

0502

Solidity及一些開發工具簡介

最基本的那種介紹

by 劉其峰、黃俊閔

課程投影片及程式碼：

<https://goo.gl/1TxxTZ>

Outline

- Solidity簡介
- 開發環境 - Remix (Browser - Solidity)
- Ether 錢包 - Metamask 插件安裝及界面介紹
- 智慧合約的撰寫 using Solidity
- 合約部署
- 查看合約在以太坊測試鏈中的狀態

Solidity 簡介

- 最常用的智能合約語言
- 可以編譯成Bytecode在EVM上執行
- 程式只能在blockchain上測試和執行
- 一經佈署, 程式將永久存在
- 可以透過Mist、Parity、Metamask等錢包軟體來佈署

Solidity特性

- 物件導向程式語言
- 有各種型態, 但沒有float([原因](#))
- [說明文件](#)很完整

其他語言

- Serpent - Python-like
- LLL - Lisp-like

Remix(Browser - Solidity)環境介紹

- 線上編譯器，程式碼寫好後可以在自己的記憶體，也可以透過某些錢包軟體進行合約佈署。

檔案頁籤

寫程式的地方

```
ballot.sol x
1 pragma solidity ^0.4.0;
2 contract Ballot {
3
4     struct Voter {
5         uint weight;
6         bool voted;
7         uint8 vote;
8         address delegate;
9     }
10    struct Proposal {
11        uint voteCount;
12    }
13
14    address chairperson;
15    mapping(address => Voter) voters;
16    Proposal[] proposals;
17
18    /// Create a new ballot with $( _numProposals ) different proposals.
19    function Ballot(uint8 _numProposals) {
20        chairperson = msg.sender;
21        voters[chairperson].weight = 1;
22        proposals.length = _numProposals;
23    }
24
25    /// Give $(voter) the right to vote on this ballot.
26    /// May only be called by $(chairperson).
27    function giveRightToVote(address voter) {
28        if (msg.sender != chairperson || voters[voter].voted) return;
29        voters[voter].weight = 1;
30    }
31
32    /// Delegate your vote to the voter $(to).
33    function delegate(address to) {
34        Voter sender = voters[msg.sender]; // assigns reference
35        if (sender.voted) return;
36        while (voters[to].delegate != address(0) && voters[to].delegate != msg.sender)
37            to = voters[to].delegate;
38        if (to == msg.sender) return;
39        sender.voted = true;
```

Settings Files Contract Debugger Analysis Docs

Current Solidity version:

0.4.10+commit.f0d539ae.Emscripten.clang

Switch version: [Click to select new compiler version](#)

☐ Text Wrap ☐ Enable Optimization ☒ Auto Compile

Compile

設定編譯器處：

setting頁面可選擇solidity版本

ballot.sol

```
1 pragma solidity ^0.4.0;
2 contract Ballot {
3
4     struct Voter {
5         uint weight;
6         bool voted;
7         uint8 vote;
8         address delegate;
9     }
10    struct Proposal {
11        uint voteCount;
12    }
13
14    address chairperson;
15    mapping(address => Voter) voters;
16    Proposal[] proposals;
17
18    /// Create a new ballot with $( _numProposals ) different proposals.
19    function Ballot(uint8 _numProposals) {
20        chairperson = msg.sender;
21        voters[chairperson].weight = 1;
22        proposals.length = _numProposals;
23    }
24
25    /// Give $(voter) the right to vote on this ballot.
26    /// May only be called by $(chairperson)
27    function giveRightToVote(address voter) {
28        if (msg.sender != chairperson || voters[voter].voted) return;
29        voters[voter].weight = 1;
30    }
31
32    /// Delegate your vote to the voter $(to).
33    function delegate(address to) {
34        Voter sender = voters[msg.sender]; // assigns reference
35        if (sender.voted) return;
36        while (voters[to].delegate != address(0) && voters[to].delegate != msg.sender)
37            to = voters[to].delegate;
38        if (to == msg.sender) return;
39        sender.voted = true;
40        sender.delegate = to;
41        Voter delegate = voters[to];
42        if (delegate.voted)
43            proposals[delegate.vote].voteCount += sender.weight;
44        else
45            delegate.weight += sender.weight;
46    }
47
48    /// Give a single vote to proposal $(proposal).
49    function vote(uint8 proposal) {
50        Voter sender = voters[msg.sender];
51        if (sender.voted || proposal >= proposals.length) return;
52        sender.voted = true;
53        sender.vote = proposal;
```

