



中山大學  
SUN YAT-SEN UNIVERSITY

# 基于区块链的供应链金融平台 (前期热身报告)

姓 名: 张欢

学 号: 1834216

年 级: 18 级

专 业: 软件工程

2020 年 11 月 29 日

## 目录

1. 项目要求.....	4
2. 项目环境.....	4
3. 项目内容.....	4
3.1. 单群组 FISCO BCOS 联盟链的搭建(使用开发部署工具 build_chain.sh 脚本在本地搭建一条 4 节点的 FISCO BCOS 链) .....	4
3.1.1. 安装依赖.....	4
3.1.2. 创建操作目录, 下载安装脚本.....	4
3.1.3. 搭建单群组 4 节点联盟链.....	4
3.1.4. 启动 FISCO BCOS 链.....	5
3.1.5. 使用命令 bash nodes/127.0.0.1/start_all.sh 启动所有节点.....	5
3.1.6. 检查进程.....	5
3.1.7. 检查日志输出.....	5
3.2. 添加新的节点到搭建的链中.....	7
3.2.1. 为新节点生成私钥证书.....	7
3.2.2. 准备配置文件.....	7
3.2.3. 为机构生成新的 SDK 证书.....	8
3.2.4. 生成新机构证书.....	8
3.3. 部署及调用 HelloWorld 合约.....	9
3.3.1. 编写 HelloWorld 合约.....	9
3.3.2. 部署 HelloWorld 合约.....	9
3.3.3. 调用 HelloWorld 合约.....	9
3.3.4. 执行指令 quit, 退出控制台.....	10
4. 使用命令查看一个区块, 并对各个字段进行解释.....	10
4.1. 使用命令 getBlockByNumber 查看区块 2.....	10
4.2. 各个字段包含的信息(按上图显示顺序) .....	10
4.2.1. transactions: 交易列表, 当 includeTransactions 为 false 时, 显示交易的哈希。当 includeTransactions 为 true 时, 显示交易详细信息。 .....	10
4.2.2. number: 区块高度.....	11
4.2.3. hash: 区块哈希.....	11
4.2.4. parentHash: 父区块哈希.....	11
4.2.5. logsBloom: log 的布隆过滤器值.....	11
4.2.6. transactionsRoot: 区块内所有交易的 merkle 根.....	11
4.2.7. receiptsRoot: 区块内所有交易回执的 merkle 根.....	11
4.2.8. dbHash: 记录交易数据变更的哈希.....	11
4.2.9. stateRoot: 状态根哈希.....	11
4.2.10. sealer: 共识节点序号.....	11
4.2.11. sealerList: 共识节点列表.....	11
4.2.12. extraData: 附加数据.....	11
4.2.13. gasLimit: 区块中允许的 gas 最大值.....	11
4.2.14. gasUsed: 区块中所有交易消耗的 gas.....	11
4.2.15. timestamp: 时间戳.....	11
4.2.16. signatureList: PBFT 共识的签名列表.....	11
5. 参考文档.....	11

5.1. JSON-RPC API.....	11
5.2. 安装——FISCO BCOS V2.7.0 文档.....	11

## 1. 项目要求

- 1.1. 使用已有的开源区块链系统 FISCO-BCOS，完成私有链的搭建以及新节点的加入。（截图说明搭建流程）
- 1.2. 自行编写一个智能合约并部署到私有链上，同时完成合约调用。（截图说明部署流程）
- 1.3. 使用命令查看一个区块，并对各个字段进行解释。

## 2. 项目环境

Ubuntu 16.04 64bit

## 3. 项目内容

- 3.1. 单群组 FISCO BCOS 联盟链的搭建（使用开发部署工具 build\_chain.sh 脚本在本地搭建一条 4 节点的 FISCO BCOS 链）

