**Trinity 接口说明**

接口通过 JSON-RPC 的方式提供，底层使用 HTTP 协议进行通讯

测试服务器：

106.15.91.150

**端口**

主网: 10552

测试网：20552

JSON-RPC HTTP 10552 20552

**GET** 请求示例

一次典型的 JSON-RPC GET 请求格式如下：

下面以注册地址方法为例。

请求 URL：

[http://somewebsite.com:20552?jsonrpc=2.0&method=registeaddress&params=["AY8r7uG6rH7MRLhABALZvf8jM4bCSfn3YJ"]&id=1](http://somewebsite.com:20552?jsonrpc=2.0&method=registeaddress&params=%5B%22AY8r7uG6rH7MRLhABALZvf8jM4bCSfn3YJ%22%5D&id=1)

发送请求后，将会得到如下的响应：

"jsonrpc": "2.0",

"id": 1,

"result": [0,"SUCCESS"]

}

**POST** 请求示例

一次典型的 JSON-RPC Post 请求的格式如下：

请求 URL：

http:// 106.15.91.150:20552

请求 Body：

{

"jsonrpc": "2.0",

"method": "registeaddress",

"params": ["AY8r7uG6rH7MRLhABALZvf8jM4bCSfn3YJ"],

"id": 1

}

发送请求后，将会得到如下的响应：

{

"jsonrpc": "2.0",

"id": 1,

"result": [0,"SUCCESS"]

}

**registeaddress**

地址注册至网关S

**参数说明**

address: 地址，string

port:监听端口号,string

**调用示例**

请求正文：

{+

"jsonrpc": "2.0",

"method": "registeaddress",

"params": ["AY8r7uG6rH7MRLhABALZvf8jM4bCSfn3YJ","5678"],

"id": 1

}

响应正文：

{

"jsonrpc": "2.0",

"id": 1,

"result": [0,"SUCCESS"]

}

响应说明：

[0,"SUCCESS"] 注册成功

[error\_code,"error message"]

**getbalanceonchain**

获取地址的链上资产

**参数说明**

Local\_addr: 本地的地址， string

asset\_type：资产类型（代币编码），string 参数为可选参数，不输入返回所有资产类型的链上资产情况

**调用示例**

请求正文：

{

"jsonrpc": "2.0",

"method": "getchannelstate",

"params": ["AY8r7uG6rH7MRLhABALZvf8jM4bCSfn3YJ"],

"id": 1

}

响应正文：

{

"jsonrpc": "2.0",

"id": 1,

"result":[{"AssetType":"TNC",

"Balance":10000},

{"AssetType":"NEO",

"Balance":10000},

{"AssetType":"NEOGAS",

"Balance":10000}]

}

**getchannelstate**

获取channale 状态

**参数说明**

Local\_addr: 本地的地址， string

**调用示例**

请求正文：

{

"jsonrpc": "2.0",

"method": "getchannelstate",

"params": ["AY8r7uG6rH7MRLhABALZvf8jM4bCSfn3YJ"],

"id": 1

}

响应正文：

{

"jsonrpc": "2.0",

"id": 1,

"result":

[{"channel\_name":"AUvupHdCKhaSsfPpzo6f4He4c9wFwhkLmZAY8r7uG6rH7MRLhABALZvf8jM4bCSfn3YJ",

"channel\_state":"OPEN",

"open\_block": 1000,

"tx\_info":[{"address":"AY8r7uG6rH7MRLhABALZvf8jM4bCSfn3YJ",

"despoit": 100,

"balance": 20

},

{"address":"AUvupHdCKhaSsfPpzo6f4He4c9wFwhkLmZ",

"despoit": 0,

"balance": 80

}]

},

{"channel\_name":"Axxxxxxxxxxx8r7uG6rH7MRLhABALZvf8jM4bCSfn3YJ",

"channel\_state":"OPEN",

"open\_block": 12000,

"tx\_info":[{"address":"AY8r7uG6rH7MRLhABALZvf8jM4bCSfn3YJ",

"despoit": 100,

"balance": 20

},

{"address":"AUvupHdCKhaSsfPpzo6f4He4c9wFwhkLmZ",

"despoit": 0,

"balance": 80

}]

}

]

