國立政治大學資訊管理學系所
Department of Management Information Systems, NCCU

區塊鏈與智能合約應用

從技術細節到商業模式

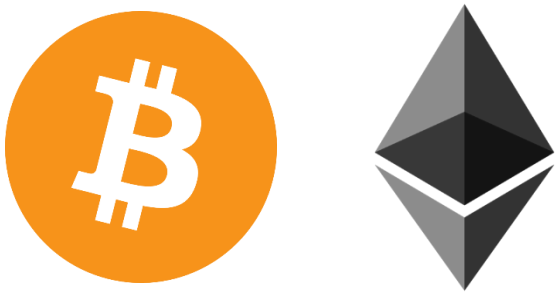
蕭舜文 Shun-Wen Hsiao Ph.D.
hsiaom@nccu.edu.tw

Assistant Professor
Department of Information Management Systems
National Chengchi University, Taiwan

NATIONAL CHENGCHI UNIVERSITY

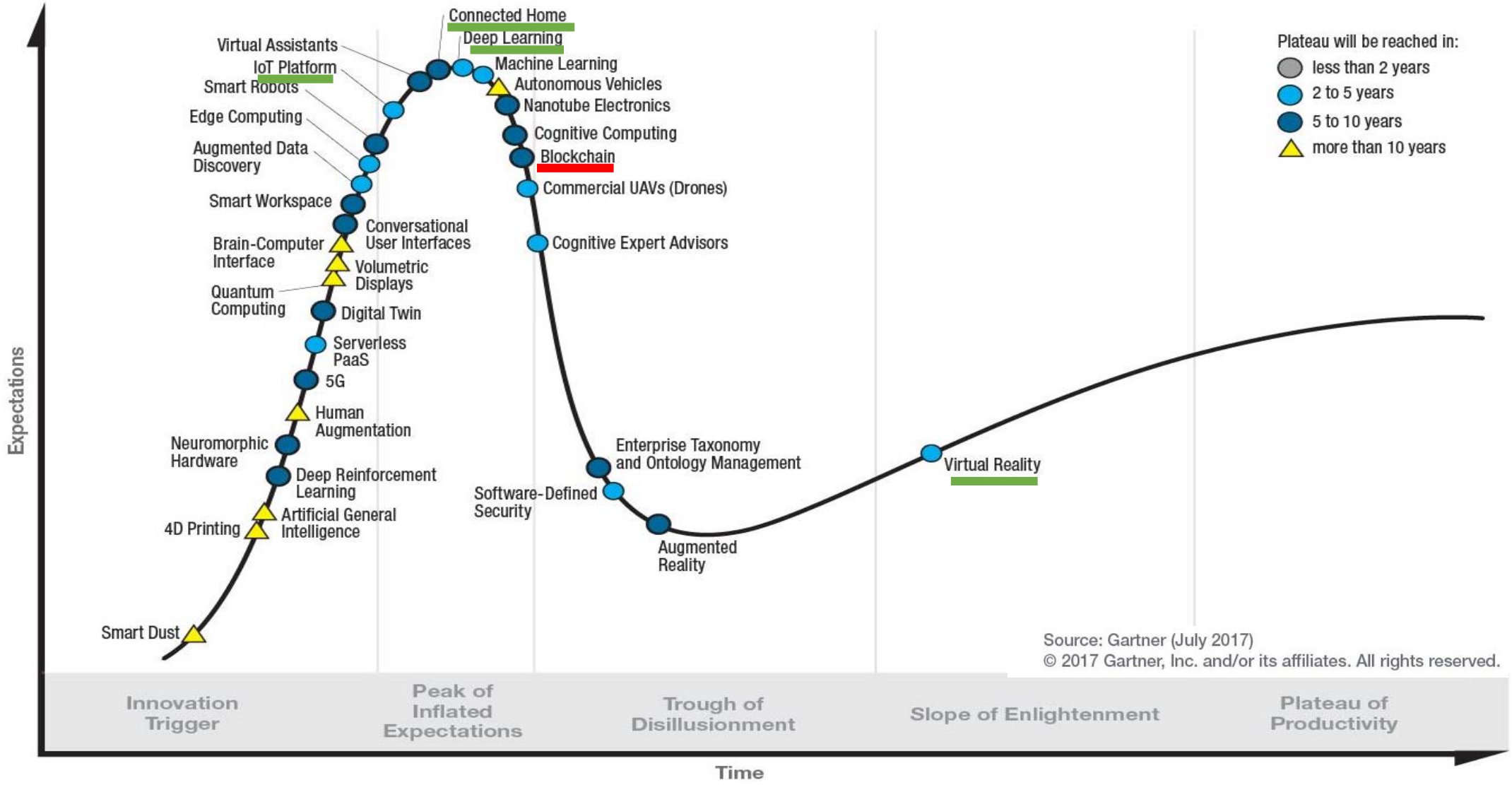
FinTech

- **[Wiki]** Financial technology, also known as FinTech, is an industry composed of companies that use new technology and innovation to leverage available resources in order to compete in the marketplace of traditional financial institutions and intermediaries in the delivery of financial services.



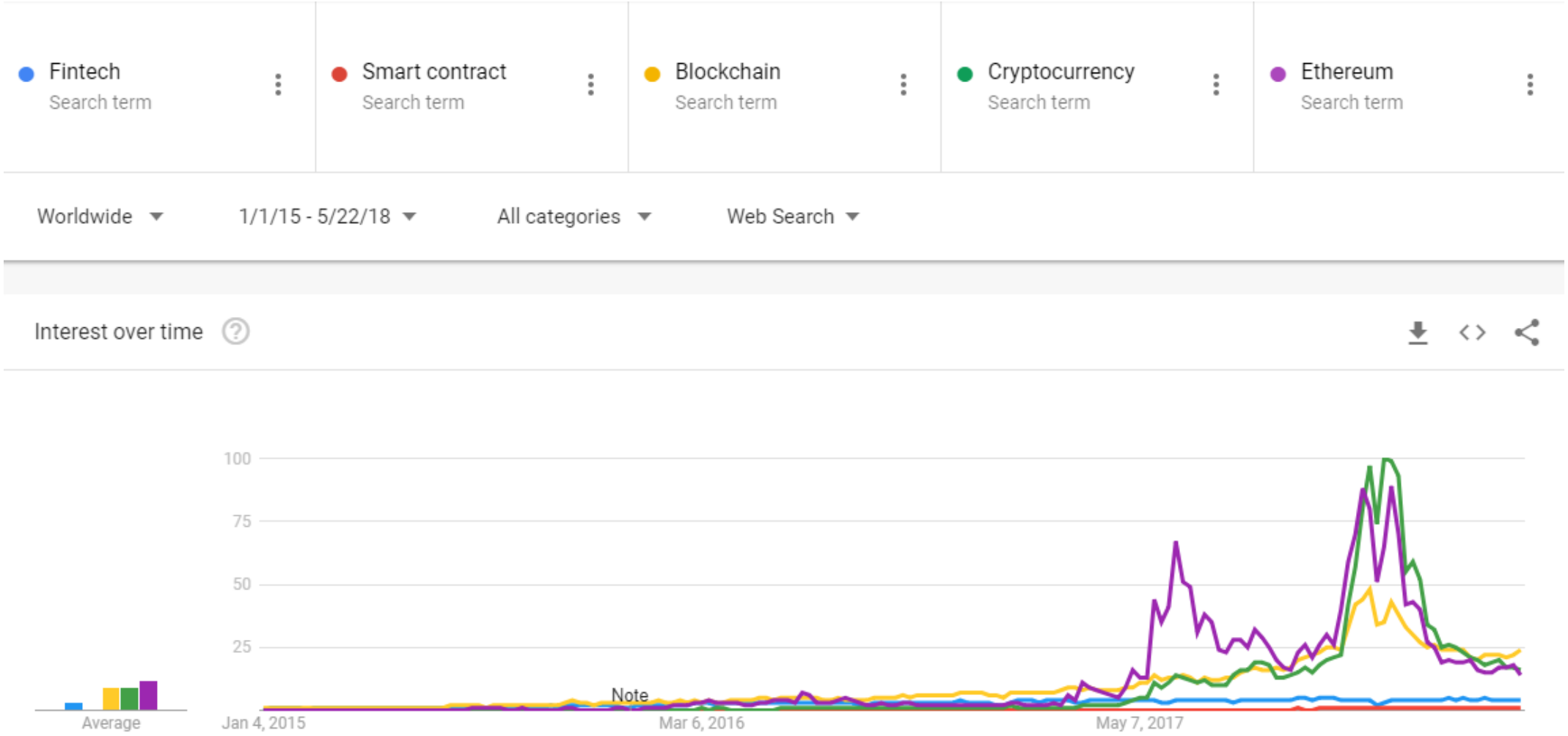
https://en.wikipedia.org/wiki/Financial_technology

Gartner Hype Cycle for Emerging Technologies



Source: Gartner (July 2017)

Google Trends



https://www.google.com.tw/trends/ @ 2018/05/22

Hashcash

- Computing power is the valuable resource, and it should not be wasted.
- If I can prove that I already consume certain amount of my computation power in order to do something next, then you should pay more attention to what I say. But how can I prove that?

