以太坊去中心化应用（DApp）开发

— 第1课 —

熊丽兵（Tiny熊）

公众号：登链学院　　微信号：xlbxiong

登链学院
edu.upchain.pro

自我介绍

- 熊丽兵（Tiny熊）
- 深入浅出区块链站长 learnblockchain.cn
- 《精通以太坊智能合约开发》作者 电子工业出版社
- 登链科技CTO

登链学院
edu.upchain.pro

登链学院

- 专注区块链技术普及推广
- 让每个程序员都懂区块链

登链学院
edu.upchain.pro

课程介绍

- 课程特点
- 面向人群
- 课程内容
- 学习收获

登链学院
edu.upchain.pro

课程特点

- 10分钟/每课
- 20节以上
- 案例教学
- 源码公开

登链学院
edu.upchain.pro

面向人群

想从事区块链开发

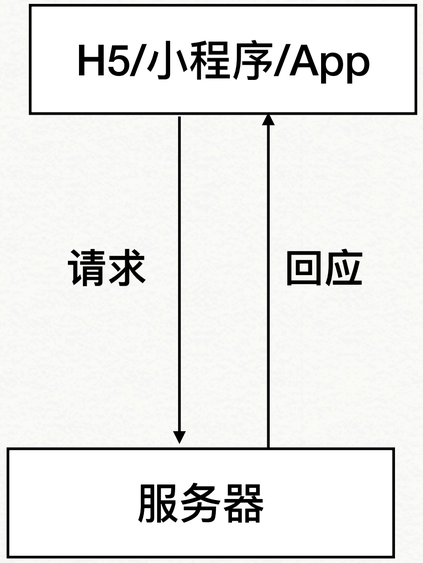
- 对区块链、以太坊、有基本了解
- 了解智能合约开发、Solidity语言

推荐：
《从0到1学区块链》
《深入详解以太坊智能合约语言solidity》
《通过代币学智能合约》

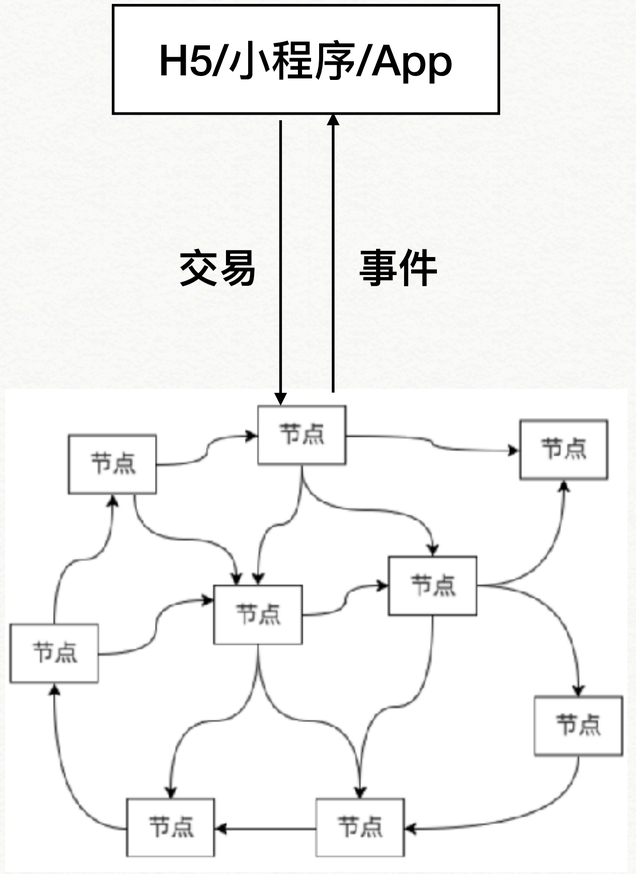
登链学院
edu.upchain.pro

去中心化应用

中心化应用
web2.0



去中心化应用
web3.0



登链学院
edu.upchain.pro

去中心化应用

中心化应用
web2.0

去中心化应用
web3.0

UI表现

http请求

后端服务程序

服务器Nginx/Apache

UI表现

钱包/DApp浏览器

rpc请求

web3.js

智能合约

节点EVM

登链学院
edu.upchain.pro

课程内容

Geth 安装与使用

- Geth 搭建私有链
- Geth 搭建联盟链
- Ganache 模拟节点
- Geth 账号管理、部署合约

登链学院
edu.upchain.pro

课程内容

智能合约

- 智能合约编写（Solidity）
- 智能合约测试
- 智能合约部署

登链学院
edu.upchain.pro

课程内容

Web3.js

- Web3.js 原理
- Web3.js API 查询
- Web3.js API 使用（控制台、应用）
- 案例：如何应用如何调用合约函数，监听合约状态

登链学院
edu.upchain.pro

课程内容

应用UI

- HTML5/CSS
- JavaScript(JQuery)

