Table of Contents

简介 1.1

fisco-bcos介绍 1.2

- 白皮书介绍 1.2.1
- 项目介绍 1.2.2

环境搭建 1.3

- solc 1.3.1
- remix-ide 1.3.2

节点搭建 1.4

- 单节点搭建 1.4.1
- 多节点搭建 1.4.2
- 多群组多节点搭建 1.4.3
- 浏览器搭建 1.4.4

模块介绍 1.5

合约 1.6

- 以太坊合约部署 1.6.1
- fisco-bcos合约部署 1.6.2
- 理解合约一 1.6.3
- 理解合约二 1.6.4
- 理解合约三 1.6.5

fisco-bcos

FISCO BCOS 是一个稳定、高效、安全的区块链底层平台，经过多家机构、多个应用，长时间在生产环境运行的实际检验

一、fisco介绍和节点简单搭建 demo:节点快速搭建 二、控制台和浏览器介绍和搭建,控制台操作等 demo:环境搭建 三 四，多群组组网搭建，机构/群组/节点/证书/安全控制/准入等概念介绍和区分, demo:复制节点搭建 五 共识和区块执行流程 demo:通过日志或者gdb还原区块执行流程 六 合约详细 demo:资产管理 七 新特性和不同 demo:新特性的演示或者源码查看 八:总结 对比主流链和解决问题的初心

权限控制 https://mp.weixin.qq.com/s?__biz=MzA3MTI5Njg4Mw==&mid=2247485317&idx=1&sn=4a7cf90cc727382af2099d11f67d8b0e&scene=19#wechat_redirect

仲裁节点环境搭建 https://mp.weixin.qq.com/s?__biz=MzA3MTI5Njg4Mw==&mid=2247485337&idx=1&sn=622e88b631ae1bfe5789b2fe21576779&scene=19#wechat_redirect

Executive::initialize Executive::verifyTransaction bool Executive::execute Executive::create m_t->isCreation() Executive::createOpcode Executive::executeCreate Executive::go

echo 52 | disasm

Executive::initialize Executive::verifyTransaction bool Executive::execute Executive::create m_t->isCreation() Executive::createOpcode Executive::executeCreate Executive::go

echo 52 | disasm

Executive::initialize Executive::verifyTransaction bool Executive::execute Executive::create m_t->isCreation() Executive::createOpcode Executive::executeCreate Executive::go

echo 52 | disasm

环境搭建

solc

```
brew update
brew upgrade
brew tap ethereum/ethereum
brew install solidity
```

https://solidity.readthedocs.io/en/v0.4.24/installing-solidity.html

remix-ide

> 合约开发和测试工具

git clone https://github.com/ethereum/remix-ide.git cd remix-ide npm install npm run build && npm run serve ```

名称介绍

单节点搭建

mac下安装 目前我们官方文档给出的环境搭建都是在linux下测试运行的,实际上调整一下,也能在mac上运行。mac环境需要先安装openssl curl的环境;

```
下载自动脚本
curl -LO https://github.com/FISCO-BCOS/FISCO-BCOS/releases/
初始化环境
./build_chain.sh -l "127.0.0.1:1"
启动
./nodes/127.0.0.1/start_all.sh
```

查看进程

```
ps -ef | grep -v grep | grep fisco-bcos
```

日志输出

```
tail -f nodes/127.0.0.1/node0/log/log_2020072613.05.log

info|2020-07-26 13:11:03.890250|[g:1][CONSENSUS][SEALER]++

info|2020-07-26 13:11:03.890799|[g:1][CONSENSUS][PBFT]check
info|2020-07-26 13:11:03.890821|[g:1][CONSENSUS][PBFT]check
```

控制台

```
cd nodes/127.0.0.1/
./download_console.sh
cp sdk/* console/conf/
cp  console/conf/applicationContext-sample.xml console/conf
./console/start.sh
```

```
There is no hello.sol in the directory of contracts/solidit
```

在操作控制台命令,获取区块高度

```
getBlockNumber
```

查看到区块高度是0;我们看ETH的公链高度每几秒机会增加,fisco-bcos的高度没有随着时间的增加而增加。

部署一个测试合约，发现高度增加了1

```
deploy HelloWorld
getBlockNumber
getCode 0x16e8178ad85c1a820ace341637a28b8bff278540
call HelloWorld 0x16e8178ad85c1a820ace341637a28b8bff278540
call HelloWorld 0x16e8178ad85c1a820ace341637a28b8bff278540
call HelloWorld  0x16e8178ad85c1a820ace341637a28b8bff27854(
```

账本的状态没有改变,所以区块高度一直没有增加;当部署合约之后，账本状态发送了改变,高度也发送了改变。ETH公链使用POW机制,需要时时给矿工正反馈的机制,在每个高度产生2ETH的奖励给某个挖矿用户,即在每个高度账本的状态都发生了改变,所以需要产生区块,高度也发生了变化.