}

}

响应说明：

"channel\_name" channel 标识

" channel\_state " channel 状态，APP侧只展示 OPEN和SETTLING状态的通道

" open\_block " channel 有效块高度

" tx\_info " channel 双方的资产信息

" tx\_info ".address: 通道地址

" tx\_info ".despoit:该地址对应的押金

" tx\_info ".balance: 该地址对应的可用额度（已包含押金）

**registchannle**

新建channel请求 （用于A端发起全新状态通道）

**参数说明**

sender\_addr: sender的地址， string

receiver\_addr: receiver的地址， string

asset\_type:资产类型，string

deposit： lock的代币个数（押金）， string

open\_blockchain: 通道开启时间，在多少个块后自动关闭， string

**调用示例**

请求正文：

{

"jsonrpc": "2.0",

"method": "registchannle",

"params": ["AY8r7uG6rH7MRLhABALZvf8jM4bCSfn3YJ"，"AUvupHdCKhaSsfPpzo6f4He4c9wFwhkLmZ","tnc","100", "50000"],

"id": 1

}

响应正文：

{

"jsonrpc": "2.0",

"id": 1,

"result": {"channel\_name":"3129844fc317d3f7d7cb7f30ac23b180",

"trad\_info":"AUvupHdCKhaSsfPpzo6f4He4c9wFwhkLmZ"

}

响应说明： 返回channal\_name和（基于sender和合约地址的交易数量为deposit的）待签名交易信息

[error\_code,"error message"]

**sendrawtransaction**

请求对已签名的交易信息进行上链广播，请求中包含待广播的签名交易

**参数说明**

sender\_addr: sender的地址， string

Channel\_name: 状态通道标识，string

signaturetrading：签名的交易信息 string

**调用示例**

请求正文：

{

"jsonrpc": "2.0",

"method": "getassetproof",

"params": ["AY8r7uG6rH7MRLhABALZvf8jM4bCSfn3YJ"，"AUvupHdCKhaSsfPpzo6f4He4c9wFwhkLmZ"，"9wFwhkLmZ"],

"id": 1

}

响应正文：

{

"jsonrpc": "2.0",

"id": 1,

"result": [0,"SUCCESS"]

}

**udpatedeposit**

Channel押金更新请求 （押金只可增加不可减少）

**参数说明**

local\_addr: 本地地址， string

channel\_name:状态通道标识，string

asset\_type:押金资产类型，string

value：新增押金的数量， string

**调用示例**

请求正文：

{

"jsonrpc": "2.0",

"method": "registchannle",

"params": ["AY8r7uG6rH7MRLhABALZvf8jM4bCSfn3YJ"，"9wFwhkLmZ","tnc","100"],

"id": 1

}

响应正文：

{

"jsonrpc": "2.0",

"id": 1,

"result": {"channel\_name":"3129844fc317d3f7d7cb7f30ac23b180",

"trad\_info":"AUvupHdCKhaSsfPpzo6f4He4c9wFwhkLmZ"

}

响应说明： 返回channal\_name和基于新增押金数量的待签名交易信息

[error\_code,"error message"]

**sendertoreceiver**

sender to receiver转账

**参数说明**

sender\_addr: sender的地址， string

receiver\_addr: receiver的地址， string

channel\_name:状态通道标识，string

assettype:交易资产类型，string

count：转账的数量， string

**调用示例**

请求正文：

{

"jsonrpc": "2.0",

"method": "sendertoreceiver",

"params": ["AY8r7uG6rH7MRLhABALZvf8jM4bCSfn3YJ"，"AUvupHdCKhaSsfPpzo6f4He4c9wFwhkLmZ"，"9wFwhkLmZ"，"tnc"，“20”],

"id": 1

}

响应正文：

{

"jsonrpc": "2.0",

"id": 1,

"result": "20"

}

响应说明：

"20": 转账金额

"no enough" : 余额不足

**closechannel**

关闭通道

**参数说明**

sender\_addr: sender的地址， string

receiver\_addr: receiver的地址， string

channel\_name：channel\_name

**调用示例**

请求正文：

{

"jsonrpc": "2.0",

"method": "closechannel",

"params": ["AY8r7uG6rH7MRLhABALZvf8jM4bCSfn3YJ"，"AUvupHdCKhaSsfPpzo6f4He4c9wFwhkLmZ"，"3129844fc317d3f7d7cb7f30ac23b180"],

"id": 1

}

响应正文：

{

"jsonrpc": "2.0",

"id": 1,

"result": [0,"SUCCESS"]

}

响应说明：

[0,"SUCCESS"] 通道关闭

[error\_code,"error message"]

**场景交互图：**

