Blockchain Certification: Bitcoin & OpenTimestamps

Bicocca University, Milan, 25th January 2019
Ferdinando Maria Ametrano
Bitcoin: A Protocol and a Currency

- **Bitcoin**: protocol, software, and community
- **bitcoins**: units of the currency

*bitcoins are sent using the Bitcoin protocol*

- bitcoins are the native digital asset intrinsic to the Bitcoin protocol
Bitcoin: The Protocol

- Distributed public ledger of transactions
- Shared with peer-to-peer technology
- Massively duplicated across network nodes
- Allowing the ownership transfer of a native digital scriptural asset
- Whose native “digital token” can be exchanged, but not duplicated
- Keeps records of every transaction forever
Bitcoins: The Currency

- Only exist as public ledger documented transactions
- Are associated to public address(es) like
  
  1FEz167JCVgBvhJBahpzmrsTNewhiwgWVG

- the bitcoin distributed public ledger certifies for everybody how many bitcoins are associated to any address

  https://blockstream.info/address/1FEz167JCVgBvhJBahpzmrsTNewhiwgWVG

It is mine; you are REALLY encouraged to tip
Asymmetric Cryptography: Public/Private Key Pair

Two mathematically linked keys perform opposite digital signature functions:

- The **private** (secret) key used to generate the signature
- The **public** key used by anyone to verify the signature

- The **public** key derives from the **private** key, but the **private** key cannot be derived from the **public** one
- A bitcoin **address** is derived from a **public** key, but the **public** key cannot be derived from the **address**
- **Private** key -> **public** key -> bitcoin **address**
Asymmetric Cryptography: Public/Private Key Pair

- Private key -> public key -> bitcoin address
- The corresponding **private** key allows spending from that address

https://www.bitaddress.org
A Bitcoin Transaction: From Alice’s Address to Bob’s Address

- Transaction: amount + Bob’s address (+ Alice’s public key)
- Alice’s private key digitally signs the transaction
- The transaction is broadcasted to the network
- With Alice’s public key any network node can verify that:
  - Amount is at Alice’s address disposal
  - Digital signature is valid, the transaction has not been tampered or modified: the private key associated to Alice’s address has signed the transaction
- The transaction is then published to the public ledger
- Everybody knows that the Bob’s address has received the transacted amount
Double Spending Problem

- To securely transfer value using digital means has been possible for decades

- In digital cash schemes, a single digital token, being just a file that can be duplicated, can be spent twice

- A centralized trusted party has always been required to prevent double spending
Bitcoin Network: A Distributed Back-office

- All network nodes validate and clear all transactions
- Mining nodes provide the additional computational power required for transaction settlement

- Without a central trusted party, how do they reach distributed consensus on the transaction history?
- Consensus in a distributed asynchronous network with faulty (or malicious) nodes is a very hard problem: Computer Science even provides impossibility results
Bitcoin's Public Ledger: A Chain of Blocks

- Transactions are bundled in blocks (about one block every 10 minutes) and sequentially chained

- The cryptographic link between blocks requires computing power to be created

- A block is valid only if it includes valid transactions
Mining

- Miners compete to finalize (settle) a new block of transactions
- The winner providing proof-of-work for the finalization of a new block is rewarded with the issuance of new bitcoins in a special coinbase transaction included in that same block
- Miners solve the double spending problem:
  - A double spending transaction would invalidate the block
  - an invalid block would be rejected from the network
  - the bitcoin reward would be removed from transaction history
  - the winning miner would have wasted his work
Ledger Immutability

- Because of the *proof-of-work*, the chances of a block being altered decrease exponentially with the number of blocks chained after it.

- The chain of blocks is a history of transactions resilient to network attackers because it cannot be altered without huge resources.

- Computing power is measured in hash/s, hash being the basic operation needed for validation.
Nakamoto Distributed Consensus

- Practical Byzantine Fault Tolerant (PBFT) *distributed consensus* is achieved using *(game theory)* economic incentive for the mining nodes to be honest
- Double spending is solved without a central trusted party
- Bitcoin can resist attacks of malicious agents, as long as they do not control network majority
- Miners are compensated for their proof-of-work using seigniorage revenues, i.e. issuance of new bitcoins
- Seigniorage revenues subsidize the network
Virtuous Cycle

mining → hash
reward → power

bitcoin → Bitcoin
price → security
Validation Process: Block Generation

The proof-of-work difficulty is adapted about every 2 weeks (2016 blocks) to the overall available computing power ensuring about one block every 10 minutes.
Bitcoin Monetary Rule

- 2009: 50BTC per block, every 10 minutes
  - halving every 4Y

- This is the only way new bitcoins are released

- It is called mining because of its similarity with the progressive scarcity of gold extraction

- Supply is free of discretionary intervention
Bitcoin **Inelastic** Supply: Deterministic Decreasing Rate

- **2029**: 96.88% of all BTC issued
- **2141**: last satoshi (0.00000001 BTC) will be issued
What Makes Bitcoin Special?