-l 指定IP和启动多少台；-l过程经过以下几个关键步骤

- 下载二进制文件
- 生成根证书,节点证书,sdk证书
- 初始化节点和环境配置

多节点搭建

```
curl -LO https://github.com/FISCO-BCOS/FISCO-BCOS/releases/
 ./build_chain.sh -l "127.0.0.1:4" -p 30300,20200,8545 -v 2
 ./nodes/127.0.0.1/start_all.sh
```

启动后查看日志,看节点启动是否正常。 如果我们停止掉其中的一个，链还是能正常运行的;如果再停止一个,链就不能正常运行了。 这是拜占庭共识的特点.

多节点搭建

FISCO BCOS generator为企业用户提供了部署、管理和监控多机构多群组联盟链的便捷工具。

设计实录 达成目标

浏览器 brew install https://github.com/tebelorg/Tump/releases/download/v1.0.0/openssl.rb

浏览器搭建

官方提供的是linux版本的,mac使用可能会有环境问题,所以使用手动搭建模式

1 获取浏览器代码

```
git clone https://github.com/FISCO-BCOS/fisco-bcos-browser.

├── deploy
│.. └── comm
├── img
├── server
│.. └── fisco-bcos-browser  #后端代码
└── web
    └── fisco-bcos-browser-front #前端代码
```

fisco-bcos-browser代码使用前后端分离的方式开发,代码目录很清晰

后端部署

环境要求

环境	版本
Java	jdk1.8.0_121或以上版本
gradle	gradle-5.0或以上版本
mysql	mysql-5.6或以上版本

后端编译和配置

```
cd server/fisco-bcos-browser
#编译出dist
gradle build
#配置文件
mv conf_template conf
#修改配置文件
vim conf/application.yml
    url: jdbc:mysql://127.0.0.1:3306/db_browser?useUnicode=
    username: root
    password: xxxx
    driver-class-name: com.mysql.jdbc.Driver
#运行，默认端口为5001
sh start.sh
```

查看后端是否成功启动,因安装环境不同,可能会存在以下问题

openssl问题

```
brew install https://github.com/tebelorg/Tump/releases/down
```

gradle安装配置

```
Could not find method annotationProcessor() for arguments

https://gradle.org/releases/

gradle 需要使用5.0以上版本
```

前端配置

前端代码存在web/fisco-bcos-browser-front/dist，已经编译完直接配置nginx指向即可

```
server{
    listen       80;   #步骤1、前端nginx监听端口
    server_name  lfisco-bcos.com;           #步骤1、前端地址,
    location / {
            root    /Users/liuhaoyang/2cWorkPlace/demo-fisc
            index  index.html index.htm;
            try_files $uri $uri/ /index.html =404;
        }

    # Load configuration files for the default server block
    #步骤3、后端服务(fisco-bcos-browser server)地址及端口
    location /api {
        proxy_pass    http://127.0.0.1:5101/;
        proxy_set_header Host $host;
        proxy_set_header  X-Real-IP   $remote_addr;
        proxy_set_header  X-Forwarded-For  $proxy_add_x_for
    }
    access_log /Users/liuhaoyang/logs/php/lfisco-bcos.com_a
    error_log  /Users/liuhaoyang/logs/php/lfisco-bcos.com_e
}
```

