

Technical Report

General Description:

this document describes the detail requirement, rough design and detail design of neo test tools.

Related Document:

1. Test tools source code:

Unless otherwise stated, the latest issue of each of the above documents shall be the one used.

| | | |
|----------|---------------------------------|----------|
| 1 | CONTENTS | 4 |
| 1.1 | Revision History | 4 |
| 1.2 | Glossary and Definitions..... | 5 |
| 2 | DETAIL REQURIEMENT | 6 |
| 2.1 | Cli..... | 6 |
| 2.2 | Rpc..... | 6 |
| 3 | ROUGH DESIGN | 7 |
| 3.1 | Architecture Design..... | 7 |
| 3.2 | Sequence Definition..... | 8 |
| 3.2.1 | Test_Cli | 8 |
| 3.2.2 | Test_Rpc | 9 |
| 4 | DETAIL DESIGN | 9 |
| 4.1 | General | 9 |
| 4.2 | Common modules | 9 |
| 4.3 | API | 10 |
| 4.3.1 | Internal structure | 10 |
| 4.3.2 | Internal data | 10 |
| 4.3.3 | APIs | 11 |
| 4.3.3.1 | API manager | 11 |
| 4.3.3.2 | Cli | 13 |
| 4.3.3.3 | clirpc | 35 |
| 4.3.3.4 | node..... | 56 |
| 4.3.3.5 | RPC | 64 |
| 4.4 | Monitor | 85 |
| 4.4.1 | Internal structure | 86 |
| 4.4.2 | Internal data | 86 |
| 4.4.3 | APIs | 86 |
| 4.4.3.1 | reset..... | 86 |
| 4.4.3.2 | need_retry | 86 |
| 4.4.3.3 | recover_env..... | 87 |
| 4.4.3.4 | retry | 87 |
| 4.4.3.5 | analysis_case | 88 |
| 4.4.3.6 | run_case..... | 89 |
| 4.4.3.7 | set_retry_block | 89 |
| 4.4.3.8 | set_block | 90 |
| 4.4.3.9 | exec | 90 |
| 4.5 | Neo..... | 91 |

| | | |
|----------|---------------------------------|------------|
| 4.5.1 | Internal structure | 91 |
| 4.5.2 | Internal data | 91 |
| 4.5.3 | APIs | 91 |
| 4.5.3.1 | wallet | 91 |
| 4.6 | Resource..... | 97 |
| 4.6.1 | Internal structure | 97 |
| 4.6.2 | Internal data | 97 |
| 4.6.3 | APIs | 98 |
| 4.7 | Test..... | 98 |
| 4.7.1 | Internal structure | 98 |
| 4.7.2 | Internal data | 98 |
| 4.7.3 | APIs | 98 |
| 4.7.3.1 | Test_cli..... | 98 |
| 4.7.3.2 | Test_rpc..... | 98 |
| 4.8 | Utils | 98 |
| 4.8.1 | Internal structure | 98 |
| 4.8.2 | Internal data | 99 |
| 4.8.3 | APIs | 99 |
| 4.8.3.1 | Common..... | 99 |
| 4.8.3.2 | Connect..... | 102 |
| 4.8.3.3 | Error..... | 109 |
| 4.8.3.4 | hexstring..... | 109 |
| 4.8.3.5 | Logger | 112 |
| 4.8.3.6 | parametrizedtestcase..... | 117 |
| 4.8.3.7 | Taskdata..... | 119 |
| 4.8.3.8 | Taskrunner | 126 |
| 4.8.3.9 | Taskthread | 127 |
| 5 | APPENDIX—QUESTIONS | 128 |

FIGURES

| | |
|---------------------------------------|---|
| Figure 1 Test Tool Architecture | 7 |
| Figure 2 Test_cli | 8 |
| Figure 3 Test_rpc..... | 9 |

CONTENTS

1.1 Revision History

| Revision History | | | | | |
|------------------|---------------------|---------|-------------|----------|----------|
| Date | PIC | Version | Description | Reviewer | Approver |
| 2018-12-28 | Zhong Chongguang | 0.0.01 | Initial | | |

1.2 Glossary and Definitions

| Term | Description |
|------|-----------------------|
| DDT | Data-Driven Tests |
| RPC | Remote Procedure Call |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |

2 DETAIL REQUIREMENT

Neo test tools is used for check neo network provide the api and each function. Neo test tools can through CLI, RPC to communication with the neo network. Also can communication with the service tool. Current neo test tools developed with python. Data driven test. Request is using JSON.

Neo test tools can not only check the normal data and normal sequence for neo but also can check the abnormal data and abnormal sequence for neo.

Neo test tools is automatic test framework. it can automatic execute the test suit. And automatic check the result.

Software requirement:

OS: ubuntu 16.04.2

Python: python3

WireShark 2.2.6

Zabbix 3.4

Hardware(server) requirement:

CPU: E5-2673 V3 @ 2.40GHz

Memory: 8G

HD: 64G

2.1 CLI

current neo test tools support CLI related test , it need cover all follow requirements:

1. CLI all API—neo test tools should support all CLI apis process check and result check .

2.2 RPC

current neo test tools support RPC related test , it need cover all follow requirements:

1. RPC all API—neo test tools should support all RPC apis process check and result check .

3 ROUGH DESIGN

3.1 Architecture Design

See following picture about the test tool architecture:

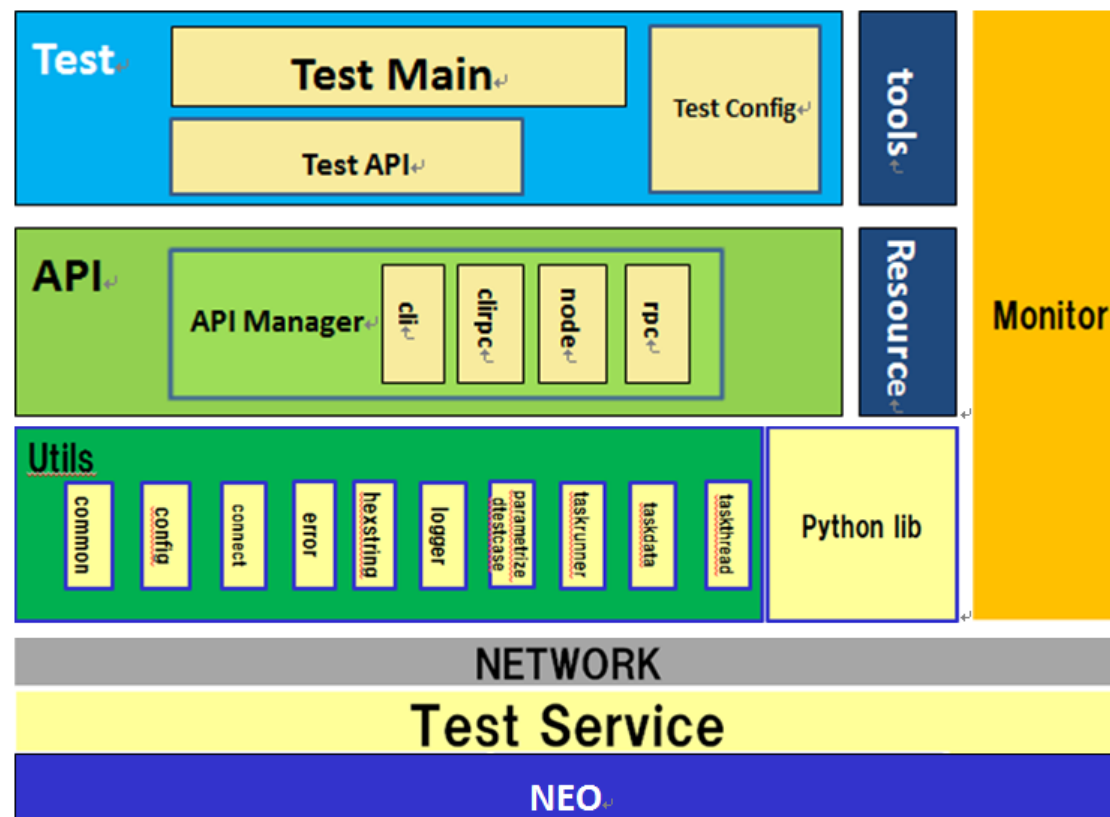


Figure 1 Test Tool Architecture

The Test, API, Utils, Tools, Resource, Monitor blocks are tool modules.