### 3.1.1. 安装依赖

开发部署工具 build\_chain.sh 脚本依赖于 openssl, curl，使用命令 `sudo apt install -y openssl curl` 安装依赖。

```
alisa@alisa-virtual-machine:~$ sudo apt install -y openssl curl
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libcurl4 libgssapi-krb5-2 libkrb5-3 libkrb5support0 libssh-4 remmina
  remmina-common remmina-plugin-rdp remmina-plugin-secret remmina-plugin-vnc
Suggested packages:
  krb5-doc krb5-user remmina-plugin-exec remmina-plugin-kwallet
  remmina-plugin-nx remmina-plugin-spice remmina-plugin-www
  remmina-plugin-xdncp
The following NEW packages will be installed:
  curl libcurl4
The following packages will be upgraded:
  libgssapi-krb5-2 libkrb5-3 libkrb5support0 libssh-4 openssl remmina
  remmina-common remmina-plugin-rdp remmina-plugin-secret remmina-plugin-vnc
10 upgraded, 2 newly installed, 0 to remove and 1397 not upgraded.
```

### 3.1.2. 创建操作目录，下载安装脚本

- 3.1.2.1. 使用命令 `cd ~ && mkdir -p fisco && cd fisco` 创建操作目录

- 3.1.2.2. 使用命令 `curl -#LO`

[https://github.com/FISCO-BCOS/FISCO-BCOS/releases/download/v2.7.0/build\\_chain.sh](https://github.com/FISCO-BCOS/FISCO-BCOS/releases/download/v2.7.0/build_chain.sh)  
&& `chmod u+x build_chain.sh` 下载脚本

```
alisa@alisa-virtual-machine:~/fisco$ curl -#LO https://github.com/FISCO-BCOS/FISCO-BCOS/releases/download/v2.7.0/build_chain.sh && chmod u+x build_chain.sh
##### 100.0%
##### 100.0%
```

### 3.1.3. 搭建单群组 4 节点联盟链

- 3.1.3.1. 在 fisco 目录下执行指令 `bash build_chain.sh -l 127.0.0.1:4 -p 30300,20200,8545`，生成一条单群组 4 节点的 FISCO 链。

```

alisa@alisa-virtual-machine:~/fisco$ bash build_chain.sh -l 127.0.0.1:4 -p 3030
0,20200,8545
[INFO] Downloading fisco-bcos binary from https://github.com/FISCO-BCOS/FISCO-BCOS/releases/download/v2.7.0/fisco-bcos.tar.gz ...
##### 100.0%
##### 100.0%
=====
Generating CA key...
=====
Generating keys and certificates ...
Processing IP=127.0.0.1 Total=4 Agency=agency Groups=1
=====
Generating configuration files ...
Processing IP=127.0.0.1 Total=4 Agency=agency Groups=1
=====
[INFO] Start Port      : 30300 20200 8545
[INFO] Server IP       : 127.0.0.1:4
[INFO] Output Dir       : /home/alisa/fisco/nodes
[INFO] CA Path          : /home/alisa/fisco/nodes/cert/
=====
[INFO] Execute the download_console.sh script in directory named by IP to get FISCO-BCOS console.
e.g.  bash /home/alisa/fisco/nodes/127.0.0.1/download_console.sh -f
=====
[INFO] All completed. Files in /home/alisa/fisco/nodes

```

根据显示 All completed，表明该指令执行成功。

### 3.1.4. 启动 FISCO BCOS 链

#### 3.1.5. 使用命令 bash nodes/127.0.0.1/start\_all.sh 启动所有节点

```

alisa@alisa-virtual-machine:~/fisco$ bash nodes/127.0.0.1/start_all.sh
try to start node0
try to start node1
try to start node2
try to start node3
node0 start successfully
node3 start successfully
node2 start successfully
node1 start successfully

```

### 3.1.6. 检查进程

#### 3.1.6.1. 使用命令 ps -ef | grep -v grep | grep fisco-bcos 检查进程是否启动

```

alisa@alisa-virtual-machine:~/fisco$ ps -ef | grep -v grep | grep fisco-bcos
alisa 6002 2114 3 11:13 pts/0 00:00:06 /home/alisa/fisco/nodes/127.0.0.1/node0/./fisco-bcos -c config.ini
alisa 6004 2114 3 11:13 pts/0 00:00:06 /home/alisa/fisco/nodes/127.0.0.1/node2/./fisco-bcos -c config.ini
alisa 6006 2114 3 11:13 pts/0 00:00:06 /home/alisa/fisco/nodes/127.0.0.1/node1/./fisco-bcos -c config.ini
alisa 6008 2114 3 11:13 pts/0 00:00:06 /home/alisa/fisco/nodes/127.0.0.1/node3/./fisco-bcos -c config.ini

