An Introduction to **Blockchain** and **Smart Contracts**

Giuseppe Destefanis

Disruption

THE WEB

Computer networks are vast distributed networks through which information flows in a web-like pattern. Information is carried through the network as electrical signals, radio waves or pulses of light. Computer networks crisscross the planet, transmitting vast amounts of data at high speed. Supporting giant global systems – from transport to economics – they have provided unparalleled opportunities for creativity, sharing and collaboration between individuals.

‘Basically, our goal is to organise the world’s information and make it universally accessible and useful.’

Larry Page

Larry Page
b. 1973

Sergey Brin
b. 1973

Larry Page and Sergey Brin met at Stanford University, California in 1995. Combining their interests in large data sets and academic reference systems, they created an entirely new way of searching for information on the World Wide Web. Just three years later, they launched a fledgling company, Google, from a garage in Menlo Park.

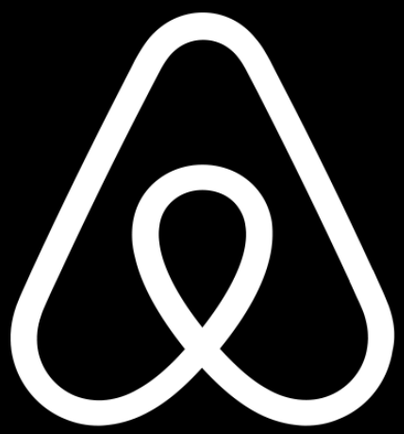
Image: Google Inc.

Google corkboard server rac

This server was an essential
first search engine. A server r
within a computer network. In
made do with inexpensive mat
off-the-shelf personal comput
corkboard for insulation. As
more sophisticated servers w
in enormous data centres.

Lent by: Google Inc. Object No: L2014-4172
Image: Connie Zhou/Google

NETFLIX



airbnb

Bitcoin: A Peer-to-Peer Electronic Cash System

Satoshi Nakamoto
satoshin@gmx.com
www.bitcoin.org

Abstract. A purely peer-to-peer version of electronic cash would allow online payments to be sent directly from one party to another without going through a financial institution. Digital signatures provide part of the solution, but the main benefits are lost if a trusted third party is still required to prevent double-spending. We propose a solution to the double-spending problem using a peer-to-peer network. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

THE TI

Max 18C, min 5C

Tuesday September 16 2008 timesonline.co.uk No 69430

Lehman collapse sends shockwave round world

hares and oil prices plunge, thousands lose jobs

ry Duncan Economics Editor

ars of a global financial meltdown w yesterday as the world's biggest kruptcy plunged markets into moil.

nvestors were left reeling as the pt demise of the Lehman Broth- investment bank sparked the est shake-up on Wall Street in ades.

nother of US capitalism's biggest tutions, Merrill Lynch, is to be owed by Bank of America in a billion takeover to save it from se.

res fell as fear spread through ancial system. Central banks un- urgent measures amid concerns e world economy was entering gerous new phase. The Bank of nd injected £5 billion of emer- lending into money markets. 5,000 Lehman staff in Britain

are now uestionably in vorst financial s since the t Depression'

aletsky, page 24

rticle page 2
ker page 5

Dow Jones industrial average was down 300 points, or 2.6 per cent. Sentiment was also bolstered by steep falls in oil prices, which dropped by more than $5 a barrel to $96, closing under $100 for the first time in six months and raising hopes that cheaper fuel would ease economic stresses on Western nations.

However, by close of trading the Dow had fallen by more than 500 points — its biggest one-day drop since the reopening after the September 11 attacks — as concerns mounted over the world's largest insurer. Shares in American International Group (AIG), which sponsors Manchester United, fell by 45 per cent after it made an unprecedented approach to the US Federal Reserve for $40 billion in emergency funding.

Last night the Fed asked Goldman Sachs and J P Morgan Chase, two of Wall Street's remaining big banks, to head a $75 billion emergency package to keep AIG afloat.