For the test tools we have following modules:

- Test :
 - a) define the special API for each test in test API file;
 - b) define the test case in test main file;
 - c) define the test special config file in test config file
 - d) execut the test case
- API:
 - a) provide the common function which is used in Test main case and Test API file;
 - b) sendback the result & response to Test module
 - c) communication with Utils module & monitor module
- Utils:
 - a) provide the basic function for test tools using;
 - b) communication with the service tool and neo network
 - c) Sendback the service tool & neo network result and response to API module

- Tools:
 - a) provide the basic tools for test tools using;
- Resource:
 - a) provide the data for test tools using
- Monitor:
 - a) Monite the test process during test case execute;
 - b) Restart and recover the test environment
 - c) Collect the the test data during test case execute.
 - d) Analysis the test result

3.2 Sequence Definition

3.2.1 Test_cli

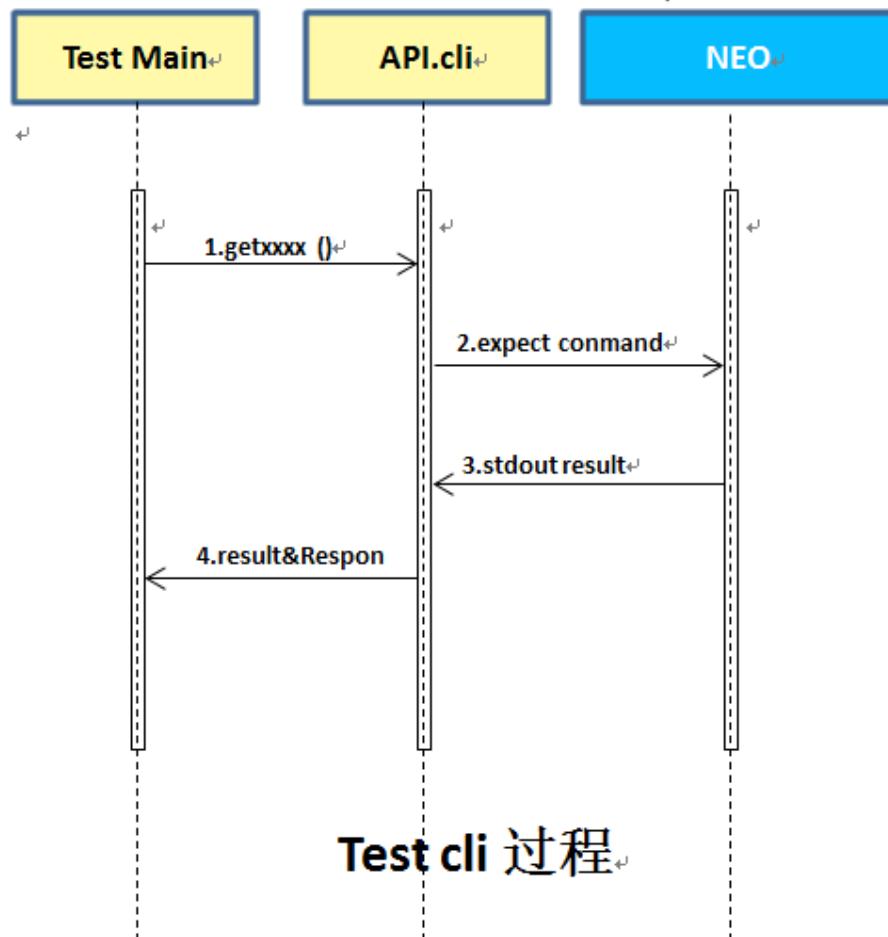


Figure 2 Test_cli

3.2.2 Test_rpc

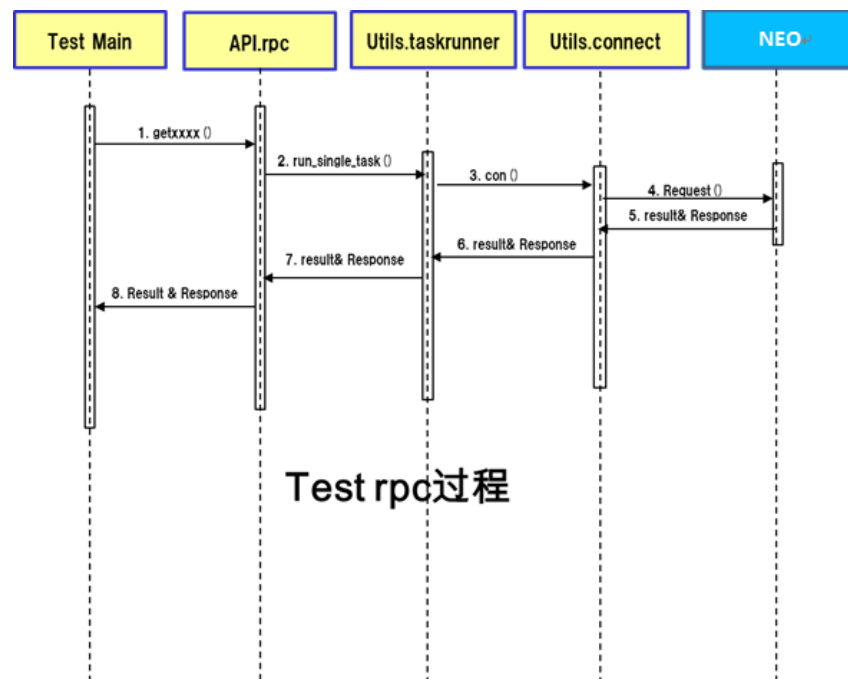


Figure 3 Test_rpc

4 DETAIL DESIGN

4.1 General

Current test tools include follow modules:

- API
- Monitor
- Resource
- Test
- neo
- Utils

4.2 Common modules

Current test tools import a lot of 3rd party modules and python modules. Follow will describe import each module's purpose.

Unittest: this is python module. we can use it to create the test case, test suit in python. Also it package init method and some result check method.

DDT: Data-Driven Tests. Because of python unittest not include the data driven function. So if you want to use data driven this library is needed.

URLlib: this is the python module for http request.

Json: JavaScript Object Notation this is python module. we can use it encoding and decoding JSON objects

OS: using OS module we can use system define the command

Sys: The sys is python module, it contains functions related to the Python interpreter and its environment

Getopt: this is python module ,we can use it to process command line parameters

Subprocess: this is python module. we can use it to fork a sub process.

Re: this is the python module. It will support of regular expressions

4.3 API

For API modules, test tools define follow the data and APIs.

4.3.1 Internal structure

| -api

```
| |  
| | -apimanager.py  
| |  
| | -cli.py  
| |  
| | -clirpc.py  
| | -node.py  
| | -rpc.py
```

4.3.2 Internal data

API module using the data like follow:

Example:

```
[ LOG ] [ REQUEST ] {  
  "NODE_INDEX": 0,  
  "REQUEST": {  
    "id": 1,  
    "jsonrpc": "2.0",  
    "method": "getblockcount",  
    "params": []  
  },  
  "RESPONSE": null,  
  "TYPE": "RPC"  
}  
  "id": 1,  
  "result": 101,
```

```
"jsonrpc": "2.0"
}
```

4.3.3 APIs

API module include follow APIs:

4.3.3.1 API manager

4.3.3.1.1 node

[功能]

调用 node api

[格式]

node(self, node=0)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------|----|--------|
| string | Node | 节点 | IN |

[返回值]

| 名称 | 说明 |
|---------|--------------|
| nodeapi | 是 nodeapi 对象 |
| | |

[Error Code]

| Error Code | Content | Note |
|------------|---------|------|
|------------|---------|------|

4.3.3.1.2 rpc

[功能]

调用 rpc api

[格式]

rpc(self, node=0)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------|----|--------|
| string | Node | 节点 | IN |

[返回值]

| 名称 | 说明 |
|--------|-------------|
| rpcapi | 是 rpcapi 对象 |
| | |

[Error Code]

none

4.3.3.1.3 clirpc

[功能]

调用 clirpc api

[格式]

clirpc (self, node=0)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------|----|--------|
| string | Node | 节点 | IN |

[返回值]

| 名称 | 说明 |
|-----------|-------------|
| clirpcapi | 是 clirpc 对象 |

[Error Code]

none

4.3.3.1.4 cli

[功能]