- **[Wiki] Hashcash** is a proof-of-work system used to limit email spam and denial-of-service attacks.
  - Email sender performs a small amount of computational work (proof-of-work) and attach the proof in each email.
  - For spammers, the aggregated work is expensive.

http://wp.xin.at/archives/tag/hashcash

X-Hashcash

recipient's email address

a magic counter to prove the spent computation

```
X-Hashcash: 1:20:1303030600:adam@cypherspace.org::McMybZIhxKXu57jd:ckvi
```

The header contains:

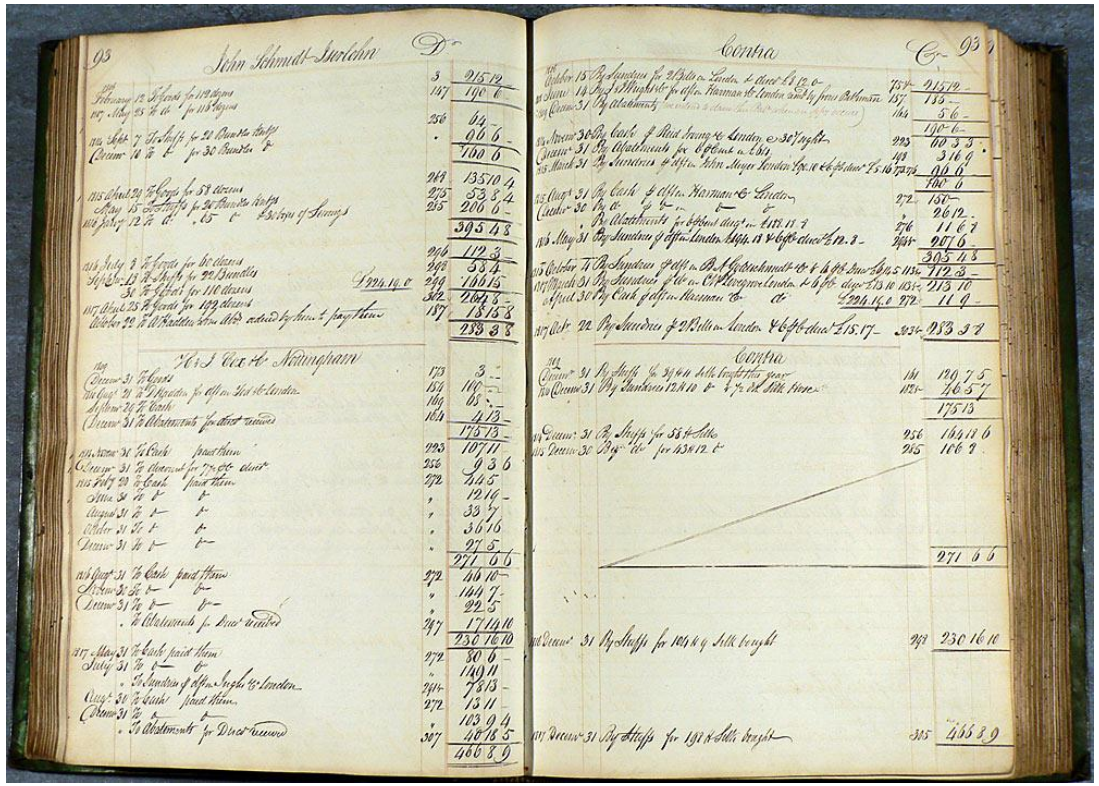
- *ver*: Hashcash format version, 1 (which supersedes version 0).
- *bits*: Number of "partial pre-image" (zero) bits in the hashed code.
- *date*: The time that the message was sent, in the format `YYMMDD[hhmm[ss]]`.
- *resource*: Resource data string being transmitted, e.g., an IP address or email address.
- *ext*: Extension (optional; ignored in version 1).
- *rand*: String of random characters, encoded in base-64 format.
- *counter*: Binary counter (up to 2^{20}), encoded in base-64 format.

https://en.wikipedia.org/wiki/Hashcash

What is blockchain?

Blockchain is not the database that you think about.

- A blockchain — originally, block chain —is a distributed database that maintains a continuously-growing list of data records hardened against tampering and revision.
- Blockchains are "an open, distributed ledger that can record transactions between two parties efficiently and in a verifiable and permanent way.



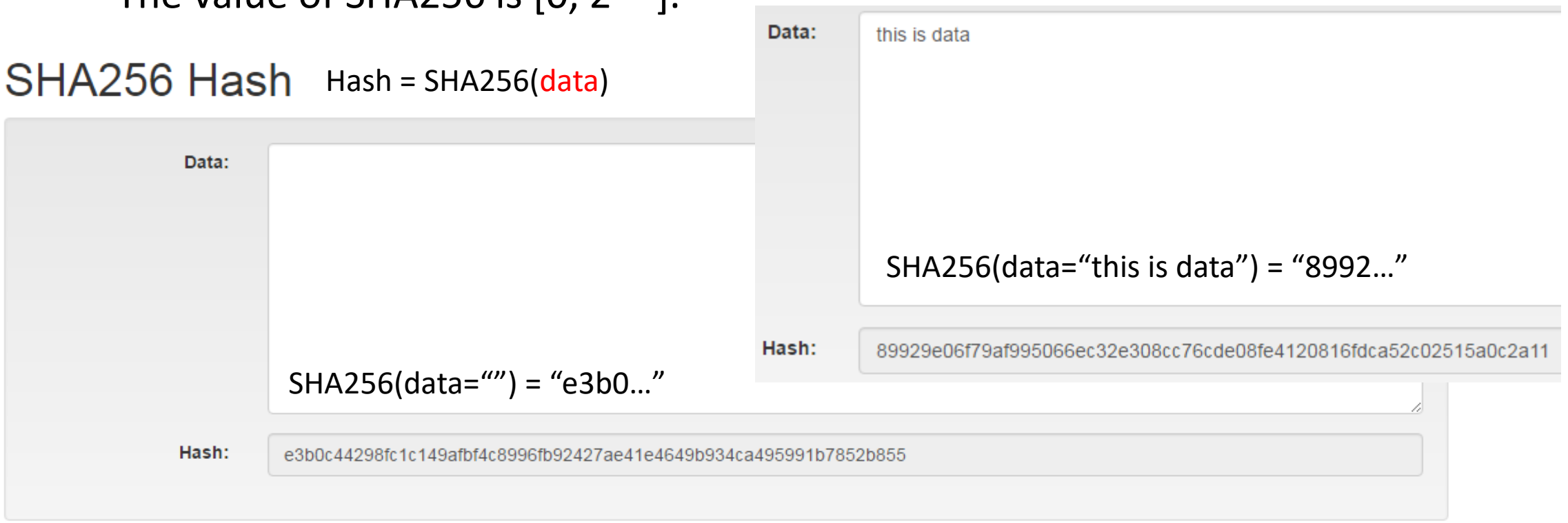
Blockchain Demo

- http://anders.com/blockchain/

- Hash
- Block
- Blockchain
- Distributed
- Tokens
- Coinbase

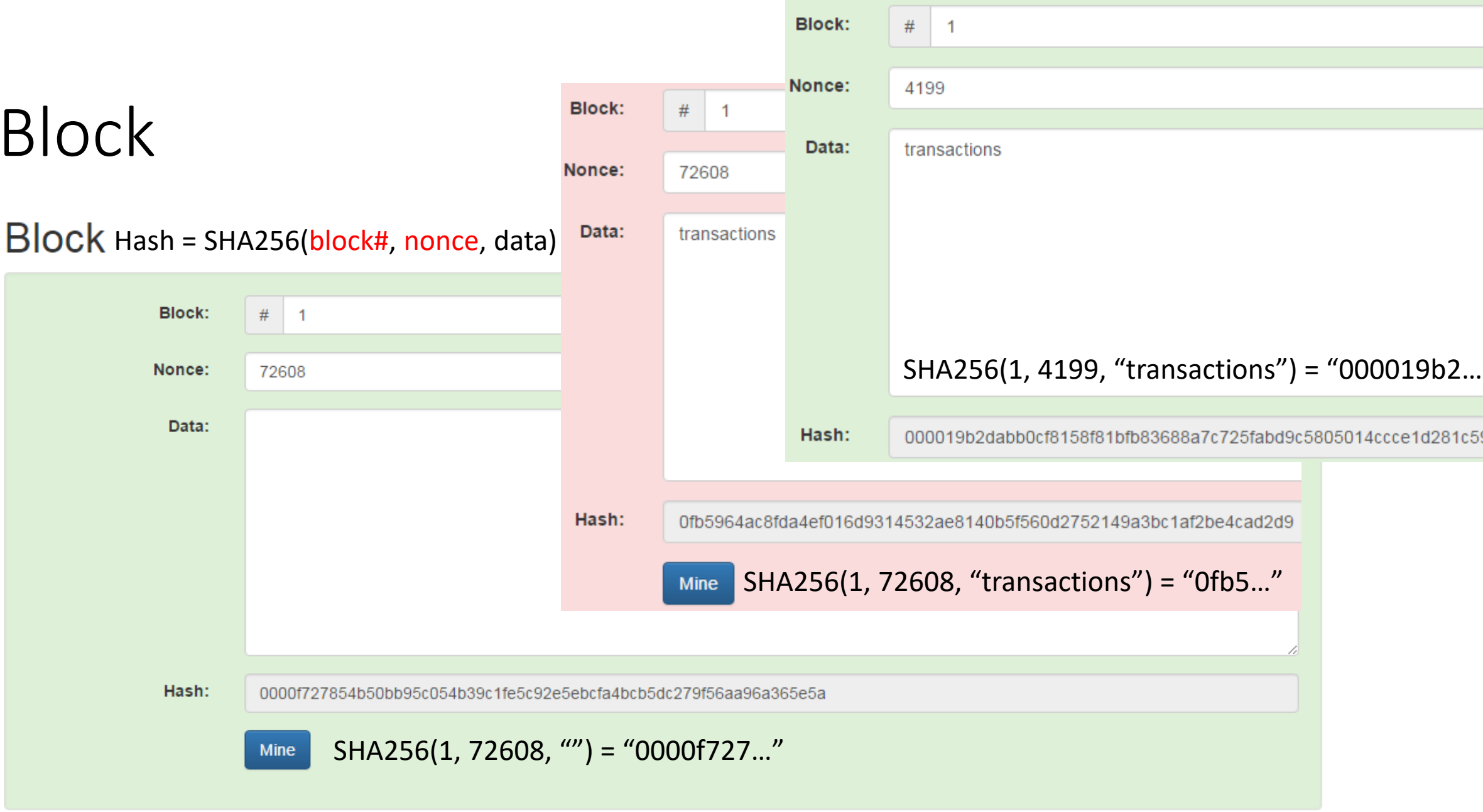
Hash: digital fingerprint

- Properties: determinism, uniformity, defined range, non-invertible.
  - The value of SHA256 is [0, 2^{256}].



