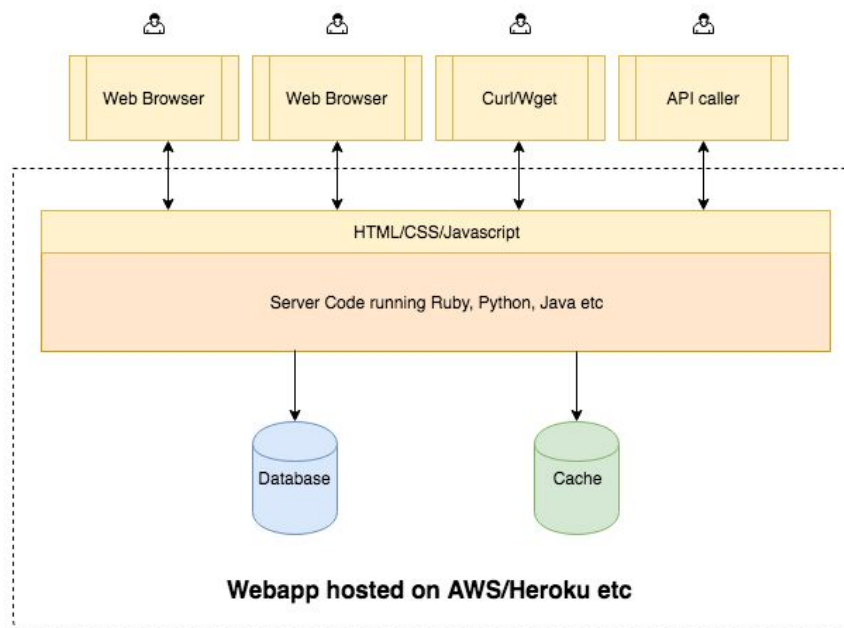


Distributed Voting System Based on Ethereum

Zhengxiong Yuan

Motivation:

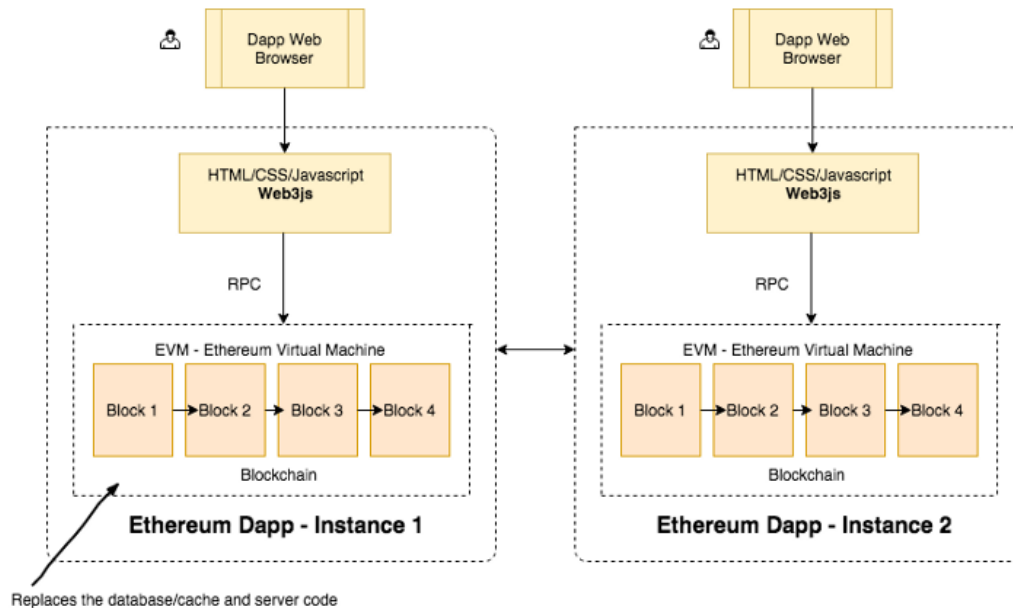
Traditional web applications usually depend on a central server. The web application is hosted on a hosting provider like AWS, Heroku or a VPS. All the clients interact with this one central application. Clients can be a browser, another api consuming your service etc. When a client makes a request to the server, the server does it's magic, talks to the database and/or cache, reads/writes/updates the database and serves the client. The following graph shows the framework for traditional web applications.



However, the above framework may have some problems in some scenarios. For example, if the web application is a voting system, how can you make sure that the central server is always stable and always has the correct information? There might be some solutions to try to fix that problem, like making many replications of the database. But, it cannot perfectly solve the problem because the users always rely on the center server instead of themselves. What if the owner of the server or some hacker made some malicious changes to the database? The users will never know if that happens. Fortunately, with the help of blockchain technology, we can build decentralized applications, which will make the system more secure and stable.

In a decentralized application, the data/transactions are stored in a blockchain. All of the users have a local copy of the blockchain. Ethereum is a global, open-source platform for

decentralized applications. Every few transactions that occur in the Ethereum network are packaged into blocks and each block is linked to the next block. This linked series of blocks which holds all the transaction data is the blockchain. The following picture shows an example of decentralized applications.