调用 cli api

[格式]

```
cli(self, node=0)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------|----|--------|
| string | Node | 节点 | IN |

[返回值]

| 名称 | 说明 |
|--------|-------------|
| cliapi | 是 clirpc 对象 |

[Error Code]

none

4.3.3.2 cli

4.3.3.2.1 init

[功能]

初始化

[格式]

```
init (self, process)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.3.3.2.2 lines

[功能]

返回读取数据

[格式]

lines(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-----------|------|
| readlines | 读取数据 |

[Error Code]

none

4.3.3.2.3 isfinish

[功能]

返回完成状态

[格式]

4.3.3.2.4 isfinish(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-------------|------|
| statefinish | 完成表述 |

[Error Code]

none

4.3.3.2.5 isblock

[功能]

判断是否阻塞

[格式]

isblock(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-------|----|
| False | 错误 |
| True | 正确 |

[Error Code]

none

4.3.3.2.6 clear

[功能]

清除屏幕

[格式]

clear(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.3.3.2.7 run

[功能]

运行

[格式]

`run(self)`

[参数]

| 类型 | 名称 | 说明 |
|----|----|----|
|----|----|----|

[返回值]

none

[Error Code]

none

4.3.3.2.8 init

[功能]

初始化

[格式]

`init(self, scriptname, neopath)`

[参数]

| 类型 | 名称 | 说明 |
|--------|-------------------|--------|
| string | <i>scriptname</i> | 脚本名 |
| string | <i>neopath</i> | neo 路径 |

[返回值]

| 名称 | 说明 |
|------------|------|
| scriptpath | 脚本路径 |

[Error Code]

none

4.3.3.2.9 readmsg

[功能]

读 msg 文件

[格式]

readmsg (self)

[参数]

| 类型 | 名称 | 说明 |
|----|----|----|
|----|----|----|

[返回值]

none

[Error Code]

none

4.3.3.2.10 clearmsg

[功能]

清除 msg

[格式]

clearmsg self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.3.3.2.11 begincmd

[功能]

开始 cmd

[格式]

```
begincmd (self, cmd)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-----|----|--------|
| String | cmd | 命令 | IN |

[返回值]

none

[Error Code]

none

4.3.3.2.12 endcmd

[功能]

结束 cmd

[格式]

```
endcmd (self, cmd)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-----|----|--------|
| string | cmd | 命令 | IN |

[返回值]

none

[Error Code]

none

4.3.3.2.13 writeline

[功能]

字符串

[格式]

writeline(self, str)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.3.3.2.14 writesend

[功能]

发送

[格式]

writesend(self, str)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.3.3.2.15 writeexcept

[功能]

[格式]

writeexcept (self, str)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.3.3.2.16 terminate

[功能]

终止

[格式]

terminate (self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.3.3.2.17 Waitnext

[功能]

等待

[格式]

waitnext(self, timeout=5, times=1):

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|---------|----|--------|
| String | Timeout | 超时 | IN |
| int | Times | 时间 | IN |

[返回值]

none

[Error Code]

none

4.3.3.2.18 Waitgenblock

[功能]

等待 genblock

[格式]

waitgenblock(self):

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.3.3.2.19 Waitsync

[功能]

等待 sync

[格式]

`waitsync(self, timeout=30):`

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|----------------------|----|--------|
| String | <code>timeout</code> | 超时 | IN |

[返回值]

| 名称 | 说明 |
|-------|----|
| False | 错误 |
| True | 正确 |

[Error Code]

none

4.3.3.2.20 exec

[功能]

执行程序

[格式]

`exec (self, exitatlast=True)`

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|------|-------------------------|------|--------|
| Bool | <code>exitatlast</code> | 最终退出 | IN |

[返回值]

| 名称 | 说明 |
|----|----|
|----|----|

| 名称 | 说明 |
|-------|----|
| False | 错误 |
| True | 正确 |

[Error Code]

none

4.3.3.2.21 version

[功能]

版本

[格式]

version(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

none

[Error Code]

none

4.3.3.2.22 help

[功能]

帮助菜单

[格式]

help(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

none

[Error Code]

none

4.3.3.2.23 clear

[功能]

清除屏幕

[格式]

clear(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.3.3.2.24 exit

[功能]

退出程序

[格式]

exit(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.3.3.2.25 create_wallet

[功能]

创建钱包文件

[格式]

create_wallet(self, filepath, password, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>filepath</i> | 文件路径 | IN |
| int | <i>passwprd</i> | 密码 | IN |
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

none

[Error Code]

none

4.3.3.2.26 open_wallet

[功能]

打开钱包文件

[格式]

open_wallet(self, filepath, password, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>filepath</i> | 文件路径 | IN |
| int | <i>passwprd</i> | 密码 | IN |
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

none

[Error Code]

none

4. 3. 3. 2. 27 upgrade_wallet

[功能]

升级旧版钱包文件

[格式]

upgrade_wallet(self, filepath, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>filepath</i> | 文件路径 | IN |
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

none

[Error Code]

none

4. 3. 3. 2. 28 rebuild_index

[功能]

重建钱包索引

[格式]

rebuild_index(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

none

[Error Code]

none

4. 3. 3. 2. 29 list address

[功能]

列出钱包中的所有账户

[格式]

list_address(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

none

[Error Code]

none

4. 3. 3. 2. 30 list asset

[功能]

列出钱包中的所有资产

[格式]

list_asset(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

none

[Error Code]

none

4. 3. 3. 2. 31 list key

[功能]

列出钱包中的所有公钥

[格式]

list_key(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

none

[Error Code]

none

4.3.3.2.32 show_utxo

[功能]

列出钱包中指定资产的 UTXO

[格式]

show_utxo(self, id_alias=None, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| int | <i>id_alias</i> | 设置指令别名 | IN |

[返回值]

none

[Error Code]

none

4.3.3.2.33 show_gas

[功能]

列出钱包中的所有可提取及不可提取的 GAS

[格式]

show_gas(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

none

[Error Code]

none

4.3.3.2.34 **claim gas**

[功能]

提取钱包中的所有可提取的 GAS

[格式]

claim_gas(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

none

[Error Code]

none

4.3.3.2.35 **create_address**

[功能]

创建地址 / 批量创建地址

[格式]

create_address(self, n=None, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| int | <i>n</i> | | IN |

[返回值]

none

[Error Code]

none

4. 3. 3. 2. 36 import_key

[功能]

导入私钥 / 批量导入私钥

[格式]

import_key(self, wif_path, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| string | <i>wif_path</i> | wif 路径 | IN |

[返回值]

none

[Error Code]

none

4. 3. 3. 2. 37 export_key

[功能]

导出私钥

[格式]

export_key(self, password, address=None, path=None, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----------------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| string | <i>path</i> | 路径 | IN |
| int | <i>password</i> | 密码 | IN |
| <i>address</i> | <i>address</i> | 地址 | |

[返回值]

none

[Error Code]

none

4. 3. 3. 2. 38 send

[功能]

向指定地址转账 参数分别为：资产 ID，对方地址，转账金额，手续费

[格式]

send(self, password, id_alias, address, value, fee=0, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----------------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| int | <i>fee</i> | 费用 | IN |
| int | <i>password</i> | 密码 | IN |
| <i>address</i> | <i>address</i> | 地址 | IN |
| <i>int</i> | <i>Id_alias</i> | 设置指令别名 | IN |
| <i>string</i> | <i>value</i> | 属性 | IN |

[返回值]

[Error Code]

none

4. 3. 3. 2. 39 import_multisigaddress

[功能]

创建多方签名合约

[格式]

import_multisigaddress(self, m, pubkeys, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| int | <i>m</i> | | IN |
| int | <i>pubkeys</i> | 密钥 | IN |

[返回值]

none

[Error Code]

none

4. 3. 3. 2. 40 sign

[功能]

签名

[格式]

sign(self, jsonobj, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| string | <i>jsonobj</i> | 记录交易内容的 json 字符串 | IN |

[返回值]

none

[Error Code]

none

4. 3. 3. 2. 41 relay

[功能]

广播

[格式]

relay(self, jsonobj, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| string | <i>jsonobj</i> | 记录交易内容的 json 字符串 | IN |

[返回值]

none

[Error Code]

none

4. 3. 3. 2. 42 show state

[功能]

显示当前区块链同步状态

[格式]

show_state(self, times=1, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| int | <i>times</i> | 时间(s) | IN |

[返回值]

none

[Error Code]

none

4.3.3.2.43 show node

[功能]