- Digital and scriptural: it only exists as validated transaction
- Asset, not liability
- Bearer instrument
- It can be transferred but not duplicated (i.e. it can be spent, but not double-spent)
- Scarce in digital realm, as nothing else before
- It mimics gold monetary policy
What Makes Bitcoin Special?

Bitcoin is digital gold with a secure uncensorable embedded settlement network

- More a crypto-commodity then a crypto-currency

- This is the groundbreaking achievement by Satoshi Nakamoto, not blockchain “technology”
## Bitcoin as (Digital) Gold in the History of (Crypto)Money

<table>
<thead>
<tr>
<th>gold</th>
<th>bitcoin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Its adoption was not centrally planned</td>
<td>Its adoption has not been centrally planned</td>
</tr>
<tr>
<td>For centuries it has been the most successful form of money</td>
<td>It is the most successful form of cryptocurrency</td>
</tr>
<tr>
<td>It has bootstrapped all monetary systems we know of</td>
<td>It is bootstrapping new monetary systems</td>
</tr>
<tr>
<td>It has been surpassed by other kind of money without becoming obsolete</td>
<td>It might be surpassed by more advanced type of cryptocurrencies without becoming obsolete</td>
</tr>
</tbody>
</table>
“Bitcoin in 2014 Is Like Internet in 1994: Weird and Scary” (Marc Andreessen)

American entrepreneur, investor, and software engineer; coauthor of Mosaic, cofounder of Netscape

Big companies desperately hoping for blockchain without Bitcoin is exactly like 1994: Can't we please have online without Internet?? 😊

https://twitter.com/pmarca/status/677658844504436737
The Wallet Garden Model

- Controlled access to web content and services
- Offered in the late '90s and early '00s by Compuserve, AOL (and to some extent MSN)

- Corporates wanted to go online, but not in the wild unregulated internet, populated by anonymous agents

- They eventually realized that perceived risks, which are real, are outweighed by benefits
What is The Blockchain?

[A hash pointer linked list of blocks]

- An append-only sequential data structure
- New blocks can only be appended at the end of the chain
- To change a block in the middle of the chain, all subsequent blocks need to be changed
- Very inefficient compared to a relational database
Blockchain Without Bitcoin

Blockchain without an *intrinsic native digital asset*

Does it make sense?

- No bitcoin
- No asset available to reward miners
- Appointed validator officials required

Central governance is required!

*Why should validators use a blockchain, i.e. a subpar data structure, instead of a database?*
“Blockchain – not bitcoin – will prove revolutionary in banking”

“When a wise man points at the moon the fool examines the finger.”

(Confucius)

“When a wise man points at the bitcoin the fool examines the blockchain.”

(Ametrano)

There is no blockchain without bitcoin
There is blockchain beyond bitcoin

Andreas Antonopoulos
Write Data On the Blockchain

There is a special Bitcoin Script operator *OP_RETURN* that can be used to store 80 bytes of arbitrary data in the blockchain.
A timestamp proves that some data existed prior to some point in time, providing a relevant document with a certain sure date, e.g. postmark.

Law requires dates to be certified by public officials and notary services.

For digital documents, timestamping is based on digital signature by certification authority.
Hash Function

- A function that maps input data of arbitrary length to a hash value, i.e. an output data of a fixed length
  - *Non-invertible* (one-way: input data can not be regenerated from the output hash value)
  - *Collision-resistant*: computationally unfeasible to find 2 inputs that produce the same output

- The resulting **hash value is a reliably unique identifier** for any input data: it can be considered its unique digital fingerprint
- The hash value does not reveal the input data
- Bitcoin uses the (Secure Hash Algorithm) SHA256 that generates a fixed size 256-bit (32-byte) output
Blockchain as Timestamping Certification Authority

- A generic data file can be hashed to produce a short unique identifier, equivalent to its digital fingerprint.
- Such a fingerprint can be associated to a bitcoin transaction (irrelevant amount) and hence attested on the blockchain.
- Blockchain immutability provides time-stamping, proving the data file existence at that moment in time in that specific status.
Blockchain Timestamping

Pro:
- Digital public proof, easily auditable by everyone
- The proof cannot be faked, manipulated, or removed
- Certification authority cannot be bribed
- Can be used along with regulatory timestamping prescription

Cons:
- Not efficient (one transaction per document)
- Lack of standardization

To solve the above limits, Peter Todd and Riccardo Casatta have proposed an open standard consisting in a set of operations for creating provable blockchain timestamps and later independently verifying them.
An Open Timestamping Standard

A timestamping proof standard

OpenTimestamps aims to be a standard format for blockchain timestamping. The format is flexible enough to be vendor and blockchain independent.

