NPTEL ONLINE CERTIFICATION COURSES

Blockchain and its applications

Bishakh Chandra Ghosh

Department of Computer Science & Engineering
Indian Institute of Technology Kharagpur

Lecture 24: Ethereum 3

CONCEPTS COVERED

- Ethereum applications - DAPPS

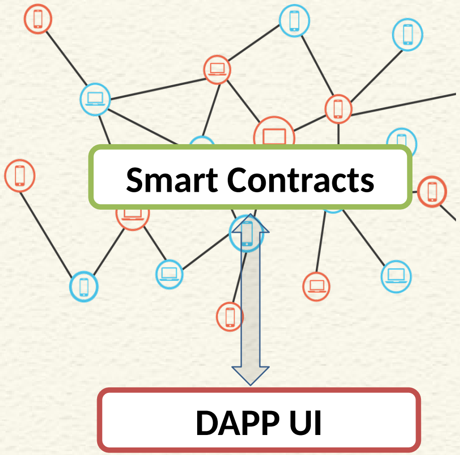
- Using web3.js to programmatically access Ethereum network

KEYWORDS

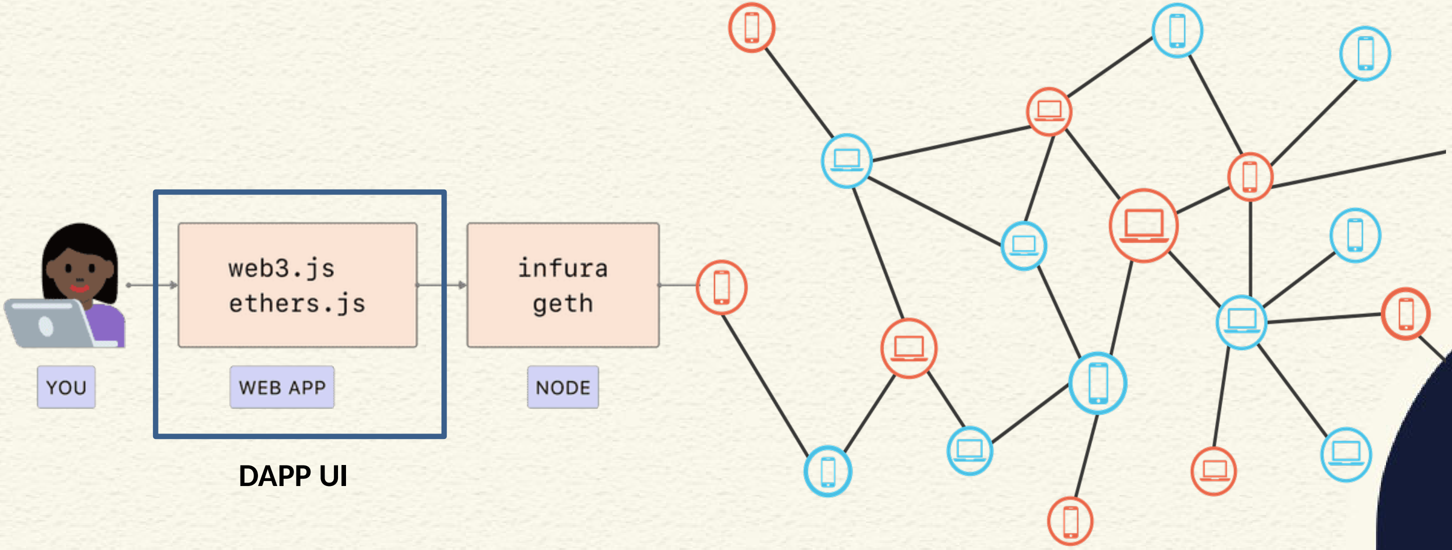
- web3.js
- DAPPS

DAPPS in Ethereum

- DAPP - decentralized application
- Application that is built for decentralized networks like Ethereum
- Combines two components:
    1. Smart Contracts
    2. User interface for executing transactions and contracts