显示当前已连接的节点地址和端口

[格式]

show_node(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

none

[Error Code]

none

4.3.3.2.44 show pool

[功能]

显示内存池中的交易（这些交易处于零确认的状态）

[格式]

show_pool(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

none

[Error Code]

none

4.3.3.2.45 export_all_blocks

[功能]

导出全部区块数据，导出的结果可以用作离线同步

[格式]

export_all_blocks(self, path=None, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| string | <i>path</i> | 路径 | IN |

[返回值]

none

[Error Code]

none

4.3.3.2.46 export blocks

[功能]

从指定区块高度导出指定数量的区块数据，导出的结果可以用作离线同步

[格式]

export_all_blocks(self, path=None, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| string | <i>count</i> | 计算 | IN |

| 类型 | 名称 | 说明 | IN/OUT |
|--------|--------------|----|--------|
| string | <i>start</i> | 启动 | IN |

[返回值]

none

[Error Code]

none

4.3.3.2.47 start_consensus

[功能]

启动共识

[格式]

start_consensus(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

none

[Error Code]

none

4.3.3.3 clirpc

4.3.3.3.1 setnode

[功能]

设置节点

[格式]

setnode(self, node)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------|----|--------|
| string | node | 节点 | IN |

[返回值]

none

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.2 simplerrun

[功能]

简单运行

[格式]

simplerrun(self, rpcmethod, params, jsonrpc='2.0', id=1)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------------------|---------|--------|
| String | <i>rpcmethod</i> | rpc 方法 | IN |
| int | <i>params</i> | 参数 | IN |
| String | <i>jsonrpc</i> | 跨语言远程调用 | IN |
| int | <i>id</i> | id | IN |

[返回值]

| 名称 | 说明 |
|--------------------|------|
| response["result"] | 响应结果 |

[Error Code]

| Error Code | Content | Note |
|------------|-------------------|------|
| 54001 | RPC CONNECT ERROR | |

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.3 readmsg

[功能]

读 msg 文件

[格式]

readmsg (self)

[参数]

| 类型 | 名称 | 说明 |
|----|----|----|
|----|----|----|

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.4 clearmsg

[功能]

清除 msg

[格式]

clearmsg (self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.5 terminate

[功能]

结束

[格式]

terminate(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.6 exec

[功能]

实行

[格式]

exec(self, exitatlast=True)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|------|-------------------|------|--------|
| Bool | <i>exitatlast</i> | 最终退出 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.7 version

[功能]

版本

[格式]

version(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.8 help

[功能]

帮助菜单

[格式]

help(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.9 wait

[功能]

等待

[格式]

wait(self, times=1)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|--------------|-------|--------|
| int | <i>times</i> | 时间（s） | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.10 clear

[功能]

清除屏幕

[格式]

clear(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|-------------------|------|
| 54001 | RPC CONNECT ERROR | |

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.11 exit

[功能]

退出程序

[格式]

exit(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.12 create_wallet

[功能]

创建钱包文件

[格式]

create_wallet(self, filepath, password, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>filepath</i> | 文件路径 | IN |
| int | <i>passwprd</i> | 密码 | IN |
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4. 3. 3. 3. 13 open_wallet

[功能]

打开钱包文件

[格式]

open_wallet(self, filepath, password, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>filepath</i> | 文件路径 | IN |
| int | <i>passwprd</i> | 密码 | IN |
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4. 3. 3. 3. 14 upgrade_wallet

[功能]

升级旧版钱包文件

[格式]

upgrade_wallet(self, filepath, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>filepath</i> | 文件路径 | IN |
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4. 3. 3. 3. 15 rebuild index

[功能]

重建钱包索引

[格式]

rebuild_index(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4. 3. 3. 3. 16 list address

[功能]

列出钱包中的所有账户

[格式]

list_address(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4. 3. 3. 3. 17 list asset

[功能]

列出钱包中的所有资产

[格式]

list_asset(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|---------|------|
|------------|---------|------|

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4. 3. 3. 3. 18 list key

[功能]

列出钱包中的所有公钥

[格式]

list_key(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4. 3. 3. 3. 19 show_utxo

[功能]

列出钱包中指定资产的 UTXO

[格式]

show_utxo(self, id_alias=None, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| int | <i>id_alias</i> | 设置指令别名 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.20 show_gas

[功能]

列出钱包中的所有可提取及不可提取的 GAS

[格式]

show_gas(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.21 claim_gas

[功能]

提取钱包中的所有可提取的 GAS

[格式]

claim_gas(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4. 3. 3. 3. 22 create_address

[功能]

创建地址 / 批量创建地址

[格式]

create_address(self, n=None, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| int | <i>n</i> | | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4. 3. 3. 3. 23 import_key

[功能]

导入私钥 / 批量导入私钥

[格式]

```
import_key(self, wif_path, exceptfunc=None)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| string | <i>wif_path</i> | wif 路径 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4. 3. 3. 3. 24 export_key

[功能]

导出私钥

[格式]

```
export_key(self, password, address=None, path=None, exceptfunc=None)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----------------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| string | <i>path</i> | 路径 | IN |
| int | <i>password</i> | 密码 | IN |
| <i>address</i> | <i>address</i> | 地址 | |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|---------|------|
|------------|---------|------|

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.25 send

[功能]

向指定地址转账 参数分别为：资产 ID，对方地址，转账金额，手续费

[格式]

send(self, password, id_alias, address, value, fee=0, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----------------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| int | <i>fee</i> | 费用 | IN |
| int | <i>password</i> | 密码 | IN |
| <i>address</i> | <i>address</i> | 地址 | IN |
| <i>int</i> | <i>id_alias</i> | 设置指令别名 | IN |
| <i>string</i> | <i>value</i> | 属性 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.26 import_multisigaddress

[功能]

创建多方签名合约

[格式]

import_multisigaddress(self, m, pubkeys, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| int | <i>m</i> | | IN |
| int | <i>pubkeys</i> | 密钥 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4. 3. 3. 3. 27 sign

[功能]

签名

[格式]

sign(self, jsonobj, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| string | <i>jsonobj</i> | 记录交易内容的 json 字符串 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4. 3. 3. 3. 28 relay

[功能]

广播

[格式]

relay(self, jsonobj, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| string | <i>jsonobj</i> | 记录交易内容的 json 字符串 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4. 3. 3. 3. 29 show state

[功能]

显示当前区块链同步状态

[格式]

show_state(self, times=1, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| int | <i>times</i> | 时间(s) | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.30 show node

[功能]

显示当前已连接的节点地址和端口

[格式]

show_node(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.31 show pool

[功能]

显示内存池中的交易（这些交易处于零确认的状态）

[格式]

show_pool(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.32 export_all_blocks

[功能]

导出全部区块数据，导出的结果可以用作离线同步

[格式]

export_all_blocks(self, path=None, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| string | <i>path</i> | 路径 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.33 export blocks

[功能]

从指定区块高度导出指定数量的区块数据，导出的结果可以用作离线同步

[格式]

export_all_blocks(self, path=None, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |
| string | <i>count</i> | 计算 | IN |
| string | <i>start</i> | 启动 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.3.34 start_consensus

[功能]

启动共识

[格式]

start_consensus(self, exceptfunc=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------------|--------|
| string | <i>exceptfunc</i> | 除 Func 函数外 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.4 node

4.3.3.4.1 setnode

[功能]

设置路径回访模式

[格式]

```
setnode(self, node)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|------|----|--------|
| Int | Node | 节点 | IN |
| | | | |

[返回值]

none

[Error Code]

none

4.3.3.4.2 simplerun

[功能]

简单运行

[格式]

```
simplerun(self, rpcmethod, params, jsonrpc='2.0', id=1)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------------------|---------|--------|
| String | <i>rpcmethod</i> | Rpc 方法 | IN |
| Int | <i>params</i> | 参数 | IN |
| String | <i>jsonrpc</i> | 跨语言远程调用 | IN |
| int | <i>id</i> | id | IN |

[返回值]

| 名称 | 说明 |
|----------|------|
| response | 响应结果 |

[Error Code]

none

4.3.3.4.3 sftp_transfer

[功能]

获取当前 block 节点数

[格式]

sftp_transfer(self, _from, _to, remotenode, op="get")

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|------|--------|
| Int | <i>remotenode</i> | 远程节点 | IN |
| String | <i>op</i> | 操作 | IN |

[返回值]

none

[Error Code]

none

4.3.3.4.4 clear

[功能]

