

# 云之家轻应用接入指南

## 1 目录

1 企业数据接入.....	2
1.1 注册云之家账号.....	2
1.2 创建自己的公司企业或团队.....	2
1.3 人员组织数据接入.....	2
1.4 第三方企业数据接入到云之家后,不是也可以使用彼此的账号互相登录云之家和第三方企业应用了? .....	3
1.5 企业注册号是什么? .....	3
2 开发者注册.....	4
3 轻应用接入 .....	5
3.1 登录.....	5
3.2 企业管理员操作界面.....	6
3.3 开发者操作界面.....	7
3.4 获取应用 ID 和 Appsecret .....	8
4 获取当前用户信息 .....	9
4.1 云之家中会附带 ticket 的场景.....	9
4.2 根据应用 ID 和 appSecret 获取 AccessToken.....	10
4.3 根据 ticket 和 accessToken 获取上下文信息 .....	12
5 企业如何创建公共号? .....	13
5.1 企业管理员登录管理中心.....	13
5.1.1 企业管理员登录管理中心.....	13

5.1.2	创建公共号.....	13
5.2	如何进入公共号管理平台? .....	14
5.3	如何获得公共号 ID 和公共号密钥.....	14
5.4	公共消息推送 .....	15

## 1 企业数据接入

云之家的“团队”，大至为一个“企业、公司、单位”，小至为一个“团队”。大方向的话，一个大老板确能拥有多个子分公司！

这里的“企业”概念，就是云之家里面提及的“团队”。企业数据接入云之家需要经过以下三个步骤：

### 1.1 注册云之家账号

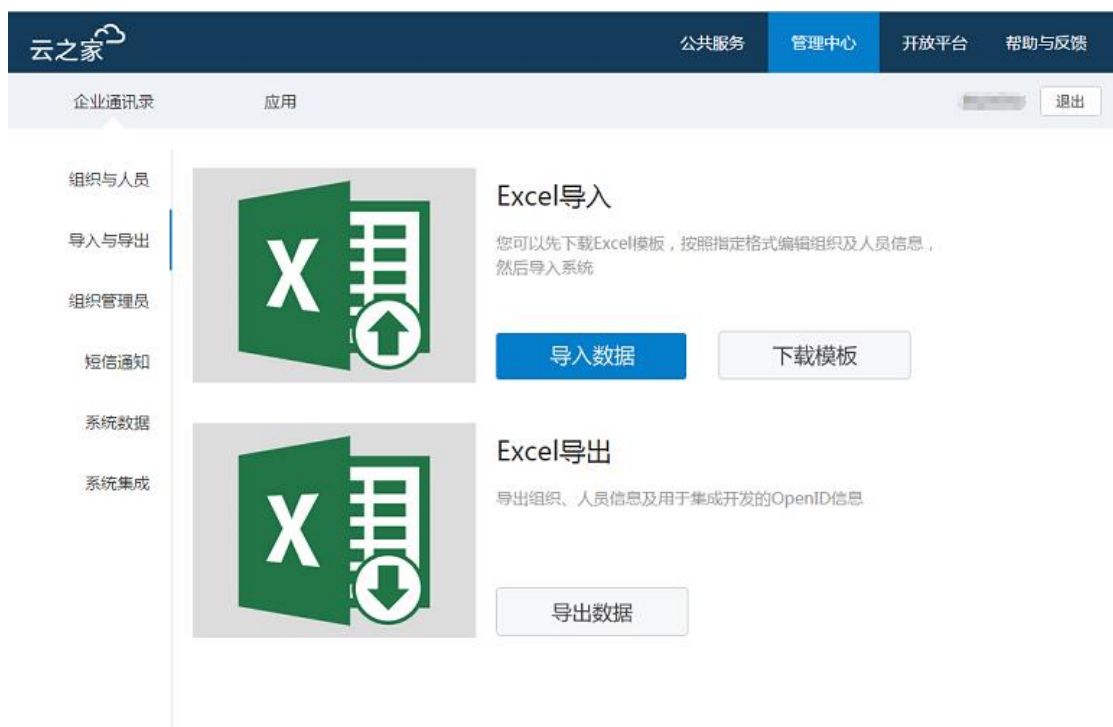
注册云之家账号可以通过[云之家 Web 首页](#)或者下载[云之家 APP](#)进行注册。

### 1.2 创建自己的公司企业或团队

以自己所在的公司，企业或团队创建团队。

### 1.3 人员组织数据接入

途径 1：使用注册的云之家账号登录管理中心进入上一步创建的工作圈中，选择“导入与导出”下载模板，按模板填写组织人员信息后，再导入 excel！实现人员组织数据的导入！如图所示。



该途径适合于“为了方便而使用云之家管理中心进行企业组织和人员的管理的第三方企业”，而对于“具备了一定开发能力的并且想把自己的应用接入到云之家的第三方企业”则更偏向于途径 2。

