

# 测试接入流程

## 1. 注册开发者账号

1. 开发者根据测试接入API调用 *开发者注册* 接口。
2. 测试平台根据开发者接口提交的邮箱地址(必须)和回调地址(非必须)，向开发者邮箱发送一份邮件，内含 `service_id`和`secret`。
3. 后续所有的测试接口请求，都需要带上此邮箱地址以及由邮箱地址、`service_id`和`secret`拼接字符串的md5值。测试平台的所有接口都将校验此md5。
4. 此接口不提供额外查询功能，如开发者信息遗失，需走客服流程找回。

## 2. 调用充值接口

1. 开发者根据测试接入API调用 *测试账号充值* 接口。
2. 调用充值接口成功后，测试平台会向接口所提交的address账户转入1个链克。
3. 测试平台的区块链环境完全独立，所以此测试链克只能用于测试环境，转入其他区块链环境无效。
4. 每个email地址每天最多发起10此转账申请。

## 3. 开发合约应用和上层业务

1. 开发者开发合约应用，并通过上层业务(App, H5等实现)功能封装对合约的调用。
2. 合约的开发可参考《迅雷区块链大赛合约开发指南》。

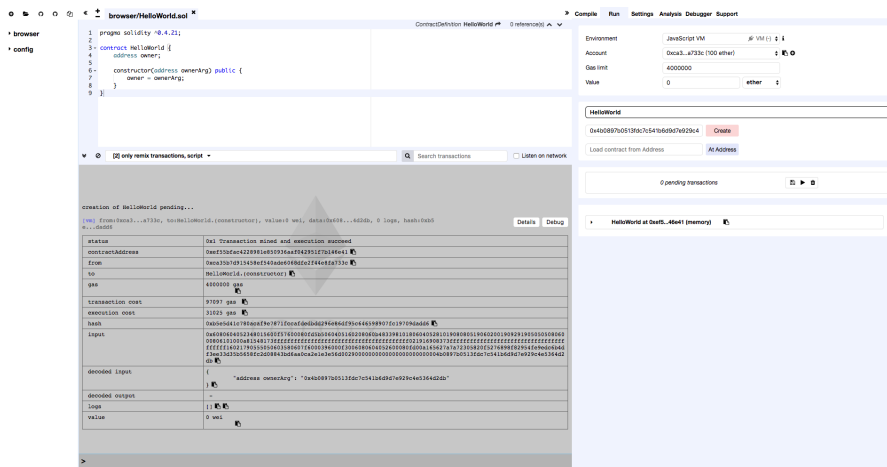
## 4. 部署合约到迅雷链测试环境

1. 迅雷链测试和正式环境不提供直接RPC连接执行合约发送交易的过程，合约的发布和调用都需要通过调用API接口实现。
2. 开发者通过调用合约发布接口，部署合约到迅雷链测试环境。
3. 由于合约的部署是由平台方执行的，所以在合约的`constructor`方法里，避免使用`msg.sender`，如有限权和调用者限制需用到`msg.sender`，可通过参数传入用户的`address`代替。

例：

```
contract HelloWorld {
    address owner;
    constructor(address ownerArg) public {
        //owner = msg.sender; 使用ownerArg代替msg.sender
        owner = ownerArg;
    }
}
```

1. 发布合约API接口需要提供合约bytecode和params，bytecode为合约编译的bytecode，params是构造函数的参数经过sign后可直接拼接在bytecode后用于部署的编码。  
这里可以使用 [remix](#) create合约，控制台里的这一条交易的detail里input即合约部署的data。但为了兼容测试环境和正式环境的输入，需要将input里的合约bytecode和最后的params sign分开传入。