清除

[格式]

clear(self, blocks=True, acc=True)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|---------------|----|--------|
| String | <i>blocks</i> | 区块 | IN |
| Bool | <i>acc</i> | 常量 | IN |

[返回值]

| 名称 | 说明 |
|----------|------|
| exec_cmd | 执行命令 |

[Error Code]

node

4.3.3.4.5 send_file

[功能]

发送文件

[格式]

send_file(self, _from, to)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.3.3.4.6 wait_gen_block

[功能]

获取当前 block 节点数

[格式]

wait_gen_block(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
| | | | |
| | | | |

[返回值]

| 名称 | 说明 |
|------|----|
| True | 正确 |

| 名称 | 说明 |
|-------|----|
| False | 错误 |

[Error Code]

node

4.3.3.4.7 wait_tx_result

[功能]

等待结果

[格式]

```
wait_tx_result(self, txhash)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|--------|--------|--------|
| int | txhash | txhash | IN |

[返回值]

| 名称 | 说明 |
|------|----|
| none | 空 |

[Error Code]

none

4.3.3.4.8 get_current_node

[功能]

获取当前节点

[格式]

```
get_current_node(self)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|------|----|
| none | 空 |

[Error Code]

none

4.3.3.4.9 check_node_block

[功能]

检查节点 block

[格式]

```
check_node_block(self, node_list)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|------------------|------|--------|
| Int | <i>node_list</i> | 节点列表 | IN |
| | | | |

[返回值]

none

[Error Code]

none

4.3.3.4.10 start_nodes

[功能]

启动多个节点

[格式]

```
start_nodes(self, indexs, start_params = Config.DEFAULT_NODE_ARGS, clear_chain = False,  
            clear_log = False, program = "neo", config = "config.json")
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|---------------------|------------|--------|
| Int | <i>indexs</i> | 索引 | IN |
| String | <i>start_params</i> | 启动参数 | IN |
| Bool | <i>clear_chain</i> | 是否清除 chain | IN |
| Bool | <i>clear_log</i> | 是否清除 log | IN |
| String | <i>program</i> | 启动程序名 | IN |
| String | <i>config</i> | 配置文件 | IN |

[返回值]

none

[Error Code]

none

4.3.3.4.11 start_node

[功能]

启动单个节点

[格式]

```
start_node(self, index, start_params = Config.DEFAULT_NODE_ARGS, clear_chain = False,
            clear_log = False, program = "neo", config = "config.json")
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|---------------------|------------|--------|
| Int | <i>index</i> | 索引 | IN |
| String | <i>start_params</i> | 启动参数 | IN |
| Bool | <i>clear_chain</i> | 是否清除 chain | IN |
| Bool | <i>clear_log</i> | 是否清除 log | IN |
| String | <i>program</i> | 启动程序名 | IN |
| String | <i>config</i> | 配置文件 | IN |

[返回值]

| 名称 | 说明 |
|----------|----|
| response | 响应 |

[Error Code]

none

4.3.3.4.12 stop_all_nodes

[功能]

停止所有节点

[格式]

```
stop_all_nodes(self)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.3.3.4.13 stop_nodes

[功能]

停止多个节点

[格式]

```
stop_nodes(self, indexs):
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|---------------|----|--------|
| Int | <i>indexs</i> | 索引 | IN |

[返回值]

none

[Error Code]

none

4.3.3.4.14 stop_node

[功能]

停止单个节点

[格式]

```
stop_node(self, index):
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|--------------|----|--------|
| Int | <i>index</i> | 索引 | IN |

[返回值]

| 名称 | 说明 |
|----------|----|
| response | 响应 |

[Error Code]

none

4.3.3.4.15 replace_configs

[功能]

替换多个配置文件

[格式]

```
replace_configs(self, indexs, config = None)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|---------------|------|--------|
| Int | <i>indexs</i> | 索引 | IN |
| Int | <i>config</i> | 配置文件 | IN |

[返回值]

none

[Error Code]

none

4.3.3.4.16 replace_config

[功能]

替换单个配置文件

[格式]

```
replace_config(self, index, config = None)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|---------------|------|--------|
| Int | <i>index</i> | 索引 | IN |
| Int | <i>config</i> | 配置文件 | IN |

[返回值]

| 名称 | 说明 |
|----------|----|
| response | 响应 |

[Error Code]

none

4.3.3.4.17 exec_cmd

[功能]

执行命令

[格式]

`exec_cmd(self, cmd, index)`

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|--------------|----|--------|
| String | <i>cmd</i> | 命令 | IN |
| Int | <i>index</i> | 索引 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

none

4.3.3.5 RPC

4.3.3.5.1 setnode

[功能]

设置节点

[格式]

```
setnode(self, node)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------|----|--------|
| string | node | 节点 | IN |

[返回值]

none

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.2 num2str

[功能]

数值数组转换为表示数字的字符数组

[格式]

```
num2str(self, num)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|------------|--------|--------|
| Int | <i>num</i> | rpc 方法 | IN |

[返回值]

| 名称 | 说明 |
|-----|----|
| num | 数字 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.3 simplerrun

[功能]

简单运行

[格式]

```
simplerrun(self, rpcmethod, params, jsonrpc='2.0', id=1)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------------------|---------|--------|
| String | <i>rpcmethod</i> | rpc 方法 | IN |
| int | <i>params</i> | 参数 | IN |
| String | <i>jsonrpc</i> | 跨语言远程调用 | IN |
| int | <i>id</i> | id | IN |

[返回值]

| 名称 | 说明 |
|----------|----|
| response | 响应 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.4 dumpprivkey

[功能]

显示私钥

[格式]

dumpprivkey(self, address)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----------------|----------------|----|--------|
| <i>address</i> | <i>address</i> | 地址 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.5 getaccountstate

[功能]

获取账户状态

[格式]

getaccountstate(self, address)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----------------|----------------|----|--------|
| <i>address</i> | <i>address</i> | 地址 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.6 getassetstate

[功能]

获取资产状态

[格式]

```
getassetstate(self, asset_id)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|------------|--------------|----|--------|
| <i>int</i> | <i>asset</i> | 资产 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.7 getbalance

[功能]

获取余额

[格式]

```
getbalance(self, asset_id)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|------------|--------------|----|--------|
| <i>int</i> | <i>asset</i> | 资产 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.8 getbestblockhash

[功能]

获取最佳 block hash

[格式]

getbestblockhash(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
| | | | |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.9 getblock

[功能]

获取 block

[格式]

```
getblock(self, hash, verbose = None)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|----------------|----------|--------|
| Int | <i>hash</i> | hash | IN |
| Int | <i>verbose</i> | 是否输出详细信息 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.10getblockcount

[功能]

获取 block 数

[格式]

```
getblockcount(self,)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.11 getblockheader

[功能]

获取 block 数据头

[格式]

getblockheader (self, hash, verbose = None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|----------------|----------|--------|
| Int | <i>hash</i> | hash | IN |
| Int | <i>verbose</i> | 是否输出详细信息 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.12 getblockhash

[功能]

获取 block hash 数

[格式]

getblockhash(self, index)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|--------------|----|--------|
| Int | <i>index</i> | 索引 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.13 getblocksysfee

[功能]

获取 block system fee

[格式]

getblocksysfee(self, index)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|--------------|----|--------|
| int | <i>index</i> | 索引 | IN |

[返回值]

| 名称 | 说明 |
|----|----|
|----|----|

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.14 getconnectioncount

[功能]

获取连接数

[格式]

getconnectioncount(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.15 getcontractstate

[功能]

获取合同状态

[格式]

```
getcontractstate(self, script_hash)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|--------------------|--------|--------|
| String | <i>script_hash</i> | Hash 表 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.16 getnewaddress

[功能]