按照自己的环境配置完重启nginx即可

模块介绍

```
├── cmake
│   ├── scripts
│   ├── secp256k1
│   └── templates
├── docs
│   └── images
├── evmc
├── fisco-bcos
│   ├── benchmark
│   ├── blockverifier
│   ├── consensus
│   ├── evm
│   ├── main
│   ├── p2p
│   ├── para
│   ├── rpc
│   ├── sync
│   └── tools
├── libblockchain
     BlockChainImp，检查构建创世块、获取区块和交易等face接口的实现
├── libblockverifier
    executeBlock
    executeTransaction
    parallelExecuteBlock
├── libchannelserver
    基于channel的sdk长连接协议定义，与web3sdk中定义类似，此处为ser
├── libconfig
    全局配置
├── libconsensus
│   ├── pbft
│   ├── raft
│   └── rotating_pbft
        共识模块
├── libdevcore
        地址, base64, log, rlp, trie等定义
├── libdevcrypto
│   ├── sm2
│   ├── sm3
│   └── sm4
    加密模块
├── libethcore
        block区块定义;
        transaction定义;
        abi定义;
        并行交易定义;
├── libeventfilter
        EventLogFilterManager
        EventLogFilterParams
```

```
        EventLogFilter: matches
├── libevm
        EVMC 提供exec接口
        ExtVmFace 提供evm的call get接口
        VMFactory
├── libexecutive
        Executive: call, create,execute
        ExtVM: 生成调用结果或部署结果
├── libflowlimit
├── libinitializer
    总的初始化，将P2P, RPC, Secure, Ledger, Log等模块的初始化集中
├── libinterpreter
        VM 虚拟机汇编操作的定义
        VMCall call操作对应汇编的定义
        VMOpt copyCode, exp256 操作工具类
├── libledger
        DBInitializer: 初始化storage相关，create实例
        创建共识引擎
        初始化blockVerifier, eventLogFilter,initBlockChain,
├── libmptstate
        Account 账户的定义，nonce, code, balance等, setCode账
        State, MPTState MPTStateFactory
        MPTStateFactory: getState
        State: addAddress, addBalance/subBalance, createAcc
        MPTState: commit操作，维持MPT State状态树，State为数据
├── libnetwork
        ASIOInterface boost的网络模块，用于socket连接, ssl的we
        Host: 节点间握手(Server/Client) 与P2P关联
        asyncConnect..
        Session session会话，监听信息
PeerWhitelist 白名单
├── libp2p
        p2p模块
├── libprecompiled
│   ├── extension
│   └── solidity
    预编译合约的实现，包含接口solidity文件。与table直接交互
├── librpc
        SafeHttpServer 接收/回复rpc API的请求
        Rpc 定义rpc的接口
        JsonHelper HttpMessage等解析工具类
├── libsecurity
    EncryptedLevelDB leveldb文件加密
    EncryptedFile: decryptContents解密文件
    KeyCenter: KeyCenterHttpClient keyServer密钥管理中心
├── libstat
├── libstorage
        BasicRocksDB  put commit
```

```
        RocksDBStorage put commit
        LevelDBStorage commit, select, put commit
        SQLStorage: 是否与AMDB相关
        ZdbStorage: 在mysql上，创建预编译、系统表等数据表
        CachedStorage: 缓存存储，具体未知
        ScalableStorage: 可扩展存储，与AMDB proxy相关?
        MemoryTable 系统表，如权限控制,crud等
        BinLog: binlog handler(decode, encode, ), binlog st
        SQLBasicAccess: buildSQL, buildConditions
        SQLConnectionPool: 连接池
        Table: table的基本操作setEntries
├── libstoragestate
        StorageState MPT操作定义对state的操作如，createAccount
        StorageStateFactory 工厂类
├── libsync
        DownloadingBlockQueue, DownloadingTxsQueue
        GossipBlockStatus 传递区块状态包
        DownloadRequest, RspBlockReq 下载区块请求
        SyncMaster 同步操作定义，send,broadcast
        SyncTransaction 广播交易
        SyncMsgEngine msg监听、收发
        SyncMsgPacket 广播的消息包
        SyncTreeTopology 网络拓扑图 nodeList, nodeInfo
├── libtxpool
        TransactionNonceCheck commonTxCheck
        TxPool insert, clear, pending, verify等对交易池的操作
├── test
│   ├── data
│   ├── tools
│   └── unittests
└── tools
    └── ci
```