As central banks battled to stabilise the system, the Fed eased its rules for emergency lending further. It announced that it would accept company shares in return for crisis loans for the first time. In Frankfurt, the European Central Bank injected €30 billion in emergency funds into eurozone markets.

A group of ten global banks also attempted to foster calm, announc-

KIRSTY WIGGLE

The most fundamental transformation on the basis of

money

What is money?

A network based platform for recording

A network based platform for recording

Ownership

Trust

Blockchain: a replicated ledger

Blockchain

Blockchain

Peer to Peer

Blockchain

Peer to Peer

Cryptography

Blockchain

Peer to Peer

Cryptography

Consensus

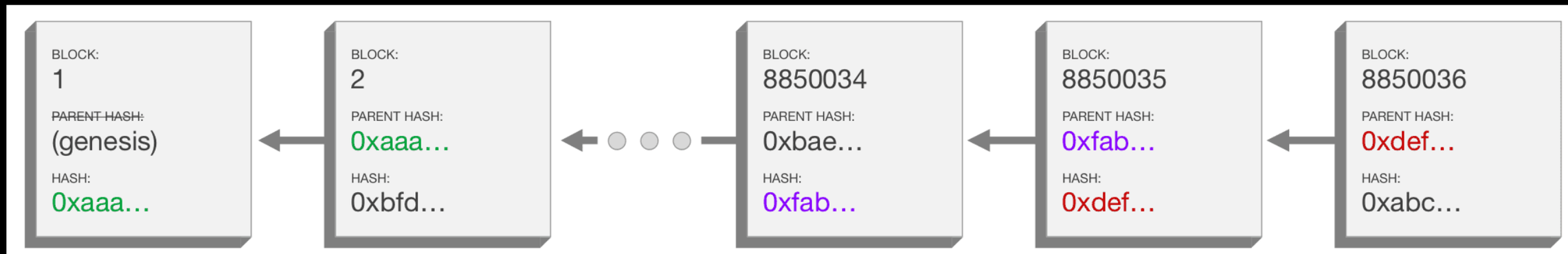
Blockchain

Peer to Peer

Cryptography

Consensus

Blockchain




```
{
    "number": 1234567,
    "hash": "0xabc123...",
    "parentHash": "0xdef456...",
    "miner": "0xa1b2c3...",
    ...,
    "transactions": [...]
}
```

Neutrality

Innovation without permission

Decentralised
Open
No borders
No identity
No control

7.5 billion

1 billion

Mobile phone repair workshop

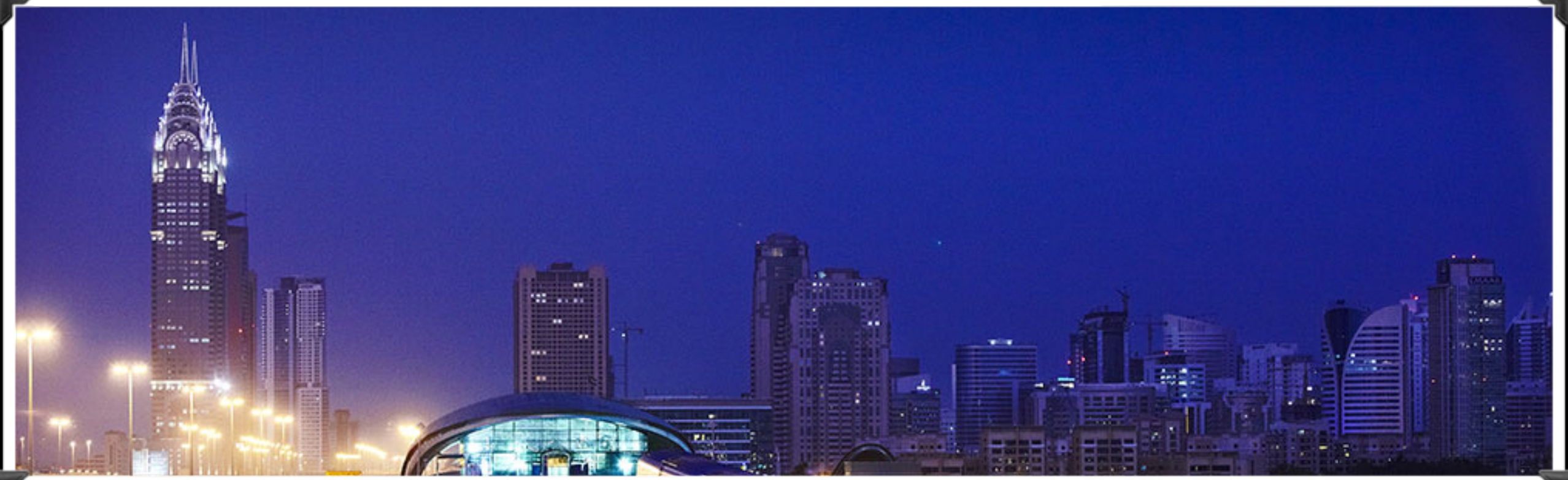
Apollinaires Tellen established a successful mobile phone repair business located near his former university in the town of Buea. From his workshop, and with the help of the online repair community, he found creative solutions for his customers' mobile phone problems, making him a key figure in the local community.

