

STROMDAO

Distributed ledger technology

An introduction

Stefan Thon

SHA256 Hash

A Hash is a fingerprint of some digital data. The same data always yields the same hash.

Data:

Hallo Leipzig

Hash:

94b26fa1568455e3b35791344d46628947f02e9eba9c8439386033405a650a52

Block

A block is a structured set of digital data to be hashed. It contains a block number, a nonce and data.

There are requirements for a block hash to be valid. To find a valid hash the nonce must be computed.

This process is called mining.

Block:

1

Nonce:

73657

Data:

Hallo Leipzig

Hash:

0000bed300d290bacd91303c0905c9da96b09f0fabb28d7e4db15fe9aca2bce5

Mine

Block chain

A block chain is a series of blocks, whereby each block references the hash of the previous block.

This makes a block chains resist data mutations.

Block:	# 2
Nonce:	8057
Data:	<p>(1) Jeder hat das Recht auf die freie Entfaltung seiner Persönlichkeit, soweit er nicht die Rechte anderer verletzt und nicht gegen die verfassungsmäßige Ordnung oder das Sittengesetz verstößt.</p> <p>(2) Jeder hat das Recht auf Leben und körperliche Unversehrtheit. Die Freiheit der Person ist unverletzlich. In diese Rechte darf nur auf Grund eines Gesetzes eingegriffen werden.</p>
Prev:	00003a4a6d561440199c698abd9daab4b782f17dce9b2f48f571
Hash:	0000af1abde0af03207ae3db6eadc5d4b68a8085d1ad8d0bc8e3
<button>Mine</button>	

Distributed block chain

A block chain with many distributed copies.

This makes it easy to filter out invalid copies of the block chain. (The majority wins.)



Transactional data, not any kind of data!

A transaction is a transfer of value

A transferable value is an asset. A digitally transferable value is a Digital Asset.

Blockchain Technology is Value Technology rather than Information Technology.

\$	10.00	From:	Anders	->	Sophia
----	-------	-------	--------	----	--------

A block is a list of transactions

There is a maximum number of transactions per block.

\$	25.00	From:	Darcy	->	Bingley
\$	4.27	From:	Elizabeth	->	Jane
\$	19.22	From:	Wickham	->	Lydia
\$	106.44	From:	Lady C.	->	Collins
\$	6.42	From:	Charlotte	->	Elizabeth

A block chain consists of multiple lists (or blocks) of transaction

Block:

#	1
---	---

Tx:

\$	25.00	From:	Darcy	->	Bingley
\$	4.27	From:	Elizabeth	->	Jane
\$	19.22	From:	Wickham	->	Lydia
\$	105.44	From:	Lady Cathor	->	Collins
\$	6.42	From:	Charlotte	->	Elizabeth

Block:

#	2
---	---

Tx:

\$	97.67	From:	Ripley	->	Lambert
\$	48.61	From:	Kane	->	Ash
\$	6.15	From:	Parker	->	Dallas
\$	10.44	From:	Hicks	->	Newt
\$	88.32	From:	Bishop	->	Burke
\$	45.00	From:	Hudson	->	Gorman
\$	92.00	From:	Vasquez	->	Apone

Chaining blocks together

Transaction lists are chained together by each list referencing its previous list.

Block:

#1

Nonce:

139358

Tx:

\$	25.00	From:	Darcy	->	Bingley
\$	4.27	From:	Elizabeth	->	Jane
\$	19.22	From:	Wickham	->	Lydia
\$	106.44	From:	Lady Cathor	->	Collins
\$	6.42	From:	Charlotte	->	Elizabeth

Prev:

00

Hash:

00000c52990ee86de55ec4b9b32beefd745d71675dc0eddfbc7b8833

Block:

#2

Nonce:

39207

Tx:

\$	97.67	From:	Ripley	->	Lambert
\$	48.61	From:	Kane	->	Ash
\$	6.15	From:	Parker	->	Dallas
\$	10.44	From:	Hicks	->	Newt
\$	88.32	From:	Bishop	->	Burke
\$	45.00	From:	Hudson	->	Gorman
\$	92.00	From:	Vasquez	->	Apone

Prev:

00000c52990ee86de55ec4b9b32beefd745d71675dc0eddfbc7b8833

Hash:

000078be183417844c14a9251ca246fb15df1074019873f5d85c1a6f

Chaining blocks together

The process of validating and referencing the previous list is called mining. It's crypto magic.

