个推 SDK 文档

getui.com

October 2016

目录

其他接口

1. 别名接口

1.1 别名说明

个推使用 clientid 来标识每个独立的用户，但 clientid 不等于开发者应用上的用户名，如果希望将消息发给应用上指定用户名的用户，则需要将用户名指定一个用户别名。

为一个或者一批 clientid 用户定义一个用户别名，通过这个用户别名对一个或一批用户进行推送。目前一个别名最多允许绑定 10 个 clientid。

别名规则说明：

1. 有效的别名组成：字母（区分大小写）、数字、下划线、汉字
2. 任务备注名长度限制为 40 字节。(UTF-8)
3. 一个别名最多允许绑定 10 个 clientid。

1.2 对应接口

1.2.1 bindAlias-单个 clientid 绑定别名

一个 clientid 只能绑定一个别名，若已绑定过别名的 clientid 再次绑定新别名，则认为与前一个别名自动解绑，绑定新别名。

函数说明：

```
bindAlias(const char *appKey, const char* appId, const char* alias, const char* cid)
```

参数说明：

参数名	类型	必需	默认值	参数描述
appKey	char*	是	无	调用 pushInit 时 APP 对应的 appKey
appId	char*	是	无	用户所属应用 id
alias	char*	是	无	用户别名
cid	char*	是	无	用户 id

代码实例

```
#include "IGtPush.h"
#include <iostream>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <string>
#ifdef WIN32
```

```
#include <windows.h>
#endif

using namespace std;
static void printResult(IPushResult &result);

//单个别名绑定
void bindAliastest();


// http的域名
//static char *host ="http://sdk.open.api.igexin.com/apiex.htm";

//https的域名
static char *host ="https://api.getui.com/apiex.htm";

//采用"C++ SDK 快速入门", "第二步 获取访问凭证 "中获得的应用配置
static char *appId = "";            //您应用的id
static char *appKey = "";           //您的appkey
static char *masterSecret = "";     //您的masterSercret
static char *cid = "";              //用户的id

int main(){

    // 注意：接口传入字符必须为UTF-8编码，因ASCII字符UTF－8编码后与原先一样，所以可以不编码,
    // 但中文等非ASCII字符必须编码
    // 如果返回的类似错误"post http data failed, code=6"，错误码可百度CURL返回的错误码是
    // 什么意思，如http://www.cnblogs.com/wainiwann/p/3492939.html
    // 程序运行前只需初始化一遍，若已经初始化过即可不用，"编码"两个字为固定写法，不需要做转换
    Result r = pushInit(host, appKey, masterSecret, "编码");
    if(r!=SUCCESS){
        printf("pushInit for app failed: ret=%d\n", r);
        return -1;
    }

    bindAliastest();
    //bindAliasListtest();
    //queryAliastest();
    //queryClientIdtest();
    //unBindAliastest();
    //unBindAliasAlltest();

    return 0;
}
```

```
void bindAliastest(){
    IPushResult result = bindAlias(appKey, appId, "test", cid);
    printResult(result);
}

static void printResult(IPushResult &result) {
    cout << "print result:-------------" << endl;
    for (int i = 0; i < result.size; i++) {
        cout << result.entry[i].key << ": " << result.entry[i].value << endl;
    }
    cout << "print end:-----------------" << endl;
}
```

1.2.2 bindAliasList-多个 clientid 绑定别名

允许将多个 clientid 和一个别名绑定，如用户使用多终端，则可将多终端对应的 clientid 绑定为一个别名，目前一个别名最多支持绑定 10 个 clientid。

函数说明：

```
bindAliasList(const char *appKey, const char* appId, Target *target, int num)
```

参数说明：

参数名	类型	必需	默认值	参数描述
appKey	char*	是	无	调用 pushInit 时 APP 对应的 appKey
appId	char*	是	无	用户所属应用 id
target	Target*	是	无	别名与 cid 对应列表
num	int	是	无	target 列表大小

代码实例

```
void bindAliasListtest(){
    Target* targetList = new Target[1];
    targetList->clientId = cid;
    targetList->alias = "test";
    IPushResult result = bindAliasList(appKey, appId, targetList, 1);
    printResult(result);
    delete []targetList;
}
// 注：只要有一个cid绑定成功，返回结果就为true
```