登链学院
edu.upchain.pro

课程内容

使用框架Truffle 简化Dapp开发流程

- 创建应用工程
- 去中心化应用测试
- 应用部署、运行
- 结合实战案例讲解

学习收获

- 提升对去中心化的理解
- 以太坊开发实战能力显著增强
- 提供代码、反复修改练习、应用到项目

登链学院
edu.upchain.pro

更多

- 一定动手练习
- 及时提问
- 课程体验师

https://learnblockchain.cn/course/

谢谢观看

欢迎关注微信号/公众号交流技术问题

登链学院

Tiny熊

登链学院
edu.upchain.pro

第一个去中心化应用

DApp：Decentralized Application

- 读取合约内名字和年龄并显示
- 名字和年龄保存在区块链（合约）上

https://learnblockchain.cn/2018/04/15/web3-html/

https://github.com/xilibi2003/DAppDemos

登链学院
edu.upchain.pro

实现第一个DApp

- 合约

  https://wiki.learnblockchain.cn/course/solidity.html

- Web 前端

  http://www.w3school.com.cn/h.asp

- Web3.js 与合约交互

  https://github.com/ethereum/web3.js

登链学院
edu.upchain.pro

初探DApp总结

关键点

- DAPP关键是Web3和智能合约交互

不足

- 使用以太坊网络，交易确认慢，消耗账户以太币
- 搭建Web服务器
- 不易修改、测试、调试、发布应用

登链学院
edu.upchain.pro

知识点

- 使用本地（局域网内）节点
- Web3.js
- 使用开发框架Truffle

以太坊去中心化应用（DApp）开发

补充：Web服务器搭建

熊丽兵（Tiny熊）

登链学院
edu.upchain.pro

Web服务器搭建

Nginx/Apache

MAMP：Mac + Apache + Mysql + Php

LAMP：Linux + Apache + Mysql + Php

WAMP：Windows + Apache + Mysql + Php

登链学院
edu.upchain.pro

Web服务器搭建

python

python -m SimpleHTTPServer

NPM

npm install lite-server --save-dev

node_modules/lite-server/bin/lite-server

以太坊去中心化应用（DApp）开发

节点搭建

熊丽兵（Tiny熊）

登链学院
edu.upchain.pro

Ganache

Ganache模拟节点

- 快速打包确认交易
- 提供多个测试账号

https://trufflframework.com/ganache

Ganache

Ganache 安装

Windows, Mac, and Linux

- 桌面版

  https://github.com/trufflesuite/ganache/releases

- 命令行版（ganache cli 以前叫TestRPC）

  https://github.com/trufflesuite/ganache-cli

  > npm install -g ganache-cli

登链学院
edu.upchain.pro

补充介绍：NPM

NPM：Node.js包管理工具

Node.js集成了NPM，安装Node.js会附带安装NPM

- 安装Node.js

  https://nodejs.org/en/download/

- NPM 使用

```
> npm -v
> npm install <Module>        ;安装到当前目录
> npm install -g <Module>     ;全局安装
```

登链学院
edu.upchain.pro

Ganache使用

连接Ganache节点

- Remix连接节点
- DApp连接节点
- MetaMask连接节点(导入账号)

登链学院
edu.upchain.pro

Ganache cli使用

- ganache-cli 启动节点
- 可以附加启动参数

https://github.com/trufflesuite/ganache-cli

Geth

Geth：go-ethereum

以太坊节点官方的go语言实现版本

功能强大

搭建节点

管理账户

发送交易

部署、运行合约

Web3.js 的执行环境

登链学院
edu.upchain.pro

Geth安装

Mac

https://github.com/ethereum/go-ethereum/wiki/Installation-Instructions-for-Mac

使用brew 安装，brew是Mac下的包管理工具，

https://brew.sh/index_zh-cn

```
> brew tap ethereum/ethereum
> brew install ethereum
> geth  启动
```