Object No: 2013-157 Image: Sjoerd Ebe Sijsma/Eyeses/Science Museum

Roadside advert
phone repair sh
2012 The repair
faulty handsets
often collabora
and tackle diffi

Object No: 2013-156

Inside the mob
As well as me
repair shops a
to enhance ph
such as FM re
most skilled
but such adv

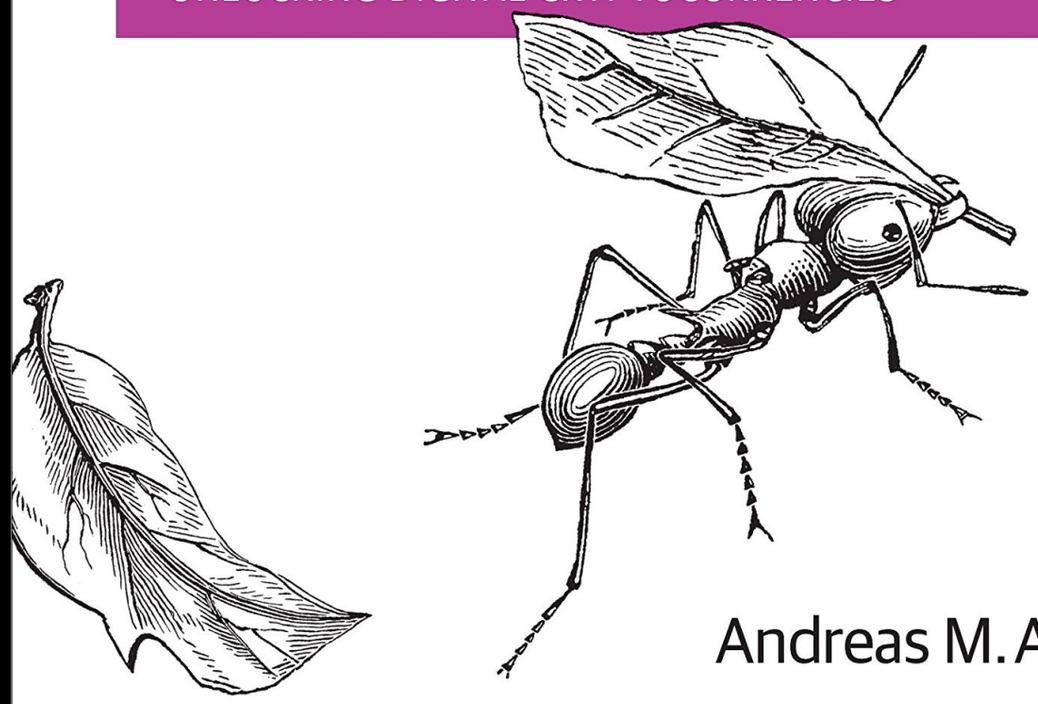


Where is the limit?