完整代码请参考第一个示例 bindAlias

1.2.3 queryClientId-根据别名获取 clientid 信息

函数说明：

```
queryClientId(const char *appKey, const char* appId, const char *alias, char** list)
```

参数说明：

参数名	类型	必需	默认值	参数描述
appKey	char*	是	无	调用 pushInit 时 APP 对应的 appKey
appId	char*	是	无	用户所属应用 id
alias	char*	是	无	用户别名
list	char**	是	无	获得别名绑定的 cid 列表

代码实例

```
void queryClientIdtest() {
    char* list = NULL;
    IPushResult result = queryClientId(appKey, appId, "test", &list);

    printResult(result);

    if(list != NULL) {
        cout << "cidlist:" << list<<endl;

        //释放资源
        releaseMem(list);
    }
}
```

完整代码请参考第一个示例 bindAlias

1.2.4 queryAlias-通过 clientid 获取别名信息

函数说明：

```
queryAlias(const char *appKey,const char *appId, const char *cid)
```

参数说明：

参数名	类型	必需	默认值	参数描述
appKey	char*	是	无	调用 pushInit 时 APP 对应的 appKey
appId	char*	是	无	用户所属应用 id
cid	char*	是	无	用户 id

代码实例

```
void queryAliastest(){
    IPushResult result = queryAlias(appKey, appId, cid);
    printResult(result);
}
```

完整代码请参考第一个示例 bindAlias

1.2.5 unBindAlias-单个 clientid 和别名解绑

函数说明：

```
unBindAlias(const char *appKey,const char *appId, const char *alias, const char* cid)
```

参数说明：

参数名	类型	必需	默认值	参数描述
appKey	char*	是	无	调用 pushInit 时 APP 对应的 appKey
appId	char*	是	无	用户所属应用 id
alias	char*	是	无	用户别名
cid	char*	是	无	用户 id

代码实例

```
void unBindAliastest() {
    IPushResult result = unBindAlias(appKey, appId, "test", cid);
    printResult(result);
}
```

完整代码请参考第一个示例 bindAlias

1.2.6 unBindAliasAll-绑定别名的所有 clientid 解绑

函数说明：

```
unBindAliasAll(const char *appKey,const char *appId, const char *alias)
```

参数说明

参数名	类型	必需	默认值	参数描述
appKey	char*	是	无	调用 pushInit 时 APP 对应的 appKey
appId	char*	是	无	用户所属应用 id
alias	char*	是	无	用户别名

代码实例：

```
void unBindAliasAlltest() {
    IPushResult result = unBindAliasAll(appKey, appId, "test");
    printResult(result);
}
```

完整代码请参考第一个示例 bindAlias

2. 停止任务接口

2.1 接口名称

pushStop(appkey, taskId)

2.2 参数描述

参数	说明
appKey	用于鉴定身份是否合法
taskId	平台返回用来标识任务（格式 OSL-yyMM_XXXXXX）

2.3 stop 代码实例

```
#include "IGtPush.h"
#include <iostream>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <string>
#ifdef WIN32
#include <windows.h>
#endif

using namespace std;

void stopTaskTest();

// http的域名
//static char *host ="http://sdk.open.api.igexin.com/apiex.htm";

//https的域名
static char *host ="https://api.getui.com/apiex.htm";

//采用"C++ SDK 快速入门", "第二步 获取访问凭证 "中获得的应用配置
static char *appKey = "";           //您的appkey
```

```
static char *masterSecret = "";  //您的masterSecret

int main(){
    // 注意：接口传入字符必须为UTF-8编码，因ASCII字符UTF－8编码后与原先一样，所以可以不编码,
    // 但中文等非ASCII字符必须编码
    // 如果返回的类似错误"post http data failed, code=6"，错误码可百度CURL返回的错误码是
    // 什么意思，如http://www.cnblogs.com/wainiwann/p/3492939.html
    // 程序运行前只需初始化一遍，若已经初始化过即可不用，"编码"两个字为固定写法，不需要做转换
    Result r = pushInit(host, appKey, masterSecret, "编码");
    if(r!=SUCCESS){
        printf("pushInit for app failed: ret=%d\n", r);
        return -1;
    }

    stopTaskTest();
    return 0;
}

void stopTaskTest() {
        Result result = pushStop(appKey,"OSA-1126_lsJPsulDip7Pcd5JqWgUe8");
        printf("%d\n", result);
}
```