登链学院
edu.upchain.pro

Geth安装

Windows

https://github.com/ethereum/go-ethereum/wiki/Installation-instructions-for-Windows

- 下载zip https://geth.ethereum.org/downloads/
- 解压出geth.exe
- 命令行定位到geth.exe 打开

登链学院
edu.upchain.pro

Geth安装

Linux（Ubuntu）

https://github.com/ethereum/go-ethereum/wiki/Installation-instructions-for-Windows

```
> sudo apt-get install software-properties-common
> sudo add-apt-repository -y ppa:ethereum/ethereum
> sudo apt-get update
> sudo apt-get install ethereum
```

登链学院
edu.upchain.pro

Geth使用

基本用法

- 查看帮助 > geth help
- 启动 > geth
- 启动控制台 > geth console
- 数据目录 > geth --datadir mydir
- 开发者模式 > geth --dev
- 日志控制 > geth 2>> test.log
  > geth — verbosity

https://github.com/ethereum/go-ethereum/wiki/Command-Line-Options

https://learnblockchain.cn/2017/11/29/geth_cmd_options/

登链学院
edu.upchain.pro

Geth使用

启用 API服务

- HTTP-RPC　　RPC： Remote Procedure Call 远程过程调用

  --rpc —rpcapi etc, web3　　--rpccorsdomain

- WebSocket

  —ws —wsapi etc, web3

- IPC（进程间通信）

  默认开启

https://github.com/ethereum/go-ethereum/wiki/Management-APIs

登链学院
edu.upchain.pro

Geth控制台API使用

- 账号管理
- 获取区块信息
- 转账
- 合约部署

https://github.com/ethereum/wiki/wiki/JavaScript-API

https://web3.learnblockchain.cn

登链学院
edu.upchain.pro

Geth 搭建私有链

为什么需要私有链

定制：调整挖矿难度，区块大小，预分配以太币

初始创世纪块

https://github.com/ethereum/go-ethereum

geth init genesis.json

可使用工具puppeth生成创世纪块配置文件

登链学院
edu.upchain.pro

Geth 节点集群

开启多个节点

本地测试注意点：

1.每个节点都有不同的**--datadir**

2.每个节点要运行在不同的端口，使用**--port**及**--rpcport**控制

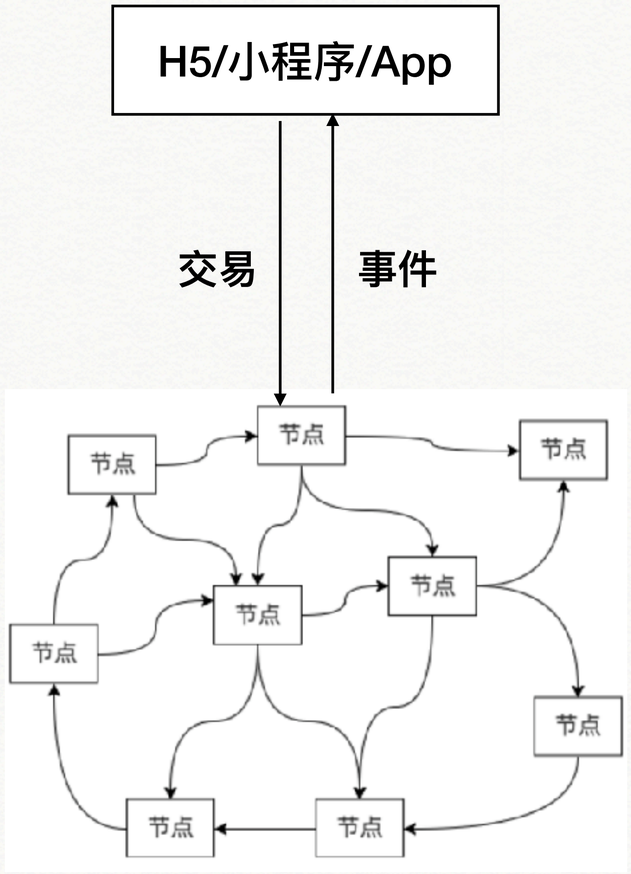
3.每个节点ipc唯一或禁用ipc，使用参数**—ipcpath**或**--ipcdisable**