获取新地址

[格式]

```
getnewaddress(self)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.17 getrawmempool

[功能]

获取 raw mem pool

[格式]

getrawmempool(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.18 getrawtransaction

[功能]

获取 raw transaction

[格式]

getrawtransaction(self, txid, verbose = None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|----------------|----------|--------|
| Int | <i>Txid</i> | 交易编号 | IN |
| Int | <i>verbose</i> | 是否输出详细信息 | IN |
| | | | |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.19 getstorage

[功能]

获取存储

[格式]

getstorage(self, script_hash, key)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|--------------------|--------|--------|
| String | <i>script_hash</i> | Hash 表 | IN |
| String | <i>key</i> | 关键字 | IN |
| | | | |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|---------|------|
|------------|---------|------|

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.20 gettxout

[功能]

获取 txout

[格式]

gettxout(self, txid, n=0)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|-------------|------|--------|
| Int | <i>txid</i> | 交易编号 | IN |
| int | <i>n</i> | | IN |
| | | | |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.21 getpeers

[功能]

获取 peers

[格式]

getpeers(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.22 getvalidators

[功能]

获取验证

[格式]

getvalidators(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.23 getversion

[功能]

获取版本

[格式]

getversion(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.24 getwalletheight

[功能]

获取钱包高度

[格式]

getwalletheight(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.25 invoke

[功能]

调用

[格式]

invoke(self, scripthash, params)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|--------|--------|
| String | <i>scripthash</i> | Hash 表 | IN |
| Int | <i>params</i> | 参数 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.26 invokefunction

[功能]

调用函数

[格式]

invokefunction(self, scripthash, operation, params)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------|---------|--------|
| String | <i>scripthash</i> | 脚本 hash | IN |
| String | <i>operation</i> | 构造函数 | IN |
| Int | <i>params</i> | 参数 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.27 invokescript

[功能]

调用脚本

[格式]

`invokescript(self, script)`

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|---------------|----|--------|
| String | <i>script</i> | 脚本 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|---------|------|
|------------|---------|------|

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.28 listaddress

[功能]

地址列表

[格式]

listaddress(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.29 sendfrom

[功能]

发件地址

[格式]

sendfrom(self, asset_id, _from, to, value, fee=None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

| 类型 | 名称 | 说明 | IN/OUT |
|---------------|--------------|----|--------|
| int | <i>asset</i> | 资产 | IN |
| <i>string</i> | <i>value</i> | 属性 | IN |
| int | <i>fee</i> | 费用 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.30 sendrawtransaction

[功能]

发送 raw transaction

[格式]

sendrawtransaction(self, hex)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------|---------|--------|
| String | <i>_hex</i> | 交易 hash | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.31 sendtoaddress

[功能]

发送的地址

[格式]

```
sendtoaddress(self, asset_id, address, value, fee)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|---------------|----------------|----|--------|
| int | <i>asset</i> | 资产 | IN |
| <i>string</i> | <i>value</i> | 属性 | IN |
| int | <i>fee</i> | 费用 | IN |
| address | <i>address</i> | 地址 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.32 sendmany

[功能]

多发送

[格式]

```
sendmany(self, params)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|---------------|----|--------|
| Int | <i>params</i> | 参数 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.3.3.5.33 validateaddress

[功能]

确认地址

[格式]

validateaddress(self, address)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|---------|----------------|----|--------|
| address | <i>address</i> | 地址 | IN |

[返回值]

| 名称 | 说明 |
|-----------|----|
| simplerun | 运行 |

[Error Code]

| Error Code | Content | Note |
|------------|--------------------------------|------|
| 54001 | RPC CONNECT ERROR | |
| 54005 | SERVICE TOOL RPC CONNECT ERROR | |

4.4 Monitor

For Monitor, test tools define the follow datas and APIs.

4.4.1 Internal structure

| - Monitor

```
| |
| | monitor.py
```

4.4.2 Internal data

Monitor module test using the data like follow:

Example:

TRY_RECOVER_TIMES = 2

CHECK_LOOP = 50

FAILED_RADIO = 40 # 40%

4.4.3 APIs

Monitor module will include follow APIs:

4.4.3.1 reset

[功能]

重启

[格式]

reset(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.4.3.2 need_retry

[功能]

需要重试

[格式]

need_retry(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-------|----|
| False | 错误 |
| True | 正确 |
| False | 错误 |

[Error Code]

none

4.4.3.3 recover_env

[功能]

恢复环境

[格式]

recover_env(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.4.3.4 retry

[功能]

重试

[格式]

retry(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.4.3.5 analysis_case

[功能]

分析 case

[格式]

analysis_case(self, case, logpath)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|----------------|--------|--------|
| String | <i>case</i> | Case 名 | IN |
| String | <i>logpath</i> | Log 路径 | IN |
| | | | |
| | | | |

[返回值]

| 名称 | 说明 |
|-------|----|
| False | 错误 |
| False | 错误 |
| True | 正确 |

[Error Code]

none

4.4.3.6 run_case

[功能]

运行 case

[格式]

```
run_case(self, case)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------|--------|--------|
| String | <i>case</i> | Case 名 | IN |
| | | | |

[返回值]

| 名称 | 说明 |
|----------|----|
| True | 正确 |
| analysis | 分析 |

[Error Code]

none

4.4.3.7 set_retry_block

[功能]

设置重试 block

[格式]

```
set_retry_block(self)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
| | | | |

[返回值]

none

[Error Code]

none

4.4.3.8 set_block

[功能]

设置 block

[格式]

```
set_block(self, case)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------|----|--------|
| string | <i>case</i> | 实例 | IN |

[返回值]

none

[Error Code]

none

4.4.3.9 exec

[功能]

执行

[格式]

```
exec(self, runner, testcases, monitor = True)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------------------|-------------|--------|
| Point | <i>runner</i> | 执行进程 | IN |
| String | <i>testcases</i> | Test case 名 | IN |
| Bool | <i>monitor</i> | 是否监控 | IN |
| | | | |

[返回值]

none

[Error Code]

none

4.5 Neo

For neo module, test tools define follow the datas and APIs.

4.5.1 Internal structure

```
| - neo
|   |   |
|   |   | - wallet.py
```

4.5.2 Internal data

neo module no special data internal use

4.5.3 APIs

neo Module will include follow APIs:

4.5.3.1 wallet

4.5.3.1.1 path

[功能]

路径

[格式]

path(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|------|----|
| Path | 路径 |

[Error Code]

none

4.5.3.1.2 name

[功能]

名字

[格式]

name (self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|---------------|----------|
| ijson["name"] | Ijson 名称 |

[Error Code]

node

4.5.3.1.3 version

[功能]

版本

[格式]

version (self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|------------------|----------|
| ijson["version"] | Ijson 版本 |

[Error Code]

none

4.5.3.1.4 script_n

[功能]

加密

[格式]

scrypt_n(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|----------------------|----------|
| ijson["scrypt"]["n"] | Ijson 加密 |

[Error Code]

node

4.5.3.1.5 scrypt_r

[功能]

加密

[格式]

scrypt_r(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|----------------------|----------|
| ijson["scrypt"]["r"] | Ijson 加密 |

[Error Code]

none

4.5.3.1.6 scrypt_p

[功能]

加密

[格式]

script_p(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|----------------------|----------|
| ijson["script"]["p"] | Ijson 加密 |

[Error Code]

none

4.5.3.1.7 account

[功能]

账目

[格式]

account(self, index=0)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|---------|----|
| account | 账户 |

[Error Code]

node

4.5.3.1.8 address

[功能]

地址

[格式]

address(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|------------------|----------|
| ijson["address"] | Ijson 地址 |

[Error Code]

none

4.5.3.1.9 label

[功能]

标签

[格式]

label(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|----------------|----------|
| ijson["lable"] | Ijson 标签 |

[Error Code]

none

4.5.3.1.10 isDefault

[功能]

默认显示

[格式]

isDefault(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|--------------------|------------|
| ijson["isdefault"] | Ijson 默认显示 |

[Error Code]

none

4.5.3.1.11 lock

[功能]

锁

[格式]

lock(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|---------------|---------|
| ijson["lock"] | Ijson 锁 |

[Error Code]

none

4.5.3.1.12 key

[功能]

密钥

[格式]

key(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|--------------|----------|
| ijson["key"] | Ijson 密钥 |

[Error Code]

none

4.5.3.1.13 contract

[功能]

合约

[格式]

contract(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|-------------------|----------|
| ijson["contract"] | Ijson 合约 |

[Error Code]

none

4.6 Resource

For resource module, test tools define follow the datas and APIs.

4.6.1 Internal structure

| - resource

| |

| | -wallet

| | -1~5.json

4.6.2 Internal data

Authenticate function test using the data like follow:

Example:

resource module no special data internal use

4.6.3 APIs

resource module no API for other modules using. Rpcserver API please refer to service tools TN

4.7 Test

For Test Module, test tools define follow the datas and APIs.

4.7.1 Internal structure