Block

Block Hash = SHA256(block#, nonce, data)



A valid (signed) block must has a hash value below a target value.

Uniformity

Hash = SHA256(block#, nonce, data)

Nonce	Hash
0	baa53f13fd719bd4783df3f5a701307fe07f58f0429b7cf354df675574155700
1	ed1aa5ac024adacc4c24dc36e5cd80686e98182cd243ca2c88da045879c73c0a
2	78b9a7ce2aa42804d0467928ced9379c1d46e0be90c682cc964239ec6ea4376a
3	c2a14d556096530956169 4b612cd20eba80680512aae807a450f6a81ba51a6f5
4	1c36ff1639ea413d1f0dfffab93ee30ac150c86f95c6d3b838691f0655c3e8e6
5	226dbc8b98fc30c11b54396eee60706ff00114b20eade097c83bd43fe19b9dc1
6	40f27b239f0e2318c38cc0e658d58b318e318723656e4a6e1236c830725e6367
7	562574f76f44baa039b0146ac1ddfea791ad4197d31c78bb5f30132bf4bc9216
8	df143a97d757d582060362247c92f11cec1063a213059d01cbed4b2eedc8c49d
9	585971d555cafcd1746f1d22b2dc35c6409f7b3d6cec3f0d929bc0552c1c2ff4
⋮	

Nonce	Hash
4194	fe090988c2a7222829c0a43e35e4a4c324529dd3a55bdaf21f4664313da5420c
4195	52c385a291350a2a7972869a8363438900ee592980d30994de5373486ada6201
4196	9a2509ec716a31b40d931834fe26dc6eaec1feb433930f91630b1990abf08a35
4197	54d573689aa78f34cf0bfe3f1298c6b6217aefc3284b3e79fe536a8924a0e2b5
4198	34b2dcffea74ecb4778db49ccba4235a7be8b06d6af6a5e642de82950f685c3a
4199	**000019b2dabb0cf8158f81bfb83688a7c725fabd9c5805014ccce1d281c59333**
4200	06eddd5bf4748ac7205ec6acd16a4dd75dc34a9657e2246d71c9b37fe3d9d93d
4201	5042fb839757a7f6fd11f82b263978185e0280607dd6fa01823c32ea45baa4ab
4202	e31220f70538a429a939da8aede308452f8d32b7d2b3671e5b708c062107cc22
4203	5e7d84f9b7220cf064670141112297edf4f1a97ee7ed7fcb904079caf21260ee
⋮	

Block (cont'd) Hash = SHA256(block#, nonce, data)

- A valid (signed) block
  - A valid block must has a hash value below a target value, e.g., leading 32 bits are zero.
  - Since you cannot change block index and data, you can create a valid block by only selecting a nonce value in $[0, 2^{32}]$. The process is called "mining".
  - Because of "uniformity", mining is a mathematic lottery.
  - Easy to verify; difficult to mine.

https://en.bitcoin.it/wiki/Difficulty

Sudoku

- Sudoku can be viewed as a mathematical problem that is difficult to find its solution.
- It is easy to verify the correctness of a solution independently.
- The difficulty of a sudoku problem can be adjusted easily.

- Proof of Work (PoW): if a solution is found, he/she must perform some computational works.

		3	9			7	6	
	4				6			9
6		7		1				4
2			6	7			9	
		4	3		5	6		
	1			4	9			7
7				9		2		1
3			2				4	
	2	9			8	5		

Genesis Block

- A genesis block is the first block of a block chain.
  - Modern versions of Bitcoin assign it block number 0, though older versions gave it to number 1.
- The genesis block is almost always hardcoded into the software.
- It is a special case in that it does not reference a previous block.
- This block contains the dated title of a Times article.


THE TIMES

£1.50

Eat Out from £5

More than 900 great restaurants, including four Gordon Ramsay favourites from £15

Israel prepares to send tanks and troops into Gaza

Michael Sheen: Frost, Nixon and me

Working mums: So that's how she does it

Detox in style: The best spas on the planet

Salman Rushdie: I won't marry again

Giant killing? Guide to the FA Cup third round

Chancellor on brink of second bailout for banks

Billions may be needed as lending squeeze tightens

99p

The Times 03/Jan/2009 Chancellor on brink of second bailout for banks

https://en.bitcoin.it/wiki/Genesis_block

Block #0

Summary	
Number Of Transactions	1
Output Total	50 BTC
Estimated Transaction Volume	0 BTC
Transaction Fees	0 BTC
Height	0 (Main Chain)
Timestamp	2009-01-03 18:15:05
Received Time	2009-01-03 18:15:05
Relayed By	Unknown
Difficulty	1
Bits	486604799 (0x1d00ffff)
Size	0.285 KB
Version	1
Nonce	2083236893
Block Reward	50 BTC

Hashes	
Hash	000000000019d6689c085ae165831e934ff763ae46a2a6c172b3f1b60a8ce26f
Previous Block	0000000000000000000000000000000000000000000000000000000000000000
Next Block(s)	00000000839a8e6886ab5951d76f411475428afc90947ee320161bbf18eb6048
Merkle Root	4a5e1e4baab89f3a32518a88c31bc87f618f76673e2cc77ab2127b7afdeda33b

Network Propagation

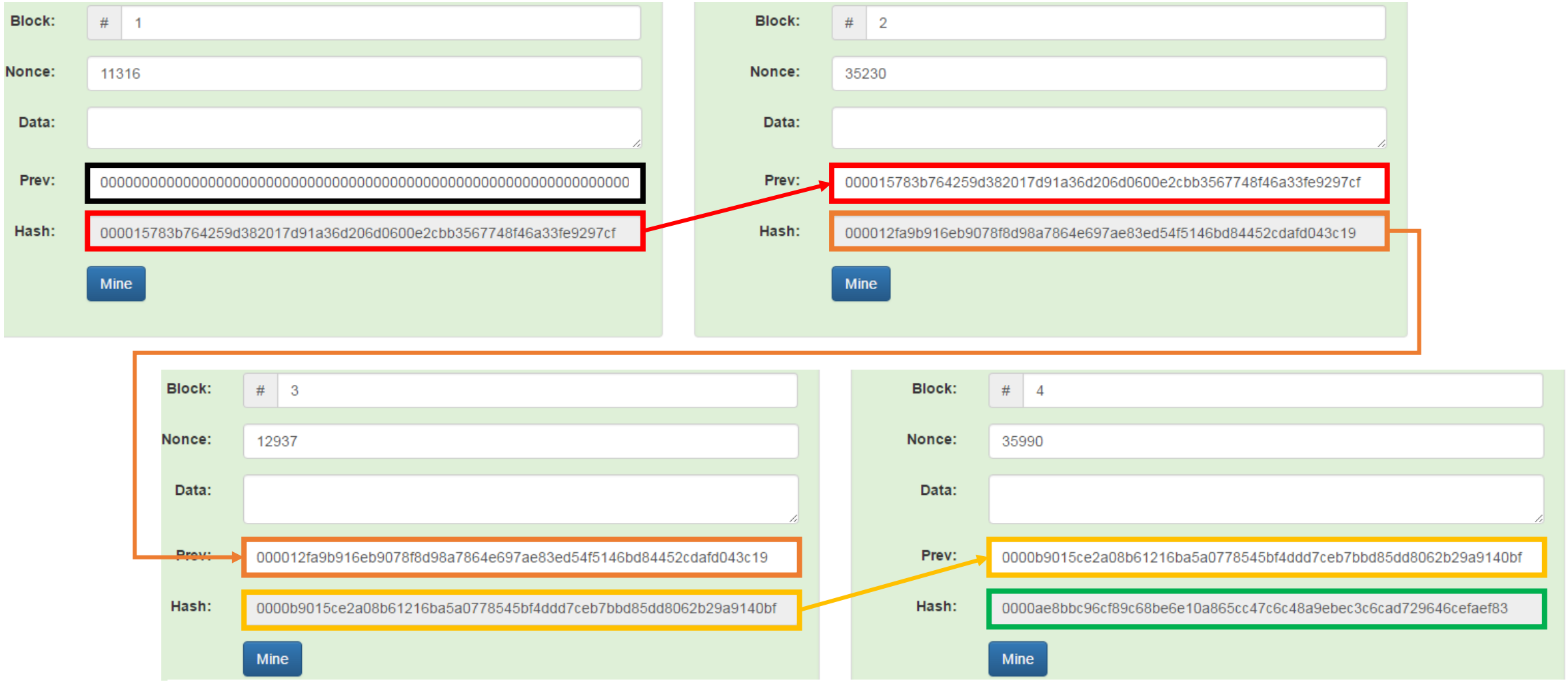
https://blockchain.info/block-height/0

Transactions

4a5e1e4baab89f3a32518a88c31bc87f618f76673e2cc77ab2127b7afdeda33b			2009-01-03 18:15:05
No Inputs (Newly Generated Coins)	→	1A1zP1eP5QGefi2... (Genesis of Bitcoin)	50 BTC

Blockchain

Hash = SHA256(block#, nonce, data, prev)



Chain Reaction

- If you change/mutate the data in a block, you have to re-mine all the following blocks to keep all blocks valid.