- Third party auditability (suitable for regulatory prescriptions)
- Scalable: timestamp an unlimited number of documents with a single transaction
- Convenient: public server provides the service for free
OpenTimestamps: Distributed, Trust-minimizing, Scalable, Convenient

- **Trust**: OpenTimestamps uses decentralized, publicly auditable, blockchains, removing the need for trusted authorities; OpenTimestamps’s architecture is designed to support multiple, cross-checked, notarization methods.

- **Scalability**: OpenTimestamps scales indefinitely, allowing timestamps to be created for free by combining an unlimited number of timestamps into one blockchain transaction by leveraging Merkle-tree.

- **Convenience**: OpenTimestamps can create a third-party-verifiable timestamp in about a second; you don’t need to wait for a blockchain confirmation.

https://petertodd.org/2016/opentimestamps-announcement
OpenTimestamps: Trust

- Decentralized, independent, uncensorable, cross-jurisdictional
- Third party auditable
- Blockchain agnostic

Please note that a timestamp is as reliable as the used blockchain:

- very reliable when using Bitcoin because that blockchain is secured by huge computational power (proof-of-work);
- much less reliable with other public permissionless blockchain;
- when used with private permissioned blockchain its reliability depends on the reliability of the chain governance: in that case traditional certification authorities are probably better.
OpenTimestamps: Scalability

An OpenTimestamps calendar server provides “aggregation before attestation”:

1. aggregation of multiple documents in a Merkle tree data structure
2. attestation of all documents at the same time using just a single blockchain transaction

- Moreover, a calendar server can offer its services to multiple remote OpenTimestamps clients
Merkle Tree: Hash Pointer Binary Tree

- Merkle tree can efficiently summarize large sets of data into one single hash
  1. Hash all documents
  2. Calculate the hash of the $H_A || H_B$ concatenation to obtain $H_{AB}$, the next level of the tree
  3. Iterate the process
- The membership proof is $O(\log N)$: to prove that DOC$_B$ is in the tree only 2 data are needed: $H_A$ and $H_{CD}$
- Timestamp the tree root only

**Merkle root**

- $H_{AB} = \text{hash}(H_A || H_B)$
- $H_{CD} = \text{hash}(H_C || H_D)$
- $H_A = \text{hash}(A)$
- $H_B = \text{hash}(B)$
- DOC$_B$
- $\text{Root } H = \text{hash}(H_{AB} || H_{CD})$
OpenTimestamps: Convenience

- While anyone can timestamp with permissionless blockchain(s) by paying the transaction fees, OpenTimestamps provides public servers free to use without any registration or API key.

- Verifiable timestamp are created in about a second.

- Public format: no vendor lock-in.

- Independently verifiable: no need for calendar server after timestamping.
Drop below a file to **stamp** it: its `.ots` OpenTimestamps receipt will be downloaded automatically. The hash of the file will be calculated inside your browser preserving your privacy.

Alternatively, drop below an `.ots` receipt/proof file to **verify** it.

Drop here a file to **stamp** OR an `.ots` receipt/proof file to **verify**
Drop below a file to **stamp** it: its `.ots` OpenTimestamps receipt will be downloaded automatically. The hash of the file will be calculated inside your browser preserving your privacy.

Alternatively, drop below an `.ots receipt/proof file to **verify** it.

---

**SUCCESS!**

OpenTimestamps receipt created and download started

---

**SHA256:**

610b0a4b2769898674a2624e9330fbd60bbee200db2b57514be49d9a8b63dc25

**Bitcoin is digital gold.txt 24 B**
Drop below a file to **stamp** it: its `.ots` OpenTimestamps receipt will be downloaded automatically. The hash of the file will be calculated inside your browser preserving your privacy.

Alternatively, drop below an `.ots` receipt/proof file to **verify** it.

**Bitcoin is digital gold.txt.ots** 1.9 kB

**Stamped SHA256 hash:**
610b0a4b2769898674a2624e9330fb960beec200db2b57514be49d9a8b63dc25

**VERIFIED!**

Bitcoin block 471809 attests existence as of 2017-06-18 CEST

Drop here the original file to check that its hash matches the stamped one.
Bitcoin is digital gold. txt 1.9 kB

Stamped SHA256 hash:
610b0a4b2769898674a2624e9330fd60bbe200db2b57514be49d9a8b63dc25

☑️ VERIFIED!

