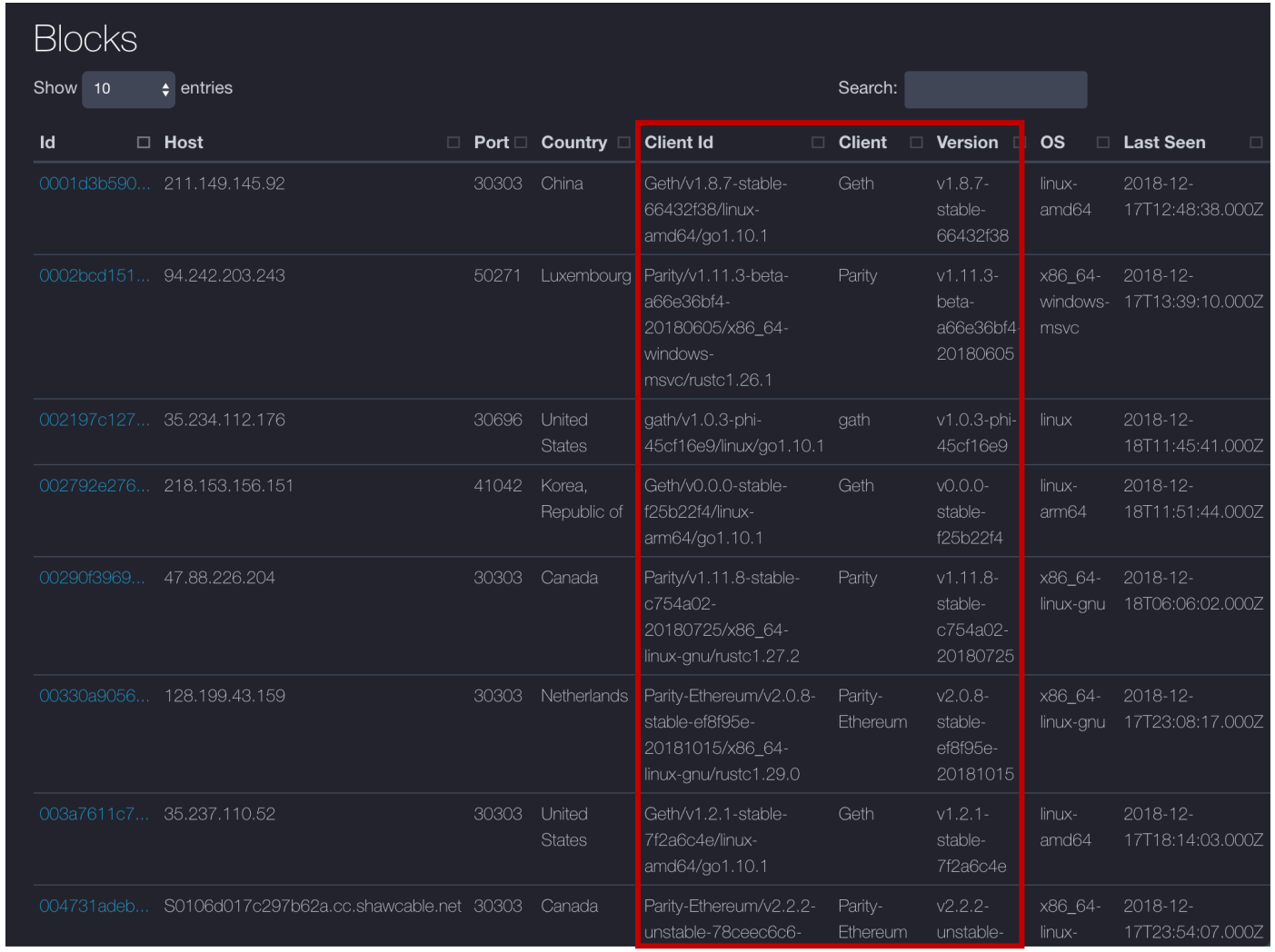
Full node

Ethereum protocol implementation

- Ethereum Client
  - Full node
  - Light client
- geth (go-ethereum)
  - https://ethereum.github.io/go-ethereum/
- parity (Rust)
  - https://www.parity.io/

parity

Ethereum nodes



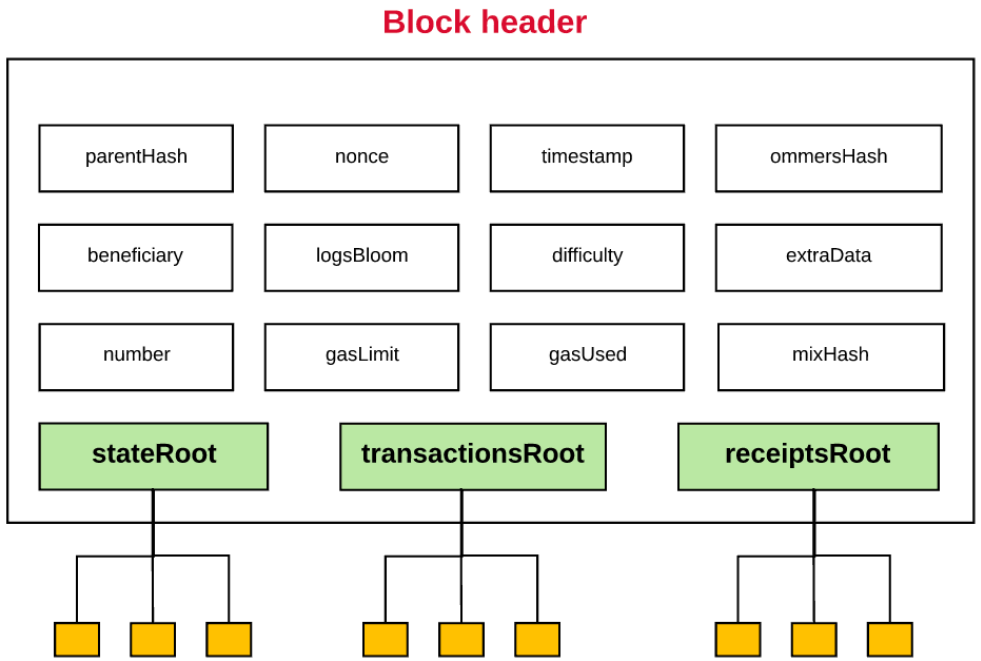
Blocks

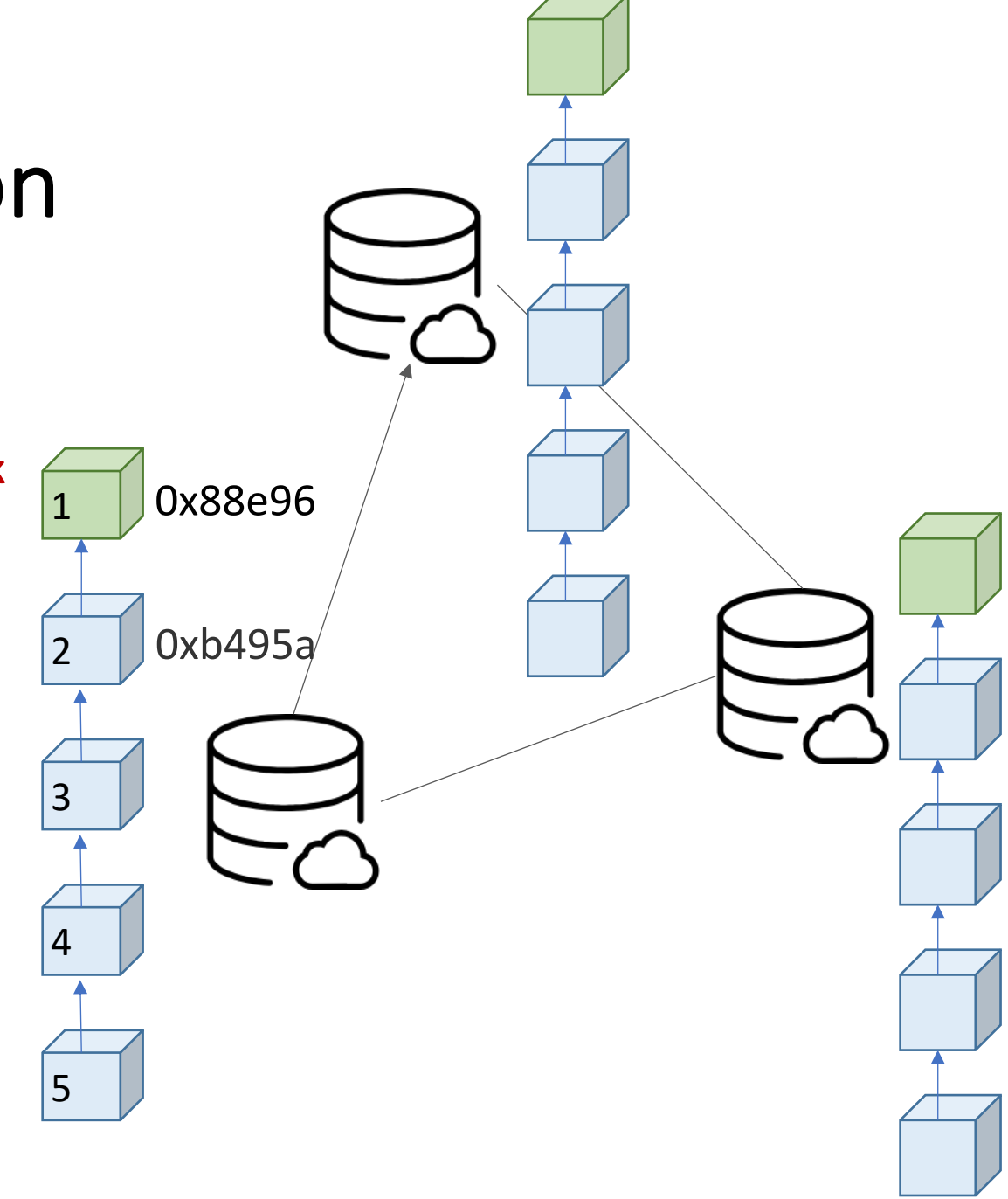
Show 10 entries Search:

Id	Host	Port	Country	Client Id	Client	Version	OS	Last Seen
0001d3b590...	211.149.145.92	30303	China	Geth/v1.8.7-stable-66432f38/linux-amd64/go1.10.1	Geth	v1.8.7-stable-66432f38	linux-amd64	2018-12-17T12:48:38.000Z
0002bcd151...	94.242.203.243	50271	Luxembourg	Parity/v1.11.3-beta-a66e36bf4-20180605/x86_64-windows-msvc/rustc1.26.1	Parity	v1.11.3-beta-a66e36bf4-20180605	x86_64-windows-msvc	2018-12-17T13:39:10.000Z
002197c127...	35.234.112.176	30696	United States	gath/v1.0.3-phi-45cf16e9/linux/go1.10.1	gath	v1.0.3-phi-45cf16e9	linux	2018-12-18T11:45:41.000Z
002792e276...	218.153.156.151	41042	Korea, Republic of	Geth/v0.0.0-stable-f25b22f4/linux-arm64/go1.10.1	Geth	v0.0.0-stable-f25b22f4	linux-arm64	2018-12-18T11:51:44.000Z
00290f3969...	47.88.226.204	30303	Canada	Parity/v1.11.8-stable-c754a02-20180725/x86_64-linux-gnu/rustc1.27.2	Parity	v1.11.8-stable-c754a02-20180725	x86_64-linux-gnu	2018-12-18T06:06:02.000Z
00330a9056...	128.199.43.159	30303	Netherlands	Parity-Ethereum/v2.0.8-stable-ef8f95e-20181015/x86_64-linux-gnu/rustc1.29.0	Parity-Ethereum	v2.0.8-stable-ef8f95e-20181015	x86_64-linux-gnu	2018-12-17T23:08:17.000Z
003a7611c7...	35.237.110.52	30303	United States	Geth/v1.2.1-stable-7f2a6c4e/linux-amd64/go1.10.1	Geth	v1.2.1-stable-7f2a6c4e	linux-amd64	2018-12-17T18:14:03.000Z
004731adeb...	S0106d017c297b62a.cc.shawcable.net	30303	Canada	Parity-Ethereum/v2.2.2-unstable-78ceec6c6-	Parity-Ethereum	v2.2.2-unstable-	x86_64-linux-	2018-12-17T23:54:07.000Z

https://www.ethernodes.org/network/1/nodes

Node synchronization



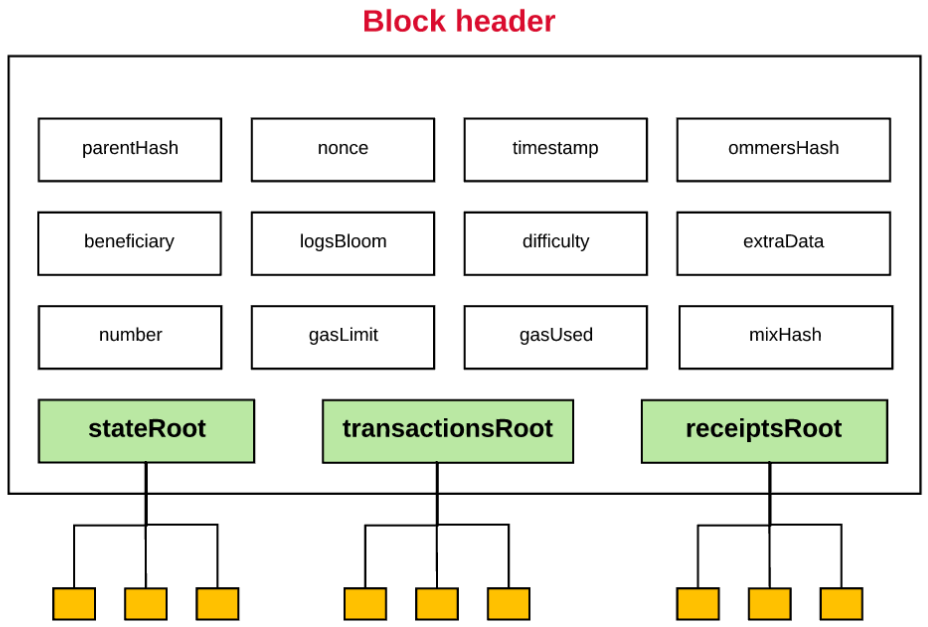


Block header

1. geth attach http://127.0.0.1:8545 (geth)