多节点建立连接

admin.addPeer

登链学院
edu.upchain.pro

节点搭建小结



- Ganache 模拟节点
- Geth 节点

开发者模式

私有链（网络）

节点集群（联盟链）

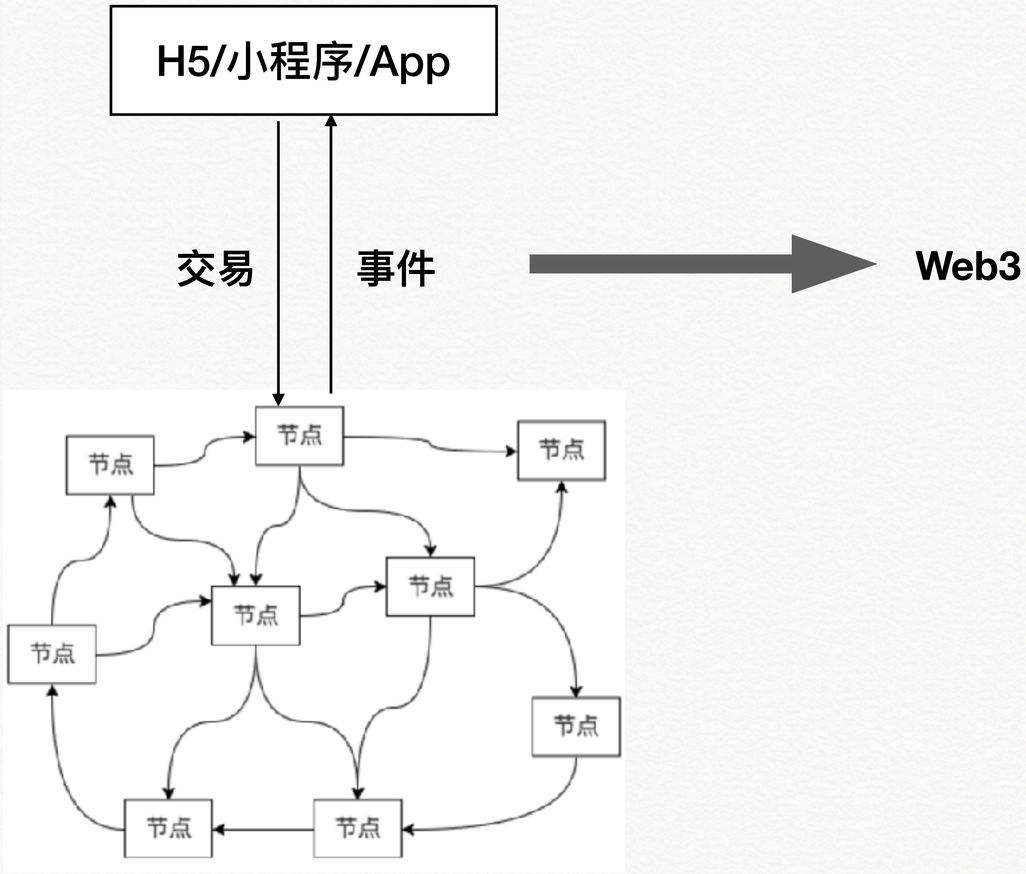
以太坊去中心化应用（DApp）开发

Web3 概述

熊丽兵（Tiny熊）

登链学院
edu.upchain.pro

什么是Web3



登链学院
edu.upchain.pro

什么是Web3

JSON-RPC 的封装

```
// 请求：
curl -X POST
--data '{"jsonrpc":"2.0","method":"eth_accounts","params":[],"id":1}'
-H "Content-Type: application/json”
http://127.0.0.1:8545

// 返回：
{
  "id":1,
  "jsonrpc": "2.0",
  "result": ["0xc94770007dda54cF92009BFF0dE90c06F603a09f"]
}
```

https://github.com/ethereum/wiki/wiki/JSON-RPC

```
// 请求：
> eth.accounts

// 返回：
["0xc94770007dda54cF92009BFF0dE90c06F603a09f"]
```