Programmatically access Ethereum networks



DAPP UI

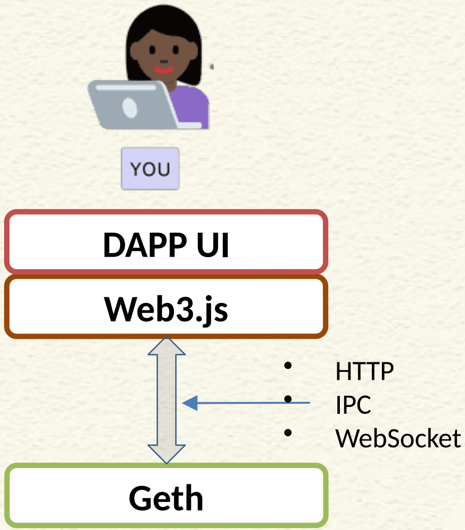
Mainnet / Testnet

web3.js

- Ethereum JavaScript API
  https://web3js.readthedocs.io/

- Collection of libraries that allow you to interact with a local or remote ethereum nodes.

- It can connect using:
  - HTTP
  - IPC
  - WebSocket



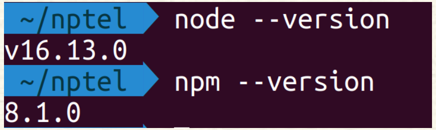
Install web3.js

- Install Node.js
  https://nodejs.org/en/

```
wget https://nodejs.org/dist/v16.13.0/node-v16.13.0-linux-x64.tar.xz

tar -xvf node-v16.13.0-linux-x64.tar.xz

export PATH=/home/<username>/nptel/node-v16.13.0-linux-x64/bin:$PATH
```

- Check node and npm version

```
node --version

npm --version
```



```
~/nptel  node --version
v16.13.0
~/nptel  npm --version
8.1.0
```

Install web3.js

- Install web3.js

```
npm install web3
```

- Verify that you can import web3

```
node
var Web3 = require('web3');
console.log(Web3.version)
```

Connecting web3.js to Geth

```
// Import web3
var Web3 = require('web3');

// Connect to local Geth client
var web3 = new Web3('http://localhost:8545');
```

Query Balance

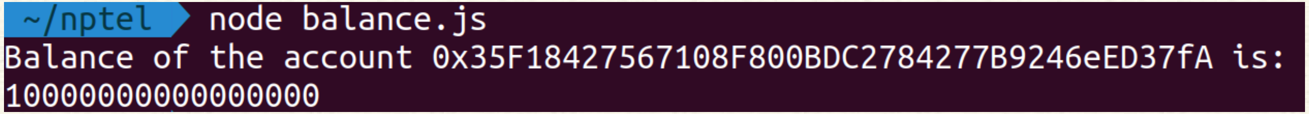
- Create a file with .js extension. Example: "balance.js"

```
// Import web3
var Web3 = require('web3');

// Connect to local Geth client
var web3 = new Web3('http://localhost:8545');

// Query balance of an account
console.log("Balance of the account
0x35F18427567108F800BDC2784277B9246eED37fA is:");
web3.eth.getBalance("0x35F18427567108F800BDC2784277B9246eED37fA").then(
console.log);
```

Query Balance

- Create a file with .js extension. Example: "balance.js"

```
// Import web3
var Web3 = require('web3');

// Connect to local Geth client
var web3 = new Web3('http://localhost:8545');

// Query balance of an account
console.log("Balance of the account
0x35F18427567108F800BDC2784277B9246eED37fA is:");
web3.eth.getBalance("0x35F18427567108F800BDC2784277B9246eED37fA").then(
console.log);
```