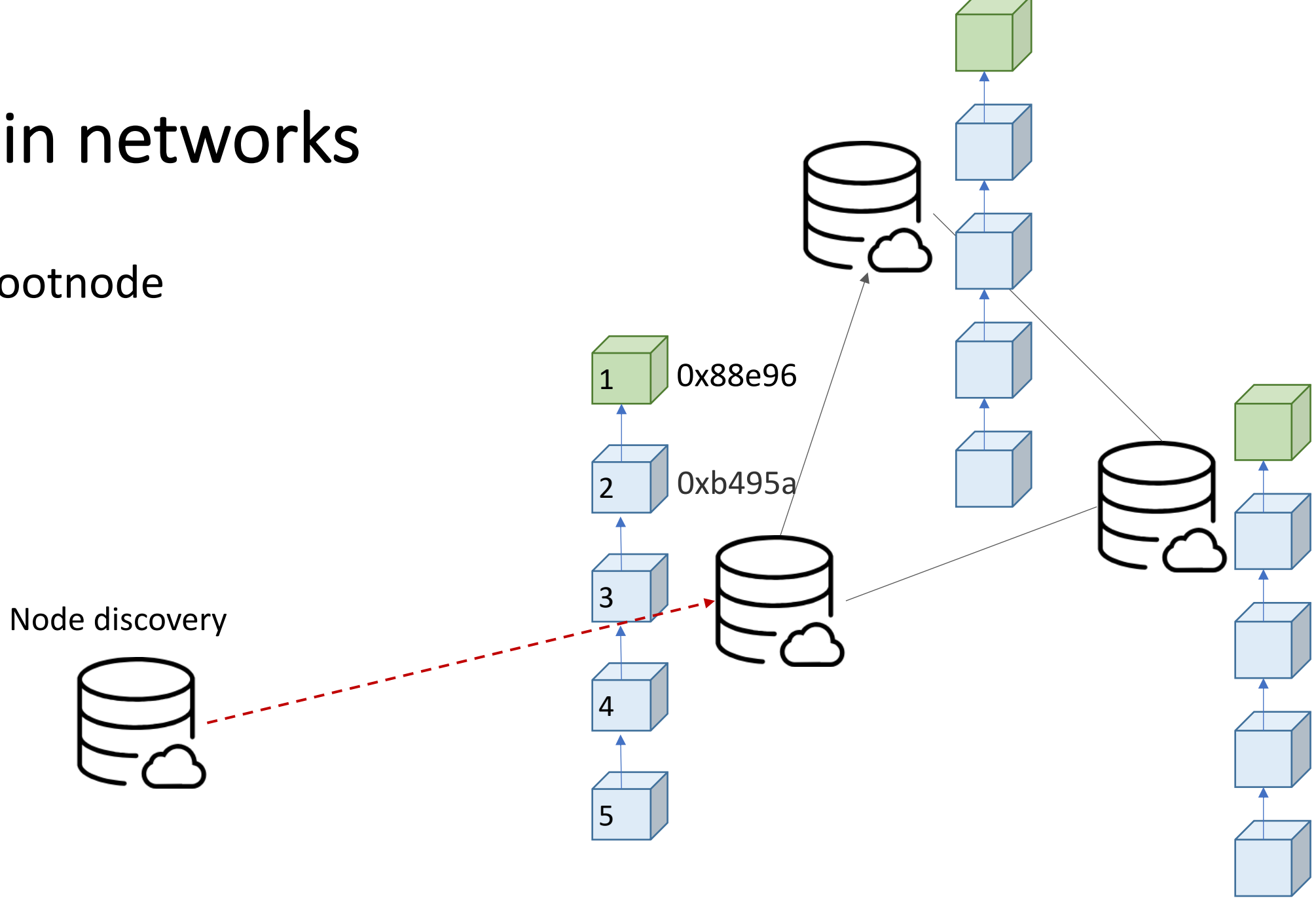
geth (geth) ⌘1 | geth (geth) ⌘2 | ~ (zsh) ⌘3 | ~ (zsh) ⌘4


```
{
  difficulty: 2,
  extraData: "0xd783010810846765746886676f312e3131866461727769 6e0000000000000000bf481b42df532442d9cfeab2828863df0b4efb
0544e169db7f03069828f72801365711137eeb9a7c6a423ae60fee5f0b41182003eff46259d0a6f0f1f7e7e69500",
  gasLimit: 6289319,
  gasUsed: 0,
  hash: "0x8cd93ff67645d3a1b96952ca3f55296a8ebebf66c4f63eb5daf7e7bc0492459c",
  logsBloom: "0x00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000",
  miner: "0x0000000000000000000000000000000000000000",
  mixHash: "0x0000000000000000000000000000000000000000000000000000000000000000",
  nonce: "0x0000000000000000",
  number: 1,
  parentHash: "0xd8e5944bddc81104674d8446c44402e90bbd75930f6917460b22eeff04b1263f",
  receiptsRoot: "0x56e81f171bcc55a6ff8345e692c0f86e5b48e01b996cadc001622fb5e363b421",
  sha3Uncles: "0x1dcc4de8dec75d7aab85b567b6ccd41ad312451b948a7413f0a142fd40d49347",
  size: 606,
  stateRoot: "0xeedbe7f7c84bec5eb68c548b47d7d905516e19117f080ffa3ed44b8def3acd4c",
  timestamp: 1545135093,
  totalDifficulty: 3,
  transactions: [],
  transactionsRoot: "0x56e81f171bcc55a6ff8345e692c0f86e5b48e01b996cadc001622fb5e363b421",
  uncles: []
}
> _
```

Join networks

- Bootnode



https://github.com/ethereum/go-ethereum/wiki/Connecting-to-the-network

Bootnode list

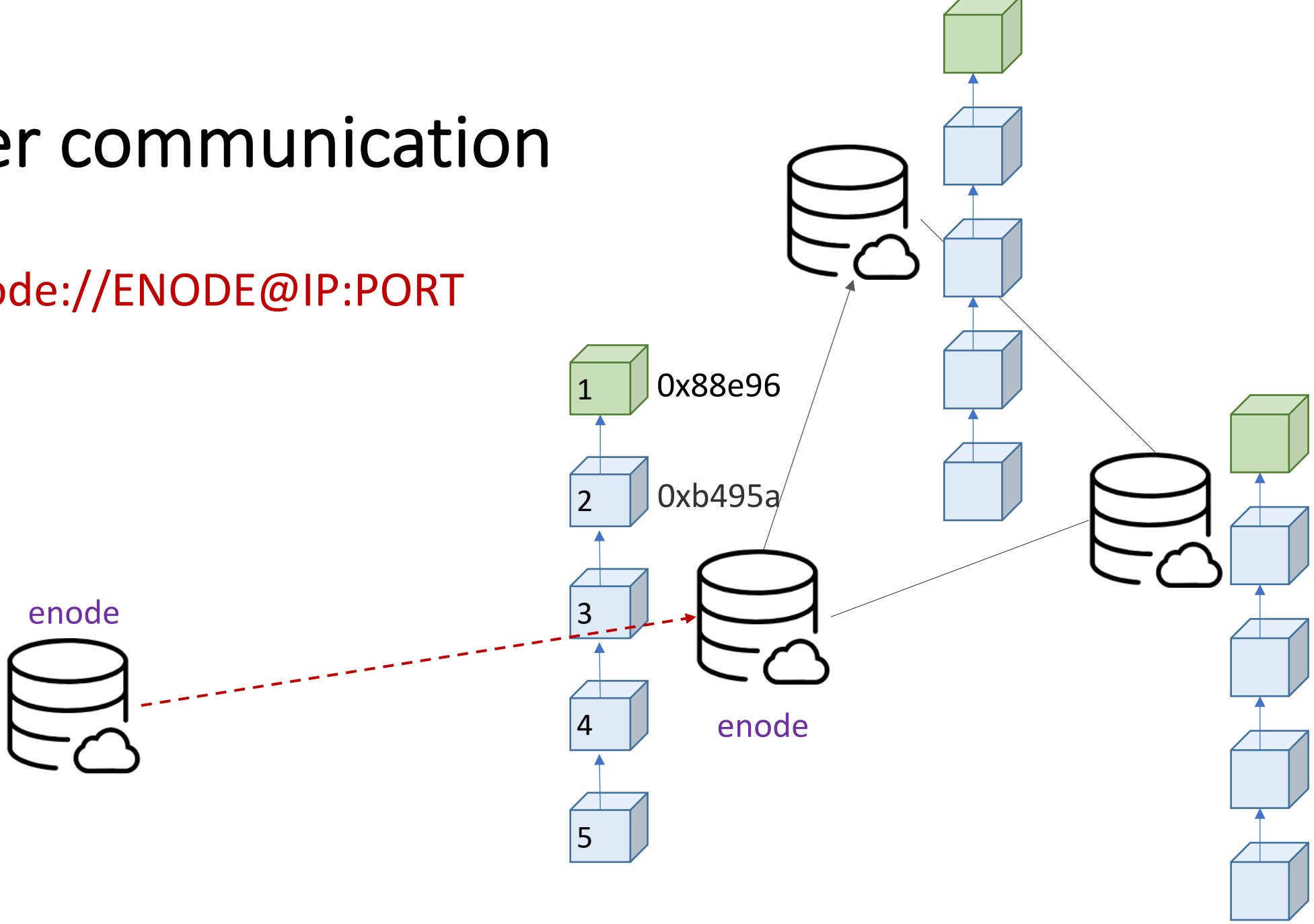
- Hardcode
- DNS

```
var MainnetBootnodes = []string{
    // Ethereum Foundation Go Bootnodes
    "enode://a979fb575495b8d6db44f750317d0f4622bf4c2aa3365d6af7c284339968eef29b69ad0dce72a4d8db5ebb4968de0e3bec910127f134779fbcb0cb6d3331163c@52.16.188.185:30303", // IE
    "enode://3f1d12044546b76342d59d4a05532c14b85aa669704bfe1f864fe079415aa2c02d743e03218e57a33fb94523adb54032871a6c51b2cc5514cb7c7e35b3ed0a99@13.93.211.84:30303", // US-WEST
    "enode://78de8a0916848093c73790ead81d1928bec737d565119932b98c6b100d944b7a95e94f847f689fc723399d2e31129d182f7ef3863f2b4c820abbf3ab2722344d@191.235.84.50:30303", // BR
    "enode://158f8aab45f6d19c6cbf4a089c2670541a8da11978a2f90dbf6a502a4a3bab80d288afdbeb7ec0ef6d92de563767f3b1ea9e8e334ca711e9f8e2df5a0385e8e6@13.75.154.138:30303", // AU
    "enode://1118980bf48b0a3640bdba04e0fe78b1add18e1cd99bf22d53daac1fd9972ad650df52176e7c7d89d1114cfef2bc23a2959aa54998a46afcf7d91809f0855082@52.74.57.123:30303", // SG

    // Ethereum Foundation C++ Bootnodes
    "enode://979b7fa28feeb35a4741660a16076f1943202cb72b6af70d327f053e248bab9ba81760f39d0701ef1d8f89cc1fbd2cacba0710a12cd5314d5e0c9021aa3637f9@5.1.83.226:30303", // DE
}
```


https://github.com/ethereum/go-ethereum/blob/master/params/bootnodes.go

Peer communication

- enode://ENODE@IP:PORT



Bootnode command

```
$ cd go-ethereum
$ make all
$ cd build/bin


// bootnode command
$ bootnode -nodekeyhex [privkey] -writeaddress

// example
$ bootnode -nodekeyhex af713077fa1244509d9872b989f7776e428a0a6afd4a77f83deabd15d80ec869 -writeaddress
f6449e7acaef2a59c2bb269260a89f6782c58976abf94ea3f01f67e6693b795cb2ec1287678f3ea615004f5993e78b0fe7e8739ffe2ac3
2a7fbdb5cbf62f81ef
```

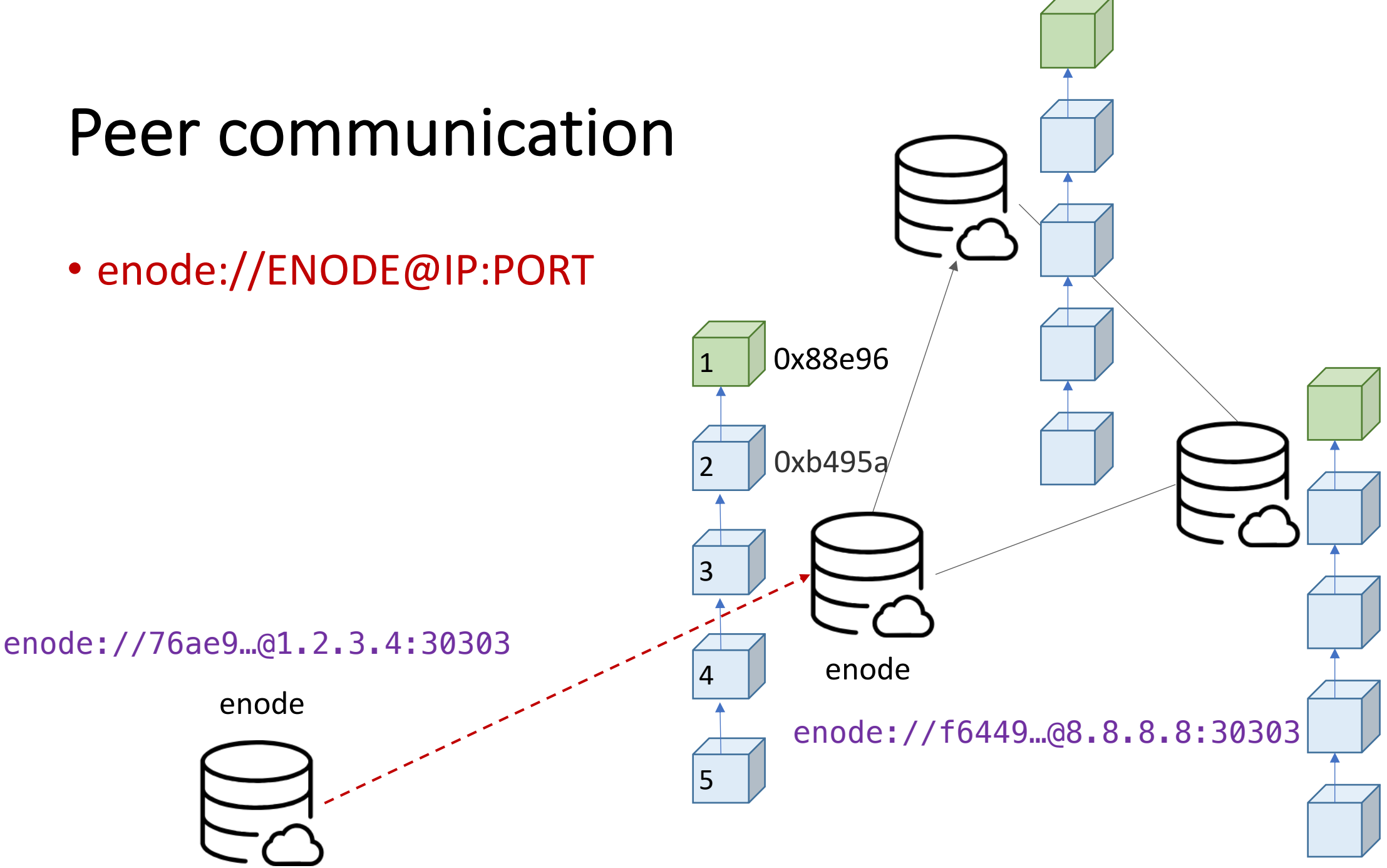
enode://**ENODE**@IP:PORT

```
enode://f6449e7acaef2a59c2bb269260a89f6782c58976abf94ea3f01f67e6693b795cb2ec1287678f
3ea615004f5993e78b0fe7e8739ffe2ac32a7fbdb5cbf62f81ef@8.8.8.8:30303
```


https://ethereum.stackexchange.com/questions/28970/how-to-produce-enode-from-node-key/28978

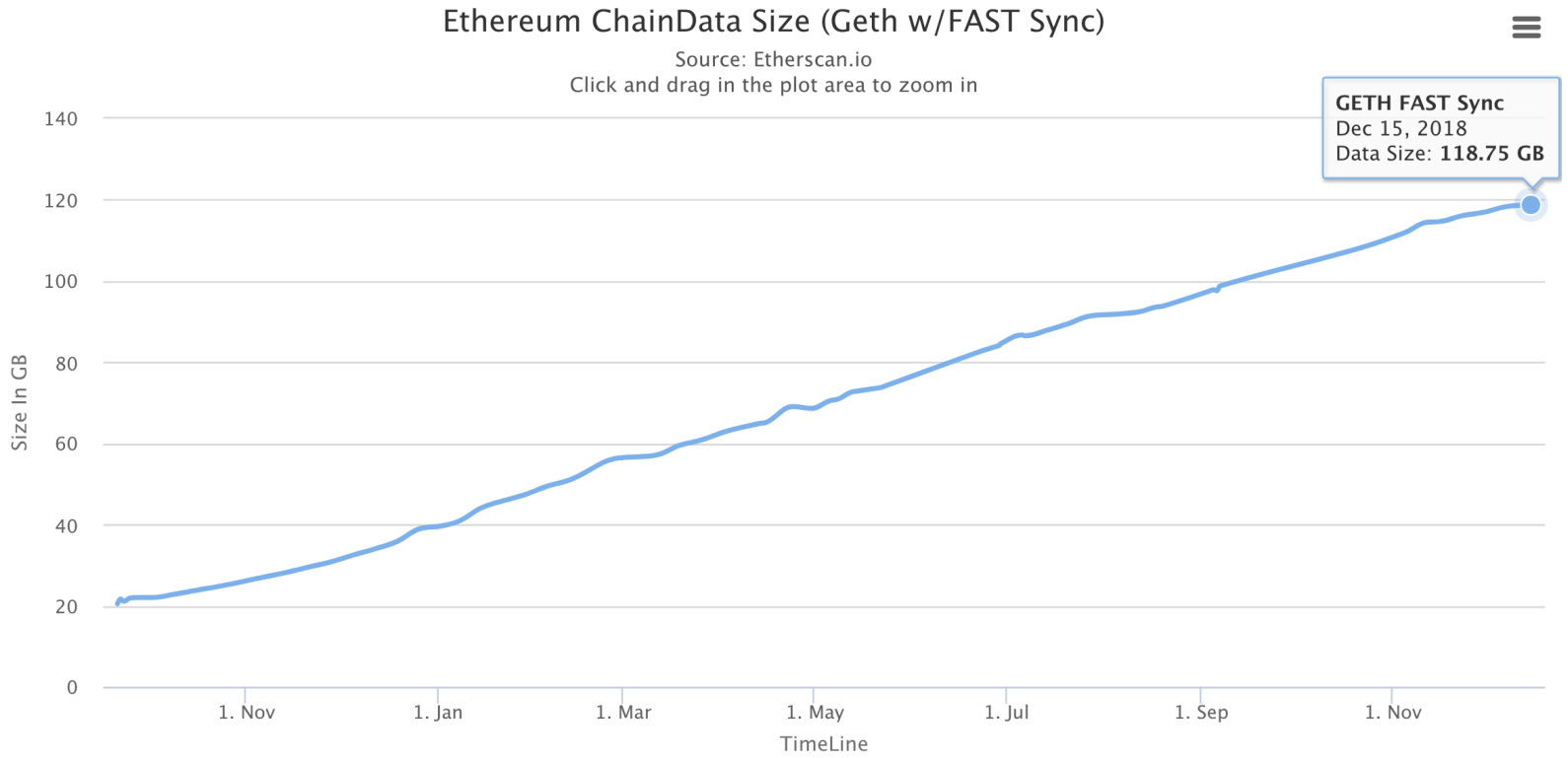
Peer communication

- enode://ENODE@IP:PORT

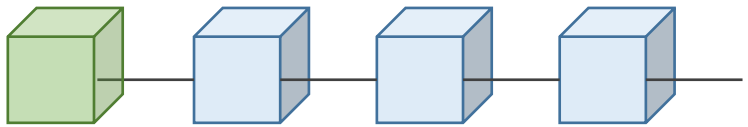


geth modes

- Disk: **SSD**
- --syncmode
  - **fast**
  - full
  - light
- --gcmode
  - **full**
  - archive