名称介绍

名称介绍

EVM

名称介绍

合约是什么

合约是一段在链上执行的代码

字节码(ByteCode)是什么

源码通过编译可以形成字节码

字节码是一种包含执行程序、由一序列 op 代码/数据对组成的二进制文件

字节码与硬件无关,需要在特定的虚拟机中执行

```
   solc --bin --opcodes  Demo.sol

   Opcodes:
PUSH1 0x80 PUSH1 0x40 MSTORE PUSH1 0x0 DUP1 SSTORE CALLVALU
Binary:

608060405260008055348015610014576000 80fd5b50604051610100380
```

ABI(Application Binary Interface)是什么

ABI是定义以太坊合约调用的一种格式。

定义调用的函数签名，参数编码，返回结果编码等。

```
solc --bin --abi    Demo.sol

[{"inputs":[{"internalType":"int256","name":"y","type":"int
```

ABI上链么

在执行合约或者调用合约的过程种使用到了ABI ABI本身不上链

为什么用到ABI

既然不上链为什么要用ABI呢?

创建合约和调用合约本身有想通的地方,都是发一笔交易,改变账户或者账本状态。

在创建合约的时候,如果有构造方法,那构造方法的参数必须要通过ABI才能实现编码.

执行合约的时候,参数需要编码才能放入交易.

```
https://github.com/ethereum/web3.js/blob/1.x/packages/web3-

//发布部署合约
Contract.prototype.deploy = function(options, callback){

    options = options || {};

    options.arguments = options.arguments || [];
    options = this._getOrSetDefaultOptions(options);


    // throw error, if no "data" is specified
    if(!options.data) {
        if (typeof callback === 'function'){
            return callback(errors.ContractMissingDeployDat
        }
        throw errors.ContractMissingDeployDataError();
    }

    var constructor = _.find(this.options.jsonInterface, fu
        return (method.type === 'constructor');
    }) || {};
    constructor.signature = 'constructor';

    return this._createTxObject.apply({
        method: constructor,
        parent: this,
        deployData: options.data,
        _ethAccounts: this.constructor._ethAccounts
    }, options.arguments);

};


//创建交易
Contract.prototype._createTxObject =  function _createTxObj
    var args = Array.prototype.slice.call(arguments);
    var txObject = {};

    if(this.method.type === 'function') {

        txObject.call = this.parent._executeMethod.bind(txO
        txObject.call.request = this.parent._executeMethod.

    }

    txObject.send = this.parent._executeMethod.bind(txObjec
    txObject.send.request = this.parent._executeMethod.bind
```

```
    txObject.encodeABI = this.parent._encodeMethodABI.bind(
    txObject.estimateGas = this.parent._executeMethod.bind(

    if (args && this.method.inputs && args.length !== this.
        if (this.nextMethod) {
            return this.nextMethod.apply(null, args);
        }
        throw errors.InvalidNumberOfParams(args.length, thi
    }

    txObject.arguments = args || [];
    txObject._method = this.method;
    txObject._parent = this.parent;
    txObject._ethAccounts = this.parent.constructor._ethAcc

    if(this.deployData) {
        txObject._deployData = this.deployData;
    }

    return txObject;
};

//encode ABI
Contract.prototype._encodeMethodABI = function _encodeMetho
    var methodSignature = this._method.signature,
        args = this.arguments || [];

    var signature = false,
        paramsABI = this._parent.options.jsonInterface.filt
            return ((methodSignature === 'constructor' && !
                ((json.signature === methodSignature || jso
        }).map(function (json) {
            var inputLength = (_.isArray(json.inputs)) ? js

            if (inputLength !== args.length) {
                throw new Error('The number of arguments is
            }

            if (json.type === 'function') {
                signature = json.signature;
            }
            return _.isArray(json.inputs) ? json.inputs :
        }).map(function (inputs) {
            return abi.encodeParameters(inputs, args).repla
        })[0] || '';

    // return constructor
    if(methodSignature === 'constructor') {
```

```
        if(!this._deployData)
            throw new Error('The contract has no contract d

        if(!this._deployData.startsWith('0x')) {
            this._deployData = '0x' + this._deployData;
        }

        return this._deployData + paramsABI;

    }

    // return method
    var returnValue = (signature) ? signature + paramsABI :

    if(!returnValue) {
        throw new Error('Couldn\'t find a matching contract
    }

    return returnValue;
};
```

合约是如何运行的

调研合约

```
//input
0x87db03b700000000000000000000000000000000000000000000000000000000

//获取结果
eth.getStorageAt("0xde2F6913ED3cdfD3220F5A6706DfA68AD325B01
```