- The more blocks that go by, the harder and harder it is to make a change to the past data. That is how a blockchain resists mutation and change.

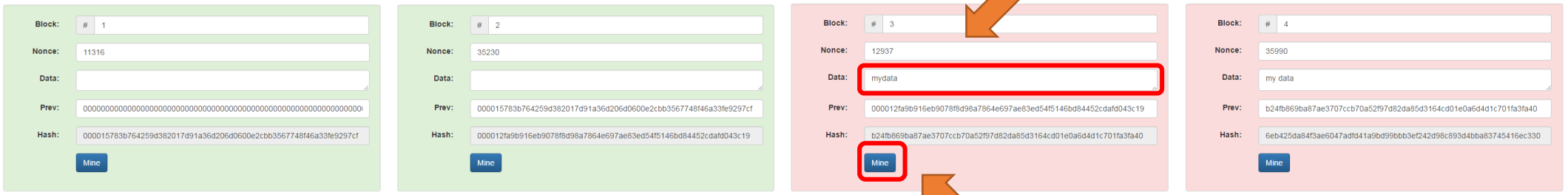
- Question: How do you know your blockchain is re-mined?

Distributed Blockchain

Peer A

Peer B

Peer C

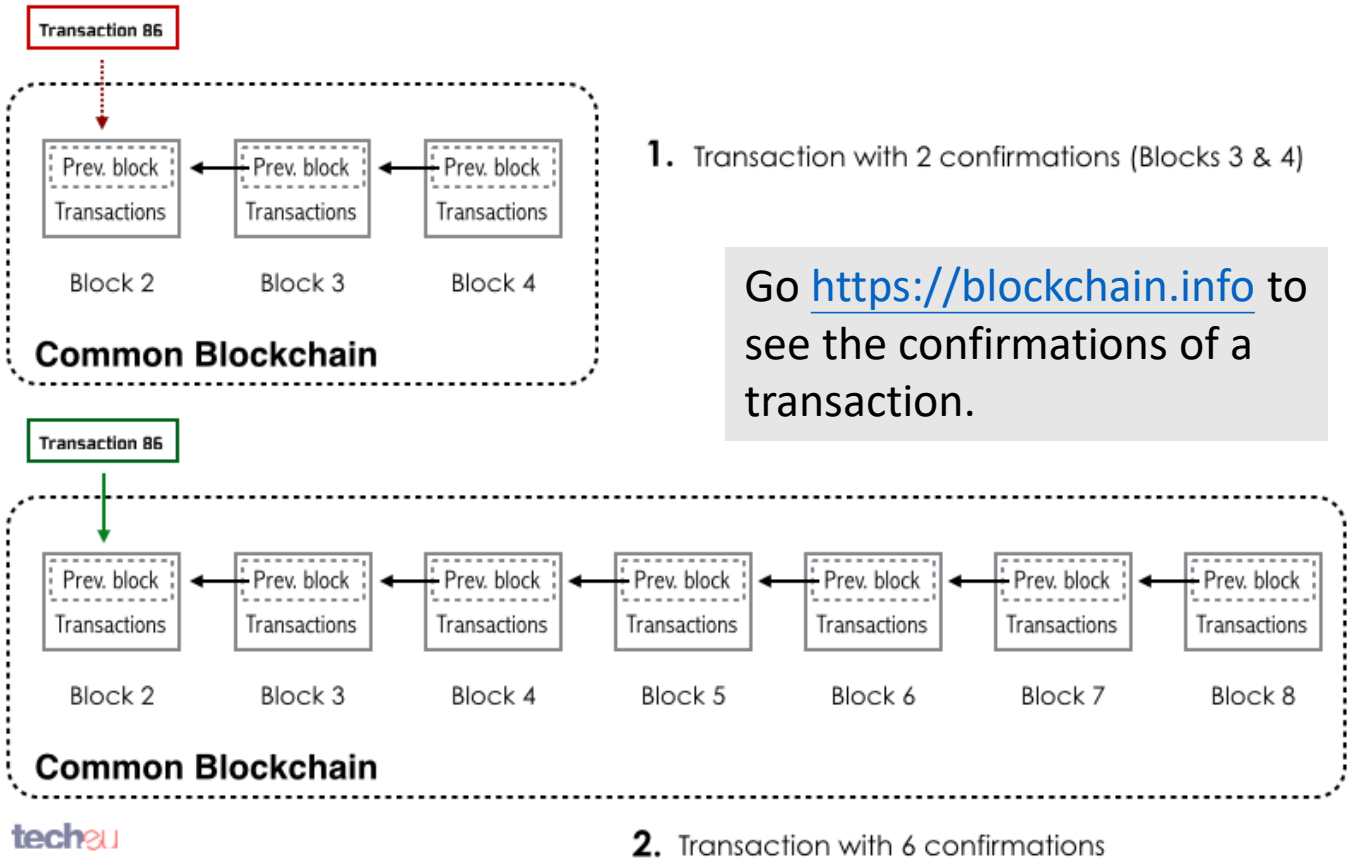


Invalid block!

Byzantine Generals' Problem!

Confirmations

- Each time a new block is added to the blockchain after this one, the confirmation count grows.
- Since the blockchain might have a fork. If we accept a transaction before waiting for at least six confirmations, it might happen the network drops that branch.
  - It expose us to a fraud situation or double spending.
- Why six confirmations?
  - Because generating an alternate Blockchain branch bigger and faster than the rest of the network would require vast amounts of computational power.