O'REILLY®

Mastering Bitcoin

UNLOCKING DIGITAL CRYPTOCURRENCIES

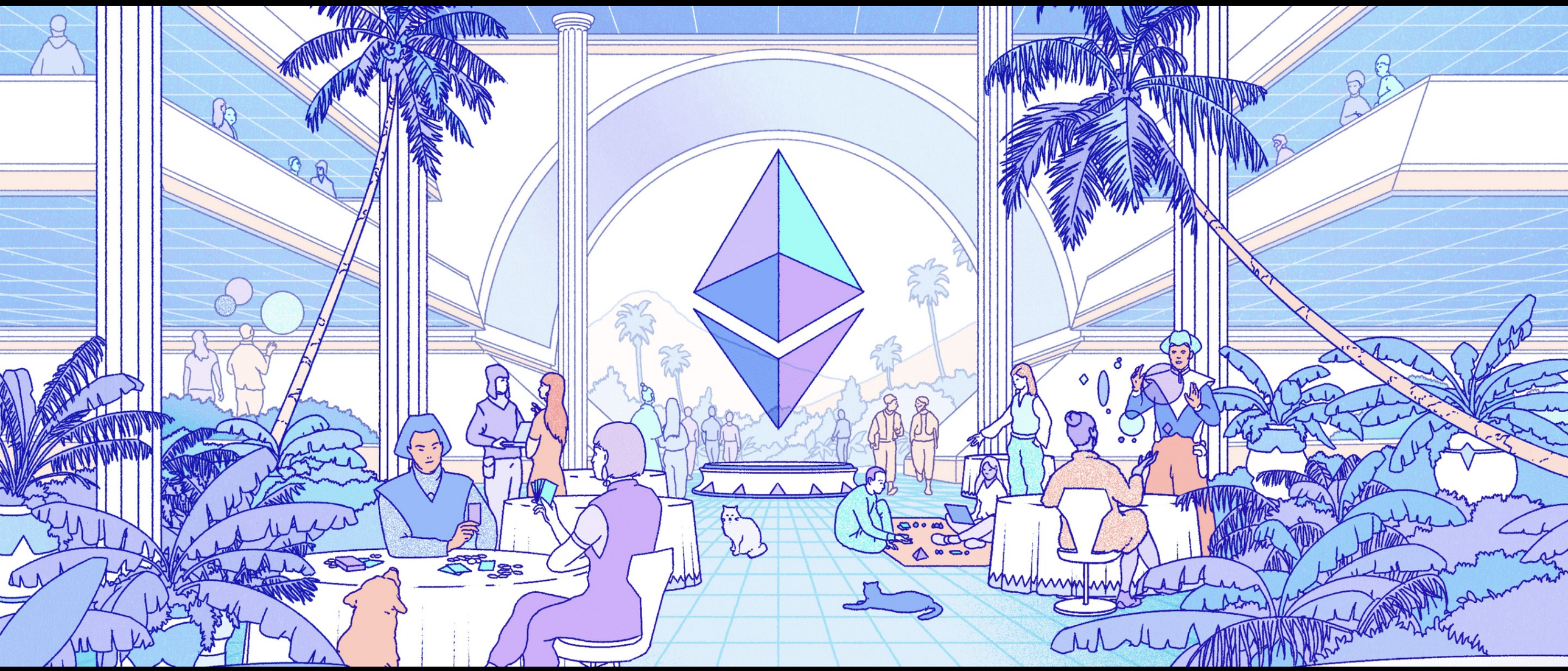


Andreas M. Antonopoulos

Smart Contracts

- Conceived in 1997 by Nick Szabo
- self-enforcing agreements (contract) expressed as a computer program whose execution enforces the terms of the contract

Ethereum



The world computer

Ethereum

- A **deterministic** (unbounded) **state machine**, consisting of a **globally accessible singleton state** and a **virtual machine** that applies changes to that state.*

*_**Mastering Ethereum**_, Antonopoulos A. (https://github.com/ethereumbook/ethereumbook)

Ethereum

- An **open source**, **globally decentralized** computing infrastructure which executes programs called **smart contracts**

- It uses a blockchain to **synchronize and store the system's state changes**, along with a cryptocurrency called **ether** to meter and constrain execution resource costs.