3. 获取推送结果

3.1 接口名称

getPushResult(const char *appKey, const char* taskId)

3.2 方法参数

参数	参数说明
Appkey	用于鉴定身份是否合法
taskId	任务唯一识别号 (格式 OSL-yyMM_XXXXXX)

3.3 返回值

回执	参数说明
	{``taskeId'':''OSA-0820_uQ7gevLuGS7Odz8FS2ZSB9'',''result'':''ok'',''msgTotal'':59,''msgProcess'':0}
	tasked：任务 ID
	msgTotal：表示有效可下发总数
	result：OK 执行成功

	result：sign_error 表示校验失败
	msgProcess：收到消息回执总数

3.4 代码实例

```
#include "IGtPush.h"
#include <iostream>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <string>
#ifdef WIN32
#include <windows.h>
#endif

using namespace std;
static void printResult(IPushResult &result);

void getresult();

// http的域名
//static char *host ="http://sdk.open.api.igexin.com/apiex.htm";

//https的域名
static char *host ="https://api.getui.com/apiex.htm";

//采用"C++ SDK 快速入门", "第二步 获取访问凭证 "中获得的应用配置
static char *appKey = "";           //您的appkey
static char *masterSecret = "";  //您的masterSecret

int main(){
    // 注意：接口传入字符必须为UTF-8编码，因ASCII字符UTF－8编码后与原先一样，所以可以不编码,
    // 但中文等非ASCII字符必须编码
    // 如果返回的类似错误"post http data failed, code=6"，错误码可百度CURL返回的错误码是
    // 什么意思，如http://www.cnblogs.com/wainiwann/p/3492939.html
    // 程序运行前只需初始化一遍，若已经初始化过即可不用，"编码"两个字为固定写法，不需要做转换
    Result r = pushInit(host, appKey, masterSecret, "编码");
    if(r!=SUCCESS){
        printf("pushInit for app failed: ret=%d\n", r);
        return -1;
    }

    getresult();
    return 0;
```

```
}

void getresult(){

    IPushResult result = {0};
    result = getPushResult(appKey,"OSA-1126_3iZaaoJ5TQ7Q6gOEQmVfe1");
    printResult(result);
}
static void printResult(IPushResult &result) {
    cout << "print result:--------------" << endl;
    for (int i = 0; i < result.size; i++) {
        cout << result.entry[i].key << ": " << result.entry[i].value << endl;
    }
    cout << "print end:-----------------" << endl;
}
```

4. 获取单日用户数据

4.1 描述

调用此接口可以获取某个应用单日的用户数据（用户数据包括：新增用户数，累计注册用户总数，在线峰值，日联网用户数）（目前只支持查询 1 天前的数据）

4.2 接口名称：

queryAppUserDataByDate(const char *appKey, const char* appId, const char* date)

4.3 接口参数：

字段	说明
appKey	用于鉴定身份是否合法
appId	应用 ID
date	查询的日期（格式：yyyyMMdd）

4.4 返回值：

字段	上级	含义
result	-	成功：ok
data	-	查询数据对象
appId	data	请求的 AppId
date	data	查询的日期（格式：yyyyMMdd）
newRegistCount	data	新注册用户数
registTotalCount	data	新注册用户数