```

### 3.1.7. 检查日志输出

#### 3.1.7.1. 使用命令 tail -f nodes/127.0.0.1/node0/log/log\* | grep connected 查看节点 node0 链接的节点数

```

alisa@alisa-virtual-machine:~/fisco$ tail -f nodes/127.0.0.1/node0/log/log* | grep connected
info|2020-11-29 11:19:33.432525|[P2P][Service] heartBeat,connected count=3
info|2020-11-29 11:19:43.432749|[P2P][Service] heartBeat,connected count=3
info|2020-11-29 11:19:53.433363|[P2P][Service] heartBeat,connected count=3
info|2020-11-29 11:20:03.433562|[P2P][Service] heartBeat,connected count=3
info|2020-11-29 11:20:13.433960|[P2P][Service] heartBeat,connected count=3
info|2020-11-29 11:20:23.434735|[P2P][Service] heartBeat,connected count=3
info|2020-11-29 11:20:33.435200|[P2P][Service] heartBeat,connected count=3
alisa@alisa-virtual-machine:~/fisco$

```

#### 3.1.7.2. 执行指令 tail -f nodes/127.0.0.1/node0/log/log\* | grep +++, 检查是否在共识



```
alisa@alisa-virtual-machine:~/fisco$ tail -f nodes/127.0.0.1/node0/log/log* | grep +++
info|2020-11-29 11:22:56.436898|[g:1][CONSENSUS][SEALER]+++++ Generating seal on,blkNum=1,tx=0,nodeIdx=1,hash=8383f989...
info|2020-11-29 11:23:00.451902|[g:1][CONSENSUS][SEALER]+++++ Generating seal on,blkNum=1,tx=0,nodeIdx=1,hash=ba42fc88...
info|2020-11-29 11:23:04.475018|[g:1][CONSENSUS][SEALER]+++++ Generating seal on,blkNum=1,tx=0,nodeIdx=1,hash=862520a9...
info|2020-11-29 11:23:08.494416|[g:1][CONSENSUS][SEALER]+++++ Generating seal on,blkNum=1,tx=0,nodeIdx=1,hash=802f3379...
info|2020-11-29 11:23:12.506563|[g:1][CONSENSUS][SEALER]+++++ Generating seal on,blkNum=1,tx=0,nodeIdx=1,hash=902eed3f...
```

根据上图显示，会不停输出++++Generating seal，表示共识正常。

### 3.1.7.3. 准备依赖

#### 3.1.7.3.1. 获取控制台并返回到 fisco 目录

```
alisa@alisa-virtual-machine:~$ cd ~/fisco
alisa@alisa-virtual-machine:~/fisco$ curl -#LO https://github.com/FISCO-BCOS/console/releases/download/v2.7.0/download_console.sh
##### 100.0%
##### 100.0%
alisa@alisa-virtual-machine:~/fisco$ bash download_console.sh
bash: download_console.sh: No such file or directory
alisa@alisa-virtual-machine:~/fisco$ bash download_console.sh
[INFO] Downloading console 2.7.0 from https://github.com/FISCO-BCOS/console/releases/download/v2.7.0/console.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100  640    100  640    0    0    745    0 --:--:-- --:--:-- --:--:--    744
0     0     0     0    0    0    0    0 --:--:-- 0:00:21 --:--:--    0
curl: (7) Failed to connect to github-production-release-asset-2e65be.s3.amazonaws.com port 443: Connection refused
[WARN] Download speed is too low, try https://osp-1257653870.cos.ap-guangzhou.myqcloud.com/FISCO-BCOS/console/releases/v2.7.0/console.tar.gz
##### 100.0%
[INFO] Download console successfully
[INFO] unzip console successfully
alisa@alisa-virtual-machine:~/fisco$
```