## 5. 调用合约方法

合约调用当前分为两种形式：通过链克口袋App扫描二维码发送合约调用的交易；通过业务方App唤起链克口袋发起合约调用的交易。

测试环境链克口袋下载地址 <https://www.pgyer.com/ALuZ>

## 5.1 唤起链克口袋

根据API实现业务APP唤起链克口袋，并传入合约调用的交易。

## 从玩客云调用链克口袋

otst://contract/?tx-

data=ZGVzYzNlLXlxbFYWFhYJnRvPTB4MTIzNDU0ZnZ5MDEeYmZQ1Njc4OTAmZmFsdWU9MTIzLjQwJmdhc2xpbWl0PTUwMDAwJmRhdGE9MHgwMTAyMDM  
source=wky&x-success=wky://x-callback-url/contractSuccess&x-error=wky://x-callback-url/contractError&x-  
cancel=wky://x-callback-url/contractCancel&&cb-data=base64编码后的回调遗传参数

### 具体解析

1. 合约执行业务 otst://contract
2. 源app名字, resource=d2t5, 解码后是wky
3. 源app回调前缀, x-source=wky
4. 成功回调 &x-success=wky://x-callback-url/contractSuccess
5. 失败回调 &x-error=wky://x-callback-url/contractError
6. 取消回调 &x-cancel=wky://x-callback-url/contractCancel
7. 回调时, 直接回传 &cb-data=abcdefg
8. 交易信息tx-  
data=ZGVzYz3nILXlvbFYWFhYJnRvPTB4MTIzNDU2Nzg5MDEyMzQ1Njc4OTAmdmFsdWU9MTIzLjQwJmdhc2xpbWI0PTUwMDAwJmRhdGE9MHgwMTAyI

解码后包含如下信息

- [illegible]

返回 (成功)

wky://x-callback-url/contractSuccess?cb-

data=abcdefg&hash=0x12345678901234567890123456789012&data=base64编码后的回调透传参数

返回(失败)

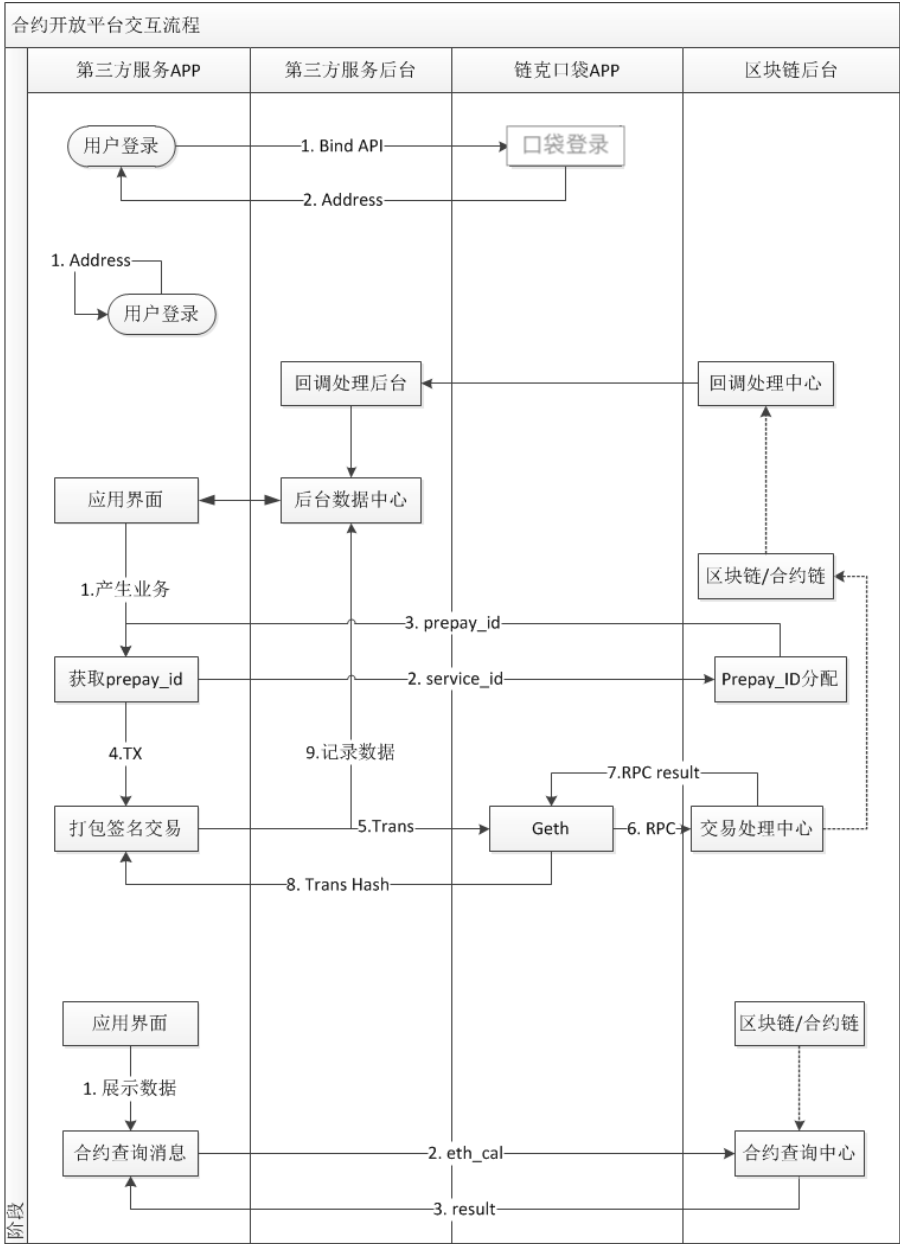
wky://x-callback-url/contractError?x-source=otc&errorCode=1&errorMessage=message&data=base64编码后的回调透传参数