Get block headers and block bodies until current block - 64(*)



https://ethereum.stackexchange.com/questions/11297/what-is-geths-light-sync-and-why-is-it-so-fast

geth

```
$ geth --syncmode "full" --gcmode "archive" --rpc --rpcapi
"db,eth,net,web3,personal" --cache=1024 --rpcport 8545 --rpcaddr 0.0.0.0 -
-rpccorsdomain "*"

$ geth --testnet --rpc --rpcapi "db,eth,net,web3,personal" --cache=1024 --
rpcport 8545 --rpcaddr 0.0.0.0 --rpccorsdomain "*"
```

- Full + archive node
  - History
  - Present state
  - All historical states

Chain types

- mainnet
- rospten
- rinkeby
  - (Proof-of-Authority network)

- 0: Olympic, Ethereum public pre-release testnet
- 1: Frontier, Homestead, Metropolis, the Ethereum public main network
- 1: Classic, the (un)forked public Ethereum Classic main network, *chain ID 61*
- 1: Expanse, an alternative Ethereum implementation, *chain ID 2*
- 2: Morden, the public Ethereum testnet, now Ethereum Classic testnet
- 3: Ropsten, the public cross-client Ethereum testnet
- 4: Rinkeby, the public Geth PoA testnet

https://ethereum.stackexchange.com/questions/17051/how-to-select-a-network-id-or-is-there-a-list-of-network-ids

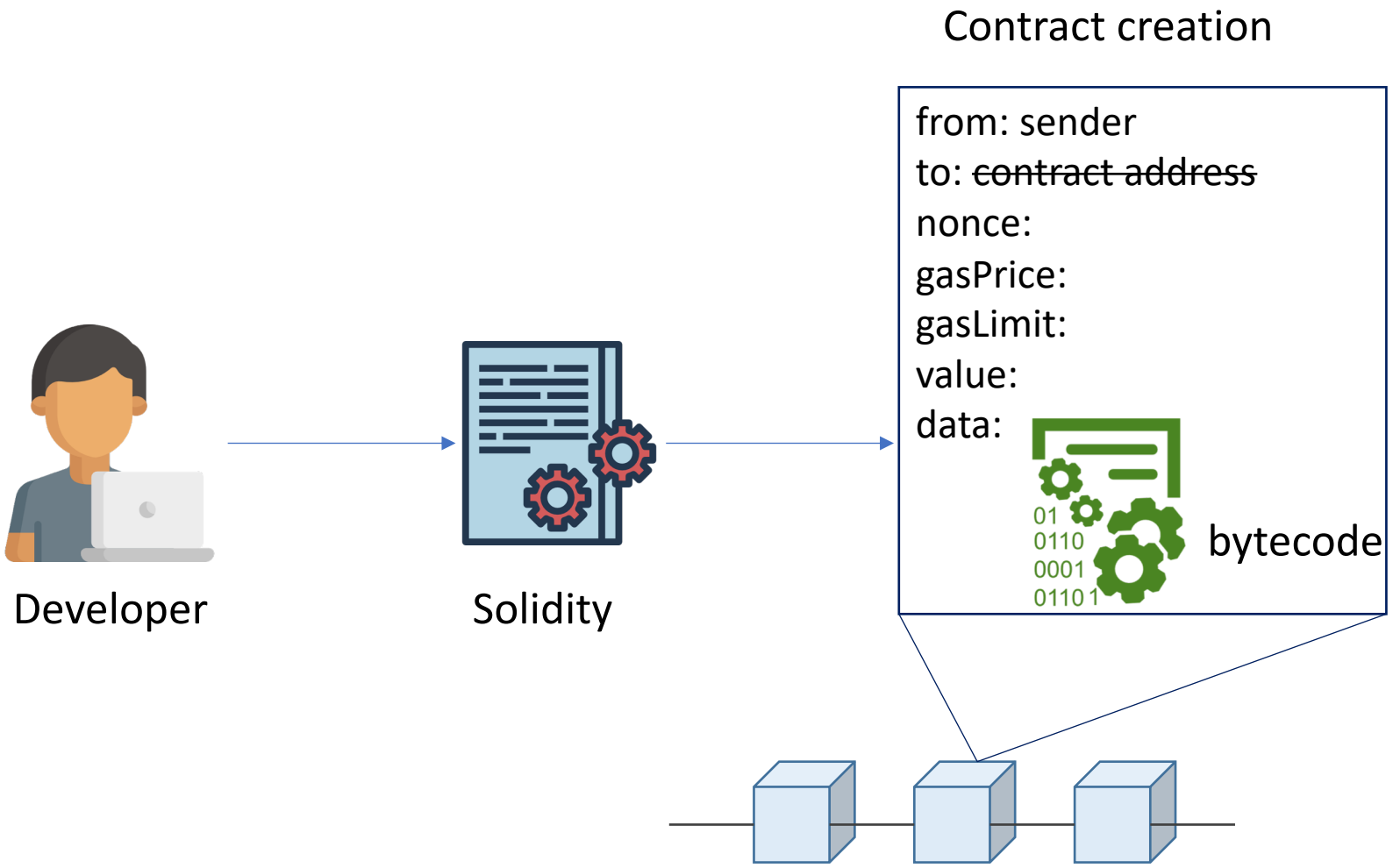
Comparison of the different TestNets

- **Ropsten**
  - PoW
  - Supported by geth and parity
  - Best reproduces the current production environment
  - Chaindata size 15 GB - Apr 2018
- **Kovan**
  - PoA (Immune to spam attacks)
  - Supported by parity only
  - Chaindata size 13 GB - Apr 2018
- **Rinkeby**
  - PoA (Immune to spam attacks)
  - Supported by geth only
  - Chaindata size 6 GB - Apr 2018
- **Sokol**
  - PoA (Immune to spam attacks)
  - Supported by parity only
  - Chaindata size 5gb - Jun 2018

testnet 資料會定期清空
千萬不要用於 production

https://ethereum.stackexchange.com/questions/27048/comparison-of-the-different-testnets

Contract development

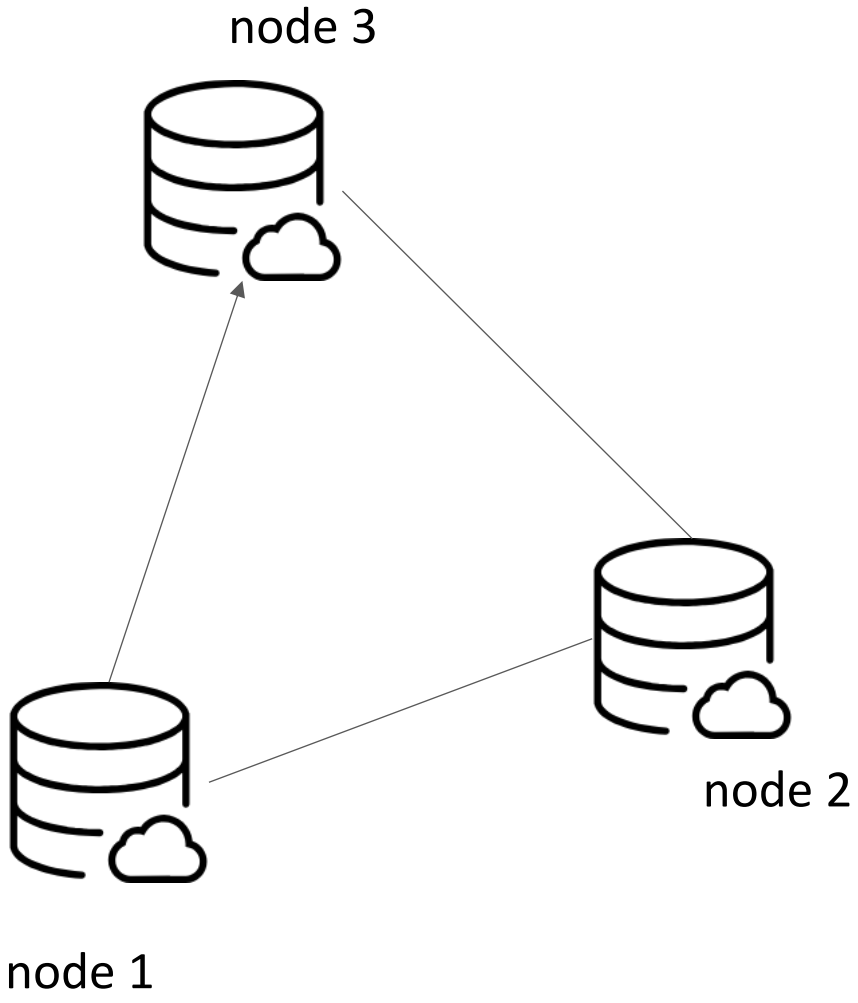


1. Solidity version
2. Contract vulnerability
3. Enough funds
4. Reasonable gas price
5. Node management
6. Contract maintenance
7. Data migration
8. Data confidentiality
9. Data privacy
10. External event
11. Optimized gas usage
12. Test

Private net

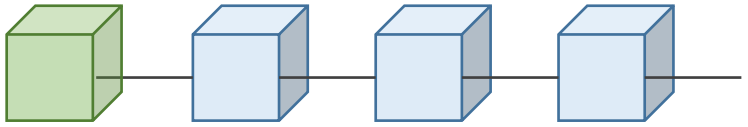
以太坊私有链

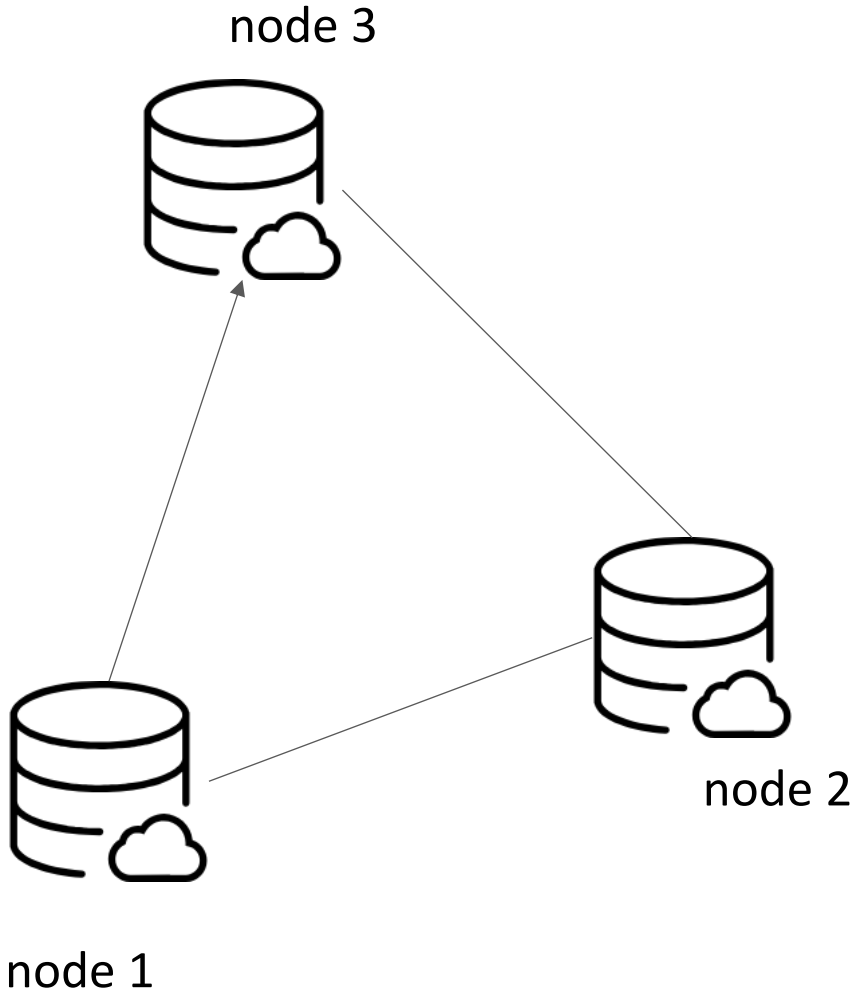
- 客製化的私有鏈
- 實驗
- 挖礦難度較易且可設定



環境設定

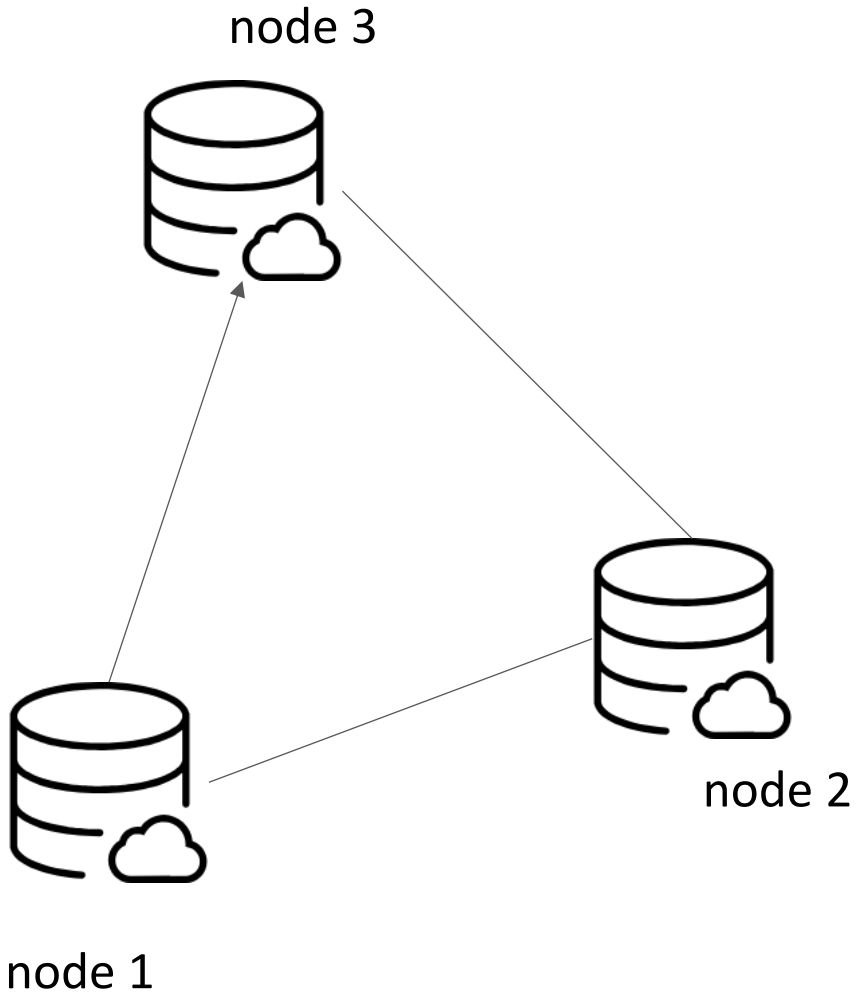
- 三台機器
- 創世區塊(genesis block)設定相同
- geth client