Transaction origin: 0xca35b7d915458e540ade6068dfe2f44e8fa733c

3000000 Transaction gas limit 交易手續費設定

0 Value (e.g. .7 ether or 5 wei, defaults to ether) 此合約所要附帶的ether數

Select execution environment:

JavaScript VM 選擇執行環境

Publish Attach Transact Transact (Payable) Call

ballot.sol:Ballot 2139 bytes

At Address Create 創造一個新合約

uint8 _numProposals

Bytecode

Interface

Web3 deploy

Metadata location

Toggle Details

6060604052341561000c57fe5b60405160208061085b8339810160405280805190602001909190501

[{"constant":false,"inputs":[{"name":"to","type":"address"}],"name":"delegate","outputs":[],"payabl

```
var _numProposals = /* var of type uint8 here */ ;
var ballot_sol_ballotContract = web3.eth.contract([{"constant":false,"inputs":[{"name":"to","type":"address"}],"name":"delegate","outputs":[],"payabl
var ballot_sol_ballot = ballot_sol_ballotContract.new(
    _numProposals,
    {
        from: web3.eth.accounts[0],
        data: '0x6060604052341561000c57fe5b60405160208061085b833981016040528080519
        gas: '4700000'
    }, function (e, contract){
        console.log(e, contract);
        if (typeof contract.address !== 'undefined') {
            console.log('Contract mined! address: ' + contract.address + ' transac
        }
    })
```

bzzr://120f7413c8d0002988e5ba45d95209cc98419237f763bc21c03ba6f34f91a9b

依據合約程式碼所產生的一串編碼，會存在創造合約的交易的資料

合約的ABI，只要知道合約地址跟此ABI就可以執行該合約

有點複雜，有機會說明




Metamask安裝及介面介紹


官網 [Chrome插件商店](#) (沒裝Chrome的話[這裡裝](#))





錢包的一種，其他還有Mist、Myethereallet、Parity也都有錢包的功能



在Metamask中申請以太幣



 Ropsten Test Net METAMASK  


 **fintech**
0x50A7A...
15.947 ETH
1330.60 USD


   


BUY **SEND**




HISTORY

 **April 26 2017 17:07**
Contract Published  0 ETH


 **April 21 2017 14:53**
Contract Published **(Rejected)** 0 ETH

 **April 21 2017 14:53**
Contract Published **(Rejected)** 0 ETH

 **April 21 2017 14:50**

 Ropsten Test Net METAMASK  

BUY ETH

 **test...**

ADDRESS 0x1b1c2788...4499
BALANCE 3.977 ETH

SELECT SERVICE

☒ **Coinbase**- Crypto/FIAT (USA only)
☐ **ShapeShift**- Crypto

COINBASE

**IN ORDER TO ACCESS THIS
FEATURE, PLEASE SWITCH
TO THE MAIN NETWORK**

OR GO TO THE
ROPSTEN TEST FAUCET

MetaMask Ether Faucet

faucet

address: 0x18a3462427bcc9133bb46e88bcbe39cd7ef0e761
balance: 1097458.3376518528 ether

request 1 ether from faucet

user

address: 0x1b1c2788a0c73c123bb68cdd9fb4beccaa1a4499
balance: 3.9775298503193155 ether
donate to faucet:

1 ether

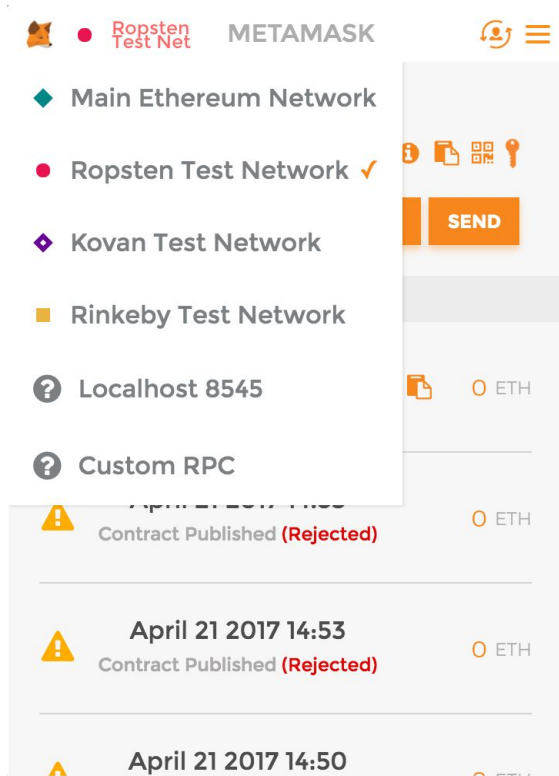
10 ether

100 ether

transactions

0x2cc0fe02c65b4d95c8c36791900b298e3cb3037b745c0ade1d02e03d7ea9ffd0

選擇網路



The screenshot shows the Metamask interface with the network selection menu open. The menu lists several options: Main Ethereum Network, Ropsten Test Network (selected with a red dot and a checkmark), Kovan Test Network, Rinkeby Test Network, Localhost 8545, and Custom RPC. Below the menu, there are transaction history entries, including one labeled 'Contract Published (Rejected)' and another dated 'April 21 2017 14:53' also labeled 'Contract Published (Rejected)'.

● Ropsten Test Net METAMASK

- ◆ Main Ethereum Network
- Ropsten Test Network ✓
- ◆ Kovan Test Network
- Rinkeby Test Network
- ⓘ Localhost 8545
- ⓘ Custom RPC

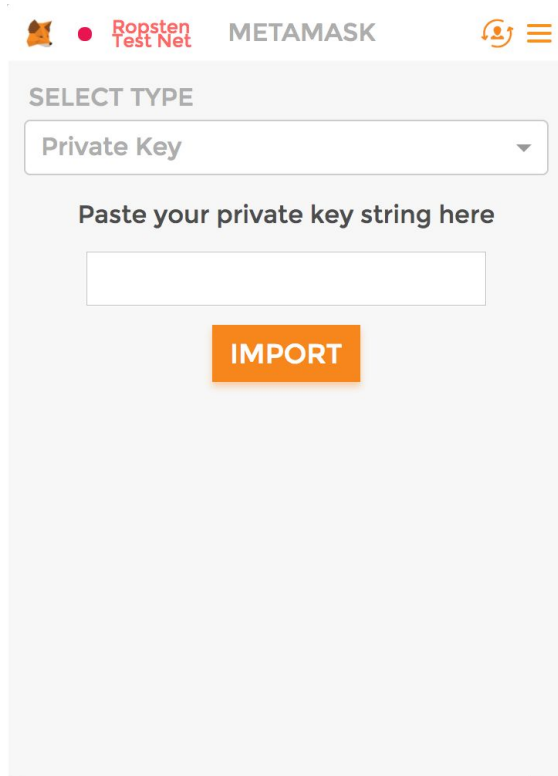
SEND

Contract Published (Rejected)

April 21 2017 14:53
Contract Published (Rejected)

April 21 2017 14:50

導入帳戶



The screenshot shows the 'Import Account' screen in Metamask. It features a 'SELECT TYPE' dropdown menu currently set to 'Private Key'. Below the dropdown is a text input field with the placeholder text 'Paste your private key string here'. An orange 'IMPORT' button is positioned below the input field.