#### 3.1.7.3.2. 拷贝控制台配置文件，配置控制台证书

```
alisa@alisa-virtual-machine:~/fisco$ cp -n console/conf/config-example.toml console/conf/config.toml
alisa@alisa-virtual-machine:~/fisco$ cp -r nodes/127.0.0.1/sdk/* console/conf/
```

### 3.1.7.4. 启动并使用控制台

#### 3.1.7.4.1. 执行指令 cd ~/fisco/console && bash start.sh 启动

```
alisa@alisa-virtual-machine:~/fisco/console$ bash start.sh
=====
Welcome to FISCO BCOS console(2.7.0)!
Type 'help' or 'h' for help. Type 'quit' or 'q' to quit console.

| $$$$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$|
| $$_| $$_| $$_| $$_| $$_| $$_| $$_| $$_| $$_| $$_|
| $$_| $$_| $$_| $$_| $$_| $$_| $$_| $$_| $$_| $$_|
| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$|
| $$_| $$_| $$_| $$_| $$_| $$_| $$_| $$_| $$_| $$_|
| $$_| $$_| $$_| $$_| $$_| $$_| $$_| $$_| $$_| $$_|
| $$_| $$_| $$_| $$_| $$_| $$_| $$_| $$_| $$_| $$_|
| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$|
| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$| $$$$|

=====
[group:1]>
```

上述显示表明控制台成功启动。

#### 3.1.7.4.2. 用控制台获取信息

##### 3.1.7.4.2.1. 获取客户端版本

```
[group:1]> getNodeVersion
ClientVersion{
  version='2.7.0',
  supportedVersion='2.7.0',
  chainId='1',
  buildTime='20201126 08:00:45',
  buildType='Linux/clang/Release',
  gitBranch='HEAD',
  gitCommitHash='0bc47666979df4766723adf1ab9c5d80f5e40537'
}
```

##### 3.1.7.4.2.2. 获取节点信息

```
[group:1]> getPeers
{
  PeerInfo{
    nodeID='19a9b325571682c8c23e517327240b3ba51dafb4e508375b9657b2c24d37e4e07a1326afc6b9f689166252589b6fd99406de1bb32ba674101bba12b78286f35b',
    IPAndPort='127.0.0.1:39116',
    node='node3',
    agency='agency',
    topic=[
    ]
  },
  PeerInfo{
    nodeID='a825a50ebd30dc9dd432b8c9cabb780df8bde5d07754af309769930ffe72cca2fd30476f9058a4021288bfd03259dd4b17bb2133c30b08de93d5ae573fe1b475',
    IPAndPort='127.0.0.1:39100',
    node='node1',
    agency='agency',
    topic=[
      _block_notify_1
    ]
  },
  PeerInfo{
    nodeID='695f196059622e13c8ca631c2e78655f575d75dc3a04a9bf97fea1534b0691f9a0047cfa2778d9a92d7a56f14fa25e74af1088edfc698a8fb38192068c74499a',
    IPAndPort='127.0.0.1:39124',
    node='node2',
    agency='agency',
    topic=[
    ]
  }
}
]
```

## 3.2. 添加新的节点到搭建的链中

### 3.2.1. 为新节点生成私钥证书

#### 3.2.1.1. 获取证书生成脚本

```
alisa@alisa-virtual-machine:~/fisco/nodes/127.0.0.1$ curl -#LO https://gitee.com/FISCO-BCOS/FISCO-BCOS/raw/master/tools/gen_node_cert.sh
##0#- #
alisa@alisa-virtual-machine:~/fisco/nodes/127.0.0.1$ la
download_bin.sh      gen_node_cert.sh    node2  start_all.sh
download_console.sh  node0               node3  stop_all.sh
fisco-bcos          node1               sdk    .transTest.sh
```