返回(取消)

wky://x-callback-url/contractCancel?x-source=otc&data=base64编码后的回调透传参数

合约调用

合约调用指需要改变合约状态的函数调用，执行函数的同时也可以转账到合约账户。用户界面输入用户请求后，触发合约调用。



步骤如下：

1. 第三方应用接受用户输入，启动合约调用流程
2. 用户使用链克区块链分配的service\_id，到区块链后台请求，分配一个prepay\_id。
3. 后台收到请求后，产生一个prepay\_id给第三方应用。
4. 第三方应用打包交易，主要包括合约地址、gas\_limit、转账金额、执行的函数和参数编码(ABI)、签名等信息。
5. 唤醒链克钱包app，将交易信息发送给链克口袋。
6. 链克口袋打包交易发送给区块链交易处理中心，最终交易转发给区块链服务器处理。
7. 返回请求结果给链克口袋。
8. 链克口袋将结果通知给第三方APP。(合约调用交互完成)

第三方应用可以同步数据到其后台服务。（第三方功能）

区块链后台将合约调用请求处理完后，如果用户有填写回调信息，回调中心通知第三方应用后台。第三方应用界面会同第三方应用后台同步信息，展示交易后的结果。

## 5.2 查询 constant 合约方法

查询合约constant状态和方法，不消耗gas，使用eth\_call API方法直接调用。

样例：

### Example

在合约部署成功的情况下，执行以下操作：

- 未部署合约

```
[root@t05f058s2 ~]# curl -k -X POST -d '{"jsonrpc":"2.0","method":"eth_call","params": [{"from":"0x7eff122b94897ea5b0e2a9abf47b86337fafebdc","to":"0xfe2587bdc1781f23d105002c328d3d2ea6cfcbdd","data":"0xb0f0c96a000000000000000000000000"}, {"latest"}], "id":1}' https://sandbox-walletapi.onethingpcs.com/call
```

```
{'error':{'message':'invalid request','code':-32600},'jsonrpc':'2.0','id':1}
```

- 已部署合约

[illegible]

- 参数说明

to: 部署合约成功之后, 获得的合约地址

data: 计算方法

[illegible]