SELECT TYPE

Private Key

Paste your private key string here

IMPORT

撰寫智能合約 using Solidity

- 宣告合約版本 `1 pragma solidity ^0.4.2;`

- 合約框架

```
3 contract example{
4
5     //---
6     //其他內容
7     //---
8
9     //建構子
10    function example(){
11
12    }
13
14    //---
15    //其他內容
16    //---
17
18 }
```

型態(type)

`bool`
`int8, int16, int24...int256`
`uint8, uint16, uint16... uint256`
`address`
`string`
`byte1, byte2~byte32`

結構(struct)

```
struct Person{  
    string name;  
    uint age;  
    bool sex;  
}
```

enum(合約狀態)

```
enum State{Left, Right, Forward}
```

HelloWorld

```
pragma solidity ^0.4.0;
```

```
contract helloworld{
```

```
    string storeData;
```

```
    function set(string x) {
```

```
        storeData = x;  }
```

```
    function get() constant returns (string data) {
```

```
        return storeData;  }
```

```
}
```

Solidity 的繼承 (以Hello World程式碼作延伸)

```
pragma solidity ^0.4.1;

contract mortal {
    address owner;

    function mortal() {
        owner = msg.sender;    }

    function kill() {
        if (msg.sender == owner)
            selfdestruct(owner);    }
}
```

```
contract helloworld is mortal{
    string storeData;

    function set(string x) {
        storeData = x;    }

    function get() constant returns (string data)
    {
        return storeData;    }
}
```

Solidity 的 return寫法

```
pragma solidity ^0.4.9;

contract Back{

    function Back(){}

    uint tempR;
    function test(uint a1) returns(uint x){
        tempR = a1;
        return tempR;
    }

    uint tempR1;
    uint tempR2;

    function test2(uint b1, uint b2) returns(uint x1, uint x2){
        tempR1 = b1;
        tempR2 = b2;
        return (tempR1, tempR2);
    }
}
```


code

goo.gl/2r0xj9

modifier語法(用以簡化程式碼)

```
modifier onlyIf{  
    if(condition) execution;  
    ;  
}
```

+

```
function imFunction() onlyIf{  
    //function contents  
}
```

||

```
function imfunction(){  
    if(condition) execution;  
    //function contents  
}
```

```
modifier onlyIf{  
    ;  
    if(condition) execution;  
}
```

+

```
function imFunction() onlyIf{  
    //function contents  
}
```

||

```
function imfunction(){  
    //function contents  
    if(condition) execution;  
}
```

如何使用他人的合約？

知道：

- Contract ABI
- Contract Address

=>就可以跟他人使用同一份合約

Etherscan.io

查看以太坊資訊瀏覽網站,

可以看到主鏈及幾條測試鏈上的區塊、交易、合約狀態。



MARKET CAP OF \$6,837,512,312

\$74.98 @ 0.0561 BTC/ETH (+3.37%)

LAST BLOCK

3627525 (15.34s Avg)

Hash Rate

21,783.75 GH/s

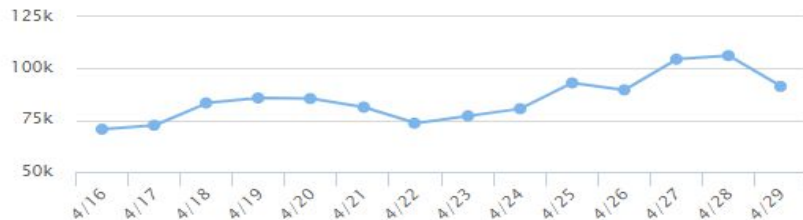
TRANSACTIONS

22472460

Network Difficulty

332.77 TH

14 day Ethereum Transaction History



Blocks

View All

Block 3627525

> 37 secs ago

Mined By [f2pool](#)

3 txns IN 2 secs

Block Reward 5.00306 Ether

Block 3627524

> 39 secs ago

Mined By [Nanopool](#)

16 txns IN 6 secs

Block Reward 5.00789 Ether

Block 3627523

> 45 secs ago

Mined By [DwarfPool1](#)

0 txns IN 3 secs

Block Reward 5 Ether

Block 3627522

Mined By [Ethernine](#)

Transactions

View All



TX# 0X65650BB8650739EAD4EC996...