途径 2：使用组织人员同步 API 接口利用企业密钥进行人员组织数据的接入。这里面提及的 eid 为企业的注册号。

#### 1.4 第三方企业数据接入到云之家后，不是也可以使用彼此的账号互相登录云之家和第三方企业应用了？

云之家提供给外部企业把数据接入云之家，一方面是方便第三方企业能够使用云之家管理中心进行其企业组织和人员的管理；另一方面是提供给自身具备了一定开发能力的第三方企业在把自己的应用接入到云之家中时，进行第三方企业自身的用户数据和云之家的用户数据有着一定的对应关系。云之家对外暴漏的是用户的 openid 字段，一个 openid 对应一个云之家用户，第三方企业可以通过云之家 openid 和自身的用户 ID 进行对应，从而实现在应用开发时，能够使用组织人员的信息，实现应用的完整开发。

所以，这里的同步并不一定就意味着“第三方企业数据接入到云之家后，就可以使用彼此的账号互相登录云之家和第三方企业应用了”。

#### 1.5 企业注册号是什么？

**企业注册号**简称 eID/eid/EID，是使用金蝶移动产品时对企业进行区分的唯一标识，云之家及其他金蝶移动产品通过企业注册号来识别并隔离不同的企业，保障企业信息的安全性。可以通过云之家 APP 进行查询，具体见如下操作。登录云之家 APP 进入上一步创建的工作圈后在“消息”页签会出现“管理员助手”的订阅消息，点击该条“管理员助手”的订阅——>点击菜单“查询注册号”得到。如图所示。



## 2 开发者注册

### [开发者注册指南](#)

注意：开发者需要对开发应用的质量、稳定性、内容负责。如出现负面影响，云之家团队可不经沟通关停开发者账号，下架此开发者账号下的所有应用，并进行责任追究！

### 3 轻应用接入

#### 3.1 登录

使用开发者账号或企业管理员到企业移动云服务平台（<http://do.kdweibo.com/mcloud/pages/>）



### 3.2 企业管理员操作界面

登录后进入应用管理页签中创建自己的轻应用，填写“应用 ID、AppSecret”等相关信息，使用应用 ID 和 AppSecret 进入应用接口层面的开发。

如图所示：



### 3.3 开发者操作界面

登录后进入轻应用注册页签中创建自己的轻应用，填写“应用 ID、AppSecret”等相关信息，在审批授权后，使用应用 ID 和 AppSecret 进入应用接口层面的开发。

如图所示：

金蝶企业移动管理云

客户管理 应用管理 轻应用注册

轻应用注册列表

应用名称: 产品领域: 全部 申请状态: 全部 搜索

应用名称	产品领域	应用ID	应用状态	申请时间	审核时间	操作
管理员助手	云之家	10050	审核通过	2014-07-15 16:15:46	2014-07-15 16:16:31	删除

注册轻应用

应用信息

产品领域: 云之家

AppSecret:

应用名称 \*: 不能与其他移动应用同名

应用类型: 轻应用

是否免费: ☐

访问地址 \*: 配置进入应用首页地址

应用说明:

应用分类

应用标签

客户端标签: ☐ 效率 ☐ 协作 ☐ 知识管理 ☐ 通知 ☐ 会议 ☐ 人力资源 ☐ 其他

保存

## 3.4 获取应用 ID 和 APPSECRET

如图所示：





## 4 获取当前用户信息

### 4.1 云之家中会附带 ticket 的场景

1. 消息中的轻应用
2. 公共号菜单中轻应用
3. 云之家应用中心的轻应用

如图所示：



## 4.2 根据应用 ID 和 appSecret 获取 AccessToken

开发者在申请到 **appid** 和 **appSecret** 后按照 [appAuth2 授权协议](#) 进行授权认证或获取

access\_token

注意：头部请求的 header 的 authorization 中传输的认证协议，具体调用见 [API 开发文档](#)

Java 示例：

```
/**
 * 说明：获取AccessToken
 *
 * { "success": true, "data": { "expires_in": 604800, "access_token":
"XXX"
 * } }
 *
 * @return
 * @throws Exception
 */
@Override
public String getAccessToken(String appId, String appSecret)
    throws Exception {
    String access_token = "";
    String version = "1.1";
    String nonce = String.valueOf(Math.random());
    String timestamp = String.valueOf(System.currentTimeMillis());
    String sign = sha(version, appId, timestamp, nonce, appSecret);
    String url = "http://do.yunzhijia.com/openauth2/api/appAuth2";
    StringBuffer sb = new StringBuffer();
    sb.append("OpenAuth2 version=");
    sb.append(version);
    sb.append(", appid=");
    sb.append(appId);
    sb.append(", timestamp=");
    sb.append(timestamp);
    sb.append(", nonce=");
    sb.append(nonce);
    sb.append(", sign=");
    sb.append(sign);
    Map<String, String> headers = new HashMap<String, String>();
```

```
headers.put("authorization", sb.toString());
Map<String, String> params = null;
String result = HttpHelper.get(headers, params, url);
JSONObject resultJson = new JSONObject().fromObject(result);
if(resultJson!=null&&resultJson.containsKey("data")){
    JSONObject dataJson = resultJson.getJSONObject("data");
    if(dataJson!=null&&dataJson.containsKey("access_token")){
        access_token = dataJson.getString("access_token");
    }
}
return access_token;
}

private static String sha(String... data)
{
    //按字母顺序排序数组
    Arrays.sort(data);
    //把数组连接成字符串（无分隔符），并sha1哈希
    return DigestUtils.shaHex(StringUtils.join(data));
}
```