檔案目錄

```
.
├── genesis.json
├── node1
└── node2

2 directories, 1 file
```



genesis.json

```
{
    "alloc": {
        "0xb36fc7a57bbbc5ba62a51427acb6ad37bc8be4a0": {
            "balance": "1000000000000000000000"
        },
        "0x1635e5014dbcd3df2f21744eaa2a868ee69d25f2": {
            "balance": "1000000000000000000000000"
        }
    },
    "config": {
        "chainID": 5566,
        "homesteadBlock": 0,
        "eip155Block": 0,
        "eip158Block": 0
    },
    "nonce": "0x0000000000000000",
    "difficulty": "0x0400",
    "mixhash": "0x0000000000000000000000000000000000000000000000000000000000000000",
    "coinbase": "0x0000000000000000000000000000000000000000",
    "timestamp": "0x00",
    "parentHash": "0x0000000000000000000000000000000000000000000000000000000000000000",
    "extraData": "0x6e636375",
    "gasLimit": "0x7a121d"
}
```

初始化 node1

```
$ geth --datadir node1 init genesis.json
```

```
$ geth --datadir node1 init genesis.json
INFO [12-18|21:37:55.025] Maximum peer count                       ETH=25 LES=0 total=25
INFO [12-18|21:37:55.035] Allocated cache and file handles         database=/Users/changwu/nccu-course/nodes/node1/get
h/chaindata cache=16 handles=16
INFO [12-18|21:37:55.041] Writing custom genesis block
INFO [12-18|21:37:55.042] Persisted trie from memory database      nodes=3 size=414.00B time=384.218µs gcnodes=0 gcsiz
e=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [12-18|21:37:55.043] Successfully wrote genesis state         database=chaindata
            hash=d7b205…1e165a
INFO [12-18|21:37:55.043] Allocated cache and file handles         database=/Users/changwu/nccu-course/nodes/node1/get
h/lightchaindata cache=16 handles=16
INFO [12-18|21:37:55.047] Writing custom genesis block
INFO [12-18|21:37:55.047] Persisted trie from memory database      nodes=3 size=414.00B time=104.212µs gcnodes=0 gcsiz
e=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [12-18|21:37:55.048] Successfully wrote genesis state         database=lightchaindata
                hash=d7b205…1e165a
```

初始化 node2

```
$ geth --datadir node2 init genesis.json
```

```
$ geth --datadir node2 init genesis.json
INFO [12-18|21:39:41.963] Maximum peer count                       ETH=25 LES=0 total=25
INFO [12-18|21:39:41.974] Allocated cache and file handles         database=/Users/changwu/nccu-course/nodes/node2/get
h/chaindata cache=16 handles=16
INFO [12-18|21:39:41.977] Writing custom genesis block
INFO [12-18|21:39:41.978] Persisted trie from memory database      nodes=3 size=414.00B time=94.999µs gcnodes=0 gcsize
=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [12-18|21:39:41.978] Successfully wrote genesis state         database=chaindata
            hash=d7b205…1e165a
INFO [12-18|21:39:41.978] Allocated cache and file handles         database=/Users/changwu/nccu-course/nodes/node2/get
h/lightchaindata cache=16 handles=16
INFO [12-18|21:39:41.982] Writing custom genesis block
INFO [12-18|21:39:41.983] Persisted trie from memory database      nodes=3 size=414.00B time=100.576µs gcnodes=0 gcsiz
e=0.00B gctime=0s livenodes=1 livesize=0.00B
INFO [12-18|21:39:41.983] Successfully wrote genesis state         database=lightchaindata
                hash=d7b205…1e165a
```

建立帳號 node1

```
$ geth --datadir "node1" account new
```

```
$ geth --datadir "node1" account new

INFO [12-18|21:48:39.434] Maximum peer count                       ETH=25 LES=0 total=25
Your new account is locked with a password. Please give a password. Do not forget this password.
Passphrase:
Repeat passphrase:
Address: {3eaff2c7bd8bb391fc843fdc7eb55d02c4a5aa35}
```

建立帳號 node2

```
$ geth --datadir "node2" account new
```

```
$ geth --datadir "node2" account new
INFO [12-18|21:48:48.789] Maximum peer count                       ETH=25 LES=0 total=25
Your new account is locked with a password. Please give a password. Do not forget this password.
Passphrase:
Repeat passphrase:

Address: {55d07fd4da7b887817331c219219cc951301dfe1}
```

啟動 geth (node1)

```
$ geth --identity "node1" --datadir "node1" --rpc --rpcport 7788 --
rpccorsdomain "*" --port 30301 --nodiscover --rpcapi
"db,eth,net,web3,personal" --networkid 1234 console
```

identity	給予當前節點命名
port	P2P 通訊 port
rpc	啟用 rpc
rpcapi	允許 client 使用的 rpc module
networkid	網路 ID
console	命令列模式

啟動 geth (node2)

```
$ geth --identity "node2" --datadir "node2" --rpc --rpcport 5566 --
rpccorsdomain "*" --port 30302 --nodiscover --rpcapi
"db,eth,net,web3,personal" --networkid 1234 console
```

identity	給予當前節點命名
port	P2P 通訊 port
rpc	啟用 rpc
rpcapi	允許 client 使用的 rpc module
networkid	網路 ID
console	命令列模式

查看節點帳號

```
(node1)
> eth.coinbase
"0x3eaff2c7bd8bb391fc843fdc7eb55d02c4a5aa35”

(node2)
> eth.coinbase
"0x55d07fd4da7b887817331c219219cc951301dfe1"
```

查看連接的節點

```
> admin.peers
[]
```

因為目前沒有任何連接數，所以為空

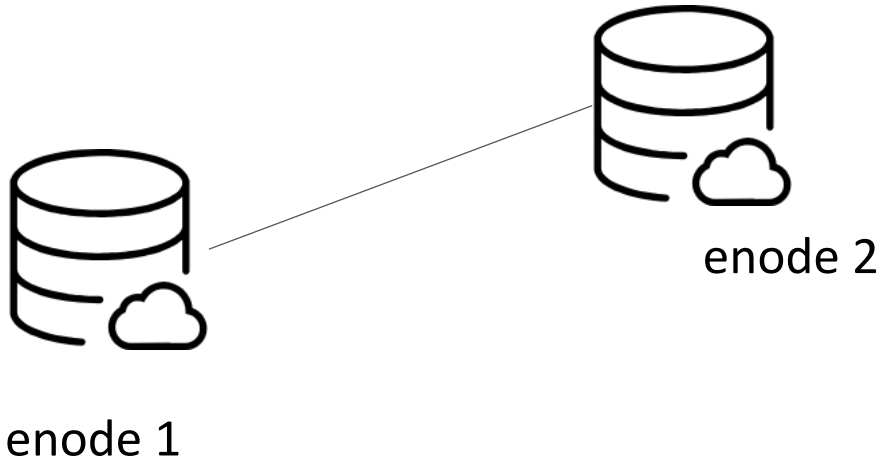
取得 node2 的 enode ID

```
(node2)
> admin.nodeInfo.enode
"enode://c7b938c499881ac1d0dfd29cb37a0c67f93e6811b6a0b0f0093fd9d81e5fd08ed
56670f689ea79e9af22f7c17da8e8323234dc26b23099e9fe2ad1b556af116a@[::]:30302
?discport=0"
```

```
enode://c7b938c499881ac1d0dfd29cb37a0c67f93e6811b6a0b0f0093fd9d81e5
fd08ed56670f689ea79e9af22f7c17da8e8323234dc26b23099e9fe2ad1b556af11
6a@127.0.0.1:30302
```

Connect node 1 with node 2

```
>
admin.addPeer("enode://c7b938c499881ac1d0dfd29cb37a0c67f93e6811b6a0b0f0093
fd9d81e5fd08ed56670f689ea79e9af22f7c17da8e8323234dc26b23099e9fe2ad1b556af1
16a@127.0.0.1:30302")
true
```



查看連接的節點

```
(node1)
> admin.peers
[{
    caps: ["eth/63"],
    id:
"c7b938c499881ac1d0dfd29cb37a0c67f93e6811b6a0b0f0093fd9d81e5fd08ed56670f689ea79e9af22f7c17da8e832
3234dc26b23099e9fe2ad1b556af116a",
    name: "Geth/node2/v1.8.16-unstable-62e94895/darwin-amd64/go1.11",
    network: {
        inbound: false,
        localAddress: "127.0.0.1:51246",
        remoteAddress: "127.0.0.1:30302",
        static: true,
        trusted: false
    },
    protocols: {
        eth: {
            difficulty: 17179869184,
            head: "0xd4e56740f876aef8c010b86a40d5f56745a118d0906a34e69aec8c0db1cb8fa3",
            version: 63
        }
    }
}]
>
```

查看區塊數目

```
(node1)
> eth.blockNumber
0
```

查看帳戶餘額

```
(node1)
> web3.fromWei(eth.getBalance(eth.accounts[0]), "ether")
0
```

```
(node1)
> web3.fromWei(eth.getBalance("0xb36fc7a57bbbc5ba62a51427acb6ad37bc8be4a0"), "ether")
10000
```

啟動挖礦

```
> miner.start()
INFO [12-18|22:43:41.865] Updated mining threads threads=4
INFO [12-18|22:43:41.865] Transaction pool price threshold updated price=1000000000
null
> INFO [12-18|22:43:41.866] Commit new mining work number=1 sealhash=27e1aa…6f5a0d
uncles=0 txs=0 gas=0 fees=0 elapsed=677.102µs
INFO [12-18|22:43:42.453] Successfully sealed new block number=1 sealhash=27e1aa…6f5a0d
hash=f36304…0090fc elapsed=587.048ms
INFO [12-18|22:43:42.453] 🔨 mined potential block number=1 hash=f36304…0090fc
INFO [12-18|22:43:42.453] Commit new mining work number=2 sealhash=46eed2…58fc64 uncles=0
txs=0 gas=0 fees=0 elapsed=242.039µs
INFO [12-18|22:43:43.557] Successfully sealed new block number=2 sealhash=46eed2…58fc64
hash=8b2f4f…f39249 elapsed=1.104s
```

attach to node1

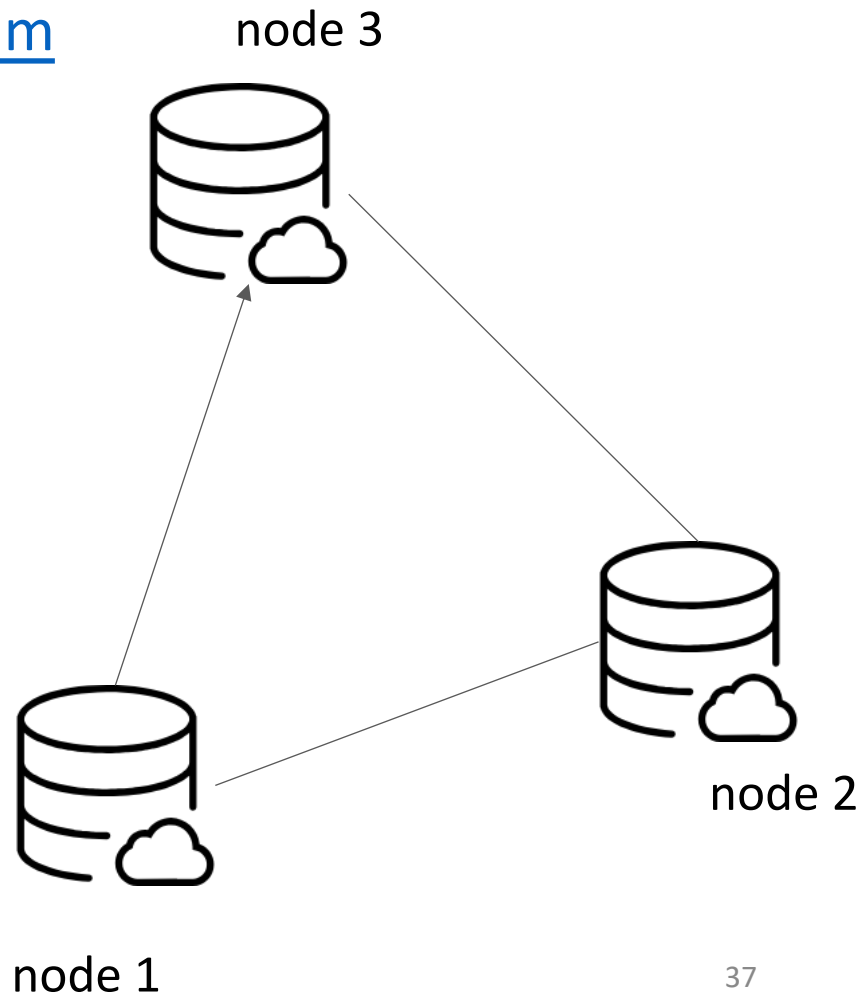
```
$ geth attach http://127.0.0.1:5566
Welcome to the Geth JavaScript console!

instance: Geth/node2/v1.8.16-unstable-62e94895/darwin-amd64/go1.11
coinbase: 0x55d07fd4da7b887817331c219219cc951301dfe1
at block: 38 (Tue, 18 Dec 2018 22:44:25 CST)
modules: eth:1.0 net:1.0 personal:1.0 rpc:1.0 web3:1.0

>
```

attach to node2

```
$ geth attach http://127.0.0.1:7788
Welcome to the Geth JavaScript console!

instance: Geth/node1/v1.8.16-unstable-62e94895/darwin-amd64/go1.11
coinbase: 0x3eaff2c7bd8bb391fc843fdc7eb55d02c4a5aa35
at block: 29 (Tue, 18 Dec 2018 22:44:15 CST)
modules: eth:1.0 net:1.0 personal:1.0 rpc:1.0 web3:1.0

>
```

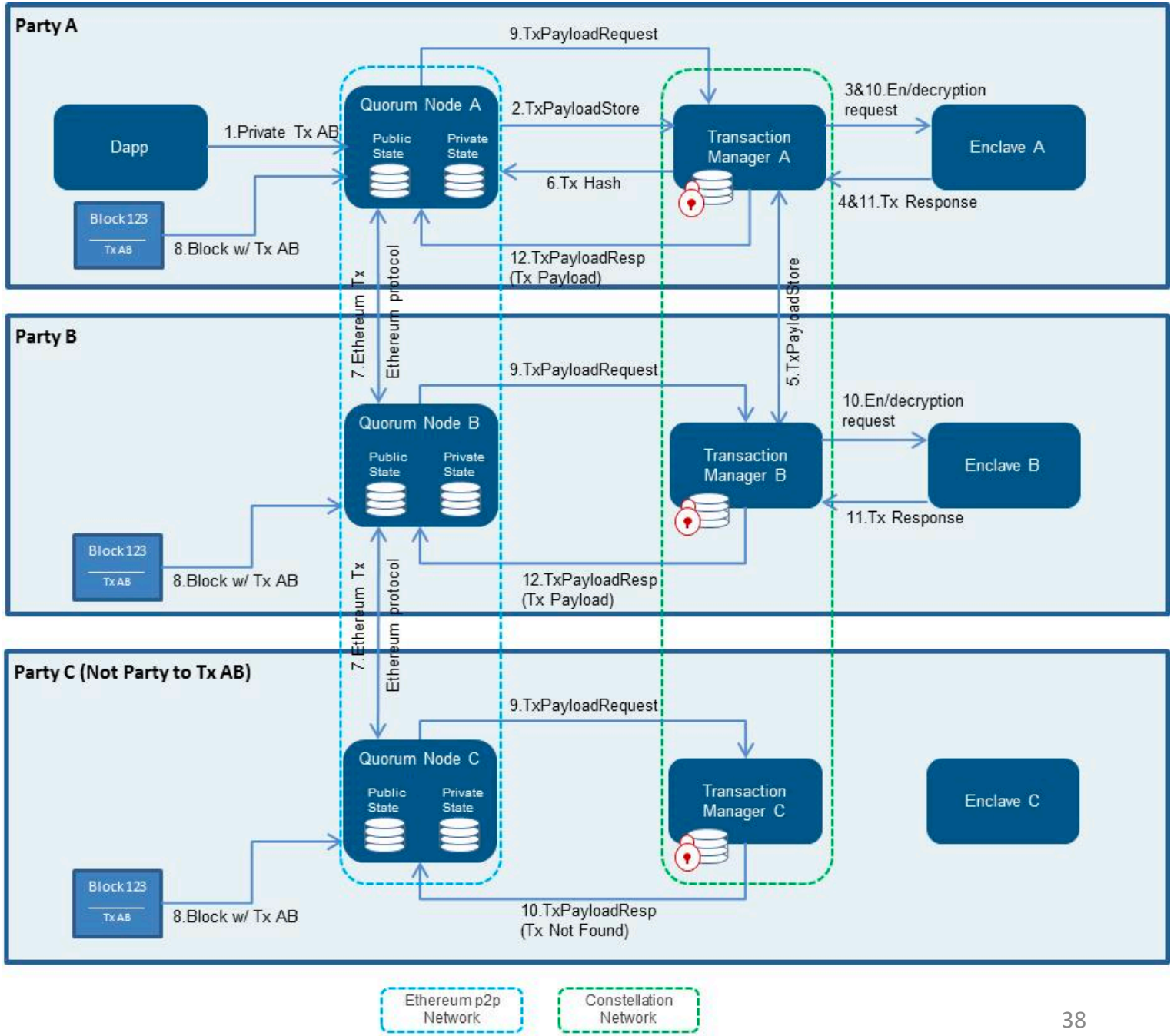
Clique

- https://ethfans.org/posts/Clique-Consensus-Algorithm
- 不同共識演算法，非 PoW
- 輪流出塊
- 認證節點 (block proposer)
- 非認證節點 (only sync block)



quorum

- 企業私有鏈



https://github.com/jpmorganchase/quorum

References

- https://feelncut.com/2018/03/05/102.html
- https://arvanaghi.com/blog/how-to-set-up-a-private-ethereum-blockchain-using-geth/
- https://arvanaghi.com/blog/explaining-the-genesis-block-in-ethereum/

devcon4 workshop

- http://www.cryptokube.io/
- Slide
  - http://www.cryptokube.io/Devcon4-ArchitectingWithEthereum.pdf
- Exercise
  - https://github.com/CryptoKube-io/devcon4-workshop#exercises