Every client/user/browser communicates with its own instance of the application. There is no central server to which all clients connect to. This means, every person who wants to interact with a dapp (Decentralized Application) will need a full copy of the blockchain running on their computer/phone etc. In this case, users will not rely on any central server. Instead, every user can have the whole blockchain.

Instructions:

1. Npm and nodejs:

```
zyuan@pop-os:~/develop/DVote$ npm -v
7.0.14
zyuan@pop-os:~/develop/DVote$ node -v
v15.3.0
```

You can install nodejs and npm from the official site (<https://nodejs.org/en/>).

2. Install packages:

After having nodejs and npm installed on your computer, you can download the following packages using "npm install <package>@<version>".

```
{
  "dependencies": {
    "ganache-cli": "^6.12.1",
    "solc": "^0.6.4",
    "web3": "^1.3.0"
  }
}
```

3. Run ganache-cli (a local blockchain) (NOTE: keep it running):

```
zyuan@pop-os:~/develop/DVote$ ./node_modules/.bin/ganache-cli
Ganache CLI v6.12.1 (ganache-core: 2.13.1)

Available Accounts
=====
(0) 0x78341b23a71180627d9f2C8cb6bd03bcA3D58B1c (100 ETH)
(1) 0x6b05f1Ba2AD1ca751b1411F72325f90ffdaC27Ee (100 ETH)
(2) 0x8F8FfC63880EAc4062e16B84BB385Faafb198C8E (100 ETH)
(3) 0xbce2a9EB6D39A06E5000D54711a766490c52c2F4 (100 ETH)
(4) 0xF1Ce4b19f9F33Ee3e3D5028FeE0bd8F9C79d8206 (100 ETH)
(5) 0x97afEA693d1D50a7c5Dc7eB95b4e46d56f7e9a6e (100 ETH)
(6) 0xE79b1Eeb649eF0b05e10a2B1484ecB73b3cB9C77 (100 ETH)
(7) 0xaD19B8d3cd8c64210a852697665Bed5e23728dF6 (100 ETH)
(8) 0x43ba350A1D7C299b018b9a9E556B0471e6137e7d (100 ETH)
(9) 0x5E392168C887C49A559177BA3Af602F4E03522f6 (100 ETH)

Private Keys
=====
(0) 0x659242f2cddcf5391f972e01f76de7a2cd088eea3f102d87941650553550bbee
(1) 0xa5e12fea137709e1148aa41c163ac8883ea6d77f617f8af0c6fe7fe6361b0bd7
(2) 0x8281d8a84c1a2b3a61802d5d68da58f4e1803bd3acaab25705b1d3dcf8d974cb
(3) 0x334bc202779b296e82e096be837c4e009d479d7cf4c95f4039171c385dc9056a
(4) 0x08c538f75be61685401f4e8441590d3ab2087acb3ba1c3cb92290d145f3c3f7f
(5) 0x1b95f13b3731c5c9be0a91611fa46a12b12bbb9dc9820f2fd8bee54d3fbec556
(6) 0x129ebbf135a5fab12957a73f162b7fc224d23564020b8c0bd65797c0308343a8
(7) 0x3d1f8803b88fd84a0d816eb1866b0e8c5f89c630d7fd4d0bc42d00afab27a26c
(8) 0x4a889f69c8f5a83adf7673e714cd059b31ad47fb44cf97e1f6c2cdd3b94b7e81
(9) 0xf1a81593766784ae72a8d7fab27165810f4ccf3db62f31a080b0aeda6141cf67

HD Wallet
=====
Mnemonic:      jar rather wolf stove bottom soup sure capital sign dizzy please casino
Base HD Path:  m/44'/60'/0'/0/{account_index}

Gas Price
=====
20000000000

Gas Limit
=====
6721975

Call Gas Limit
=====
9007199254740991

Listening on 127.0.0.1:8545
```

4. Deploy contract:

```
zyuan@pop-os:~/develop/DVote$ node contract.js  
0x81e18c1195b3e744A8a60a492e9FFC1cb85a6C10
```

5. Open "index.html" using your browser:



Distributed Voting

Candidates	Votes
Jason	
Amy	
John	

6. Set contract address:

Copy the output in step 5 to the first input box, then click "Set". The output in step 5 is the address of the contract.

7. Now you can vote for the candidates using the second input box. You can also see the transactions in the "ganache-cli" window.

Distributed Voting

Candidates	Votes
Jason	3
Amy	1
John	5

```
Transaction: 0x7088b3e912ac737a698deca1a40cf39eac9a5d0fa4d87f8f154e9fc21cd43d43
Gas usage: 35396
Block Number: 9
Block Time: Sun Nov 29 2020 18:07:20 GMT-0600 (Central Standard Time)
```

```
eth_getTransactionReceipt
eth_call
eth_gasPrice
eth_sendTransaction
```

```
Transaction: 0x8a60a68b9efe4a7c08dabafd16b32cb25f4d1b5db14046f0cc44855f4f6a8b25
Gas usage: 35396
Block Number: 10
Block Time: Sun Nov 29 2020 18:07:20 GMT-0600 (Central Standard Time)
```

```
eth_getTransactionReceipt
eth_call
```