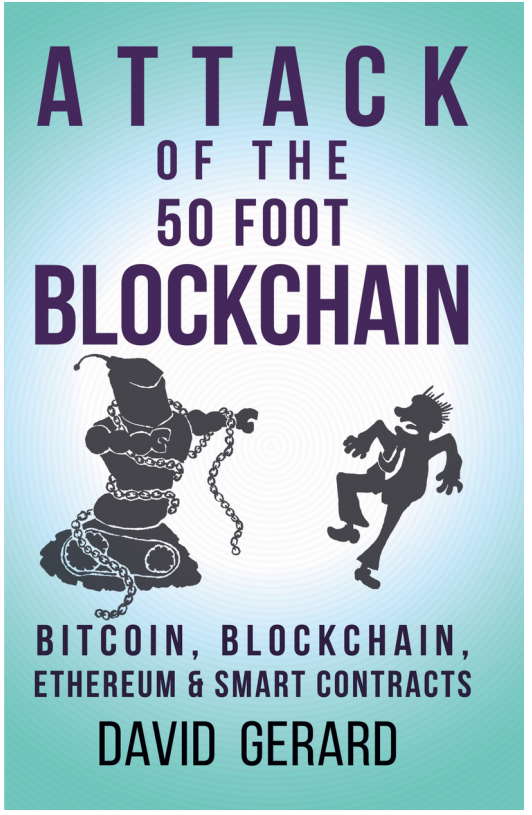
The Blockchain: Magic (probably) doesn't happen

How to sell a hash tree as a tech revolution

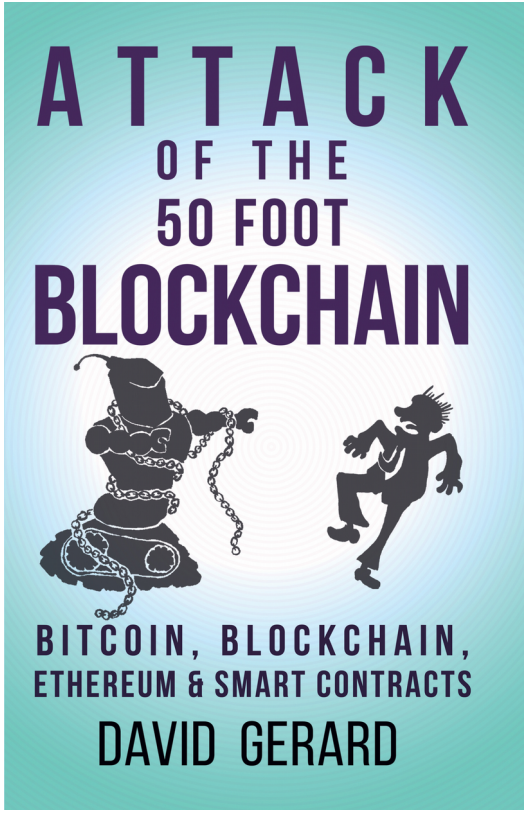
David Gerard



David Gerard

- Started as music journalist
- Moved to IT, Unix sysadmin
- Started following Bitcoin in 2011
- *Attack of the 50 Foot Blockchain* released 2017

  — *well-timed for the bubble!*



What on earth is a “blockchain”?

Simple accounting ledger

- Just a log of transactions

From	To	Date	Amount
Satoshi	Hal	09 January 2009	$50.00
Vitalik	Gavin	09 January 2009	$1,000.00
Craig	Ian	10 January 2009	$0.02
Vitalik	Eliezer	12 January 2009	$300,000.00
Mark	Aleksandr	13 January 2009	$400,000,000.00

- But — how can we ensure against errors?

Simple ledger with hashes

- Let’s attach a hash to every record!

From	To	Date	Amount	Hash
Satoshi	Hal	09 January 2009	$50.00	8227fb49
Vitalik	Gavin	09 January 2009	$1,000.00	d64ad954
Craig	Ian	10 January 2009	$0.02	85e19b86
Vitalik	Eliezer	12 January 2009	$300,000.00	9749ce74
Mark	Aleksandr	13 January 2009	$400,000,000.00	5c397c18

- So we know each record is correct
- But — what if we have a *lot* of entries?

Let’s hash all the hashes!

From	To	Date	Amount	Hash
Satoshi	Hal	09 January 2009	$50.00	8227fb49
Vitalik	Gavin	09 January 2009	$1,000.00	d64ad954
Craig	Ian	10 January 2009	$0.02	85e19b86
Vitalik	Eliezer	12 January 2009	$300,000.00	9749ce74
Mark	Aleksandr	13 January 2009	$400,000,000.00	5c397c18
				d8eb1c14

- So if we know that last hash — we know that the whole block has to come to that hash!
- If we have 1000 entries and add a new one, we don’t have to rehash all the entries — just their hashes

Tamper-evident append-only ledger!

- If you distribute the ledger, you can quickly verify the hashes of your copy
- And — it’d be impossibly slow to fake
- This hash-of-hashes construct is called a Merkle Tree (1979)
- Used in BitTorrent, ZFS, git … and Bitcoin

Let’s chain the blocks!