使用0.4.25的编译器

智能合约编译后的字节码，分为三个部分：部署代码、runtime代码、auxdata。

部署代码：以上面的输出结果为例，EVM虚拟机在创建合约的时候，会先创建合约账户，然后运行部署代码。运行完成后它会将runtime代码+auxdata 存储到区块链上。之后再把二者的存储地址跟合约账户关联起来(也就是把合约账户中的code hash字段用该地址赋值)，这样就完成了合约的部署。

runtime代码：运行时代码。

auxdata：每个合约最后面的43字节就是auxdata，它会紧跟在runtime代码后面被存储起来，是源码的加密指纹，用来验证。这只是数据，永远不会被EVM执行。

solc命令的--bin-runtime选项，输出了runtime代码和auxdata，省略了部署代码，使用remix-ide也可以

bytecode

```
608060405260008055348015610014576000080fd5b506040516020806100

6080604052600436106049576000357c0100000000000000000000000000000
//auxdata
a165627a7a72305820902c0ea7bc58fa4d3a23cebb43ad421601d9d35a2
```

runtime bytecode

```
6080604052600436106049576000357c0100000000000000000000000000000

a165627a7a72305820902c0ea7bc58fa4d3a23cebb43ad421601d9d35a2
```

bytecode 种包含了部署代码 和 runtime

还使用这一段代码做实验保持为Demo.sol

```
pragma solidity ^0.4.22;
contract Demo{
    int m  = 0;
    constructor(int y) public {
        m = 15 + y;
    }

    function add( int x ) public {
        m = m + x + 14 ;
    }
    function get( ) public view returns (int) {
        return m;
    }
}
```

拷贝到控制台目录下

```
cp ~/Documents/learn/discuss-fisco-bcos/code_and_doc/Demo.s
```

部署合约，查看交易详情

```
deploy Demo  1

//部署代码
0x6080604052600080553480156100145760008fd5b506040516020806

//runtime
6080604052600436106049576000357c0100000000000000000000000000

//输入
0000000000000000000000000000000000000000000000000000000000000
```

最后64位表示输入参数;