```
| - test
|   |
|   | test_cli
|
.....
|   |   |-test_config.py
|   |   |-test_main.py
```

*other test projects structure almost same as test cli.

4.7.2 Internal data

Test module using the data like follow:

Example:

```
address="AXbevnGZBypZrcQwQfdYZxXj5zcFTtiMEA"
cost1="tasks/cost_1.json"
filterfile="tasks/004.json"
```

4.7.3 APIs

Test Module will include follow APIs:

4.7.3.1 Test_cli

Test_cli not define special APIs.

4.7.3.2 Test_rpc

Test_rpc not define special APIs.

4.8 utils

For utils, it will provide basic function to support test running.

4.8.1 Internal structure

```
| -utils
```

```

|   |
|   common.py
|   |
|   config.py
|   |
|   connect.py
|   |
|   error.py
|   |
|   hexstring.py
|   |
|   logger.py
|   |
|   parametrizedtestcase.py
|   |
|   taskdata.py
|   |
|   taskrunner.py
|   |
|   taskthread.py

```

4.8.2 Internal data

Utils module using the data like follow:

Example:

```

NEO_ERR_CODE = {
    0 : "SUCCESS",
    54001   RPC CONNECT ERROR
    54005   SERVICE TOOL RPC CONNECT ERROR
}

```

4.8.3 APIs

Utils Module will include follow APIs:

4.8.3.1 Common

4.8.3.1.1 Cmp

[功能]

比较

[格式]

cmp(expect_data, cmp_data)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|--------------------|------|--------|
| String | <i>expect_data</i> | 期望数据 | IN |
| String | <i>cmp_data</i> | 比对数据 | IN |
| | | | |

[返回值]

| 名称 | 说明 |
|-------|----|
| False | 错误 |
| False | 错误 |
| False | 错误 |
| True | 正确 |
| False | 错误 |
| False | 错误 |
| False | 错误 |
| True | 正确 |
| False | 错误 |
| True | 正确 |

[Error Code]

none

4.8.3.1.2 Pause

[功能]

暂停

[格式]

pause(msg)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------------|----|--------|
| String | <i>msg</i> | 消息 | IN |
| | | | |

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
| | | | |

[返回值]

| 名称 | 说明 |
|---------|----|
| command | 命令 |

[Error Code]

none

4.8.3.1.3 bl_reserver

[功能]

bl 保留

[格式]

bl_reserver(input)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|--------------|----|--------|
| String | <i>input</i> | 输入 | IN |
| | | | |
| | | | |

[返回值]

| 名称 | 说明 |
|--------|----|
| none | 空 |
| input | 输入 |
| output | 输出 |

[Error Code]

none

4.8.3.1.4 bl_address

[功能]

bl 地址

[格式]

bl_address(input)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|--------------|----|--------|
| String | <i>input</i> | 输入 | IN |

[返回值]

| 名称 | 说明 |
|---|-------------------------|
| Common.bl_reserver(Common.base58_to_address(input)) | 输入 base58_to_address |

[Error Code]

none

4.8.3.2 Connect

4.8.3.2.1 multithread_run

[功能]

多线程执行

[格式]

multithread_run(logger, cfg_request, cfg_response)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|---------------------|--------|--------|
| String | <i>logger</i> | Logger | IN |
| String | <i>cfg_request</i> | 请求 | IN |
| String | <i>cfg_response</i> | 结果 | IN |

[返回值]

| 名称 | 说明 |
|---------------------------|------|
| return (result, response) | 响应结果 |

[Error Code]

none

4.8.3.2.2 con

[功能]

连接

[格式]

con(itype, ip, request)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|----------------|-------|--------|
| String | <i>itype</i> | 连接类型 | IN |
| String | <i>ip</i> | Ip 地址 | IN |
| String | <i>request</i> | 请求 | IN |

[返回值]

| 名称 | 说明 |
|------|----|
| none | 空 |

[Error Code]

none

4.8.3.2.3 con_cli

[功能]

连接 cli

[格式]

con_cli(ip, request)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-----------|-------|--------|
| String | <i>ip</i> | Ip 地址 | IN |

| 类型 | 名称 | 说明 | IN/OUT |
|--------|----------------|----|--------|
| String | <i>request</i> | 请求 | IN |

[返回值]

| 名称 | 说明 |
|---------------|---------|
| response.json | 响应 json |

[Error Code]

none

4.8.3.2.4 con_rpc

[功能]

连接 rpc

[格式]

con_rpc(ip, request)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|----------------|-------|--------|
| String | <i>ip</i> | Ip 地址 | IN |
| String | <i>request</i> | 请求 | IN |

[返回值]

| 名称 | 说明 |
|---------------|---------|
| response.json | 响应 json |

[Error Code]

none

4.8.3.2.5 con_restful

[功能]

连接 restful

[格式]

`con_restful(ip, api_request)`

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|--------------------|-------|--------|
| String | <i>ip</i> | Ip 地址 | IN |
| String | <i>api_request</i> | 请求 | IN |

[返回值]

| 名称 | 说明 |
|------------|---------|
| json.loads | Json 负载 |
| json.loads | Json 负载 |

[Error Code]

none

4.8.3.2.6 con_ws

[功能]

连接 websocket

[格式]

`con_ws(ip, request)`

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|----------------|-------|--------|
| String | <i>ip</i> | Ip 地址 | IN |
| String | <i>request</i> | 请求 | IN |

[返回值]

| 名称 | 说明 |
|------------|---------|
| json.loads | Json 负载 |

[Error Code]

none

4.8.3.2.7 con_test_service

[功能]

连接 test service

[格式]

con_test_service(ip, request)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|----------------|-------|--------|
| String | <i>ip</i> | Ip 地址 | IN |
| String | <i>request</i> | 请求 | IN |

[返回值]

| 名称 | 说明 |
|---------------|---------|
| response.json | 响应 json |

[Error Code]

none

4.8.3.2.8 on_message

[功能]

[格式]

on_message(ws, message)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|----------------|-----------|--------|
| String | <i>ws</i> | Websocket | IN |
| String | <i>message</i> | 消息 | IN |

[返回值]

none

[Error Code]

none

4.8.3.2.9 on_error

[功能]

[格式]

`on_error(ws, error)`

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|--------------|-----------|--------|
| String | <i>ws</i> | Websocket | IN |
| String | <i>Error</i> | 错误 | IN |

[返回值]

none

[Error Code]

none

4.8.3.2.10 on_close

[功能]

[格式]

`on_close(ws)`

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-----------|-----------|--------|
| String | <i>ws</i> | Websocket | IN |

[返回值]

none

[Error Code]

none

4.8.3.2.11 on_open

[功能]

[格式]

on_open(ws)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-----------|-----------|--------|
| String | <i>ws</i> | Websocket | IN |

[返回值]

none

[Error Code]

none

4.8.3.2.12 ws_heartbeat_thread

[功能]

Websocket 心跳检测线程

[格式]

ws_heartbeat_thread(self, heartbeat_gap = 5)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|----------------------|--------|--------|
| Int | <i>heartbeat_gap</i> | 心跳检测间距 | IN |

[返回值]

| 名称 | 说明 |
|-------|----|
| False | 错误 |

[Error Code]

none

4.8.3.2.13 exec

[功能]

执行

[格式]

```
exec(self, heartbeat_gap = 5, message_cb = None)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|-----|----------------------|--------|--------|
| Int | <i>heartbeat_gap</i> | 心跳检测间距 | IN |
| Int | <i>message_cb</i> | 回调函数 | IN |

[返回值]

| 名称 | 说明 |
|-------|----|
| Flase | 错误 |

[Error Code]

none

4.8.3.3 Error

Error no special public APIs

4.8.3.4 hexstring

4.8.3.4.1 FileToHex

[功能]