- Each block’s hash is also hashed with the next block
- This gives us a hash of the whole … chain of blocks
- It’s … a blockchain!
- Note lack of magic

From	To	Date	Amount	Hash
Satoshi	Hal	09 January 2009	$50.00	8227fb49
Vitalik	Gavin	09 January 2009	$1,000.00	d64ad954
Craig	Ian	10 January 2009	$0.02	85e19b86
Vitalik	Eliezer	12 January 2009	$300,000.00	9749ce74
Mark	Aleksandr	13 January 2009	$400,000,000.00	5c397c18
				d8eb1c14

From	To	Date	Amount	Hash
Satoshi	Hal	09 January 2009	$50.00	8227fb49
Vitalik	Gavin	09 January 2009	$1,000.00	d64ad954
Craig	Ian	10 January 2009	$0.02	85e19b86
Vitalik	Eliezer	12 January 2009	$300,000.00	9749ce74
Mark	Aleksandr	13 January 2009	$400,000,000.00	5c397c18
				d8eb1c14

From	To	Date	Amount	Hash
Satoshi	Hal	09 January 2009	$50.00	8227fb49
Vitalik	Gavin	09 January 2009	$1,000.00	d64ad954
Craig	Ian	10 January 2009	$0.02	85e19b86
Vitalik	Eliezer	12 January 2009	$300,000.00	9749ce74
Mark	Aleksandr	13 January 2009	$400,000,000.00	5c397c18
				d8eb1c14

Bitcoin

Bitcoin

- Digital cash would be a useful thing
- We could use this hard-to-fake Merkle tree ledger for our new digital cash!
- But — who gets to add new entries?
- Obvious answer: central authority (bank)
- But ...

Bitcoin’s founders had certain political requirements

- Founded in ideology — *extremist libertarianism*
  — *see “The Politics of Bitcoin” by David Golumbia (2016)*
- No central authority at all — *no trust requirement*
- A completely rigid gold standard! — *digital version*
- Credit is bad too — *use the actual “gold” as money*

How bitcoins are issued

- 21 million Bitcoins total, released slowly
- New bitcoins issued every ~10 minutes
- How to do this with no central authority?
- *Make it a lottery!*

How Bitcoin mining works

- Get a block of transactions
- Guess a random number (“nonce”), add to end
- Take the hash!

From	To	Date	Amount	Hash
Satoshi	Hal	09 January 2009	$50.00	8227fb49
Vitalik	Gavin	09 January 2009	$1,000.00	d64ad954
Craig	Ian	10 January 2009	$0.02	85e19b86
Vitalik	Eliezer	12 January 2009	$300,000.00	9749ce74
Mark	Aleksandr	13 January 2009	$400,000,000.00	5c397c18
			nonce	12132341
			hash	00000032

How Bitcoin mining works

- If the hash is a small enough number —
  you win the bitcoins!
- If you don’t — guess again
- Literally — just guessing numbers very fast

  — *no “complex calculations”, just simple ones fast*

  — *36,000,000,000,000,000,000,000 guesses every 10 minutes, 1 winner*

“Proof of Work” — Proof of Waste

- If too many people win — make it harder!
- Ends up in a Red Queen’s race
  — *more and more power to stay in the same place*
- As much power as Ireland or Austria — 0.1-0.5% of world
  — *literally wasted guessing numbers*
- Still only does 7 transactions/second — *same since 2009*
- Bitcoin is anti-efficient
- So … what does all this get us?

The fabulous promises of Bitcoin!

- Decentralised! Trustless!
- Fast and free!
- Uncensorable and irreversible!
- No “just printing money” — limited supply!

How the promises worked out

- Bitcoin had recentralised by early 2014
- Proof of Work has economies of scale
  — *so it recentralises*
- Four mining pools issue most of the bitcoins
- Bitmain owns 50% of mining, makes 80% of mining chips

How the promises worked out

- Bitcoin was fast and near-free up to mid-2015

  … when the transactions reached capacity
- Bitcoin transactions have been slow, unpredictable and expensive since
- Peaked at ~$55 average fee in Dec 2017

How the promises worked out

- Uncensorable! Irreversible!
- This turns out not to be what users want
  — *consumers like chargebacks, they increase confidence*
- Errors, fraud, thefts not easily reversible
  — *irreversibility is a fraudster's charter*
- Brittle!
  — *one mistake and you've lost your coins*

How the promises worked out

- You can’t “just print” bitcoins
- BUT — anyone can copy the code
  — *and they did — 1000+ altcoins*
- Market treats all these as one pool, “cryptos”
- Bitcoin is just like gold! … if you could create new gold mines by cut’n’paste

Can altcoins do better?

- Bitcoin was the first paper/string mock-up, pressed into service
- Other proof-of-work coins have similar throughput
  — *Ethereum runs 16 transactions/second*
  — *already having transaction clogs — CryptoKitties!*
- Users hop from coin to coin as old ones clog
- Markets treat all this as one pool of “crypto”

Can altcoins do better?