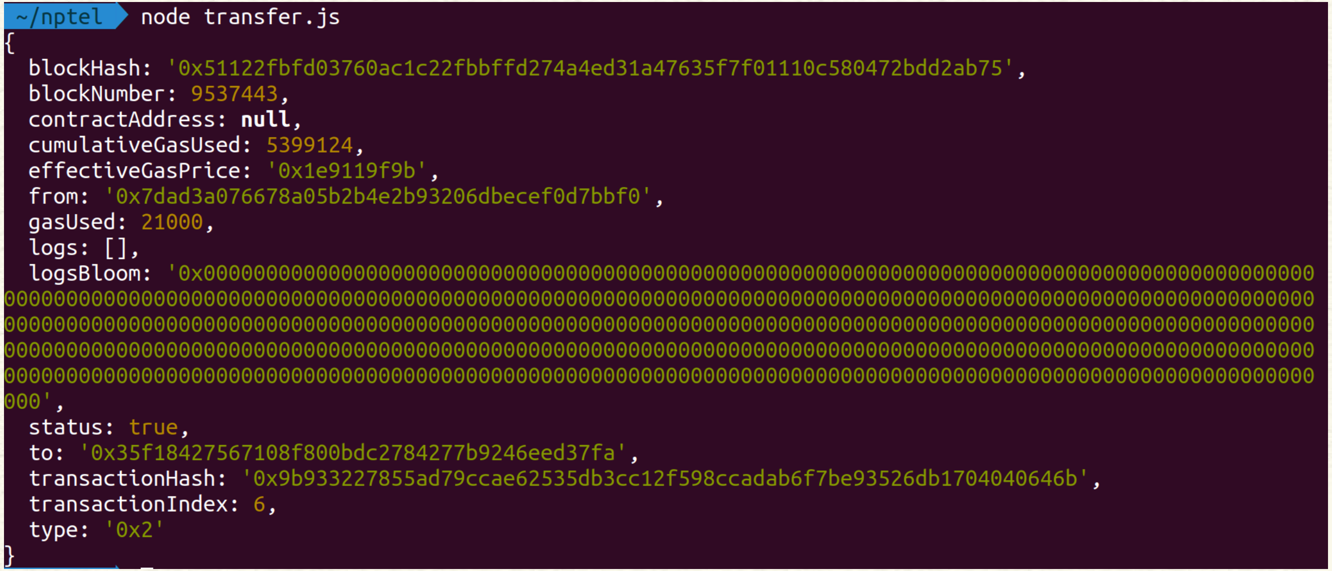


```
 ~/nptel  node balance.js
Balance of the account 0x35F18427567108F800BDC2784277B9246eED37fA is:
1000000000000000
```

Transfer Ether

```
// Import web3
var Web3 = require('web3');

// Connect to local Geth client
var web3 = new Web3('http://localhost:8545');

// Prepare transaction
var transaction = {
from: "0x7dad3a076678a05b2b4e2b93206dbecef0d7bbf0",
to: "0x35F18427567108F800BDC2784277B9246eED37fA",
value: Web3.utils.numberToHex(1000000000000000000)
}

// Send the transaction
web3.eth.sendTransaction(transaction).then(console.log);
```

Transfer Ether



```
~/nptel  node transfer.js
{
  blockHash: '0x51122fbfd03760ac1c22fbbffd274a4ed31a47635f7f01110c580472bdd2ab75',
  blockNumber: 9537443,
  contractAddress: null,
  cumulativeGasUsed: 5399124,
  effectiveGasPrice: '0x1e9119f9b',
  from: '0x7dad3a076678a05b2b4e2b93206dbecef0d7bbf0',
  gasUsed: 21000,
  logs: [],
  logsBloom: '0x00000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
000',
  status: true,
  to: '0x35f18427567108f800bdc2784277b9246eed37fa',
  transactionHash: '0x9b933227855ad79ccae62535db3cc12f598ccadab6f7be93526db1704040646b',
  transactionIndex: 6,
  type: '0x2'
}
```

Conclusion

- Web3.js for DAPP Development
- Build interface for Ethereum applications

Thank you