "To_Account":["id1","id2","id3","id4"],   "TagList":   [     "Tag_Profile_IM_Nick"   ],   "LastStandardSequence":0 }</pre>
返回结果示例	<pre>{   "UserProfileItem":   [     {       "To_Account":"id1",       "ProfileItem":       [         {           "Tag":"Tag_Profile_IM_Nick",           "Value":"NickNameTest1"         }       ],       "ResultCode":0,       "ResultInfo":""     },     {       "To_Account":"id2",       "ProfileItem":</pre>

	<pre>[     {         "Tag": "Tag_Profile_IM_Nick",         "Value": "NickNameTest2"     } ], "ResultCode": 0, "ResultInfo": "", }, {     "To_Account": "id3",     "ResultCode": 10001,     "ResultInfo": "帐号不存在" }, {     "To_Account": "id4",     "ResultCode": 10002,     "ResultInfo": "帐号非法" } ], "CurrentStandardSequence": 2, "Fail_Account": ["id3"], "Invalid_Account": ["id4"], "ActionStatus": "FAIL", "ErrorCode": 10001, "ErrorInfo": "Err_Profile_PortraitGet_Msg", "ErrorDisplay": "" }</pre>
接口说明	✓ 这个请求的语义是：以帐号 id 的身份来拉取帐号名为 id1、id2、id3 和 id4 的用户的昵称。

	<ul style="list-style-type: none"> <li>✓ <b>From_Account:</b> 请求发起方的帐号，一般情况下为用户本人帐号，在管理员代替用户发起请求的情况下，这里应该填写被代替的用户的帐号。</li> <li>✓ 目前可以拉取的字段包括： <ul style="list-style-type: none"> <li>昵称:"Tag_Profile_IM_Nick"，限制长度为 64 字节，UTF8 格式</li> <li>加好友设置:"Tag_Profile_IM_AllowType"，该字段有且仅有以下三种取值： <ul style="list-style-type: none"> <li>需要确认="AllowType_Type_NeedConfirm"</li> <li>允许任何人="AllowType_Type_AllowAny"（默认）</li> <li>拒绝任何人="AllowType_Type_DenyAny"。</li> </ul> </li> </ul> </li> <li>✓ <b>LastStandardSequence:</b> 用户资料数据的版本号，如果用户修改了自己的资料那么在服务器端就给该 Sequence 自增一次，客户端第一次拉取时，LastStandardSequence 填写为 0，并且记录本次拉取资料回包里面的 CurrentStandardSequence，以便下次拉取作为 LastStandardSequence 来发起调用，如果客户端带上的 LastStandardSequence 和服务器端保存的 Sequence 相同，那么就不返回数据给客户端，因为服务器认为客户端拥有上一次的历史数据，如果客户端带上的 LastStandardSequence 和服务器端保存的 Sequence 不相同，那么就返回数据，因为服务器认为客户端的历史数据已经不是最新的了，有必要给客户端下发最新的资料。</li> <li>✓ <b>CurrentStandardSequence:</b> 用户资料数据的最新版本号，客户端需要将此值保存下来下次请求时带上给服务器端。</li> <li>✓ 拉取资料协议提供批量的命令，可以拉取任何一个帐号的资料。</li> <li>✓ 如果调用方在调用时填写了 3 个帐号，那么在回包中一定会有 3 个帐号，并且每个帐号都有自己的 ResultCode，ResultCode 为 0 则表示成功，ResultCode 不为 0 表示有异常，具体的意义参考错误码描述章节部分。</li> <li>✓ <b>Fail_Account:</b> 如果某个帐号的 ResultCode 不为 0，那么就会在这个地方记录一次，Fail_Account 的信息和包体里面 ResultCode 字段是冗余的，这里只是为了提供调用的遍历性，调用方不需要逐个遍历</li> </ul>
--	---

	<p>ResultCode，只要检查 Fail_Account 就可以知道哪些账户获取资料失败了。</p> <p>✓ Invalid_Account: 如果调用的请求包里面所携带的 To_Account 中有非法帐号，比如注销过的账户等，就会把对应的 To_Account 放到 Invalid_Account 中。</p> <p>✓ ErrorCode 的意义请参考错误码描述章节部分。</p>
--	--

## 3 错误码说明

### 3.1 异常错误码说明

命令名	错误码	错误描述
setProfilePortrait	40602	设置资料读取数据失败
	40603	设置资料非法字段
	40604	设置资料系统错误非法字段长度
	40605	设置资料请求参数非法
	40606	设置资料设置标准数据失败
	40607	设置资料非法字段
	40608	设置资料系统错误非法字段长度
	40609	设置资料设置自定义数据失败
getProfilePortrait	40801	获取资料字段非法
	40802	获取资料系统执行标配任务失败
	40803	获取资料系统解析数据失败
	40804	获取资料非法字段
	40805	获取资料系统执行自定义任务列表失败
	40806	获取资料系统错误自定义字段长度非法
	40807	获取资料标配主 Key 不存在
	40808	获取资料标配读取数据失败
	40809	获取资料自定义主 Key 不存在
	40810	获取资料自定义读取数据失败
	40811	获取资料时 Account 不正确