Bitcoin block 471809 attests existence as of 2017-06-18 CEST

Bitcoin is digital gold. txt 24 B

SHA256:
610b0a4b2769898674a2624e9330fd60bbe200db2b57514be49d9a8b63dc25

☑️ SUCCESS!
The provided file is the stamped one: its hash value matches the stamped one
What Timestamping is Not

It should be obvious, but it is worth mentioning that timestamping:

- can be selectively revealed to show convenient evidence and hiding inconvenient evidence (e.g. timestamping a bet outcome and its opposite, later revealing only the right one)
- does not prove authorship (that should be proved using a digital signature);
- can be repudiated ("it was not me...") if not digitally signed;
- does not ensure veracity, validity, correctness, or accuracy of the timestamped document.
The Foolish Blockchain Certification

- IBM Food Trust
- EY Wine Blockchain
- Carrefour chicken
- Etc.

- just dishonest marketing gimmick, i.e. misleading advertising.
Use Case 1: Digital Signature without Timestamping

- What if a signing private key is stolen?
- The key revocation certificate is issued to signal that signatures after the theft should be considered invalid
  
  **WRONG!!**

- Every signature performed with that key should be considered invalid because the thief can backdate documents
Use Case 1: Digital Signature with Timestamping

- Traditional timestamping relies on a third-party central authority signing with its private key.
- What if the timestamper’s private key is stolen?
- Every timestamp created by that key must be considered invalid because the thief can backdate timestamps.

*Time*

- $T_0$
- $T_1$
Use Case 1: Digital Signature with Blockchain Timestamping

- Blockchain notarization is an effective hardening approach
- What if the traditional timestamper’s private key is stolen?

**Blockchain timestamps cannot be backdated!**

© 2019 Digital Gold Institute
Use Case 1: Hardened Digital Signature

Hardened digital signature

💡 Timestamping that cannot be backdated

https://gist.github.com/RCasatta/6824c80e3de137f0d8d230f622e4bbaa
Use Case 2: Timestamp Internet

- OpenTimestamps is used to timestamp the whole Internet Archive [https://archive.org/](https://archive.org/)

- This has been possible thanks to the high scalability of the OpenTimestamps protocol

- For the first time historical archived data cannot be altered without being noticed

Use Case 3: Regulatory Compliance

- Broker-dealers have started using notarization to satisfy the regulatory prescriptions for storing required records exclusively in non-rewriteable and non-erasable electronic storage media.

- WORM (write once read many) optical media has been used so far, but it is quite impractical, especially for large data set

- Compliance can be achieved anchoring rewritable data sources to the blockchain, providing accurate and secure time-stamping resilient to manipulation

https://drive.google.com/drive/folders/0B8tGDTaBY4-Nb32uRmgzRXJXOUk
Use Case 4: Publicly Verifiable Certificates

It is easy to verify documents:
- signed by the issuer
- timestamped on blockchain

It would be easy to provide public web-portals for drag-and-drop verification
Blockchain Certification: the Italian Law

- AGID will have to provide technical specification
- Let’s hope for the best...
- Blockchain cannot be used to track provenance of Italian tomatoes as Di Maio wishes...

Anchoring: A New Security Paradigm

- Bitcoin blockchain network security is preserved by a computation power unparalleled in human history
- Other networks can tap into this security via anchoring (i.e. periodic time-stamping of their network status)
- Any “stateful system with global memory” can outsource its security to the bitcoin network, piggybacking its resilience
- Bitcoin seigniorage revenues might provide security for all transactional networks
- Bitcoin mining as global outsourced decentralized security
Digital Gold Jewelry

What jewelry is for gold, notarization could be for bitcoin:

not essential

but effective at leveraging its beauty
Bibliography

- OpenTimestamps protocol [https://opentimestamps.org/]
- Bitcoin & Blockchain Technology, Milano-Bicocca University, Spring 2019 [https://www.ametrano.net/bbt/]
- Bitcoin, YouTube videos, [https://goo.gl/byVNqP]
- Ferdinando Ametrano, Bitcoin as Digital Gold (2018), United Nations Department of Economic and Social Affairs; video: [https://goo.gl/NkEC9w]; slides: [https://goo.gl/szzBXh]
Takeaways

- Blockchain timestamping is the decentralized digital alternative to traditional certification authorities
- Its OpenTimestamps standardization is trust-minimizing, scalable, convenient and free
- Timestamping is not magic: it provides proof of existence at a given date, it does not convey authorship, non-repudiation, veracity, guaranteed origin, etc.
- Most of the time, timestamping only makes sense if coupled with digital signature or alternative authorship proofs
- Centralized timestamping on private permissioned blockchain is no different from traditional Certification Authority
- For a decentralized timestamp to be reliable, it must use bitcoin
- Timestamping, notarization, and anchoring are digital gold jewelry
Ferdinando M. Ametrano
Executive Director
ferdinando@dgi.io

To Be Announced Soon
Chief Operating Officer
info@dgi.io

info@dgi.io
www.dgi.io
www.dgi.org/feed.xml
www.github.com/dginst
www.twitter.com/DigitalGoldInst
www.facebook.com/DigitalGoldInstitute
www.linkedin.com/company/digital-gold-institute

"Scarcity in the Digital Domain"