Solidity

Smart contract language

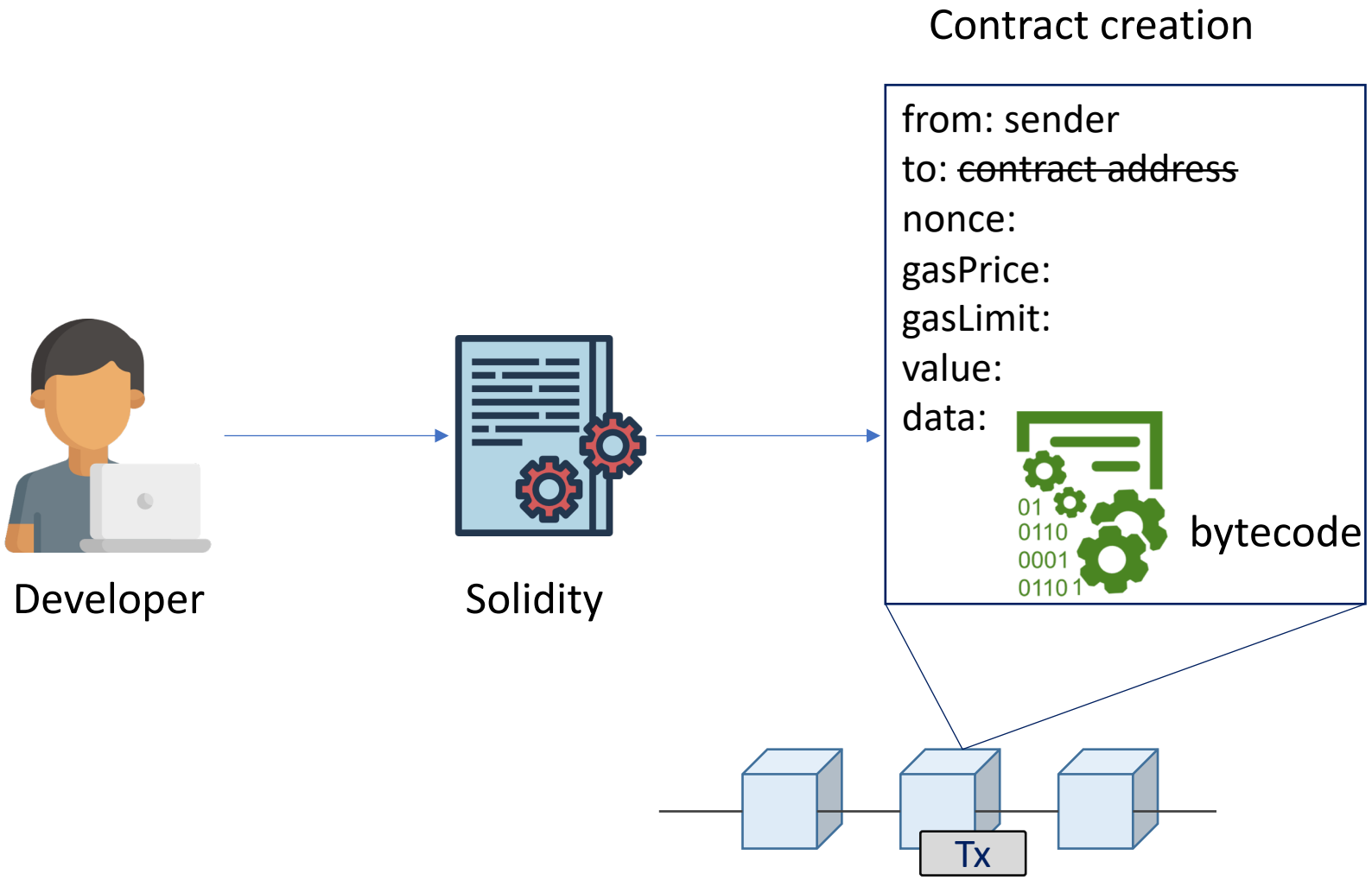
- Solidity
  - https://github.com/ethereum/solidity
- Vyper
  - https://github.com/ethereum/vyper

What is solidity

- An object-oriented, high-level language for implementing smart contracts.
- Smart contracts are programs which govern the behaviour of accounts within the Ethereum state.

https://solidity.readthedocs.io/en/v0.5.11/index.html

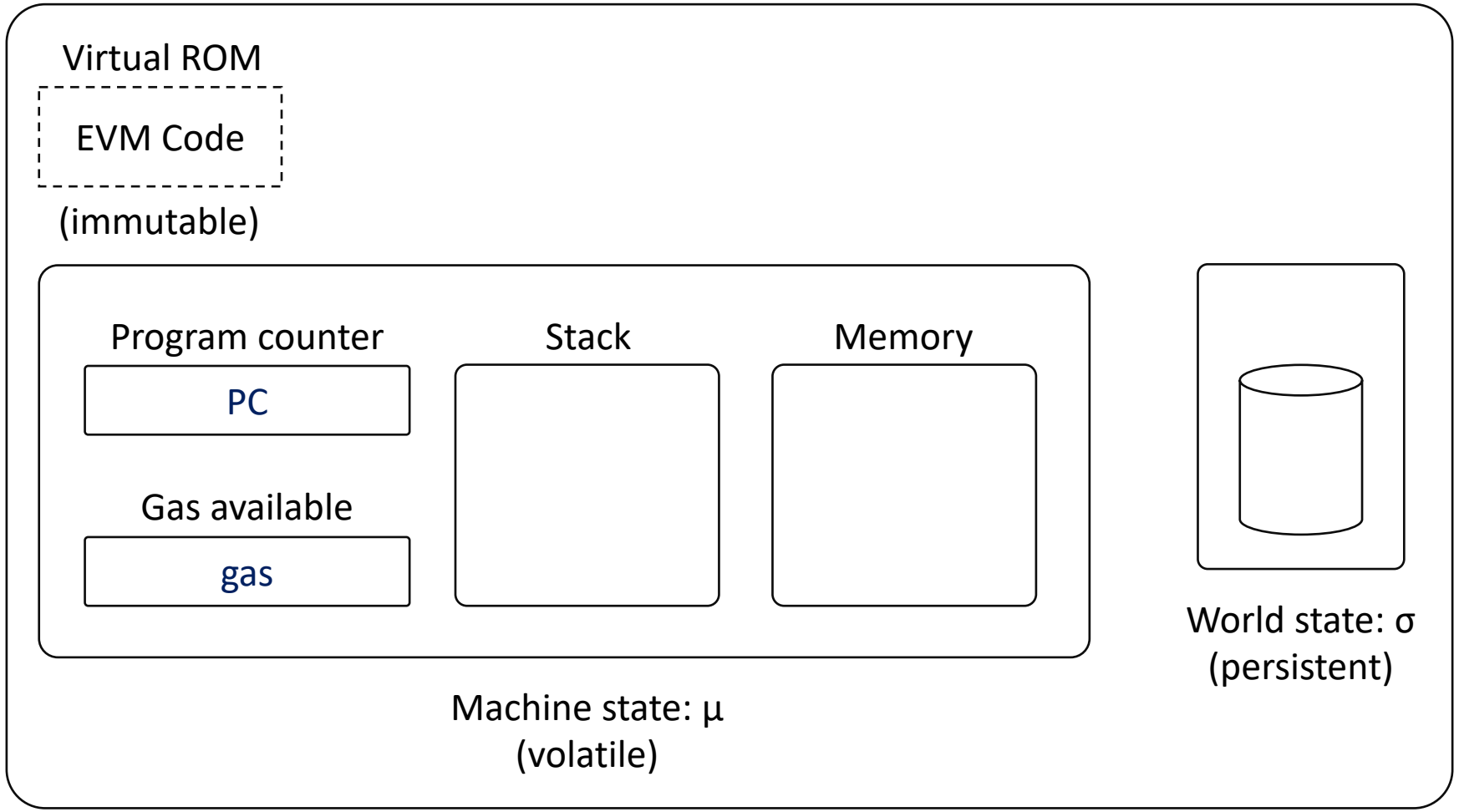
Contract development



1. Solidity version
2. Contract vulnerability
3. Enough funds
4. Reasonable gas price
5. Node management
6. Contract maintenance
7. Data migration
8. Data confidentiality
9. Data privacy
10. External event
11. Optimized gas usage
12. Test

Stack-based architecture

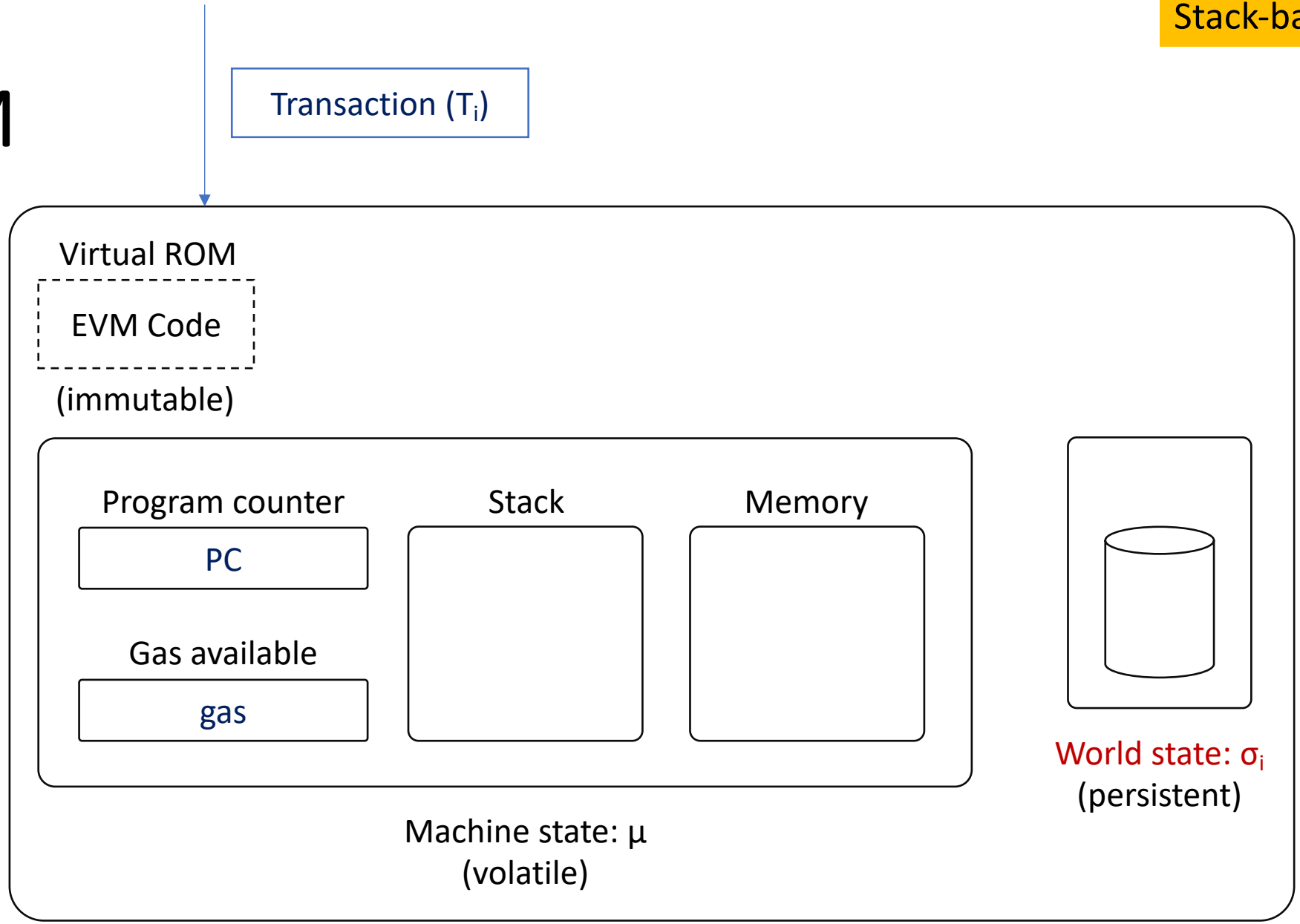
EVM



Ethereum Virtual Machine (EVM)

Stack-based architecture

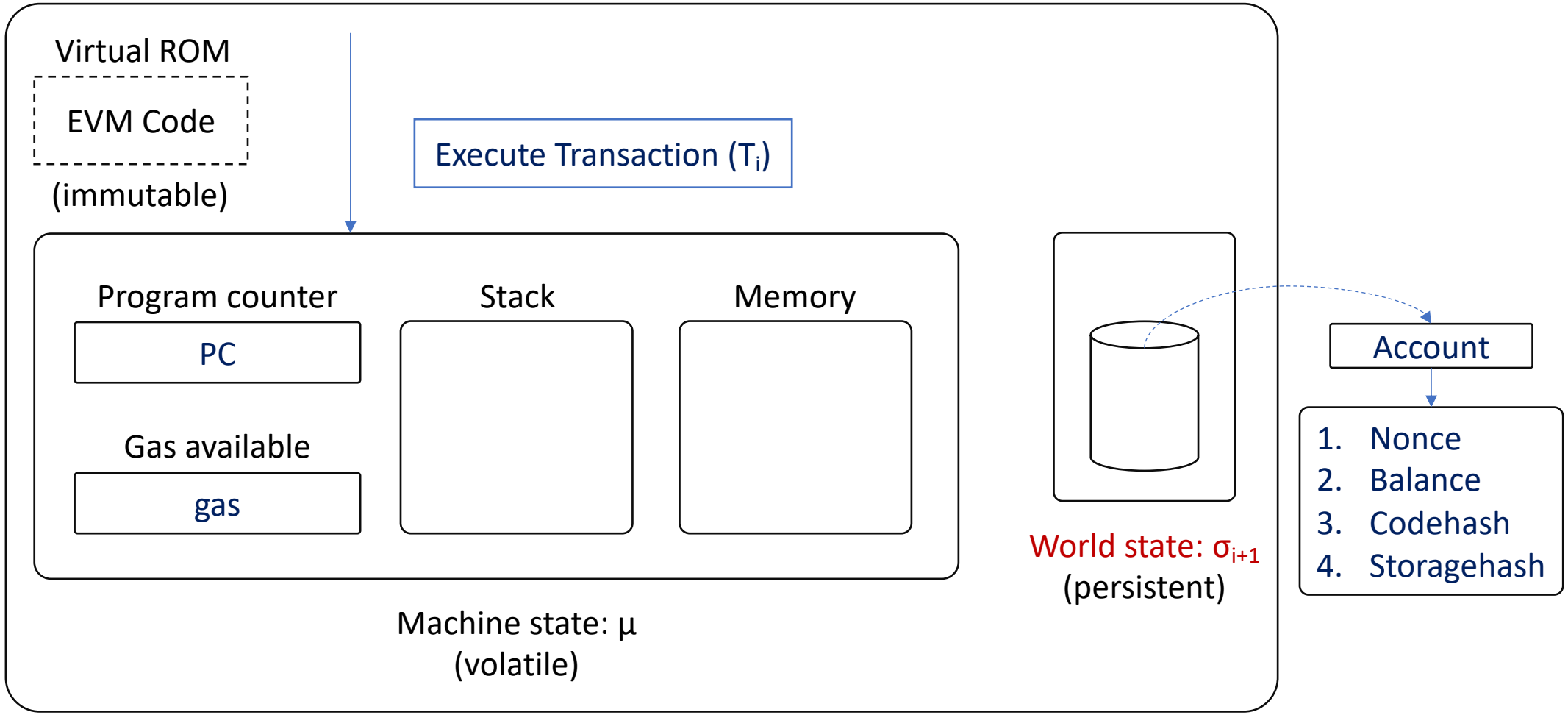
EVM



Ethereum Virtual Machine (EVM)