#### 4.3 根据 ticket 和 accessToken 获取上下文信息

获取上下文信息见 [API 开发文档](#)

Java 示例:

```
/**
 * 说明: 根据ticket获取上下文信息
 * @param ticket
 * @param appId
 * @param appSecret
 * @return
 * @throws Exception
 */
private static String getContextByTicket(String ticket, String appId,
    String appSecret) throws Exception {
    String token = getAccessToken(appId, appSecret);
    Map<String, String> params = new HashMap<String,String>();
    params.put("access_token", token);
    params.put("ticket", ticket);
    String url="http://do.yunzhijia.com/openauth2/api/getcontext";
    String result = HttpHelper.get(params, url);
    return result;
}
```

## 5 企业如何创建公共号？

### 5.1 企业管理员登录管理中心

企业公共号只能由企业管理员（工作圈管理员）负责创建。开发者需要要求企业管理员进行企业公共号的创建。企业管理员通过如下步骤进行企业公共号的创建：

#### 5.1.1 企业管理员登录[管理中心](#)

企业管理员使用其云之家账号( 可以通过[云之家 Web 首页](#)或者下载[云之家 APP](#) 进行注册 ) 登录[管理中心](#)进入需要创建公共号的企业。

#### 5.1.2 创建公共号

进入[管理中心](#)后选择应用页签出现“公共服务号列表”，点击“+”号进行公共号的创建。如图所示。



创建好公共号后，该创建公共号的企业管理员便为该公共号的公共号管理员。企业管理员可以在此再任命更多的人成为该公共号的公共号管理员。

## 5.2 如何进入公共号管理平台？

企业管理员创建完公共号后可以点击“公共服务”页签进入公共号管理平台，也可以和公共号管理员一样通过公共号管理平台进入公共号管理平台。注意：如果企业管理员没有创建任何公共号，则是不能进入公共号管理平台进入公共号管理平台的。

## 5.3 如何获得公共号 ID 和公共号密钥

企业管理员和公共号管理员进入公共号管理平台后选择“开发者”下的“API 接口”。如图所示。



## 5.4 公共消息推送

发公共号消息见 [API 开发文档](#)

Java 示例:

```
/**
 * 说明: 推送公共号消息
 * @param pubaccId 公共号Id
 * @param pubaccKey 公共号密钥
 * @param eId 企业eId
 * @param code 其中, "to":[{"no":"10001","code":"all"}] 表示企业10001
    所有订阅的用户。"code":"all"一定不要漏, 否则报错
 * @param text 消息内容
 * @param todo 是否推送待办消息, 格式为整型, 默认1=推送到待办消息; 0=推送原公
    共号消息"
 * @param type 定义: 2:纯文本信息 5:文本链接信息 6:图文混排信息

 * @return
 * @throws Exception
 */
private static String pushPubaccMsg(
    String pubaccId, String pubaccKey, String eId, String code,
    String text, int todo, int type) throws Exception {
    String random = String.valueOf(Math.random());
    long time_1 = System.currentTimeMillis();
    //form
    JSONObject from = new JSONObject();
    from.put("no", eId); //工作圈id, 与eid值相同
    from.put("pub", pubaccId); //公共号编码
    from.put("time", time_1);
    from.put("nonce", random);
    String[] data={eId,pubaccId,pubaccKey,random,
String.valueOf(time_1)};
    from.put("pubtoken",sha(data));

    //to
    JSONArray tos = new JSONArray();
    JSONObject to = new JSONObject();
    to.put("no",eId);
    to.put("code", code);
```

```
tos.add(to);

JSONObject msg = new JSONObject();
msg.put("text", text);
msg.put("todo", todo);

JSONObject content = new JSONObject();
content.put("from", from);
content.put("to", tos);
content.put("type", type);
content.put("msg", msg);
System.out.println(content.toString());
String pushUrl = "http://do.yunzhijia.com/pubacc/pubsend";
String result = HttpClientHelper.sendPost(pushUrl,
content.toString());
    return result;
}
public static String sha(String[] data) {
    Arrays.sort(data); // 按字母顺序排序数组
    return DigestUtils.shaHex(StringUtils.join(data)); // 把数组连接成字符串无分隔符), 并sha1哈希
}
```