文件转换 16 进制

[格式]

```
FileToHex(path)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------|----|--------|
| String | <i>path</i> | 路径 | IN |
| | | | |

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
| | | | |

[返回值]

| 名称 | 说明 |
|------------------|------------------------|
| binascii.b2a_hex | 将对象从 str 转换为 hexstring |

[Error Code]

none

4.8.3.4.2 ByteToHex

[功能]

byte 转换 16 进制

[格式]

ByteToHex(bytes_value)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|--------------------|--------|--------|
| String | <i>bytes_value</i> | Byte 值 | IN |
| | | | |
| | | | |

[返回值]

| 名称 | 说明 |
|------------------|------------------------|
| binascii.b2a_hex | 将对象从 str 转换为 hexstring |

[Error Code]

none

4.8.3.4.3 HexToByte

[功能]

16 进制转换 byte

[格式]

HexToByte(hexStr)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|---------------|--------|--------|
| String | <i>hexStr</i> | 16 进制值 | IN |
| | | | |
| | | | |

[返回值]

| 名称 | 说明 |
|-----------|------|
| Bytearray | 字节数组 |
| result | 结果 |

[Error Code]

none

4.8.3.4.4 recoveryAddress

[功能]

恢复地址

[格式]

recoveryAddress(m, hexArr, checkMultiSig="ae")

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|----------------------|---------|--------|
| String | <i>M</i> | | IN |
| Array | <i>hexArr</i> | 16 进制数组 | IN |
| String | <i>checkMultiSig</i> | 检查签名 | IN |

[返回值]

| 名称 | 说明 |
|------------------|----|
| false | 错误 |
| ripemd_160_value | 加密 |

[Error Code]

none

4.8.3.4.5 numSeq

[功能]

[格式]

numSeq (m)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|----------|----|--------|
| String | <i>M</i> | | IN |
| | | | |
| | | | |

[返回值]

| 名称 | 说明 |
|---------|--------|
| 00 | |
| teststr | 测试参数 |
| test222 | 测试 222 |

[Error Code]

none

4.8.3.5 Logger

4.8.3.5.1 setPath

[功能]

设置路径

[格式]

setPath(self, path)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------|----|--------|
| String | <i>path</i> | 路径 | IN |
| | | | |
| | | | |

[返回值]

none

[Error Code]

none

4.8.3.5.2 logPath

[功能]

log 路径

[格式]

logPath(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|---------|--------|
| logpath | Log 路径 |

[Error Code]

none

4.8.3.5.3 open

[功能]

打开 log

[格式]

open(self, filepath, title = None)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-----------------|--------|--------|
| String | <i>filepath</i> | Log 路径 | IN |
| String | <i>title</i> | Title | IN |
| | | | |

[返回值]

none

[Error Code]

none

4.8.3.5.4 print

[功能]

打印 log

[格式]

print(self, str)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------------|------|--------|
| String | <i>Str</i> | 打印内容 | IN |
| | | | |
| | | | |

[返回值]

none

[Error Code]

none

4.8.3.5.5 error

[功能]

Error log

[格式]

error (self, str)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------------|------|--------|
| String | <i>Str</i> | 打印内容 | IN |
| | | | |
| | | | |

[返回值]

none

[Error Code]

none

4.8.3.5.6 info

[功能]

info log

[格式]

info(self, str)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------------|------|--------|
| String | <i>Str</i> | 打印内容 | IN |
| | | | |
| | | | |

[返回值]

none

[Error Code]

none

4.8.3.5.7 close

[功能]

关闭 log

[格式]

```
close(self, result = None, msg = None)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|---------------|----|--------|
| String | <i>result</i> | 结果 | IN |
| String | <i>msg</i> | 消息 | IN |
| | | | |

[返回值]

none

[Error Code]

none

4.8.3.5.8 append_record

[功能]

追加记录

[格式]

```
append_record(self, name, status, logpath)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|----------------|--------|--------|
| String | <i>name</i> | 名字 | IN |
| String | <i>status</i> | 状态 | IN |
| String | <i>logpath</i> | log 路径 | IN |

[返回值]

none

[Error Code]

none

4.8.3.6 parametrizedtestcase

4.8.3.6.1 setup

[功能]

设置

[格式]

setUp(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.8.3.6.2 result

[功能]

结果

[格式]

result(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.8.3.6.3 ASSERT

[功能]

ASSERT

[格式]

ASSERT(self, result, info = "")

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|---------------|----|--------|
| String | <i>result</i> | 结果 | IN |
| String | <i>info</i> | 信息 | IN |
| | | | |

[返回值]

none

[Error Code]

none

4.8.3.6.4 BLOCK

[功能]

Block

[格式]

BLOCK(self, result, info = "")

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|---------------|----|--------|
| String | <i>result</i> | 结果 | IN |

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------|----|--------|
| String | <i>info</i> | 信息 | IN |
| | | | |

[返回值]

none

[Error Code]

none

4.8.3.7 Taskdata

4.8.3.7.1 Path

[功能]

Path

[格式]

path(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|------|----|
| path | 路径 |

[Error Code]

none

4.8.3.7.2 Log_path

[功能]

Log 路径

[格式]

log_path(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|--------------|------|
| path.replace | 路径替换 |

[Error Code]

none

4.8.3.7.3 Dir

[功能]

Dir

[格式]

dir(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|--------------|------|
| path.replace | 路径替换 |

[Error Code]

none

4.8.3.7.4 set_type

[功能]

设置类型

[格式]

set_type(self, itype)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|--------------|----|--------|
| String | <i>itype</i> | 类型 | IN |
| | | | |
| | | | |

[返回值]

none

[Error Code]

none

4.8.3.7.5 type

[功能]

类型

[格式]

type(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|------|----|
| type | 类型 |

[Error Code]

none

4.8.3.7.6 name

[功能]

名字

[格式]

name(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|------|----|
| name | 名称 |

[Error Code]

none

4.8.3.7.7 data

[功能]

数据

[格式]

data(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|------|----|
| data | 数据 |

[Error Code]

none

4.8.3.7.8 to_json

[功能]

To json

[格式]

to_json(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|----------|---------|
| taskjson | 分派 json |

[Error Code]

none

4.8.3.7.9 request

[功能]

请求

[格式]

request(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|------------------|---------|
| data ["REQUEST"] | 数据” 结果” |

[Error Code]

none

4.8.3.7.10set_request

[功能]

设置请求

[格式]

```
set_request(self, request)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|----------------|----|--------|
| String | <i>request</i> | 请求 | IN |
| | | | |
| | | | |

[返回值]

none

[Error Code]

none

4.8.3.7.11 node_index

[功能]

节点索引

[格式]

```
node_index (self)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|---------------|------------|
| taskjson[key] | 分派 json 密钥 |
| None | 空 |

[Error Code]

none

4.8.3.7.12load_cfg

[功能]

读取配置文件

[格式]

load_cfg(self, cfg)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------------|------|--------|
| String | <i>cfg</i> | 配置文件 | IN |
| | | | |
| | | | |

[返回值]

| 名称 | 说明 |
|------|------|
| json | json |

[Error Code]

none

4.8.3.7.13 tasks

[功能]

分派

[格式]

tasks(self, recursive=True)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|------------------|----|--------|
| String | <i>recursive</i> | 递归 | IN |

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
| | | | |
| | | | |

[返回值]

| 名称 | 说明 |
|-----|-------|
| ret | 子程序返回 |

[Error Code]

none

4.8.3.8 Taskrunner

4.8.3.8.1 run_single_task

[功能]

执行单一任务

[格式]

```
run_single_task(task, judge = True, process_log = True)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|--------------------|-----|--------|
| String | <i>task</i> | 任务 | IN |
| Bool | <i>judge</i> | 判断 | IN |
| Bool | <i>process_log</i> | Log | IN |

[返回值]

| 名称 | 说明 |
|------------------|--------|
| result, response | 结果, 响应 |

[Error Code]

none

4.8.3.8.2 run_pair_task

[功能]

执行一对任务

[格式]

```
run_pair_task(task1, task2, compare_src_key = None, compare_dist_key = None)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|--------|-------------------------|------|--------|
| String | <i>Task1</i> | 任务 1 | IN |
| String | <i>Task2</i> | 任务 2 | IN |
| Int | <i>compare_src_key</i> | | IN |
| Int | <i>compare_dist_key</i> | | IN |

[返回值]

| 名称 | 说明 |
|---------|------|
| result1 | 结果 1 |
| result2 | 结果 2 |
| result | 结果 |

[Error Code]

none

4.8.3.9 Taskthread

4.8.3.9.1 run

[功能]

执行

[格式]

```
run(self)
```

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

none

[Error Code]

none

4.8.3.9.2 get_result

[功能]

获取结果

[格式]

get_result(self)

[参数]

| 类型 | 名称 | 说明 | IN/OUT |
|----|----|----|--------|
|----|----|----|--------|

[返回值]

| 名称 | 说明 |
|--------|----|
| result | 结果 |
| None | 空 |

[Error Code]

none

5 APPENDIX—QUESTIONS

Give a list of questions/issues listed in the document

END OF DOCUMENT