#### 3.2.1.2. 生成新节点私钥证书

```
alisa@alisa-virtual-machine:~/fisco/nodes/127.0.0.1$ bash gen_node_cert.sh -c ~/fisco/nodes/cert/agency -o newNode
=====
[INFO] Cert Path   : /home/alisa/fisco/nodes/cert/agency
[INFO] Output Dir  : newNode
=====
[INFO] All completed. Files in newNode
```

### 3.2.2. 准备配置文件

#### 3.2.2.1. 拷贝群组 1 中节点 node0 配置文件与工具脚本

```
alisa@alisa-virtual-machine:~/fisco/nodes/127.0.0.1$ cp node0/config.ini newNode/config.ini
alisa@alisa-virtual-machine:~/fisco/nodes/127.0.0.1$ cp node0/conf/group.1.genesis newNode/conf/group.1.genesis
alisa@alisa-virtual-machine:~/fisco/nodes/127.0.0.1$ cp node0/conf/group.1.ini newNode/conf/group.1.ini
alisa@alisa-virtual-machine:~/fisco/nodes/127.0.0.1$ cp node0/*.sh newNode/
alisa@alisa-virtual-machine:~/fisco/nodes/127.0.0.1$ cp -r node0/scripts newNode/
```

#### 3.2.2.2. 更新 newNode/config.ini 中监听的 IP 和端口，对于[rpc]模块，修改 listen\_ip、channel\_listen\_port 和 jsonrpc\_listen\_port；对于[p2p]模块，修改 listen\_port

```
[rpc]
channel_listen_ip=0.0.0.0
channel_listen_port=20204
jsonrpc_listen_ip=127.0.0.1
jsonrpc_listen_port=8549

[p2p]
listen_ip=0.0.0.0
listen_port=30304
; nodes to connect
node.0=127.0.0.1:30300
node.1=127.0.0.1:30301
node.2=127.0.0.1:30302
node.3=127.0.0.1:30303
node.4=127.0.0.1:30304
```

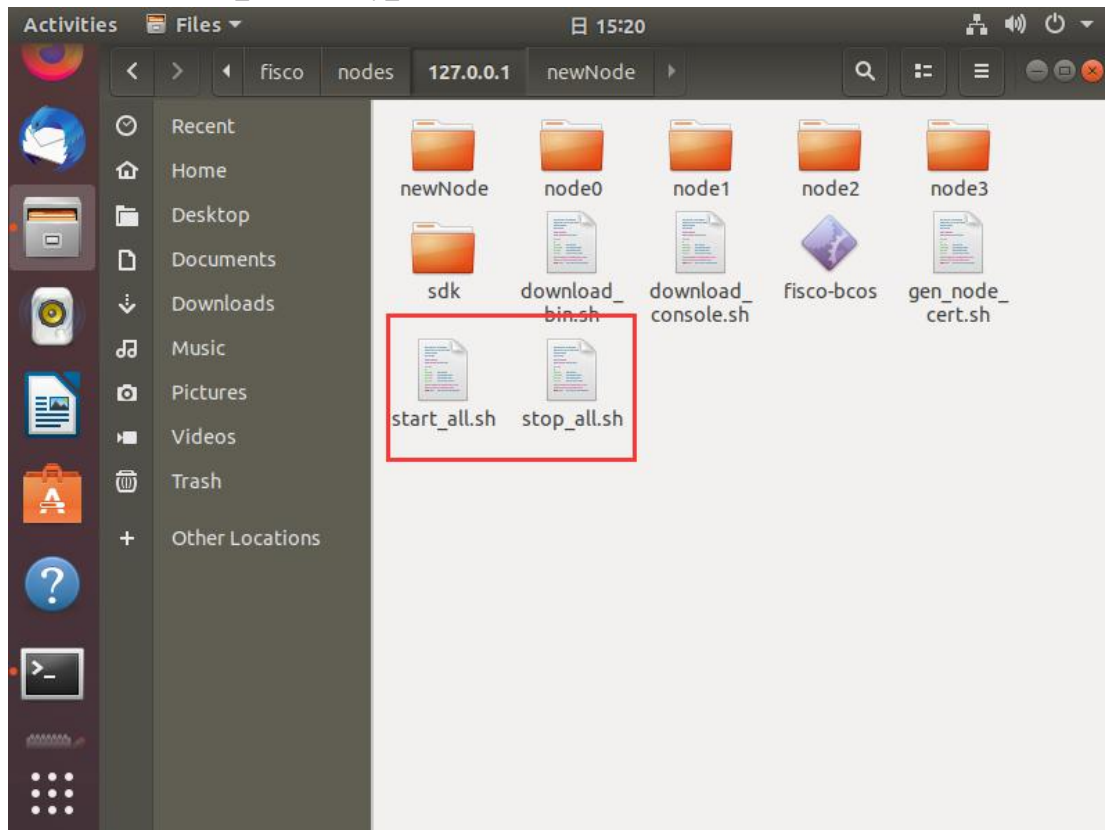
#### 3.2.2.3. 将新节点的 P2P 配置中的 IP 和 Port 加入原有节点的 config.ini 中的[p2p]字段