> 37 secs ago

From 0xfbb1b73c4f0bda4f... To 0xf7b098298f7c69fc...

Amount 0 Ether



TX# 0X62A99E3B631ECC981EFC2B89..

> 37 secs ago

From 0xfbb1b73c4f0bda4f... To 0xa74476443119a94...

Amount 0 Ether



TX# 0X52B028EE9A576AF215503DEA..

> 37 secs ago

From 0x12054334971d034... To 0x57b174839cbd0a5...

Amount 0.25493812 Ether



TX# 0XFA07902CC2BACC7547CBD43...

> 39 secs ago

區塊查詢 <https://etherscan.io/block/3627485>



LOGIN

Search by Address / Txhash / Block / Token

GO

LANGUAGE

HOME

BLOCKCHAIN

ACCOUNT

TOKEN

CHART

MISC

Block #3627485

Home / Blocks / Block Information

Overview

Block Information

Height:	< Prev 3627485 Next >
TimeStamp:	9 hrs 48 mins ago (Apr-30-2017 05:15:56 PM +UTC)
Transactions:	34 transactions and 2 contract internal transactions in this block
Hash:	0xcaae3a49e428c8e185f81dd39028fc8fc9156ceb549e2e85c4aca1e397afdbf1
Parent Hash:	0x476a0ea70273ce4b739979ae253a661033c90019a5724fc2d3043152cef3e14a
Sha3Uncles:	0x1dcc4de8dec75d7aab85b567b6ccd41ad312451b948a7413f0a142fd40d49347
Mined By:	0xea674fdde714fd979de3edf0f56aa9716b898ec8 (Ethermine) IN 25 secs
Difficulty:	332,500,562,504,872

交易查詢

[LOGIN](#)[GO](#)[LANGUAGE](#)[HOME](#)[BLOCKCHAIN](#)[ACCOUNT](#)[TOKEN](#)[CHART](#)[MISC](#)

Transactions :: For Block 3627485

[Home](#) / [Transactions](#)

A total of 34 transactions found

[First](#) [Prev](#) [Page 1 of 2](#) [Next](#) [Last](#)

TxHash	Block	Age	From		To	Value ↓	[TxFee]
0x6ba6df720bb45ef4...	3627485	9 hrs 51 mins ago	Ethermine	➡	0x8e162421aceb34f...	1.0009483 Ether	0.0004589
0x6a2957281d58f87...	3627485	9 hrs 51 mins ago	Ethermine	➡	0x273b099884ec557...	0.09617263 Ether	0.00042
0xa1102bd2aade27c...	3627485	9 hrs 51 mins ago	Ethermine	➡	0xc1c427cd8e6b7ee...	1.01271693 Ether	0.00042
0xe0ce04743944ce0...	3627485	9 hrs 51 mins ago	Ethermine	➡	0x790b03860dce2f2...	1.00024728 Ether	0.00042
0x342098441fd4b32f...	3627485	9 hrs 51 mins ago	Ethermine	➡	0x0d0171d19cc259d...	0.27728389 Ether	0.00042
0xb355d7510b1d81b...	3627485	9 hrs 51 mins ago	Ethermine	➡	0xbdd8fee4c509e3d...	1.00382994 Ether	0.00042
0xca4e9cefb67d9cf9...	3627485	9 hrs 51 mins ago	Ethermine	➡	0x4f6847cd4a7c4fa9...	0.09738316 Ether	0.00042
0x94c57f872bc5434...	3627485	9 hrs 51 mins ago	Ethermine	➡	0x482d68e98537f72...	1.00006538 Ether	0.00042

進階開發環境

- 安裝Parity
- 進入開發鏈: `parity ui --chain dev`
- 進入測試鏈(Koven): `parity ui --testnet`
- 進入測試鏈(Ropsten): `parity ui --chain=ropsten`

比較進階的例子

- 牽涉以太幣交易的合約
- 使用Event來記錄合約的log
- 使用enum來改變合約的狀態