查看整个汇编代码

```
.code
  PUSH 80           contract Demo{\n\tint m  = 0;\...
  PUSH 40           contract Demo{\n\tint m  = 0;\...
  MSTORE            contract Demo{\n\tint m  = 0;\...
  PUSH 0           0
  DUP1            int m  = 0
  SSTORE            int m  = 0
  CALLVALUE            constructor(int y) public {\n\...
  DUP1            olidity ^
  ISZERO            a
  PUSH [tag] 1           a
  JUMPI            a
  PUSH 0           a
  DUP1            n
  REVERT            .22;\ncontrac
tag 1           a
  JUMPDEST           a
  POP            constructor(int y) public {\n\...
  PUSH 40           constructor(int y) public {\n\...
  MLOAD           constructor(int y) public {\n\...
  PUSH 20           constructor(int y) public {\n\...
  DUP1            constructor(int y) public {\n\...
  PUSHSIZE           constructor(int y) public {\n\...
  DUP4            constructor(int y) public {\n\...
  CODECOPY           constructor(int y) public {\n\...
  DUP2            constructor(int y) public {\n\...
  ADD            constructor(int y) public {\n\...
  DUP1            constructor(int y) public {\n\...
  PUSH 40           constructor(int y) public {\n\...
  MSTORE            constructor(int y) public {\n\...
  DUP2            constructor(int y) public {\n\...
  ADD            constructor(int y) public {\n\...
  SWAP1            constructor(int y) public {\n\...
  DUP1            constructor(int y) public {\n\...
  DUP1            constructor(int y) public {\n\...
  MLOAD            constructor(int y) public {\n\...
  SWAP1            constructor(int y) public {\n\...
  PUSH 20           constructor(int y) public {\n\...
  ADD            constructor(int y) public {\n\...
  SWAP1            constructor(int y) public {\n\...
  SWAP3            constructor(int y) public {\n\...
  SWAP2            constructor(int y) public {\n\...
  SWAP1            constructor(int y) public {\n\...
  POP            constructor(int y) public {\n\...
  POP            constructor(int y) public {\n\...
  POP            constructor(int y) public {\n\...
  DUP1            y
  PUSH F           15
```

```
  ADD             15 + y
  PUSH 0            m
  DUP2             m = 15 + y
  SWAP1             m = 15 + y
  SSTORE             m = 15 + y
  POP             m = 15 + y
  POP             constructor(int y) public {\n\...
  PUSH #[$] 0000000000000000000000000000000000000000000000000
  DUP1             contract Demo{\n\tint m  = 0;\...
  PUSH [$] 0000000000000000000000000000000000000000000000000
  PUSH 0            contract Demo{\n\tint m  = 0;\...
  CODECOPY             contract Demo{\n\tint m  = 0;\...
  PUSH 0            contract Demo{\n\tint m  = 0;\...
  RETURN             contract Demo{\n\tint m  = 0;\...
.data
  0:
    .code
      PUSH 80            contract Demo{\n\tint m  = 0;\...
      PUSH 40            contract Demo{\n\tint m  = 0;\...
      MSTORE             contract Demo{\n\tint m  = 0;\...
      PUSH 4            contract Demo{\n\tint m  = 0;\...
      CALLDATASIZE              contract Demo{\n\tint m  = 0
      LT              contract Demo{\n\tint m  = 0;\...
      PUSH [tag] 1            contract Demo{\n\tint m  = 0;
      JUMPI             contract Demo{\n\tint m  = 0;\...
      PUSH 0            contract Demo{\n\tint m  = 0;\...
      CALLDATALOAD              contract Demo{\n\tint m  = 0
      PUSH 100000000000000000000000000000000000000000000000
      SWAP1             contract Demo{\n\tint m  = 0;\...
      DIV              contract Demo{\n\tint m  = 0;\...
      PUSH FFFFFFFF             contract Demo{\n\tint m  = 0
      AND             contract Demo{\n\tint m  = 0;\...
      DUP1              contract Demo{\n\tint m  = 0;\...
      PUSH 6D4CE63C             contract Demo{\n\tint m  = 0
      EQ              contract Demo{\n\tint m  = 0;\...
      PUSH [tag] 2            contract Demo{\n\tint m  = 0;
      JUMPI             contract Demo{\n\tint m  = 0;\...
      DUP1              contract Demo{\n\tint m  = 0;\...
      PUSH 87DB03B7             contract Demo{\n\tint m  = 0
      EQ              contract Demo{\n\tint m  = 0;\...
      PUSH [tag] 3            contract Demo{\n\tint m  = 0;
      JUMPI             contract Demo{\n\tint m  = 0;\...
    tag 1             contract Demo{\n\tint m  = 0;\...
      JUMPDEST              contract Demo{\n\tint m  = 0;\..
      PUSH 0            contract Demo{\n\tint m  = 0;\...
      DUP1              contract Demo{\n\tint m  = 0;\...
      REVERT              contract Demo{\n\tint m  = 0;\...
    tag 2             function get( ) public view re...
```

```
  JUMPDEST              function get( ) public view re..
  CALLVALUE              function get( ) public view re.
  DUP1               olidity ^
  ISZERO               a
  PUSH [tag] 4              a
  JUMPI              a
  PUSH 0             a
  DUP1               n
  REVERT               .22;\ncontrac
tag 4              a
  JUMPDEST               a
  POP              function get( ) public view re...
  PUSH [tag] 5              function get( ) public view r
  PUSH [tag] 6              function get( ) public view r
  JUMP               function get( ) public view re...
tag 5              function get( ) public view re...
  JUMPDEST               function get( ) public view re..
  PUSH 40              function get( ) public view re...
  MLOAD              function get( ) public view re...
  DUP1               function get( ) public view re...
  DUP3               function get( ) public view re...
  DUP2               function get( ) public view re...
  MSTORE               function get( ) public view re...
  PUSH 20              function get( ) public view re...
  ADD              function get( ) public view re...
  SWAP2                function get( ) public view re...
  POP              function get( ) public view re...
  POP              function get( ) public view re...
  PUSH 40              function get( ) public view re...
  MLOAD              function get( ) public view re...
  DUP1               function get( ) public view re...
  SWAP2              function get( ) public view re...
  SUB              function get( ) public view re...
  SWAP1              function get( ) public view re...
  RETURN               function get( ) public view re...
tag 3              function add( int x ) public {...
  JUMPDEST               function add( int x ) public {..
  CALLVALUE              function add( int x ) public {.
  DUP1               olidity ^
  ISZERO               a
  PUSH [tag] 7              a
  JUMPI              a
  PUSH 0             a
  DUP1               n
  REVERT               .22;\ncontrac
tag 7              a
  JUMPDEST               a
  POP              function add( int x ) public {...
```

```
  PUSH [tag] 8            function add( int x ) public
  PUSH 4                   function add( int x ) public {...
  DUP1                    function add( int x ) public {...
  CALLDATASIZE                     function add( int x ) public
  SUB                    function add( int x ) public {...
  DUP2                    function add( int x ) public {...
  ADD                    function add( int x ) public {...
  SWAP1                    function add( int x ) public {...
  DUP1                    function add( int x ) public {...
  DUP1                    function add( int x ) public {...
  CALLDATALOAD                     function add( int x ) public
  SWAP1                    function add( int x ) public {...
  PUSH 20                   function add( int x ) public {...
  ADD                    function add( int x ) public {...
  SWAP1                    function add( int x ) public {...
  SWAP3                    function add( int x ) public {...
  SWAP2                    function add( int x ) public {...
  SWAP1                    function add( int x ) public {...
  POP                    function add( int x ) public {...
  POP                    function add( int x ) public {...
  POP                    function add( int x ) public {...
  PUSH [tag] 9            function add( int x ) public
  JUMP                    function add( int x ) public {...
tag 8                   function add( int x ) public {...
  JUMPDEST                  function add( int x ) public {..
  STOP                    function add( int x ) public {...
tag 6                   function get( ) public view re...
  JUMPDEST                  function get( ) public view re..
  PUSH 0                   int
  DUP1                    m
  SLOAD                    m
  SWAP1                    return m
  POP                    return m
  SWAP1                    function get( ) public view re...
  JUMP [out]                function get( ) public view re.
tag 9                   function add( int x ) public {...
  JUMPDEST                  function add( int x ) public {..
  PUSH E                   14
  DUP2                    x
  PUSH 0                   m
  SLOAD                    m
  ADD                    m + x
  ADD                    m + x + 14
  PUSH 0                   m
  DUP2                    m = m + x + 14
  SWAP1                    m = m + x + 14
  SSTORE                    m = m + x + 14
  POP                    m = m + x + 14
```

```
        POP             function add( int x ) public {...
        JUMP [out]            function add( int x ) public {
    .data
```