Smart Contracts

- real programs running on the blockchain.
- The most important ecosystem for the development and distribution of SCs, is currently the Ethereum blockchain, through the Solidity programming language.
- It is enjoying an increasing popularity, and several applications in the real world have already been developed

Smart Contracts

- The development of SCs has proved being very different from traditional software development, and involves a series of new problems and challenges

- The issue of Smart Contracts security is crucial as many of them are critical and deal with large sums of money

- Programming languages and virtual machines for the development on blockchain are still limited, as well as programming resources and development environments

- Turing-complete systems can run in infinite loops;

- Turing-complete systems can run in infinite loops;

- unintended never-ending loops can arise without warning! (coding is difficult!);

- Turing-complete systems can run in infinite loops;
- unintended never-ending loops can arise without warning! (coding is difficult!);
- every participating node (client) must validate every transaction (running any smart contracts it calls);

- Turing-complete systems can run in infinite loops;
- unintended never-ending loops can arise without warning! (coding is difficult!);
- every participating node (client) must validate every transaction (running any smart contracts it calls);
- Ethereum can’t predict if a smart contract will terminate;

- Turing-complete systems can run in infinite loops;
- unintended never-ending loops can arise without warning! (coding is difficult!);
- every participating node (client) must validate every transaction (running any smart contracts it calls);
- Ethereum can’t predict if a smart contract will terminate;
- How to deal with this possible situation? (It would be a DoS attack);

Gas Mechanism

GAS

- The EVM executes a smart contract, and it accounts for every instruction (computation, data access, etc.);
- Each instruction has a predetermined cost in units of gas;
- When a transaction triggers the execution of a smart contract, it must include an amount of gas that sets the upper limit of what can be consumed running the smart contract;
- The EVM will terminate the execution if the amount of gas consumed by the computation exceeds the gas available in the transaction;
- The required gas is proportional to the amount of computational power required to perform the operation

A Smart Contract should be:

- **deterministic:** given the same input it must always provide the same output. To this purpose, it must not call non-deterministic functions, and it must not use non-deterministic data resources;

- **terminable:** by definition the SC must be able to finish within a certain time limit;

- **isolated:** each contract must be kept isolated to avoid corrupting the entire system in the event of a virus or bug;

- **immutable:** once deployed on the blockchain, a SC cannot be changed (it can be disabled forever). This property, together with the ability to publish the source code of the SC, guarantees the highest level of transparency and trust.

When a SC is compiled, it is converted into a sequence of “operation codes”

Operation	Gas	Description
ADD/SUB	3	Arithmetic operation
JUMP	8	Unconditional Jump
SSTORE	5,000/20,000	Storage operation
BALANCE	400	Get balance of an account
CALL	25,000	Create a new account
CREATE	32,000	Create a new account

- Before performing an operation, the user sets a gas limit, which corresponds to the maximum amount of gas that they wants to pay.

- If the gas actually required by the operations overcomes the gas limit, that operation will be aborted, each change will be rolled-back, but all the gas will be spent and therefore lost;

- If the user sets a gas limit higher than the one required, the operation will be carried out, and only the used gas will be spent

Ethereum Fees @EtherFees · 44m

Ethereum fees ETH | ERC20
Fast: 146 gwei $5.46 $16.89
Average: 138 gwei $5.16 $15.97
Safe Low: 118 gwei $4.41 $13.65
Pending Transactions: 164,272



Blockchain Oriented SE

GAS Saving Design Patterns

- We collected 24 design patterns, which we assigned to 5 categories:

1. **External Transactions**
2. **Storage**
3. **Saving Space**
4. **Operations**
5. **Miscellaneous**

External Transactions

Name:	**Proxy**
Problem:	SCs are immutable. If a SC must be changed due to a bug or a needed extension, you must deploy a new contract, and also update all SCs making direct calls to the old SC, thus deploying also new versions of these. This can be very expensive.
Solution:	Use Proxy delegate pattern. Proxy patterns are a set of SCs working together to facilitate upgrading of SCs, despite their intrinsic immutability. A Proxy holds the addresses of referred SCs, in its state variables, which can be changed. In this way, only the references to the new SC must be updated.