#### 3.2.2.4. 启动新节点



```
alisa@alisa-virtual-machine:~/fisco/nodes/127.0.0.1$ newNode/start.sh
newNode start successfully
```

### 3.2.2.5. 拷贝 start\_all.sh、stop\_all.sh 到新节点目录下



### 3.2.2.6. 运行 ./start\_all.sh 检查连接和共识

```
alisa@alisa-virtual-machine:~/fisco/nodes/127.0.0.1$ ./start_all.sh
try to start newNode
try to start node0
try to start node1
try to start node2
try to start node3
newNode is running, pid is 11783.
node1 start successfully
node0 start successfully
node2 start successfully
node3 start successfully
alisa@alisa-virtual-machine:~/fisco/nodes/127.0.0.1$ tail -f newNode/log/log* | grep connected
info|2020-11-29 15:19:00.503844|[P2P][Service] heartBeat,connected count=4
info|2020-11-29 15:19:10.504154|[P2P][Service] heartBeat,connected count=4
info|2020-11-29 15:19:20.505241|[P2P][Service] heartBeat,connected count=4
info|2020-11-29 15:19:30.506597|[P2P][Service] heartBeat,connected count=4
^C
```

较之前相比，这里的 `count=4`，比之前多一，说明新节点成功被加入到这条链上。

## 3.2.3. 为机构生成新的 SDK 证书

### 3.2.3.1. 获取证书生成脚本

```
alisa@alisa-virtual-machine:~/fisco/nodes/127.0.0.1$ curl -#LO https://gitee.com/FISCO-BCOS/FISCO-BCOS/raw/master/tools/gen_node_cert.sh
##0=# #
```

### 3.2.3.2. 生成新节点私钥证书

```
alisa@alisa-virtual-machine:~/fisco/nodes/127.0.0.1$ bash gen_node_cert.sh -c ../cert/agency -o newSDK -s
=====
[INFO] Cert Path : ../cert/agency
[INFO] Output Dir : newSDK
=====
[INFO] All completed. Files in newSDK
```

### 3.2.4. 生成新机构证书

```
alisa@alisa-virtual-machine:~/fisco/nodes/127.0.0.1$ curl -#LO https://gitee.com/FISCO-BCOS/FISCO-BCOS/raw/master/tools/gen_agency_cert.sh
##0#- #
```



#### 3.2.4.1. 生成新机构私钥和证书

#### 3.2.4.2. 获取机构证书生成脚本

```
alisa@alisa-virtual-machine:~/fisco/nodes/127.0.0.1$ bash gen_agency_cert.sh -c ../cert/ -a newAgencyName
=====
[INFO] Cert Path : ../cert//newAgencyName
[INFO] All completed.
```

### 3.3. 部署及调用 HelloWorld 合约

#### 3.3.1. 编写 HelloWorld 合约

HelloWorld 合约提供两个接口，分别是 `get()` 和 `set()`，用于获取/设置合约变量 `name`。合约内容如下：

```
pragma solidity ^0.4.24;

contract HelloWorld {
    string name;

    function HelloWorld() {
        name = "Hello, World!";
    }

    function get() constant returns(string) {
        return name;
    }

    function set(string n) {
        name = n;
    }
}
```