https://github.com/ethereum/wiki/wiki/JavaScript-API

http://web3js.readthedocs.io/en/1.0/index.html

中文版：https://web3.learnblockchain.cn/0.2x.x/

登链学院
edu.upchain.pro

Web3 多语言实现

- JavaScript Web3.js

https://github.com/ethereum/web3.js

- Python Web3.py

https://github.com/ethereum/web3.py

- Haskell hs-web3

https://github.com/airalab/hs-web3

- Java web3j

https://github.com/web3j/web3j

- Scala web3j-scala

https://github.com/mslinn/web3j-scala

- Purescript purescript-web3

https://github.com/f-o-a-m/purescript-web3

- PHP web3.php

https://github.com/sc0Vu/web3.php

- PHP ethereum-php

https://github.com/digitaldonkey/ethereum-php

以太坊去中心化应用（DApp）开发

Web3.js 使用

熊丽兵（Tiny熊）

登链学院
edu.upchain.pro

Web3.js 版本

0.20.x

eth：区块链相关的方法

net：p2p网络状态的方法

shh：whisper协议相关

clique：POA共识相关

admin：管理节点相关的方法

miner：挖矿相关

personal：管理账户的方法

txpool：交易内存池相关

web3：包含了以上对象，还包含工具方法

https://github.com/ethereum/wiki/wiki/JavaScript-API

中文版：https://web3.learnblockchain.cn/0.2x.x/

登链学院
edu.upchain.pro

Web3 模块

1.0

对0.2x.x 版本做了重构，模块化

web3
web3-eth
web3-eth-subscribe
web3-eth-contract
web3-eth-accounts
web3-eth-personal
web3-eth-iban
web3-eth-abi
web3-net
web3-bzz
web3-shh
web3-utils
web3-admin

https://web3js.readthedocs.io/en/1.0/

登链学院
edu.upchain.pro

Web3.js 版本

1.0

引入Promise，简化异步编程，将异步操作以同步操作的流程表达出来，避免层层嵌套的回调

```
web3.eth.getBlockNumber(function callback(err, value) {
    console.log("BlockNumber:" + value)
});
```

→

```
web3.eth.getBlockNumber().then(console.log);
```

```
web3.eth.getAccounts(function callback1(error, result){
    web3.eth.getBalance(result[0], function callback2(error, value) {
        console.log("value" + value);
    });
 })
```

→

```
web3.eth.getAccounts()
   .then((res) => web3.eth.getBalance(res[0]))
   .then((value) => console.log(value) );
```

登链学院
edu.upchain.pro

Web3.js 使用

Geth 控制台

导出了web3对象，直接使用　geth 使用的0.2x.x 的版本

DApp 中使用

1. 先安装引入Web3

```
> npm install web3
> yarn add web3
> meteor add ethereum:web3

<script src="https://cdn.jsdelivr.net/gh/ethereum/web3.js/dist/web3.min.js"></script>
```

https://cdn.jsdelivr.net/gh/ethereum/web3.js/

登链学院
edu.upchain.pro

Web3.js 使用

DApp 中使用

使用MetaMask的Provider,不支持大部分同步API
https://github.com/MetaMask/faq/blob/master/DEVELOPERS.md

2. 提供Provider

```
if (typeof web3 !== 'undefined') {
  web3 = new Web3(web3.currentProvider);
} else {
  // set the provider you want from Web3.providers
  web3 = new Web3(new Web3.providers.HttpProvider("http://localhost:8545"));
}
```

https://cdn.jsdelivr.net/gh/ethereum/web3.js/

登链学院
edu.upchain.pro

Web3.js 使用

DApp 中使用

3. 使用web3 对象

如果本地节点，可以使用同步接口，
否则大部分接口仅能是使用异步接口，传递一个回调函数作为最后一个参数，
回调函数使用error-first 风格

```
web3.eth.getBlock(48, function(error, result){
   if(!error)
       console.log(JSON.stringify(result));
   else
       console.error(error);
})
```

https://github.com/xilibi2003/DAppDemos

以太坊去中心化应用（DApp）开发

Web3.js API介绍

熊丽兵（Tiny熊）

登链学院
edu.upchain.pro

Web3.js API使用

- 检查环境
- 发送交易
- 合约部署、加载合约、合约交互
- 监听合约事件

登链学院
edu.upchain.pro

Web3.js API使用

检查环境

MetaMask是否安装

```
window.addEventListener('load', function() {
  if (typeof web3 !== 'undefined') {
    web3 = new Web3(web3.currentProvider);
    if (web3.currentProvider.isMetaMask == true) {
        // "MetaMask可用";
    } else {
        // "非MetaMask环境"
    }
  } else {
        // 需要安装MetaMask
  }
}
```