Stack-based architecture

EVM



Ethereum Virtual Machine (EVM)

Solidity code

```
pragma solidity >=0.4.0 <0.7.0;

contract HelloWorld {
    function helloWorld() external pure returns (string memory) {
        return "Hello, World!";
    }
}
```

```
pragma solidity >=0.4.0 <0.7.0;

contract SimpleStorage {
    uint storedData;

    function set(uint x) public {
        storedData = x;
    }

    function get() public view returns (uint) {
        return storedData;
    }
}
```

EVM code

```
.code
    PUSH1 0x80
    PUSH1 0x40
    MSTORE
    CALLVALUE
    DUP1
    ISZERO
    PUSH2 0x10
    JUMPI
    PUSH1 0x0
    DUP1
    REVERT
```

Assembly view

```
60806040523480156100105
7600080fd5b5060ca8061001
f6000396000f3fe6080604052
600436106042577c01000000
00000000000000000000000
00000000000000000000000
00006000350463 60fe47b181
1460475780636d4ce63c1460
6f575b600080fd5b34801560
52576000…
```

Bytecode view

Smart contract

Greeter contract

```
pragma solidity ^0.4.15;

contract mortal {
   /* Define variable owner of the type address*/
   address owner;

   /* this function is executed at initialization and sets the owner of the contract */
   function mortal() { owner = msg.sender; }

   /* Function to recover the funds on the contract */
   function kill() { if (msg.sender == owner) suicide(owner); }
}

contract greeter is mortal {
   /* define variable greeting of the type string */
   string greeting;
   /* this runs when the contract is executed */
   function greeter(string _greeting) public {
       greeting = _greeting;
   }

   /* main function */
   function greet() constant returns (string) {
      return greeting;
   }
}
```

https://remix.ethereum.org/#optimize=false&version=soljson-v0.4.24+commit.e67f0147.js

Remix IDE

https://remix.ethereum.org/#optimize=false&version=soljson-v0.4.24+commit.e67f0147.js

Web3 deploy code

```
var _greeting = "nccu";
var greeterContract = web3.eth.contract([{"constant":false,"inputs":[],"name":"kill","outputs":[],"payable":false,"type":"function","st
ateMutability":"nonpayable"},{"constant":true,"inputs":[],"name":"greet","outputs":[{"name":"","type":"string"}],"payable":false,"type"
:"function","stateMutability":"view"},{"inputs":[{"name":"_greeting","type":"string"}],"payable":false,"type":"constructor","stateMutab
ility":"nonpayable"}]);
var greeter = greeterContract.new(
   _greeting,
   {
     from: web3.eth.accounts[0],
     data: '0x606060405234156100f57600080fd5b604051610323380380610323833981016040528080519091019050 5b5b60008054600160a060020a031916336
00160a060020a03161790555b600181805161005b929160200190610063565b505b50610103565b828054600181600116156101000203166002900490600052602060002
090601f016020900481019282601f106100a457805160ff1916838001178555610d1565b828001600101855582156100d1579182015b828111156100d157825182559
160200191906001019061 00b6565b5b506100de9291506100e2565b5090565b61010091905b808211156100de576000815560010161 00e8565b5090565b90565b610211
806101126000396000f300606060405263ffffffff7c0100000000000000000000000000000000000000000000000000000000600035041661 6341c0e1b58114610048578
063cfae32171461005d575b600080fd5b341561005357600080fd5b61005b6100e8565b005b341561006857600080fd5b61007061012a565b604051602080825281908
101838181518152602001915080519060200190808383600005b838110156100ad578082015181840152 5b602001610094565b505050509050908101906 01f1680156100d
a57808203805160018360200361010 00a031916815260200191505b509250505060405180910390f35b6000543373ffffffffffffffffffffffffffffffffffffffff90
81169116141561012757600054 73ffffffffffffffffffffffffffffffffffffffff16ff5b5b565b6101326101d3565b60018054600181600116156101000203166002 9
00480601f016020809104026020016040519081016040528092919081815260200182805460018160011615610100020316600290048015610 1c85780601f1061019d57
6101008083540402835291602001916101c8565b820191906000526020600020905b81548152906001019060200180831161 01ab5782900360 1f168201915b505050505
090505b90565b602060405190810160405260008152905600a165627a7a723058203c4b1b46cebbe574445df873b5a0acd2da544c3aac3fb786faef412bae1642770029
',
     gas: '4700000'
   }, function (e, contract){
    console.log(e, contract);
    if (typeof contract.address !== 'undefined') {
         console.log('Contract mined! address: ' + contract.address + ' transactionHash: ' + contract.transactionHash);
    }
 })
```

Deploy and get mined

1. geth attach /var/folders/x8/gs3w34_n4wx6cmrp35b_q4r40000gn/T/geth.ipc (geth)

geth (geth) ⌘1 | geth (geth) ⌘2

```
820380516001836020036101000a031916815260200191505b509250505060405180910390f35b60008090549061010000a900473ffffffffffffffffff
ffffffffffffffffffffffff1673ffffffffffffffffffffffffffffffffffffffffffffffff163373ffffffffffffffffffffffffffffffffffffffff16141561
0187576000809054906101000a900473ffffffffffffffffffffffffffffffffffffffff1673ffffffffffffffffffffffffffffffffffffffff16ff5
b565b606060018054600181600116156101000203166002900480601f016020809104026020016040519081016040528092919081815260200182805460
60018160011615610100020316600290048015610221578060 1f106101f657610100808354040283529160200191610221565b8201919060005260206
00020905b8154815290600101906020018083116102045782900 3601f168201915b5050505050905090560a165627a7a723058205aa75599fe31344a
4b01e86e9fa09a234ec5fd56ea5d31c48ba38b138cc4ab340029',
......      gas: '4700000'
......    }, function (e, contract){
......     console.log(e, contract);
......     if (typeof contract.address !== 'undefined') {
.........          console.log('Contract mined! address: ' + contract.address + ' transactionHash: ' + contract.transacti
onHash);
.........     }
......  })
null [object Object]
undefined
> null [object Object]
Contract mined! address: 0xfbd8e2427c9dcfe4df275d24615d850e8942900d transactionHash: 0xa570a60431fad2e1b0f8be357764e92afb
5dfcb6a471133e2885fbbd09b6c04a

> _
```

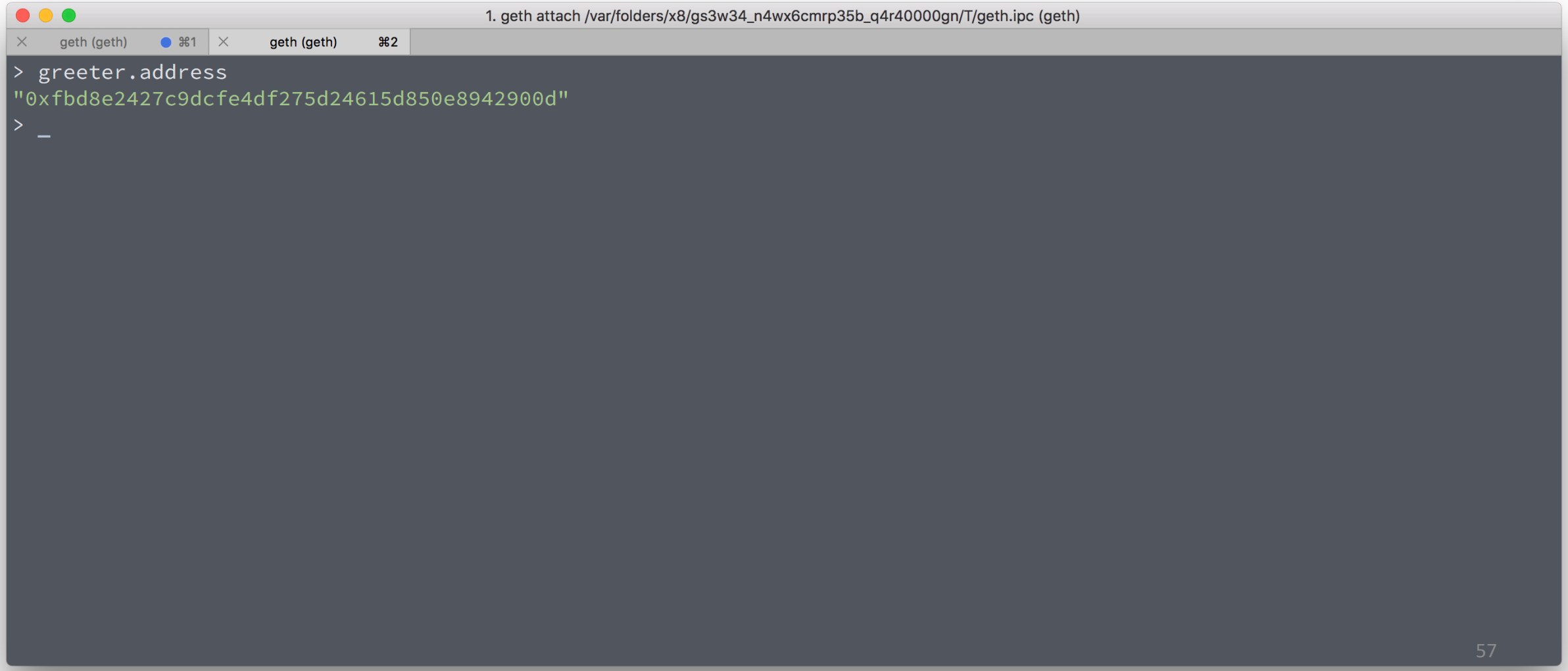
Contract ABI

1. geth attach /var/folders/x8/gs3w34_n4wx6cmrp35b_q4r40000gn/T/geth.ipc (geth)

geth (geth) ⌘1 | geth (geth) ⌘2

```
> greeter.abi
[{
    constant: false,
    inputs: [],
    name: "kill",
    outputs: [],
    payable: false,
    stateMutability: "nonpayable",
    type: "function"
}, {
    constant: true,
    inputs: [],
    name: "greet",
    outputs: [{
        name: "",
        type: "string"
    }],
    payable: false,
    stateMutability: "view",
    type: "function"
}, {
    inputs: [{
```

Contract address



Greeter contract

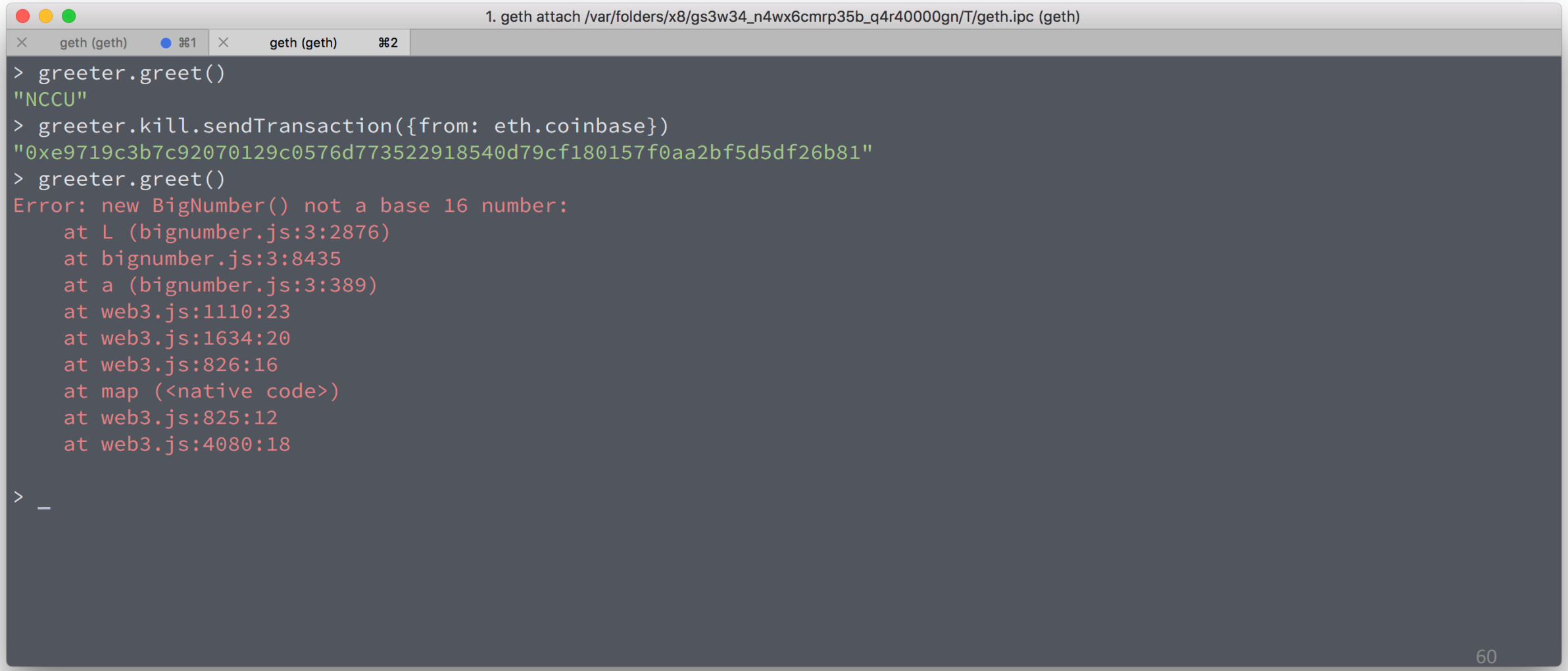
1. 有幾個合約函數?
2. 那個要花 gas 那個不用?

```
contract mortal {
    /* Define variable owner of the type address*/
    address owner;

    /* this function is executed at initialization and sets the owner of the contract */
    function mortal() { owner = msg.sender; }

    /* Function to recover the funds on the contract */
    function kill() { if (msg.sender == owner) suicide(owner); }
}

contract greeter is mortal {
    /* define variable greeting of the type string */
    string greeting;
    /* this runs when the contract is executed */
    function greeter(string _greeting) public {
        greeting = _greeting;
    }

    /* main function */
    function greet() constant returns (string) {
        return greeting;
    }
}
```

Greeter contract

```
contract mortal {
   /* Define variable owner of the type address*/
   address owner;

   /* this function is executed at initialization and sets the owner of the contract */
   function mortal() { owner = msg.sender; }

   /* Function to recover the funds on the contract */
   function kill() { if (msg.sender == owner) suicide(owner); }
}

contract greeter is mortal {
   /* define variable greeting of the type string */
   string greeting;
   /* this runs when the contract is executed */
   function greeter(string _greeting) public {
       greeting = _greeting;
   }

   /* main function */
   function greet() constant returns (string) {
      return greeting;
   }
}
```