activeCount	data	活跃用户数
onlineCount	data	在线用户数

4.5 queryAppUserDataByDate 代码示例：

```
#include "IGtPush.h"
#include <iostream>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <string>
#ifdef WIN32
#include <windows.h>
#endif

using namespace std;
static void printQueryResult(IQueryResult &result);

void getuserinfo();

// http的域名
//static char *host ="http://sdk.open.api.igexin.com/apiex.htm";

//https的域名
static char *host ="https://api.getui.com/apiex.htm";

//采用"C++ SDK 快速入门", "第二步 获取访问凭证 "中获得的应用配置
static char *appId = "";            //您应用的id
static char *appKey = "";           //您的appkey
static char *masterSecret = "";  //您的masterSecret

int main(){
    // 注意：接口传入字符必须为UTF-8编码，因ASCII字符UTF－8编码后与原先一样，所以可以不编码,
    // 但中文等非ASCII字符必须编码
    // 如果返回的类似错误"post http data failed, code=6"，错误码可百度CURL返回的错误码是
    // 什么意思，如http://www.cnblogs.com/wainiwann/p/3492939.html
    // 程序运行前只需初始化一遍，若已经初始化过即可不用，"编码"两个字为固定写法，不需要做转换
    Result r = pushInit(host, appKey, masterSecret, "编码");
    if(r!=SUCCESS){
        printf("pushInit for app failed: ret=%d\n", r);
        return -1;
    }

    getuserinfo();
```

```
    return 0;
}
void getuserinfo(){
    //初始化
    IQueryResult result = {0};
    result = queryAppUserDataByDate(appKey,appId,"20150709");
    printQueryResult(result);
}
static void printQueryResult(IQueryResult &result) {
    cout << "print result:-------------" << endl;
    cout <<"result:"<<result.result<<endl;
    for (int i = 0; i < result.size; i++) {
        cout << result.data[i].key << ": " << result.data[i].value << endl;
    }
    cout << "print end:-----------------" << endl;
}
```

5. 获取单日推送数据

描述

调用此接口可以获取某个应用单日的推送数据（推送数据包括：发送总数，在线发送数，接收数，展示数，点击数）（目前只支持查询 1 天前的数据）

5.1 接口名称：

queryAppPushDataByDate(const char *appKey, const char* appId, const char* date)

5.2 接口参数：

字段	说明
appKey	用于鉴定身份是否合法
appId	应用 ID
date	查询的日期（格式：yyyyMMdd）

5.3 返回值：

字段	上级	含义
result	-	成功：ok
data	-	查询数据对象
appId	data	请求的 AppId
date	data	查询的日期（格式：yyyyMMdd）
sendCount	data	发送总数

sendOnlineCount	data	在线发送数
receiveCount	data	接收数
showCount	data	展示数
clickCount	data	点击数

5.4 queryAppPushDataByDate 代码示例：

```
#include "IGtPush.h"
#include <iostream>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <string>
#ifdef WIN32
#include <windows.h>
#endif

using namespace std;
static void printQueryResult(IQueryResult &result);

void getappinfo();

// http的域名
//static char *host ="http://sdk.open.api.igexin.com/apiex.htm";

//https的域名
static char *host ="https://api.getui.com/apiex.htm";

//采用"C++ SDK 快速入门", "第二步 获取访问凭证 "中获得的应用配置
static char *appId = "";            //您应用的id
static char *appKey = "";           //您的appkey

int main(){
    // 注意：接口传入字符必须为UTF-8编码，因ASCII字符UTF－8编码后与原先一样，所以可以不编码,
    // 但中文等非ASCII字符必须编码
    // 如果返回的类似错误"post http data failed, code=6"，错误码可百度CURL返回的错误码是
    // 什么意思，如http://www.cnblogs.com/wainiwann/p/3492939.html
    // 程序运行前只需初始化一遍，若已经初始化过即可不用，"编码"两个字为固定写法，不需要做转换
    Result r = pushInit(host, appKey, masterSecret, "编码");
    if(r!=SUCCESS){
        printf("pushInit for app failed: ret=%d\n", r);
        return -1;
    }
```

```
    getappinfo();
    return 0;
}
void getappinfo(){
    IQueryResult result = {0};
    result = queryAppPushDataByDate(appKey,appId,"20150709");
    printQueryResult(result);
}
static void printQueryResult(IQueryResult &result) {
    cout << "print result:-------------" << endl;
    cout <<"result:"<<result.result<<endl;
    for (int i = 0; i < result.size; i++) {
        cout << result.data[i].key << ": " << result.data[i].value << endl;
    }
    cout << "print end:-----------------" << endl;
}
```