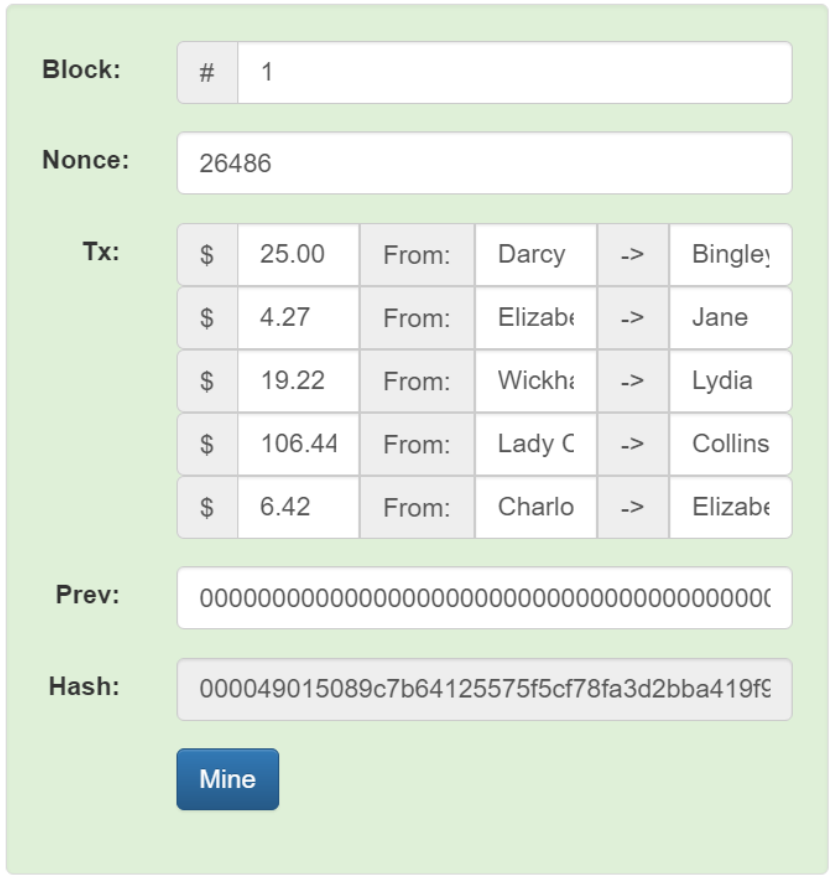


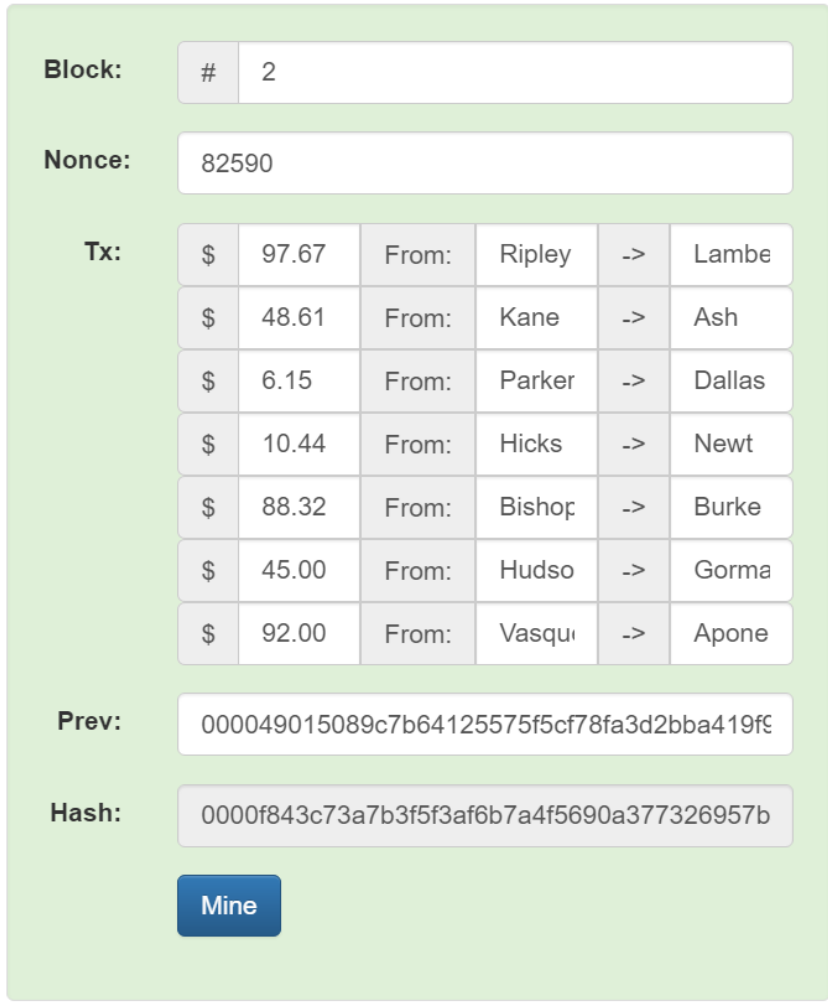
Go https://blockchain.info to see the confirmations of a transaction.

Transactions (over-simplified)

Hash = SHA256(block#, nonce, tx, prev)

- [**Wiki**] A Bitcoin transaction is a transfer of Bitcoin value that is broadcast to the network and it is collected into blocks by miners.
- For Bitcoin, it lists transactions in the block; not account balances.

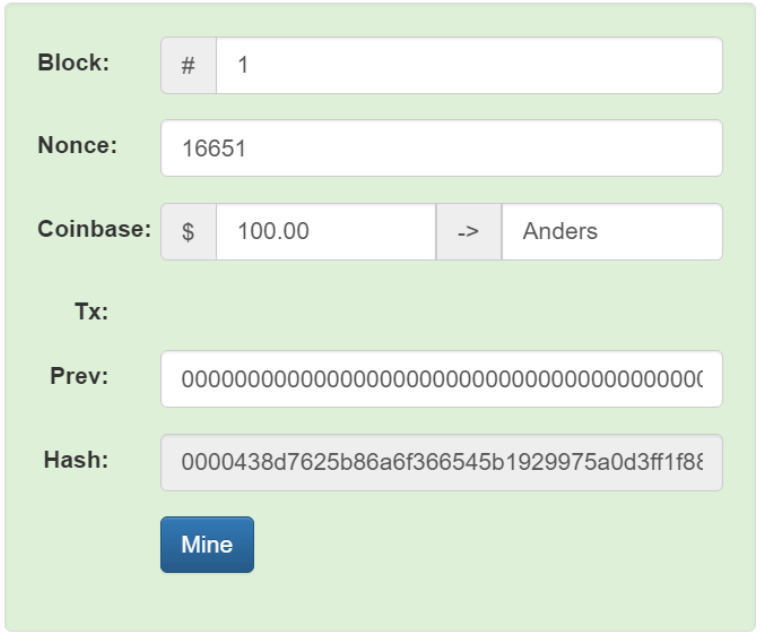


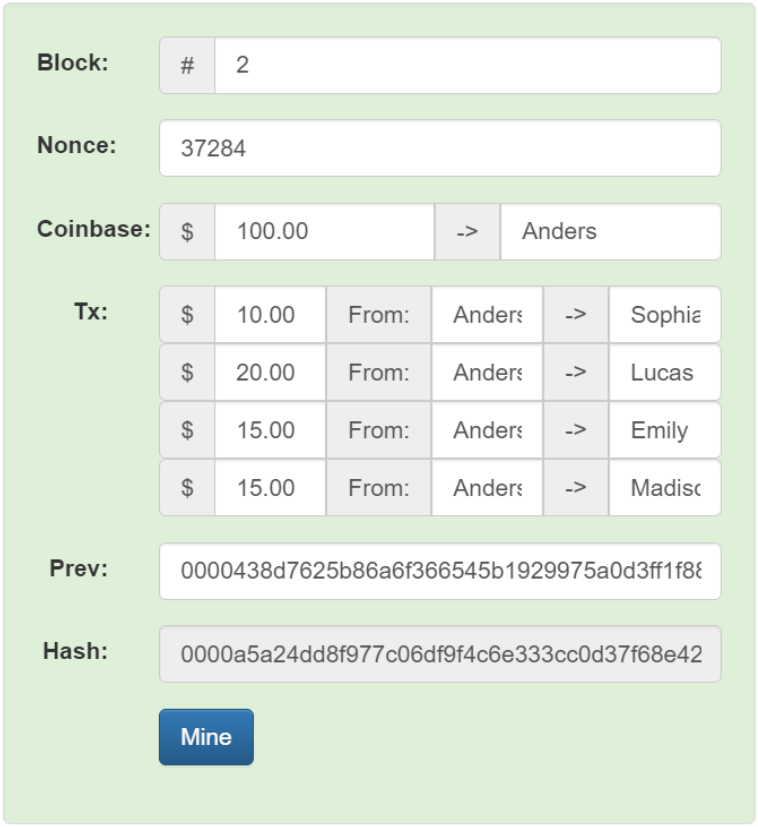


Coinbase

Hash = SHA256(block#, nonce, coinbase, tx, prev)

- A special tx.
- No ‘from*’.
- Usually ‘to’ the account of the miner.
- Create new coins from nothing to the miner.
- Now it looks like a ledger.
- Note: still no balance!





The coinbase can contain any arbitrary data. See Genesis Block.

Miners

- Need computational infrastructure to obtain high hash per second.
- CPU -> GPU -> FPGA (Field-Programmable Gate Array)
  ->ASIC (Application-Specific Integrated Circuit)
- Mining efficiency
  - Mhash/s = millions hashes per second
  - Mhash/J = millions hashes per joule (J)
  - W (watt) = 1 J/s
- Google “bitcoin mining profit”.

https://en.bitcoin.it/wiki/List_of_Bitcoin_mining_ASICs

Blockchain Application

Transaction in the data Field: Bitcoin

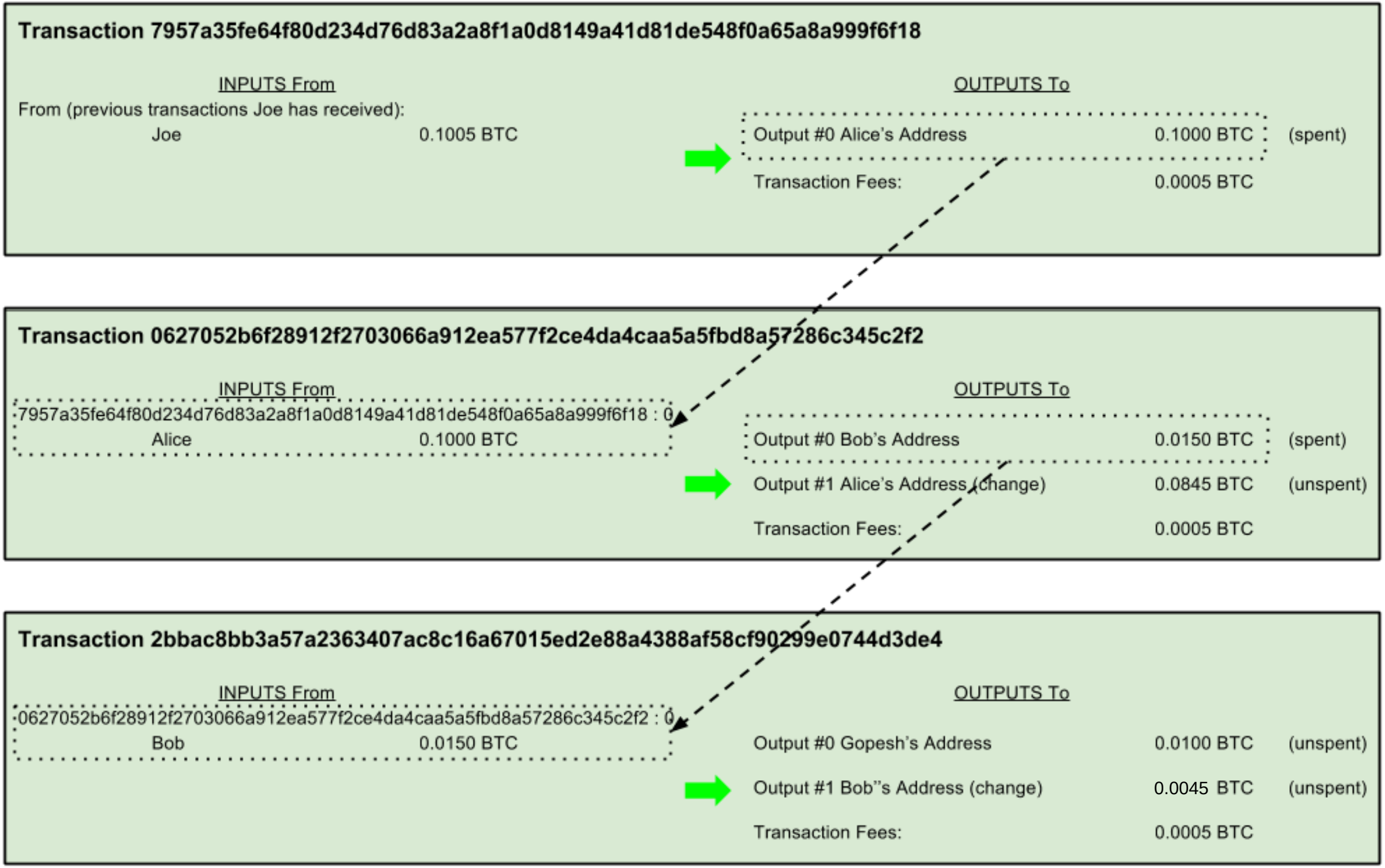
Transaction as Double-Entry Bookkeeping

Inputs	Value	Outputs	Value
Input 1	0.10 BTC	Output 1	0.10 BTC
Input 2	0.20 BTC	Output 2	0.20 BTC
Input 3	0.10 BTC	Output 3	0.20 BTC
Input 4	0.15 BTC		
Total Inputs:	0.55 BTC	Total Outputs:	0.50 BTC

	Inputs	*0.55 BTC*
-	*Outputs*	*0.50 BTC*
	Difference	*0.05 BTC (implied transaction fee)*

Check it out in
https://blockchain.info/

Transaction!
Not block!