#### 3.3.2. 部署 HelloWorld 合约

3.3.2.1. HelloWorld 合约已经内置于控制台中，位于控制台目录下 `contracts/solidity/HelloWorld.sol`，直接使用命令 `deploy HelloWorld` 部署

```
[group:1]> deploy HelloWorld
transaction hash: 0x68af542131070886cbc822f9891066fcddb3f5f0638746e7fa5dbe056d1ef5b5
contract address: 0x7cb1cbd7569232ae7f0a623832e2ce95daeeeb3f
```

#### 3.3.3. 调用 HelloWorld 合约

##### 3.3.3.1. 查看当前块高

```
[group:1]> getBlockNumber
2
```

这表明当前块高为 2。

3.3.3.2. 调用 `get` 接口获取 `name` 变量 此处的合约地址是 `deploy` 指令返回的地址

```
[group:1]> call HelloWorld 0x7cb1cbd7569232ae7f0a623832e2ce95daeeeb3f get
-----
Return code: 0
description: transaction executed successfully
Return message: Success
-----
Return values:
[
    "Hello,World!"
]
-----
```

### 3.3.3.3. 查看当前块高

```
[group:1]> getBlockNumber
2
```

块高不变，因为 `get` 接口不更改账本状态。

#### 3.3.3.4. 调用 set 设置 name

```
[group:1]> call HelloWorld 0x7cb1cbd7569232ae7f0a623832e2ce95daeeeb3f set "Hello, FISCO BCOS"
transaction hash: 0x6cb9e93258181362121f47a896d1fd87a932fa49a5e9ca17459ee63d21c24596
-----
transaction status: 0x0
description: transaction executed successfully
-----
Output
Receipt message: Success
Return message: Success
Return value: []
-----
Event logs
Event: {}
```

### 3.3.3.5. 再次查看当前块高

```
[group:1]> getBlockNumber
3
```

块高增加表示已出块，账本状态已更改。

### 3.3.3.6. 调用 get 接口获取 name 变量，检查设置是否生效

```
[group:1]> call HelloWorld 0x7cb1cbd7569232ae7f0a623832e2ce95daeeeb3f get
-----
Return code: 0
description: transaction executed successfully
Return message: Success
-----
Return values:
[
  "Hello,FISCO BCOS"
]
-----
```

values 的值被更改，设置生效。

### 3.3.4. 执行指令 quit, 退出控制台

#### 4. 使用命令查看一个区块，并对各个字段进行解释

#### 4.1. 使用命令 `getBlockByNumber` 查看区块 2

[illegible]

#### 4.2. 各个字段包含的信息（按上图显示顺序）

4.2.1. transactions: 交易列表，当 includeTransactions 为 false 时，显示交易的哈希。当 includeTransactions 为 true 时，显示交易详细信息。

- 4.2.2. number: 区块高度
- 4.2.3. hash: 区块哈希
- 4.2.4. parentHash: 父区块哈希
- 4.2.5. logsBloom: log 的布隆过滤器值
- 4.2.6. transactionsRoot: 区块内所有交易的 merkle 根
- 4.2.7. receiptsRoot: 区块内所有交易回执的 merkle 根
- 4.2.8. dbHash: 记录交易数据变更的哈希
- 4.2.9. stateRoot: 状态根哈希
- 4.2.10. sealer: 共识节点序号
- 4.2.11. sealerList: 共识节点列表
- 4.2.12. extraData: 附加数据
- 4.2.13. gasLimit: 区块中允许的 gas 最大值
- 4.2.14. gasUsed: 区块中所有交易消耗的 gas
- 4.2.15. timestamp: 时间戳
- 4.2.16. signatureList: PBFT 共识的签名列表
- 5. 参考文档
  - 5.1. [JSON-RPC API](#)
  - 5.2. [安装——FISCO BCOS V2.7.0 文档](#)