CODECOPY表示拷贝代码

RETURN 表示执行到此处时整个流程结束

CALLVALUE 获取需要给合约转账的资产；如果是0，则直接进入tag1；否则revert ，因为构造函数没有payable修饰;所以该函数不能进行转账操作。

运行合约的时候，如果是第一次执行这个合约的时候，执行完构造函数内容后有个CODECOPY指令，作用是把后续的内容拷贝到覆盖构造函数的内容，然后返回合约写入statedb中

在"安装时间"和"运行时间"之间有一个强制的分离。无法运行构造器两次

调用操作

```
call Demo 0x07456194d14c1888a6a4358aff384431b48e9159   add

getTransactionByHash 0x33359a12f1c6fd57aadd3ed5654f9f3a2644

{
    "blockHash":"0x676f254838edc813e14245989c8051ba8ab0c434
    "blockNumber":"0x37",
    "from":"0xbd10881e4d4397dd6fe3922efed68847f0aa80c2",
    "gas":"0x11e1a300",
    "gasPrice":"0x11e1a300",
    "hash":"0x33359a12f1c6fd57aadd3ed5654f9f3a2644533dba977
    "input":"0x87db03b700000000000000000000000000000000000000
    "nonce":"0x26fe90f75b671f29d9734be7b1b5de0a7b4f5bf7f21a
    "to":"0x07456194d14c1888a6a4358aff384431b48e9159",
    "transactionIndex":"0x0",
    "value":"0x0"
}
```

看input，把input拆成2部分;0x87db03b7 和 0000000000000000000000000000000000000000000000000000000000000002。第一部分是函数名,使用签名的前4个字节，在geth控制台可以验证,第二部分是参数内容.

```
web3.sha3("add(int256)")
```

Executive::initialize Executive::verifyTransaction bool Executive::execute Executive::create m_t->isCreation() Executive::createOpcode Executive::executeCreate Executive::go

echo 52 | disasm