登链学院
edu.upchain.pro

Web3.js API使用

检查环境

网络状态是否匹配

```
web3.version.getNetwork((err, netId) => {
    switch (netId) {
        case "1":
        case "2":
        default:
            console.log('This is an unknown network.');
    }
});
```

登链学院
edu.upchain.pro

Web3.js API使用

发送交易

检查钱包是否解锁

```
web3.eth.getAccounts(function (err, accounts) {
    // 检查accounts内容
})


var message = {from: fromAccount,
 to:toAccount,
 value: web3.toWei(amount, 'ether'),
 gas: gas,
 gasPrice: gasPrice
};
web3.eth.sendTransaction(message, (err, res) => {

});
```

登链学院
edu.upchain.pro

Web3.js API使用

合约部署、加载合约

```
var infoContract = web3.eth.contract(ABI);
// 加载合约
var info = infoContract.at('合约地址');
// 部署合约
infoContract.new({
        from: mySenderAddress,
        data: bytecode,
        gas: gasEstimate});
```

https://web3.learnblockchain.cn/0.2x.x/

登链学院
edu.upchain.pro

Web3.js API使用

调用合约函数

```
info.getInfo(function(error, result) {

});

info.setInfo(name, age, function(error, result) {
})
```

call调用

sendTranscation调用

https://web3.learnblockchain.cn/0.2x.x/web3.eth/#web3ethcontract

登链学院
edu.upchain.pro

Web3.js API使用

监听合约事件

合约

```
event EventName(uint param);

emit EventName(10);
```

DAPP

```
var e = contractInstance.EvevtName();
e.watch(function(err, result) {
   result.args.name;
})
```

https://web3.learnblockchain.cn/0.2x.x/web3.eth/#web3ethcontract

登链学院
edu.upchain.pro

Web3.js API总结

- 0.20.x vs 1.0
- 异步：回调使用
- MetaMask站点方式访问
- 多看文档

https://web3.learnblockchain.cn

登链学院
edu.upchain.pro

Web3.js API使用

- 检查环境
- 发送交易
- 合约部署、加载合约、合约交互
- 监听合约事件

以太坊去中心化应用（DApp）开发

补充：编辑器

熊丽兵（Tiny熊）

编辑器

Atom

https://atom.io/

插件：autocomplete-solidity、linter-solidity

VisualStudio Code

https://code.visualstudio.com/

插件：solidity、Solidity Extended、solidity-mac

以太坊去中心化应用（DApp）开发

Truffle详解

熊丽兵（Tiny熊）

登链学院
edu.upchain.pro

Truffle概述

什么是Truffle

脚手架： 集成合约编译、链接、测试、部署...

https://truffleframework.com/docs/truffle/overview

Remix 中编译拷贝ABI		truffle compile
Remix 中部署拷贝地址	⇨	truffle migrate
不易测试		truffle test

登链学院
edu.upchain.pro

Truffle概述

安装Truffle

```
npm install -g truffle
```

使用Truffle

```
truffle [command] [options]
```

init　unbox
compile　migrate
test　deploy

https://trufflframework.com/docs/truffle/reference/truffle-commands

登链学院
edu.upchain.pro

Truffle使用

使用Truffle改造前面的DAPP

创建工程

```
> mkdir firstTruffleDemo
> cd firstTruffleDemo
> truffle init
```

登链学院
edu.upchain.pro

Truffle使用

创建合约

> truffle create contract InfoContract

编写合约代码

编译合约

> truffle compile

登链学院
edu.upchain.pro

Truffle使用

部署（迁移）合约

1. 启动节点

2. 配置部署节点信息 https://truffleframework.com/docs/truffle/reference/configuration

3. 创建一个迁移脚本

```
> truffle create migration InfoContract
```

4. 编辑迁移脚本 https://truffleframework.com/docs/truffle/getting-started/running-migrations

5. 部署（迁移）

```
> truffle migrate —network
```

登链学院
edu.upchain.pro

Truffle使用

编写DAPP UI

创建src 文件夹存放前端文件

创建js 文件夹存放js代码

登链学院
edu.upchain.pro

Truffle使用

UI与合约交互

```
> npm init
> npm install truffle-contract
```

对web3进行了封装
引入了promise用法

通过生成的合约信息创建TruffleContract对象，然后deployed() 拿到合约对象。

https://truffleframework.com/docs/truffle/getting-started/interacting-with-your-contracts