Transaction 7957a35fe64f80d234d76d83a2a8f1a0d8149a41d81de548f0a65a8a999f6f18
INPUTS From
From (previous transactions Joe has received):
Joe
0.1005 BTC
OUTPUTS To
Output #0 Alice's Address
0.1000 BTC
(spent)
Transaction Fees:
0.0005 BTC
Transaction 0627052b6f28912f2703066a912ea577f2ce4da4caa5a5fbd8a57286c345c2f2
INPUTS From
7957a35fe64f80d234d76d83a2a8f1a0d8149a41d81de548f0a65a8a999f6f18 : 0
Alice
0.1000 BTC
OUTPUTS To
Output #0 Bob's Address
0.0150 BTC
(spent)
Output #1 Alice's Address (change)
0.0845 BTC
(unspent)
Transaction Fees:
0.0005 BTC
Transaction 2bbac8bb3a57a2363407ac8c16a67015ed2e88a4388af58cf90299e0744d3de4
INPUTS From
0627052b6f28912f2703066a912ea577f2ce4da4caa5a5fbd8a57286c345c2f2 : 0
Bob
0.0150 BTC
OUTPUTS To
Output #0 Gopesh's Address
0.0100 BTC
(unspent)
Output #1 Bob"s Address (change)
0.0045 BTC
(unspent)
Transaction Fees:
0.0005 BTC

Transaction at blockchain.info

Transaction!
Not block!

Transaction View information about a bitcoin transaction

0627052b6f28912f2703066a912ea577f2ce4da4caa5a5fbd8a57286c345c2f2

1Cdid9KFAaatwczBwBttQcwXYCpvK8h7FK (0.1 BTC - Output)

1GdK9UzpHBzqzX2A9JFP3Di4weBwqgmoQA - (Unspent) 0.015 BTC
1Cdid9KFAaatwczBwBttQcwXYCpvK8h7FK - (Unspent) 0.0845 BTC

0.0995 BTC

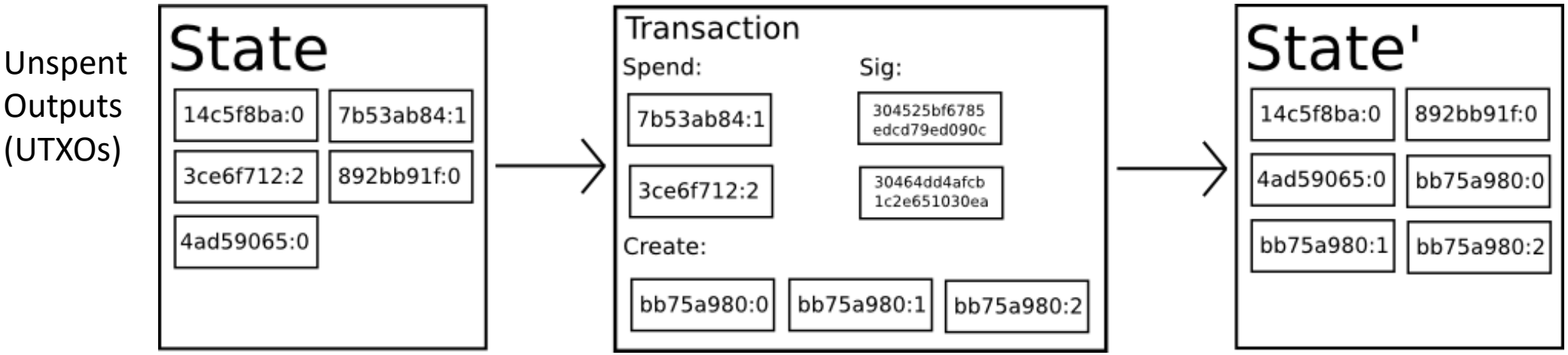
Summary	
Size	258 (bytes)
Received Time	2013-12-27 23:03:05
Included In Blocks	277316 (2013-12-27 23:11:54 + 9 minutes)
Confirmations	169137 Confirmations
Relayed by IP	Blockchain.info
Visualize	View Tree Chart

Inputs and Outputs	
Total Input	0.1 BTC
Total Output	0.0995 BTC
Fees	0.0005 BTC
Estimated BTC Transacted	0.015 BTC
Scripts	Hide scripts & coinbase

https://blockchain.info/tx/0627052b6f28912f2703066a912ea577f2ce4da4caa5a5fbd8a57286c345c2f2

Bitcoin As A State Transition System

- The ledger of Bitcoin can be thought of as a state transition system, where there is a "state" consisting of the ownership status of bitcoins and a "state transition function" that takes a state and a transaction and outputs a new state which is the result.



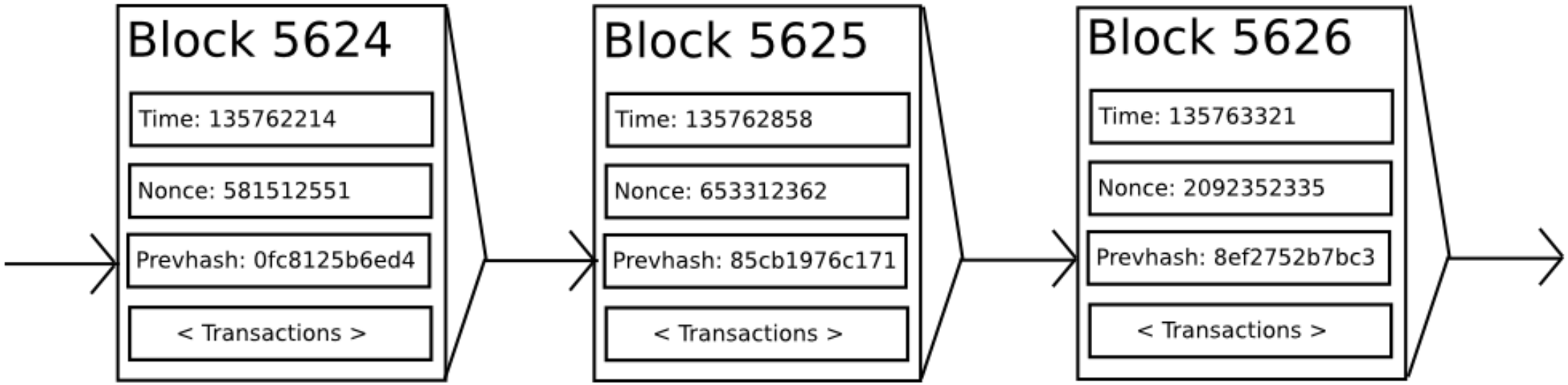
https://github.com/ethereum/wiki/wiki/White-Paper

Q: What if two transactions using the same UTXO (double spend)?

Unspent Output

- Each output has a redeem process written in the “script” which is a lock against receiver’s key.
- Receivers needs to provide the signature which unlocks the transaction output for spending the amount of Bitcoin.
- Construction a transaction can be even done completely offline.
- Most user wallets run lightweight clients that track only the user’s own unspent outputs.
  - Or it can query the bitcoin network to retrieve this information.

Mining as A State Transition System



Q: Is it possible to cheat?