Call contract function



v0.5.12

```
pragma solidity ^0.5.12;

contract mortal {
    address payable owner;

    constructor() public { owner = msg.sender; }

    function kill() public {
        require(msg.sender == owner, "Only owner can call this function.");
        selfdestruct(owner);
    }
}

contract greeter is mortal {
   string greeting;

   constructor(string memory _greeting) public { greeting = _greeting; }

   function greet() public view returns (string memory) { return greeting; }

   event show(string greeting);

   function set(string memory _greeting) public {
       greeting = _greeting;
       emit show(greeting);
   }
}
```

https://solidity.readthedocs.io/en/v0.5.3/layout-of-source-files.html#pragmas

v0.5.12

Enable certain compiler features or checks

```
pragma solidity ^0.5.12;

contract mortal {
    address payable owner;

    constructor() public { owner = msg.sender; }

    function kill() public {
        require(msg.sender == owner, "Only owner can call this function.");
        selfdestruct(owner);
    }
}

contract greeter is mortal {
   string greeting;

   constructor(string memory _greeting) public { greeting = _greeting; }

   function greet() public view returns (string memory) { return greeting; }

   event show(string greeting);

   function set(string memory _greeting) public {
       greeting = _greeting;
       emit show(greeting);
   }
}
```

https://solidity.readthedocs.io/en/v0.5.3/structure-of-a-contract.html#structure-of-a-contract

v0.5.12

```
pragma solidity >=0.4.0 <0.6.0;

contract HelloWorld {
    constructor() public {
        // ...
    }

    function bid() public payable { // Function
        // ...
    }
}
```

Structure of a contract

```
pragma solidity ^0.5.12;

contract mortal {
    address payable owner;

    constructor() public { owner = msg.sender; }

    function kill() public {
        require(msg.sender == owner, "Only owner can call this function.");
        selfdestruct(owner);
    }
}

contract greeter is mortal {
   string greeting;

   constructor(string memory _greeting) public { greeting = _greeting; }

   function greet() public view returns (string memory) { return greeting; }

   event show(string greeting);

   function set(string memory _greeting) public {
       greeting = _greeting;
       emit show(greeting);
   }
}
```

https://solidity.readthedocs.io/en/v0.5.3/contracts.html#inheritance

v0.5.12

```
pragma solidity ^0.5.12;

contract mortal {
    address payable owner;

    constructor() public { owner = msg.sender; }

    function kill() public {
        require(msg.sender == owner, "Only owner can call this function.");
        selfdestruct(owner);
    }
}

contract greeter is mortal {
   string greeting;

   constructor(string memory _greeting) public { greeting = _greeting; }

   function greet() public view returns (string memory) { return greeting; }

   event show(string greeting);

   function set(string memory _greeting) public {
       greeting = _greeting;
       emit show(greeting);
   }
}
```

Inheritance

https://solidity.readthedocs.io/en/v0.5.3/contracts.html#constructors

v0.5.12

Constructor

```
pragma solidity ^0.5.12;

contract mortal {
    address payable owner;

    constructor() public { owner = msg.sender; }

    function kill() public {
        require(msg.sender == owner, "Only owner can call this function.");
        selfdestruct(owner);
    }
}

contract greeter is mortal {
   string greeting;

   constructor(string memory _greeting) public { greeting = _greeting; }

   function greet() public view returns (string memory) { return greeting; }

   event show(string greeting);

   function set(string memory _greeting) public {
       greeting = _greeting;
       emit show(greeting);
   }
}
```

https://solidity.readthedocs.io/en/v0.5.3/structure-of-a-contract.html#state-variables

v0.5.12

address payable: Same as address, but with the additional members transfer and send.

State variables

```
pragma solidity ^0.5.12;

contract mortal {
    address payable owner;

    constructor() public { owner = msg.sender; }

    function kill() public {
        require(msg.sender == owner, "Only owner can call this function.");
        selfdestruct(owner);
    }
}

contract greeter is mortal {
   string greeting;

   constructor(string memory _greeting) public { greeting = _greeting; }

   function greet() public view returns (string memory) { return greeting; }

   event show(string greeting);

   function set(string memory _greeting) public {
       greeting = _greeting;
       emit show(greeting);
   }
}
```

https://solidity.readthedocs.io/en/v0.5.3/structure-of-a-contract.html#functions

v0.5.12

function (<parameter types>) {**internal**|**external**} [**pure**|**view**|**payable**] [**returns** (<**return** types>)]

```
pragma solidity ^0.5.12;

contract mortal {
    address payable owner;

    constructor() public { owner = msg.sender; }

    function kill() public {
        require(msg.sender == owner, "Only owner can call this function.");
        selfdestruct(owner);
    }
}

contract greeter is mortal {
   string greeting;

   constructor(string memory _greeting) public { greeting = _greeting; }

   function greet() public view returns (string memory) { return greeting; }

   event show(string greeting);

   function set(string memory _greeting) public {
       greeting = _greeting;
       emit show(greeting);
   }
}
```

Function types

https://solidity.readthedocs.io/en/v0.5.3/structure-of-a-contract.html#events

v0.5.12

```
pragma solidity ^0.5.12;

contract mortal {
    address payable owner;

    constructor() public { owner = msg.sender; }

    function kill() public {
        require(msg.sender == owner, "Only owner can call this function.");
        selfdestruct(owner);
    }
}

contract greeter is mortal {
   string greeting;

   constructor(string memory _greeting) public { greeting = _greeting; }

   function greet() public view returns (string memory) { return greeting; }

   event show(string greeting);

   function set(string memory _greeting) public {
       greeting = _greeting;
       emit show(greeting);
   }
}
```

Event

https://solidity.readthedocs.io/en/v0.5.3/structure-of-a-contract.html#function-modifiers

v0.5.12

Function Modifiers

```
pragma solidity >=0.4.22 <0.6.0;

contract Purchase {
    address public seller;

    modifier onlySeller() { // Modifier
        require(
            msg.sender == seller,
            "Only seller can call this."
        );
        _;
    }

    function abort() public view onlySeller { // Modifier usage
        // ...
    }
}
```

https://solidity.readthedocs.io/en/v0.5.3/structure-of-a-contract.html#struct-types

v0.5.12

```
pragma solidity >=0.4.0 <0.6.0;

contract Ballot {
    struct Voter { // Struct
        uint weight;
        bool voted;
        address delegate;
        uint vote;
    }
}
```

Struct Types

https://solidity.readthedocs.io/en/v0.5.3/structure-of-a-contract.html#enum-types

v0.5.12

```
pragma solidity >=0.4.0 <0.6.0;

contract Purchase {
    enum State { Created, Locked, Inactive } // Enum
}
```

Enum Types

https://solidity.readthedocs.io/en/v0.5.3/types.html#mapping-types

v0.5.12

```
pragma solidity >=0.4.0 <0.6.0;

contract MappingExample {
    mapping(address => uint) public balances;

    function update(uint newBalance) public {
        balances[msg.sender] = newBalance;
    }
}

contract MappingUser {
    function f() public returns (uint) {
        MappingExample m = new MappingExample();
        m.update(100);
        return m.balances(address(this));
    }
}
```

Mapping Types

https://solidity-by-example.org/loop/

v0.5.12

```
pragma solidity >=0.4.0 <0.6.0;

contract Loop {
    uint public count;

    // This is a demonstration of transaction gas limit.
    // Try:
    // Set the gas limit to 100000 and loop(100).
    // It should throw an error after spending all the gas.
    function loop(uint n) public returns (uint) {
        for (uint i = 0; i < n; i++) {
            count++;
        }

        return count;
    }
}
```

Loop

https://solidity-by-example.org/view-and-pure-functions/

v0.5.12

```
pragma solidity >=0.4.0 <0.6.0;

contract ViewAndPure {
    uint public x = 1;

    // Promise not to modify the state.
    function addToX(uint y) public view returns (uint) {
        return x + y;
    }

    // Promise not to modify or read from the state.
    function add(uint i, uint j) public pure returns (uint) {
        return i + j;
    }
}
```

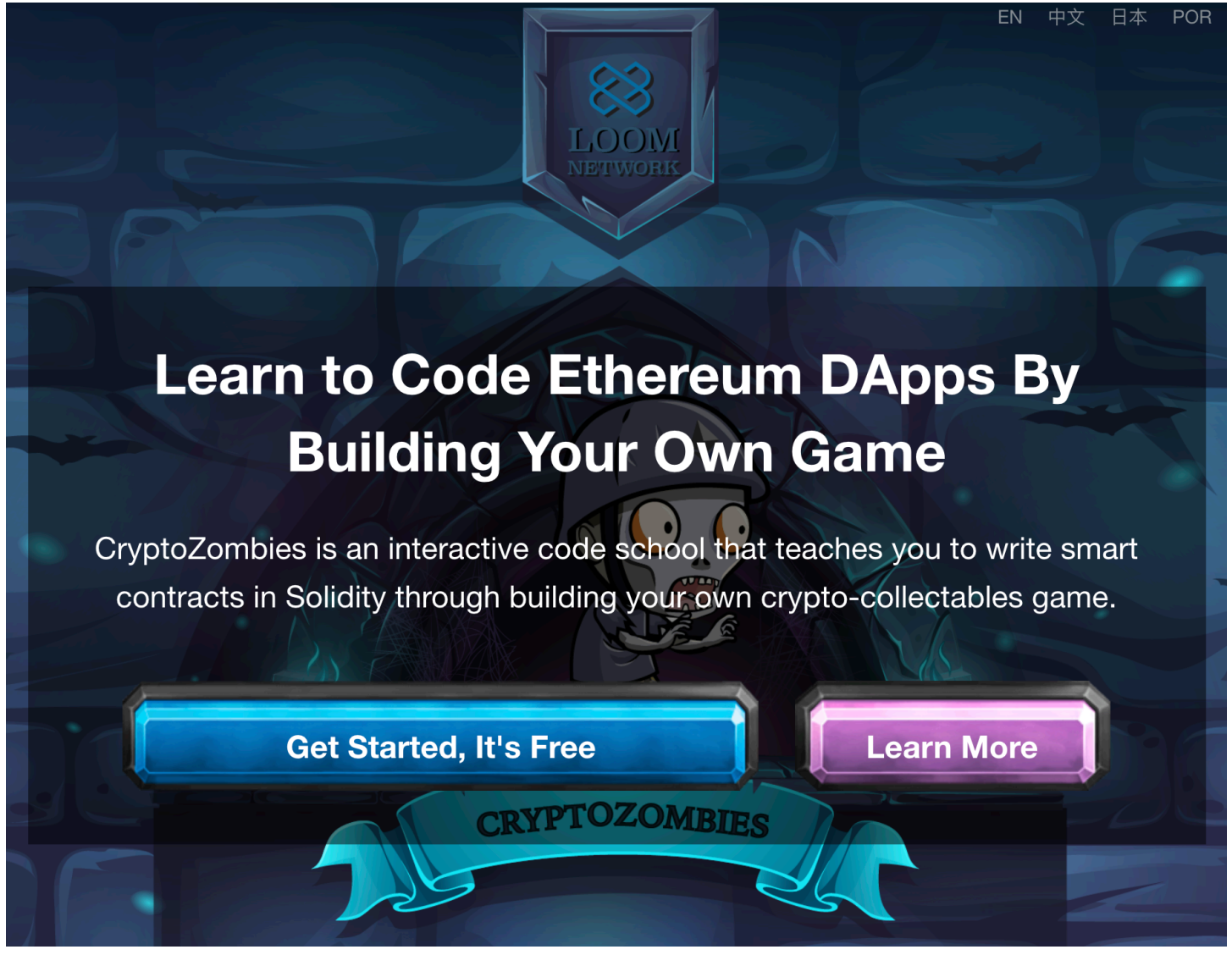
View and Pure Functions

Resources

- https://solidity.readthedocs.io/en/v0.5.12/
- https://solidity-by-example.org/
- https://github.com/OpenZeppelin/openzeppelin-contracts

https://cryptozombies.io/

Solidity Path: Beginner to Intermediate Smart Contracts



Chapter 2: Contracts

Starting with the absolute basics:

Solidity's code is encapsulated in **contracts**. A `contract` is the fundamental building block of Ethereum applications — all variables and functions belong to a contract, and this will be the starting point of all your projects.

An empty contract named `HelloWorld` would look like this:

```
contract HelloWorld {

}
```

Version Pragma

All solidity source code should start with a "version pragma" — a declaration of the version of the Solidity compiler this code should use. This is to prevent issues with future compiler versions potentially introducing changes that would break your code.

It looks like this: `pragma solidity ^0.4.25;` (for the latest solidity version at the time of this writing, 0.4.25).

Putting it together, here is a bare-bones starting contract — the first thing you'll write every time you start a new project:

Contract.sol

```
1  pragma solidity //1. Enter solidity version here
2
3  //2. Create contract here
4
```

Hints

Contract life cycle

1. 編譯合約

Compile contract to **bytecode**

2. 部署合約

```
sendTransaction({
        from: eth.coinbase,
        data: bytecode,
        gas: 470000
})
```

3. 呼叫合約函數

```
sendTransaction({
        from: eth.coinbase,
        to: contarct_addr,
        data: encode(function),
        gas: 50000
})
```

4. 摧毀合約

```
sendTransaction({
        from: eth.coinbase,
        to: contarct_addr,
        data: encode(kill),
        gas: 50000
})
```

Contract life cycle

1. 編譯合約

Compile contract to **bytecode**

2. 部署合約

```
sendTransaction({
        from: eth.coinbase,
        data: bytecode,
        gas: 470000
})
```

Contract-creating TX

3. 呼叫合約函數

```
sendTransaction({
        from: eth.coinbase,
        to: contarct_addr,
        data: encode(function),
        gas: 50000
})
```

Message call

4. 摧毀合約

```
sendTransaction({
        from: eth.coinbase,
        to: contarct_addr,
        data: encode(kill),
        gas: 50000
})
```

Message call

Contract life cycle

1. 編譯合約

Compile contract to **bytecode**

2. 部署合約

```
sendTransaction({
        from: eth.coinbase,
        data: bytecode,
        gas: 470000
})
```

3. 呼叫合約函數

```
sendTransaction({
        from: eth.coinbase,
        to: contarct_addr,
        data: encode(function),
        gas: 50000
})
```

4. 摧毀合約

```
sendTransaction({
        from: eth.coinbase,
        to: contarct_addr,
        data: encode(kill),
        gas: 50000
})
```

1. 怎麼知道跟那個合約互動?
2. 怎麼知道合約有哪些函數可以呼叫?
3. 誰可以跟合約互動?
4. 交易的執行都會花費 gas 嗎?
5. 如何估算交易執行要送的 gas?