- Ethereum “Casper” proof-of-stake — unfinished
- Experimental new work — unfinished or not fully tested
  — *IOTA, Hashgraph, Avalanche, etc*
- “If you can’t disprove my paper ... you must buy my tokens!!”
  — *Time Cube in LaTeX*
- But so far, no new solutions
- Ignore these until they survive the hostile Internet

Enterprise Blockchain

What organisations want

- Civilisation runs on bureaucracy
- Any organisation — business, non-profit, government — has bureaucracy — the machinery they run on
- Can we make this work better?
- … with ***blockchains?***

“Blockchain”

- Bitcoin losing lustre by early 2014
- So, market to business as “Blockchain technology”
- *a.k.a.* “Distributed Ledger Technology” (DLT)
  — *do shared Excel sheets count?*
- But — the promises are still Bitcoin promises!
  — *else, shared Excel sheets would count*
  — *“Blockchain” is a particular collection of marketing promises*

The fabulous promises of Blockchain!

- Literally the Bitcoin promises
  — *just change the buzzword!*
- Decentralised, fast and free!
  — *“against who” is not clear — no sensible threat model*
- Uncensorable, irreversible, immutable, incorruptible!
  — *nobody say “GDPR”*
- Smart Contracts for added magic!
  — *the hard bit is always done by “smart contracts”*

Permissioned blockchains

- Usual case in business
  — all participants known, authorised
- Don’t want your back office on the hostile Internet
- Don’t use Proof of Work (it’s silly)
- This is also called a “database”
- Even if shared — someone runs it, controls access
- No magical “blockchain” results

Smart Contracts

Smart Contracts

- Small computer programs — run automatically when something happens in the data
- You know these as database triggers, or stored procedures
  — *widely considered bad software engineering*
- Immutable, like the blockchain
  — *this is your market integrity*
- VERY hard to get right — you must deploy a perfect program
  — *all computer programs have bugs*

Smart Contracts in practice

- You'd want non-Turing-complete, functional, mathematically provable ...
- Ethereum and Solidity ignore all that (YOLO!)
- JavaScript descended language, loaded with gotchas
- ~ 100 bugs/1000 lines
- Most smart contracts are now editable by their creator for this reason

  — *so much for decentralisation*

Smart Contracts in practice

- The DAO — a Decentralised Autonomous Organisation!
- “immune to human interference”
  = “sitting duck for attackers”
- Curators warned about security hole, went ahead
- $50m of Ether stolen
- Ethereum itself was rolled back to recover the funds
  — smart contracts are “immutable” until the big boys lose money

Smart contracts on permissioned blockchains

- “Smart contract” in a closed system just means “computer program”
- Salesman: “The magic bit is done with … smart contracts!”
- Translation: “We could do it on a … computer!”
- Will be much like any other new large IT system

Blockchains in the real world

Blockchains in the real world

- Almost none in production use
- Main smart contract use case: ICO tokens
  — *and excuses why something needs a blockchain*
  — *with handwaving about blockchain economics*
- Press releases, pilot programmes
  — *a majority from IBM*

Initial Coin Offerings

1. State a problem
— *doesn't have to be a real problem*
2. Tokens can solve it!
— *add some weird Bitcoin economic reasoning*
3. There are no other steps

But the fabulous potential!

- Many solutions: fix our data and formats
- Mostly solve some bit that isn’t the problem
  — land registry, supply chain fraud
- Unable to scale to size of the problem
  — every musical blockchain proposal ever
- Don’t understand the problem
  — how to make something worse than electronic voting
- Relies on things that don’t exist yet
  — we’ll do the hard bit with ZSNARKS in smart contracts or something probably
- A panopticon of 7 billion people’s personal information
  — thankfully unfeasible, as well as a massive GDPR violation

Real world blockchain projects

- World Food Programme
  — *single-user private Ethereum — i.e., a database*
- Wal-Mart/IBM supply chain trials
  — *all nodes on IBM Cloud, administered by Wal-Mart*
  — *doesn’t exist yet*
- Maersk/IBM trials
  — *as centralised as Wal-Mart trials*
  — *vendors openly wondering what the “blockchain” bit is supposed to achieve*
- Voatz military absentee voting trial
  — *collect votes, log them on private Hyperledger cluster*
  — *use Blockchain to transmit votes from their app, print out a paper ballot*

Real world example: KSI Blockchain

- Estonia’s “blockchain revolution”
- First released 2007
- Widely touted as “blockchain success story”
- Not a blockchain at all — just the ledger
- Name is for marketing
  — *definitely worked!*
- At least it has a Merkle tree in it

Issues to consider

- You may have a use case for the Merkle tree ledger
- Even if it's marketed as "blockchain" or "DLT"
- Maybe you *really do* need a slow, very robust, distributed database
  — *one day we'll see an example*
- But it probably won't improve on git
  — *"blockchain" products that are basically a simplified git*
- Magic doesn't happen
  — *if it sounds too good to be true, it probably is*

Questions, please!

- David Gerard
- dgerard@gmail.com
- www.davidgerard.co.uk/blockchain/
- Twitter: @davidgerard