Blockchain Application

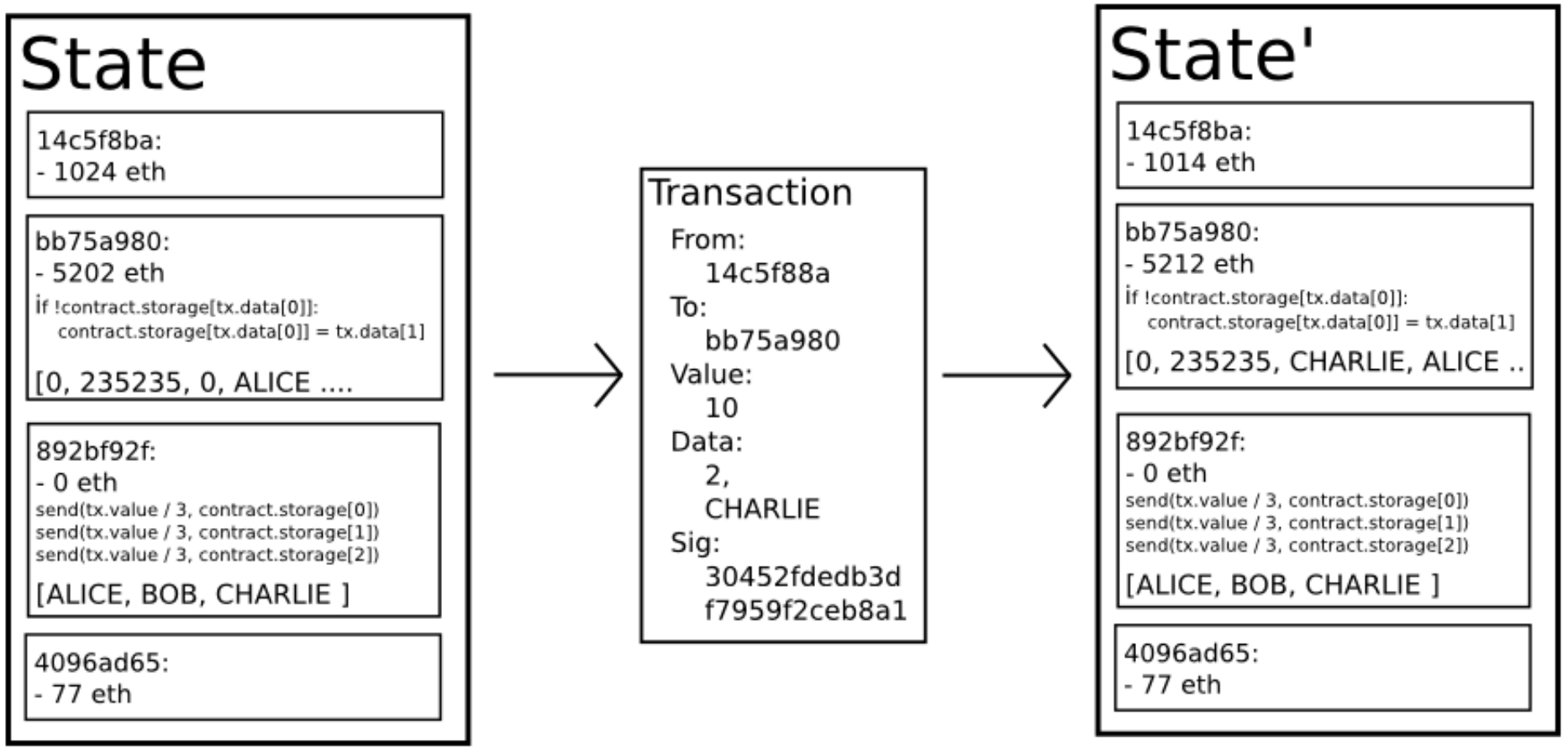
Scripts in the data field: Smart Contract

Smart Contract

- A **smart contract** is a computerized transaction protocol that executes the terms of a contract. The general objectives are to satisfy common contractual conditions, minimize exceptions both malicious and accidental, and minimize the need for trusted intermediaries. (Nick Szabo)
  - Satoshi Nakamoto did not mention anything about smart contract in his bitcoin paper.
- Blockchain makes it possible to be implemented.

一個智能合約就是能夠執行合約條款的電腦化交易協定。
設計智能合約的目的是：滿足合約條款、不出現意外的情形，無論是惡意的還是意外的；不需要信任的中間方。

Ethereum State Transition Function



Two types of accounts

- **Externally Owned Accounts** (EOAs) -> “account”
  - controlled by private keys
  - Has no code
  - one can send messages from an externally owned account by creating and signing a transaction
- **Contract Accounts** -> “contract”
  - controlled by their internal code
  - every time the contract account receives a message its code activates, allowing it to read and write to internal storage and send other messages or create contracts in turn

The popular term “smart contracts” refers to code in a Contract Account – programs that execute when a transaction is sent to that account.

Contract

- Note that "contracts" in Ethereum should **not** be seen as something that should be "fulfilled" or "complied with”.
- Rather, they are more like "autonomous agents" that live inside of the Ethereum execution environment.
- It always executing a specific piece of code when "poked" by a message or transaction.
  - Contract accounts only perform an operation when instructed to do so by an EOA.
- Contracts have direct control over their own ether balance and their own key/value store to keep track of persistent variables.

Smart Sponsor

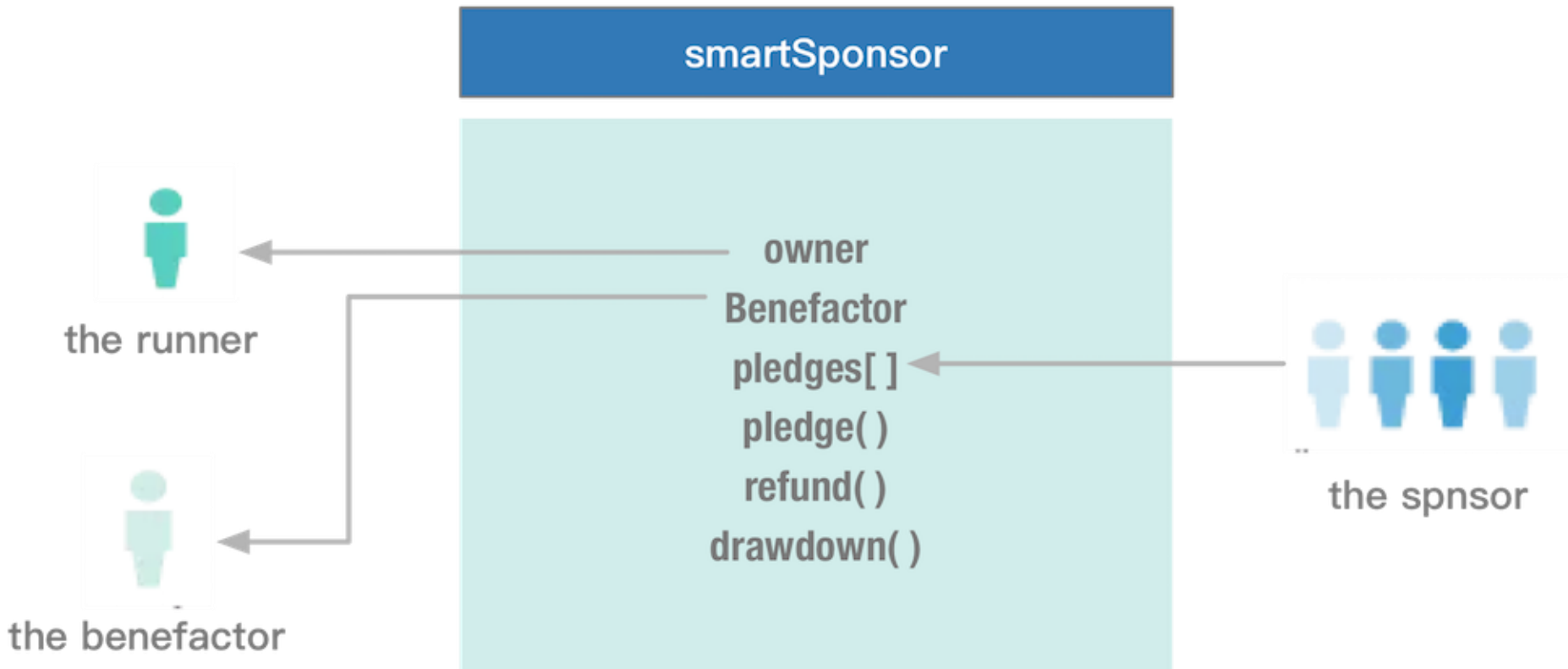
https://developer.ibm.com/clouddataservices/2016/05/19/block-chain-technology-smart-contracts-and-ethereum/

Smart Sponsor Rules 1/2

- Build a smart contract that allows the following account-holders to interact:
  - a charity holding a fund-raising activity, which we'll call **thebenefactor**
  - a sponsored runner who wants to raise money for the charity: **therunner**
  - other users who want to sponsor the runner: **thesponsor**
  - an Ethereum node that is mining the block chain, verifying transactions: **theminer**