External Transactions

Name:	**Data Contract**
Problem:	When a SC holding a significant amount of data must be updated, also all its data must be copied to the newly deployed SC, consuming a lot of gas.
Solution:	Keep the data in a separate SC, accessed by one or more SC, using the data and holding the processing logic. If this logic must be updated, the data remain in the Data Contract. This pattern usually is included also in the implementations of the Proxy pattern.

External Transactions

Name:	**Event Log**
Problem:	Often events maintain important information about the system, which must be later used by the external system interacting with the blockchain. Storing this information in the blockchain can be very expensive, if the number of events is high.
Solution:	If past events data are needed by the external system, but not by SCs, let the external system directly access the Event Log in the blockchain. Note that this Log is not accessible by SCs, and that if the event happened far in time, the time to retrieve it may be long.

Storage

Name:	**Limit Storage**
Problem:	Storage is by far the most expensive kind of memory, so its usage should be minimized.
Solution:	Limit data stored in the blockchain, always use memory for non-permanent data. Also, limit changes in storage: when executing functions, save the intermediate results in memory or stack and update the storage only at the end of all computations.

Storage

Name:	**Packing Variables**
Problem:	In Ethereum, the minimum unit of memory is a slot of 256 bits. You pay for an integer number of slots even if they are not full.
Solution:	Pack the variables. When declaring storage variables, the packable ones, with the same data type, should be declared consecutively. In this way, the packing is done automatically by the Solidity compiler. (Note that this pattern does not work for Memory and Calldata memories, whose variables cannot be packed.)

Storage

Name:	**Minimize on-chain data**
Problem:	The gas costs of Storage are very high, and much higher than the cost of Memory.
Solution:	Minimize on-chain data. The less data you put on-chain in Storage variables, the less your gas costs. Store on-chain only critical data for the SC and keep all possible data off-chain.

Operations

Name:	**Limit External Calls**
Problem:	Every call to an external SC is rather expensive, and even potentially unsafe.
Solution:	Limit external calls. In Solidity, differently from other programming languages, it is better to call a single, multi-purpose function with many parameters and get back the requested results, rather than making different calls for each data.

Operations

Name:	**Fewer functions**
Problem:	Implementing a function in an Ethereum SC costs gas.
Solution:	In general, keep in mind that implementing a SC with many small functions is expensive. However, having too big functions complicates the testing and potentially compromises the security. So, try to have fewer functions, but not too few, balancing the function number with their complexity.

Operations

Name:	**Short Circuit**
Problem:	Every single operation costs gas.
Solution:	When using the logical operators, order the expressions to reduce the probability of evaluating the second expression. Remember that in the logical disjunction (OR, ||), if the first expression resolves to true, the second one will not be executed; or that in the logical disjunction (AND, &&), if the first expression is evaluated as false, the next one will not be evaluated.

Miscellaneous

Name:	**Freeing storage**
Problem:	Sometimes, Storage variables are not longer used. Is there a way to take advantage of this?
Solution:	To help keeping the size of the blockchain smaller, you get a gas refund every time you free the Storage. Therefore, it is convenient to delete the variables on the Storage, using the keyword *delete*, as soon as they are no longer necessary.

Blockchain Oriented SE

- patterns
- testing
- architectural patterns

Design patterns for gas optimization in ethereum. Marchesi, L., Marchesi, M., Destefanis, G., Barabino, G., & Tigano, D. (2020, February). In 2020 IEEE International Workshop on Blockchain Oriented Software Engineering (IWBOSE) (pp. 9-15). IEEE.