登链学院
edu.upchain.pro

Truffle使用

```
var infoContract = web3.eth.contract(ABI);
// 加载合约
var info = infoContract.at('合约地址');
```

```
TruffleContract(infoArtifact).deployed()
  .then(function(instance) {

      });
```

登链学院
edu.upchain.pro

Truffle使用

- 加入事件监听

登链学院
edu.upchain.pro

Truffle使用

合约测试

1. 编写测试用例

2. 运行测试用例

```
> truffle test
```

https://truffleframework.com/docs/truffle/testing/writing-tests-in-solidity

登链学院
edu.upchain.pro

Truffle使用

给DAPP搭配一个服务器

```
npm install lite-server --save-dev
```

登链学院
edu.upchain.pro

Truffle小结

一个框架： 方便与合约交互、方便测试

一个脚手架： 集成合约编译、测试、部署

https://github.com/xilibi2003/DAppDemos

谢谢观看

欢迎关注微信号/公众号交流技术问题

登链学院

Tiny熊

以太坊去中心化应用（DApp）开发

案例讲解

熊丽兵（Tiny熊）

登链学院
edu.upchain.pro

案例一

发行一个Token及编写钱包DAPP

官方提供了tutorialtoken boxes

```
> truffle unbox tutorialtoken
```

登链学院
edu.upchain.pro

案例二

投票DAPP

https://github.com/maheshmurthy/ethereum_voting_dapp

登链学院
edu.upchain.pro

Truffle案例介绍

投票DAPP优化

解决投票无成本，一人多票

truffle migrate —reset

{value: weiValue}

登链学院
edu.upchain.pro

Truffle案例介绍

投票DAPP优化

引入库， 编译库，链接库

```
deployer.deploy(safemath);
deployer.link(safemath, voting);
deployer.deploy(voting, ["Rama", "Nick", "Jose"]);
```

https://github.com/xilibi2003/DAppDemos

文档链接：https://truffleframework.com/docs/truffle/getting-started/running-migrations

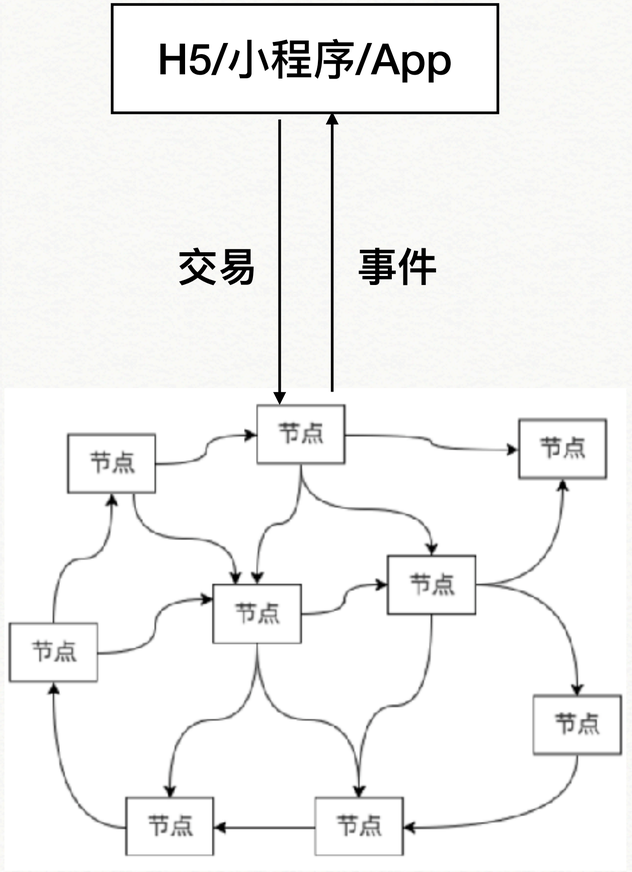
以太坊去中心化应用（DApp）开发

课程总结

熊丽兵（Tiny熊）

登链学院
edu.upchain.pro

恭喜大家学完课程



coding

https://github.com/xilibi2003/DAppDemos

https://web3.learnblockchain.cn/

https://trufflеframework.com/

谢 谢

课程助理：晓娜