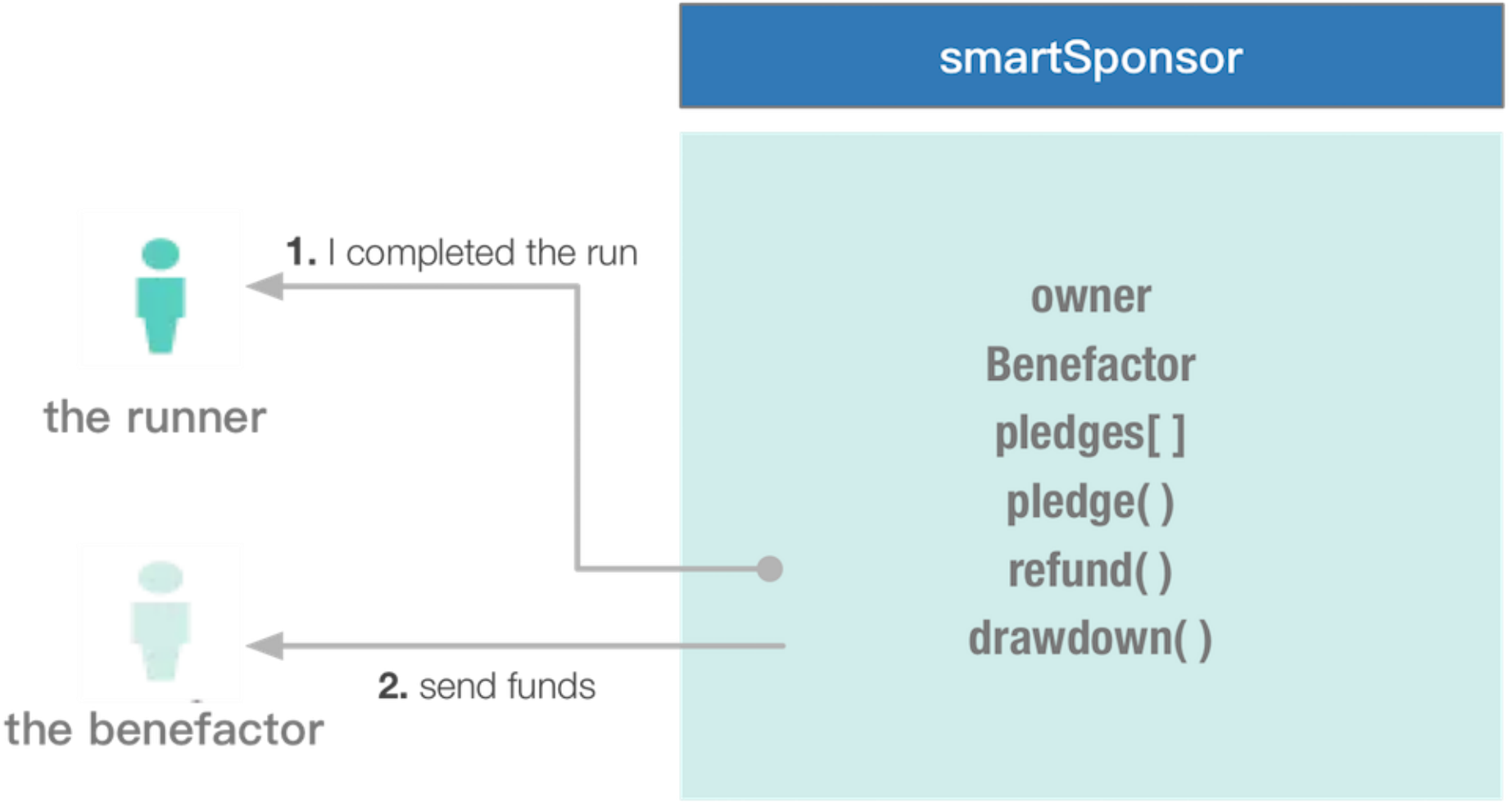
Smart Sponsor Rules 2/2

- Our contract (smartSponsor):
  - is created by a **runner** raising money for a charity by doing a sponsored run
  - when creating the contract, the **runner** nominates the **benefactor** of the money raised
  - the **runner** then invites others to sponsor the run. Users sponsor the runner by calling a function on the smart contract which transfers Ether from the **sponsor's** account to **the contract**, where it is held until further notice
  - during the lifetime of the contract everyone can see who the **benefactor** is, how much Ether has been raised and from whom (although the **sponsors** can be anonymous, of course)


smartSponsor
owner
Benefactor
pledges[ ]
pledge( )
refund( )
drawdown( )
the runner
the benefactor
the spnsor

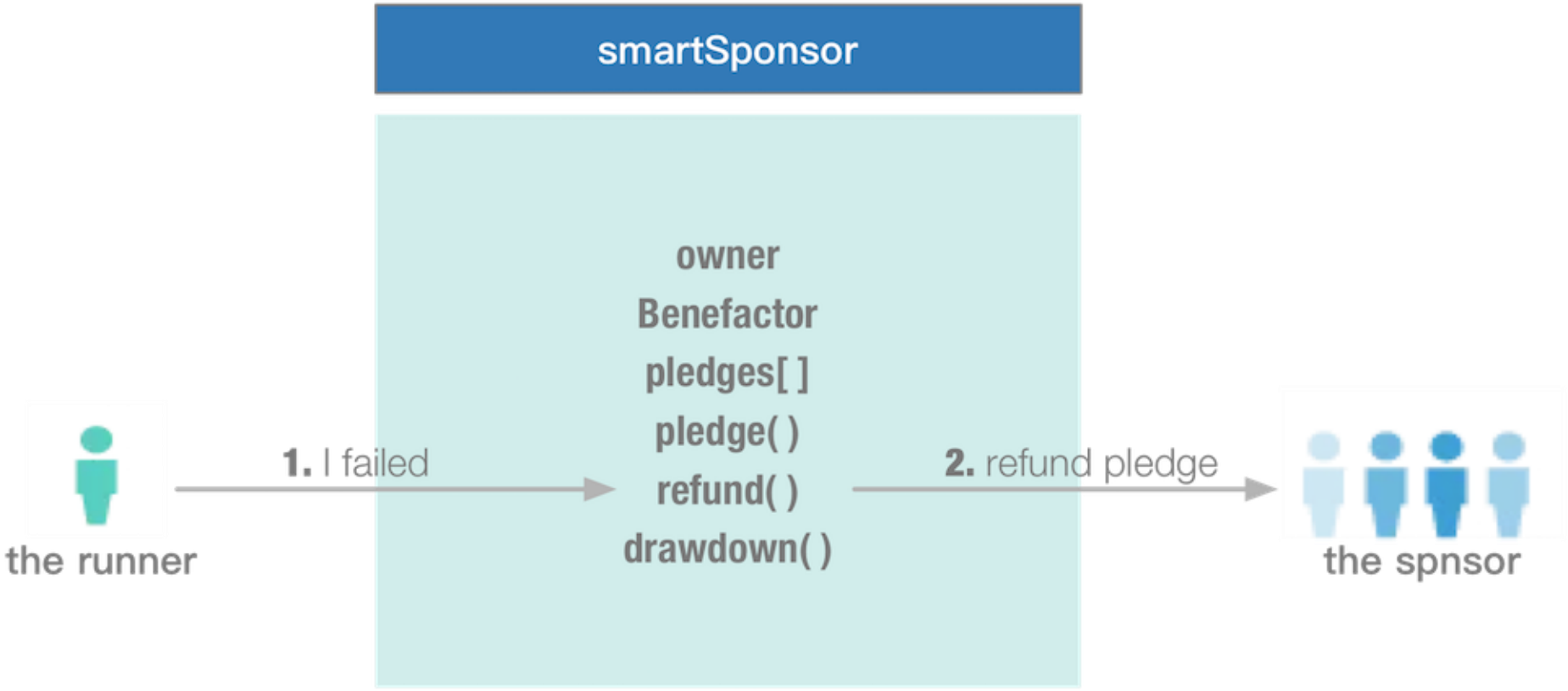
Option 1

The run goes to plan and the runner instructs the contract to transfer all of the funds to the benefactor



Option 2

The run cannot be undertaken for some reason and the runner instructs the contract to refund the sponsors' pledges



Smart Sponsor Design

- Our contract will have the following methods:
  - **smartSponsor** – the contract's constructor. It initialises the contract's state. The creator of the contract nominates the address of the account that will benefit when the contract is drawn down
  - **pledge** – can be called by anyone to donate Ether to the sponsorship fund. The sponsor supplies an optional message of support
  - **getPot** – returns the current total of Ether stored in the contract
  - **refund** – sends the sponsor money back to the sponsors. Only the contract's owner can call this function
  - **drawdown** – sends total value of the contract to the benefactor account. Again, only the contract's owner can call this function

Smart Sponsor Example 1/3

```
contract smartSponsor {
  address public owner;
  address public benefactor;
  bool public refunded;
  bool public complete;
  uint public numPledges;
  struct Pledge {
    uint amount;
    address eth_address;
    bytes32 message;
  }
  mapping(uint => Pledge) public
pledges;
```

```
 // constructor
  function smartSponsor(address
_benefactor) {
    owner = msg.sender;
    numPledges = 0;
    refunded = false;
    complete = false;
    benefactor = _benefactor;
  }
```

Smart Sponsor Example 2/3

```
 // add a new pledge
  function pledge(bytes32 _message)
{
    if (msg.value == 0 || complete
|| refunded) throw;
    pledges[numPledges] =
Pledge(msg.value, msg.sender,
_message);
    numPledges++;
  }

  // get balance
function getPot() constant returns
(uint) {
    return this.balance;
  }
```

```
 // refund the backers
  function refund() {
    if (msg.sender != owner ||
complete || refunded) throw;
    for (uint i = 0; i <
numPledges; ++i) {

pledges[i].eth_address.send(pled
ges[i].amount);
    }
    refunded = true;
    complete = true;
  }
```

Smart Sponsor Example 3/3

```
  // send funds to the contract benefactor
  function drawdown() {
    if (msg.sender != owner || complete || refunded) throw;

benefactor.send(this.balance);
    complete = true;
  }
}
```

- Then Run it.
  - https://developer.ibm.com/clouddataservices/2016/05/19/block-chain-technology-smart-contracts-and-ethereum/

Blockchain Applications

What else you would like to put in the data field?

Possible Applications

- Digital Assets
  - Security, Bond, Currency, Property (music, paint, paper)
- Digital Records
  - Domain name, Diamond, Donation, Package Tracking, Financial Service (clearing, settlement, payment), Identity (passport, birth, wedding, death certificates), Audit
- Contract
  - Insurance, Lease, Smart City (YouBike), Smart Home, P2P Landing, P2P Crowdfunding, Voting, …
- Tokens
  - Game Tokens, Reward Points, Gift Card
- Question: the current blockchain framework may not be applicable to all the applications, what additional mechanism should be included?
  - Consensus problem: difficulty, miners
  - Public or private?