web3.py

```python
from pathlib import Path

from solcx import compile_files
from web3.auto import w3


def get_contract_interface(contract_name):
    # contract path
    contract_dir = Path('contracts').absolute()
    contract_path = contract_dir / 'contracts' / f'{contract_name}.sol'
    interface_path = f'{contract_path}:{contract_name}'

    # compile contract
    compiled_sol = compile_files([contract_path])
    contract_interface = compiled_sol[interface_path]
    contract_abi = contract_interface['abi']
    contract_bytecode = contract_interface['bin']

    return contract_abi, contract_bytecode


def deploy_contract(contract_name):
    # get contract interface
    contract_abi, contract_bytecode = get_contract_interface(contract_name)

    w3.eth.defaultAccount = w3.eth.accounts[0]

    # deploy
    contract = w3.eth.contract(abi=contract_abi, bytecode=contract_bytecode)
    tx_hash = contract.constructor().transact()
    tx_receipt = w3.eth.waitForTransactionReceipt(tx_hash)
    contract_address = tx_receipt.contractAddress

    print(f'Deploy {contract_name}.sol to "{contract_address}"\n')

    return contract(address=contract_address)


def get_contract(contract_name, contract_address):
    contract_abi, contract_bytecode = get_contract_interface(contract_name)

    return w3.eth.contract(address=contract_address, abi=contract_abi)
```

deploy
1. abi
2. bytecode

interact
1. abi
2. address

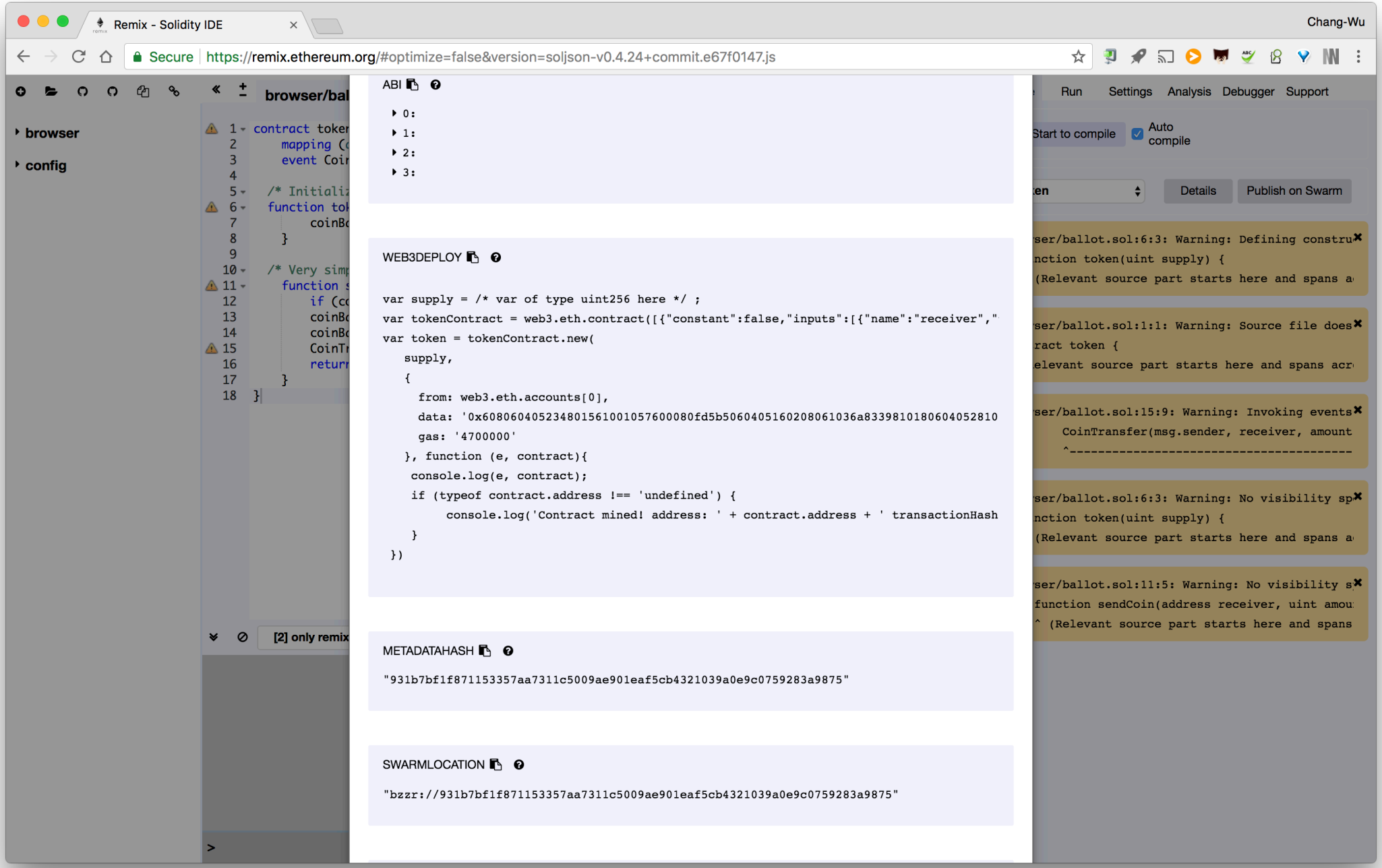
abi

- https://etherscan.io/address/0xdd974d5c2e2928dea5f71b9825b8b646686bd200#code

```
$ curl -s \
http://api.etherscan.io/api\?module\=contract\&action\=getabi\&address\=0x3be856c94c7a1ff4f433
ae95a48544b1a62ce385\&format\=raw | jq '[to_entries[] | select(.value.constant==false) | .valu
e.name]'
[
  "approve",
  "transferFrom",
  "burn",
  "burnFrom",
  "transfer",
  "emergencyERC20Drain",
  "transferOwnership"
]
```

Coin contract

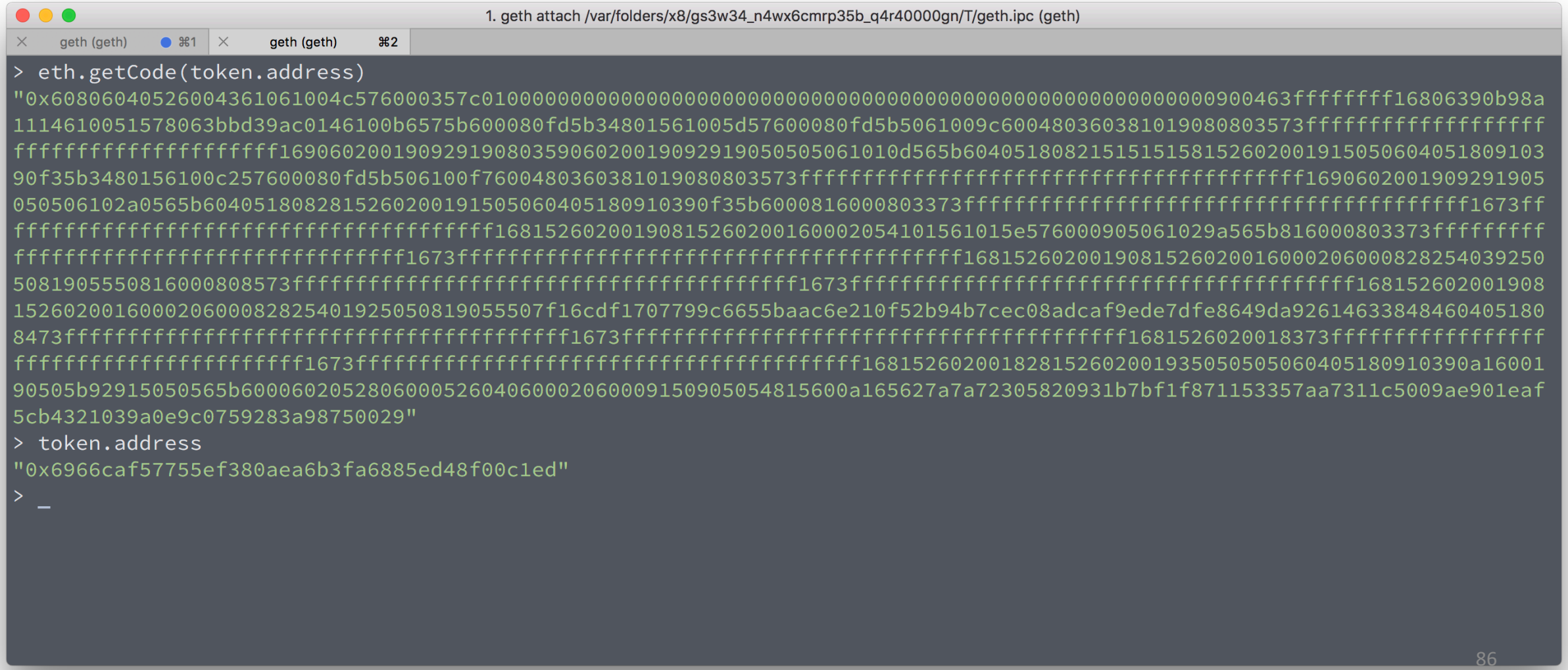
```
contract token {
    mapping (address => uint) public coinBalanceOf;
    event CoinTransfer(address sender, address receiver, uint amount);

    /* Initializes contract with initial supply tokens to the creator of the contract */
    function token(uint supply) {
        coinBalanceOf[msg.sender] = supply;
    }

    /* Very simple trade function */
    function sendCoin(address receiver, uint amount) returns(bool sufficient) {
        if (coinBalanceOf[msg.sender] < amount) return false;
        coinBalanceOf[msg.sender] -= amount;
        coinBalanceOf[receiver] += amount;
        CoinTransfer(msg.sender, receiver, amount);
        return true;
    }
}
```


Remix - Solidity IDE
Chang-Wu
Secure https://remix.ethereum.org/#optimize=false&version=soljson-v0.4.24+commit.e67f0147.js
browser
config
Run Settings Analysis Debugger Support
Start to compile
Auto compile
Details
Publish on Swarm
ABI
0:
1:
2:
3:
WEB3DEPLOY
var supply = /* var of type uint256 here */ ;
var tokenContract = web3.eth.contract([{"constant":false,"inputs":[{"name":"receiver","
var token = tokenContract.new(
supply,
{
from: web3.eth.accounts[0],
data: '0x6080604052348015610010576000080fd5b506040516020806103a833981018060405281
gas: '4700000'
}, function (e, contract){
console.log(e, contract);
if (typeof contract.address !== 'undefined') {
console.log('Contract mined! address: ' + contract.address + ' transactionHash
}
})
METADATAHASH
"931b7bf1f871153357aa7311c5009ae901eaf5cb4321039a0e9c0759283a9875"
SWARMLOCATION
"bzzr://931b7bf1f871153357aa7311c5009ae901eaf5cb4321039a0e9c0759283a9875"

Web3 deploy code

```
var supply = 10000;
var tokenContract =
web3.eth.contract([{"constant":false,"inputs":[{"name":"receiver","type":"address"},{"name":"amount","ty
pe":"uint256"}],"name":"sendCoin","outputs":[{"name":"sufficient","type":"bool"}],"payable":false,"state
Mutability":"nonpayable","type":"function"},{"constant":true,"inputs":[{"name":"","type":"address"}],"na
me":"coinBalanceOf","outputs":[{"name":"","type":"uint256"}],"payable":false,"stateMutability":"view","t
ype":"function"},{"inputs":[{"name":"supply","type":"uint256"}],"payable":false,"stateMutability":"nonpa
yable","type":"constructor"},{"anonymous":false,"inputs":[{"indexed":false,"name":"sender","type":"addre
ss"},{"indexed":false,"name":"receiver","type":"address"},{"indexed":false,"name":"amount","type":"uint2
56"}],"name":"CoinTransfer","type":"event"}]);
var token = tokenContract.new(
    supply,
    {
        from: web3.eth.accounts[0],
        data: '---------[BYTECODE]---------',
        gas: '4700000'
    }, function (e, contract){
        console.log(e, contract);
        if (typeof contract.address !== 'undefined') {
            console.log('Contract mined! address: ' + contract.address + ' transactionHash: ' +
contract.transactionHash);
    }
})
```

bytecode

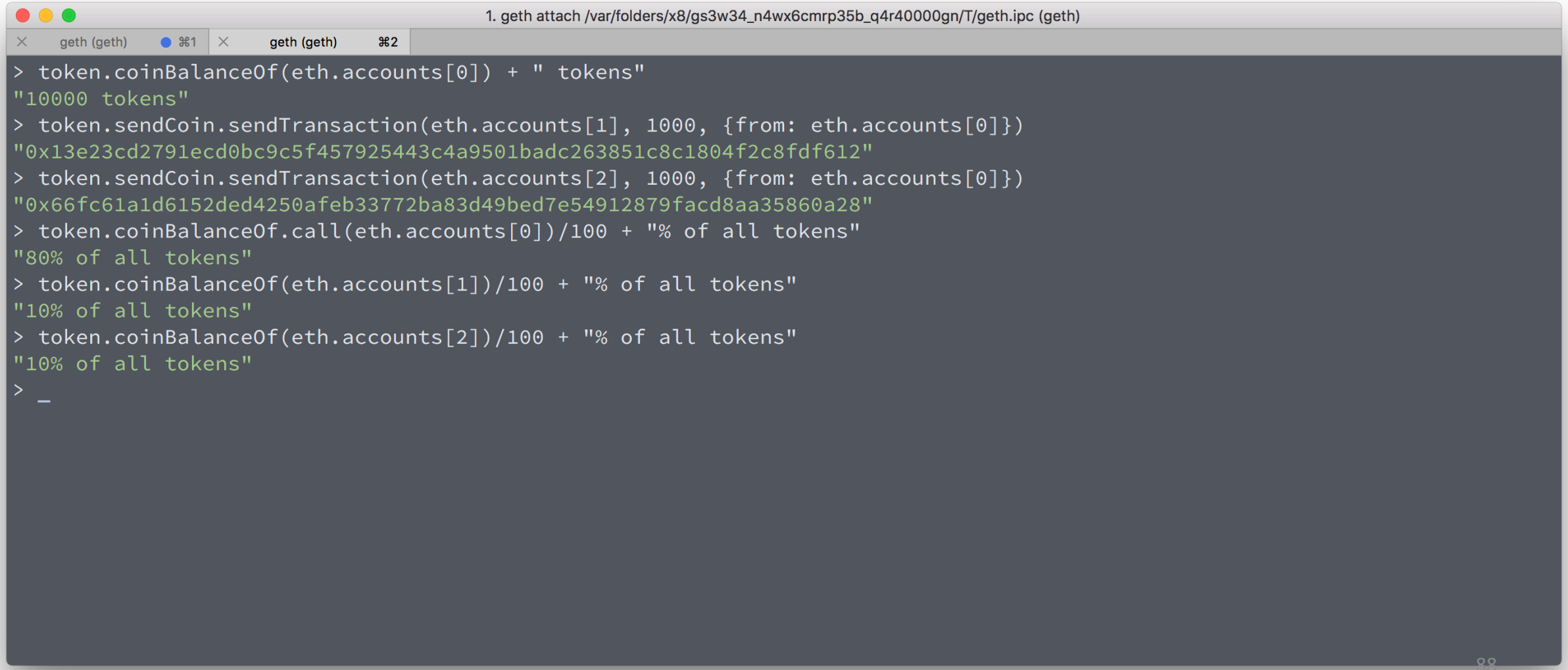

1. geth attach /var/folders/x8/gs3w34_n4wx6cmrp35b_q4r40000gn/T/geth.ipc (geth)
geth (geth) ⌘1
geth (geth) ⌘2
> eth.getCode(token.address)
"0x6080604052600436106100 4c576000357c0100000000000000000000000000000000000000000000000000000000900463ffffffff16806390b98a
11146100 51578063bbd39ac0146100b6575b600080fd5b34801561005d57600080fd5b5061009c6004803603810190808035 73ffffffffffffffffffff
ffffffffffffffffffff1690602001909291908035906020019092919050505061010d565b604051808215151515815260200191505060405180910 3
90f35b3480156100c257600080fd5b506100f760048036038101908080357 3ffffffffffffffffffffffffffffffffffffffff1690602001909291905
050506102a0565b604051808281526020019150506040518091039 0f35b60008160008033 73ffffffffffffffffffffffffffffffffffffffff1673ff
ffffffffffffffffffffffffffffffffffff1681526020019081526020016000205410 1561015e576000905061029a565b816000803373ffffffff
ffffffffffffffffffffffffffff1673ffffffffffffffffffffffffffffffffffffffff1681526020019081526020016000206000828254039250
50819055508160008085 73ffffffffffffffffffffffffffffffffffffffff1673ffffffffffffffffffffffffffffffffffffffff168152602001908
15260200160002060008282540192505081905550 7f16cdf1707799c6655baac6e210f52b94b7cec08adcaf9ede7dfe8649da926146338484604051 80
8473ffffffffffffffffffffffffffffffffffffffff1673ffffffffffffffffffffffffffffffffffffffff1681526020018373ffffffffffffffff
ffffffffffffffffffff1673ffffffffffffffffffffffffffffffffffffffff1681526020018281526020019350505050604051809103 90a16001
90505b92915050565b60006020528060005260406000206000915090505 4815600a165627a7a72305820931b7bf1f871153357aa7311c5009ae901eaf
5cb4321039a0e9c0759283a98750029"
> token.address
"0x6966caf57755ef380aea6b3fa6885ed48f00c1ed"
> _

Coin contract function

```
contract token {
    mapping (address => uint) public coinBalanceOf;
    event CoinTransfer(address sender, address receiver, uint amount);

    /* Initializes contract with initial supply tokens to the creator of the contract */
    function token(uint supply) {
        coinBalanceOf[msg.sender] = supply;
    }

    /* Very simple trade function */
    function sendCoin(address receiver, uint amount) returns(bool sufficient) {
        if (coinBalanceOf[msg.sender] < amount) return false;
        coinBalanceOf[msg.sender] -= amount;
        coinBalanceOf[receiver] += amount;
        CoinTransfer(msg.sender, receiver, amount);
        return true;
    }
}
```

https://github.com/ethereum/go-ethereum/wiki/Contract-Tutorial

sendTransaction



OpenZeppelin

- https://github.com/OpenZeppelin/openzeppelin-contracts
  - SafeMath
    - https://github.com/OpenZeppelin/openzeppelin-contracts/tree/master/contracts/math
  - ERC20
    - https://github.com/OpenZeppelin/openzeppelin-contracts/tree/master/contracts/token/ERC20

Thanks for listening!