Block:

#1

Nonce:

139358

Tx:

\$	25.00	From:	Darcy	->	Bingley
\$	4.27	From:	Elizabeth	->	Jane
\$	19.22	From:	Wickham	->	Lydia
\$	106.44	From:	Lady Cather	->	Collins
\$	6.42	From:	Charlotte	->	Elizabeth

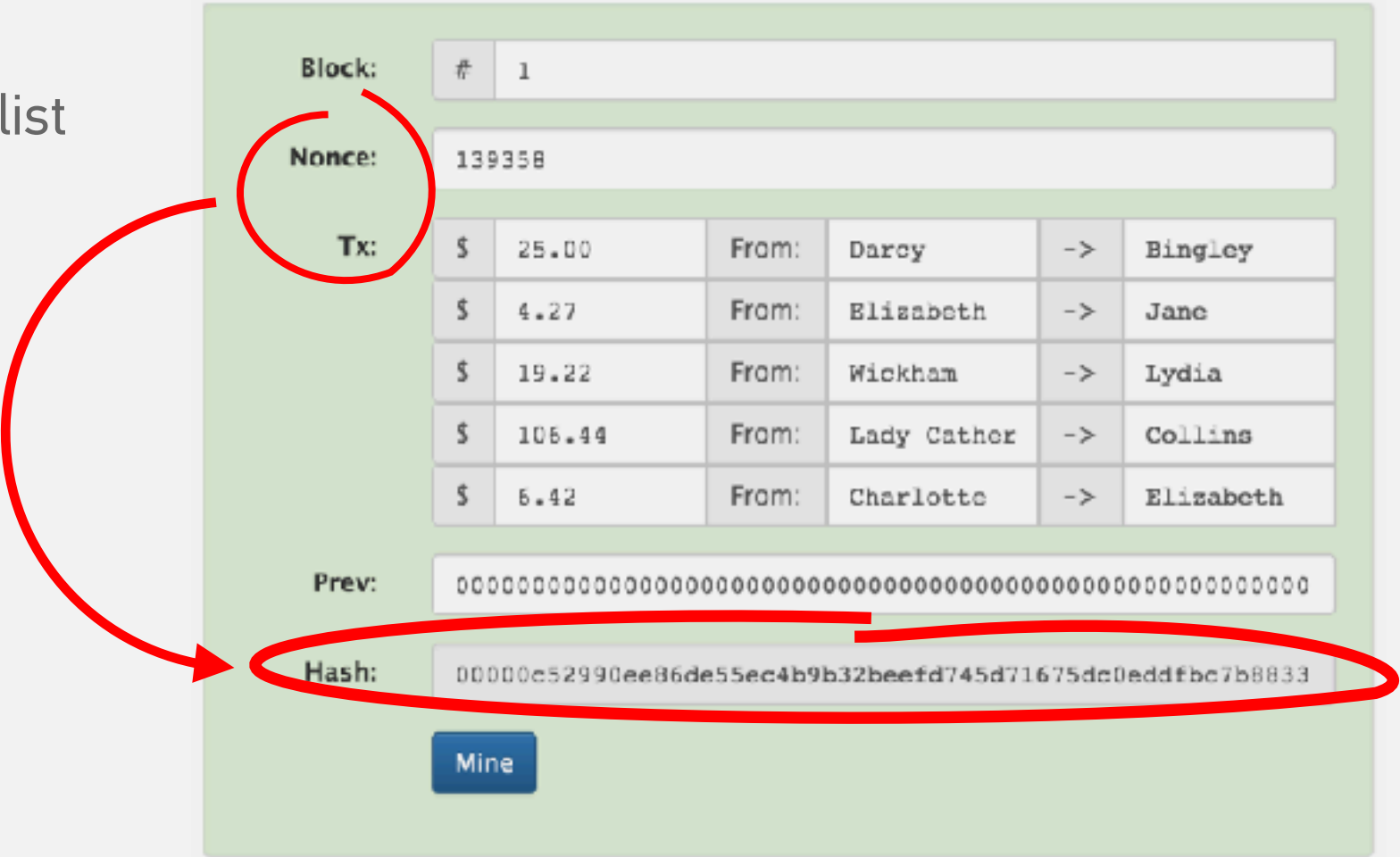
Prev:

00

Hash:

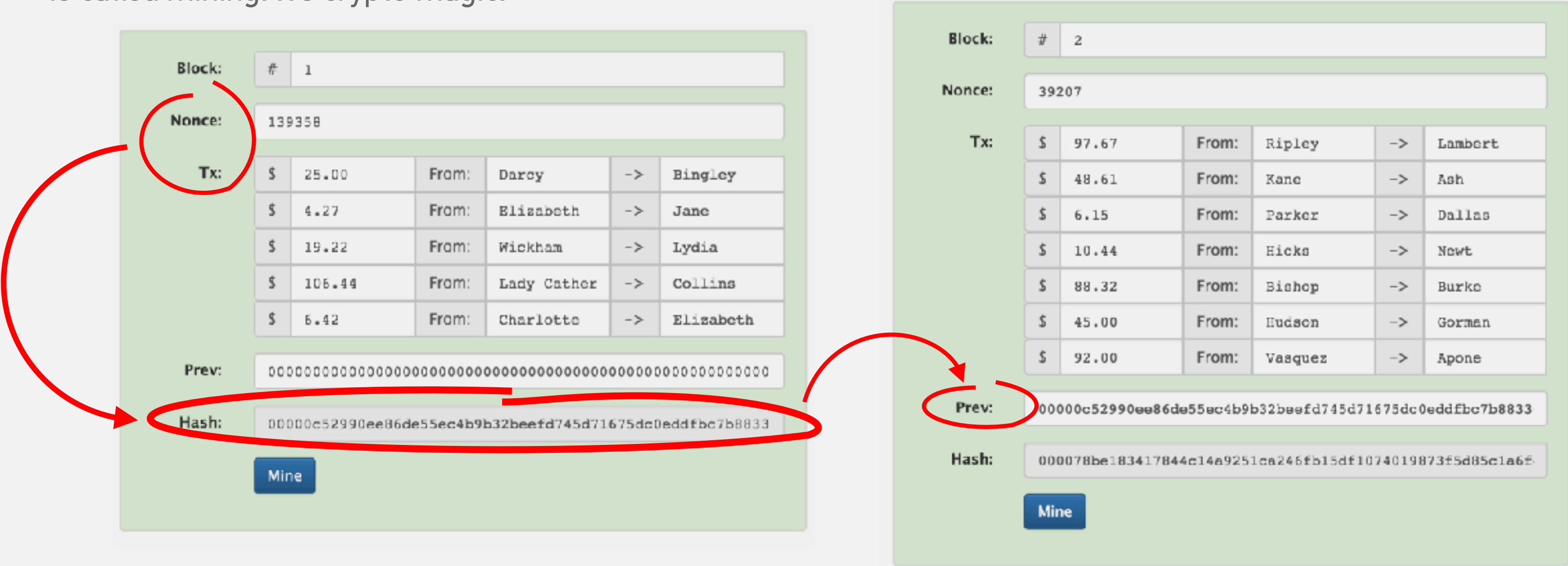
00000e52990ee86de55ec4b9b32beefd745d71675dc0eddfbc7b8833

Mine



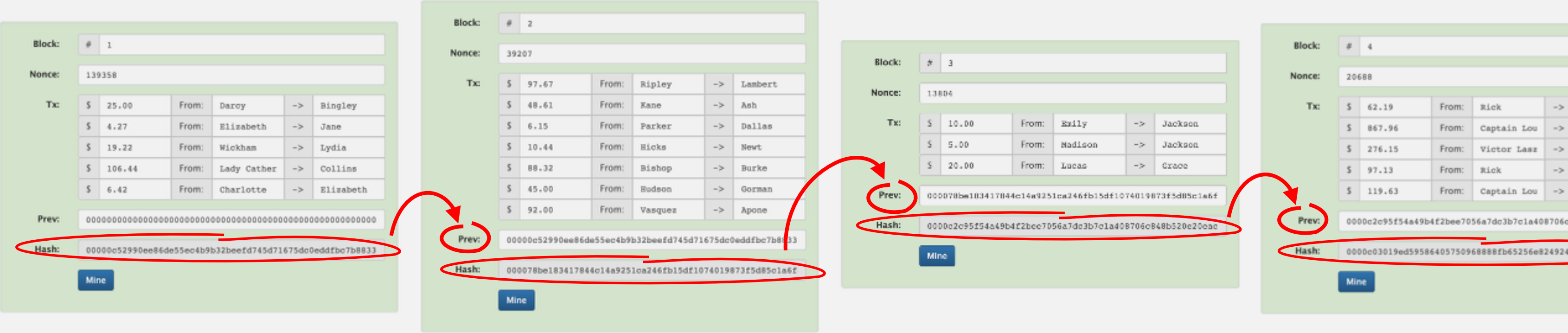
Chaining blocks together

The process of validating and referencing the previous list is called mining. It's crypto magic.



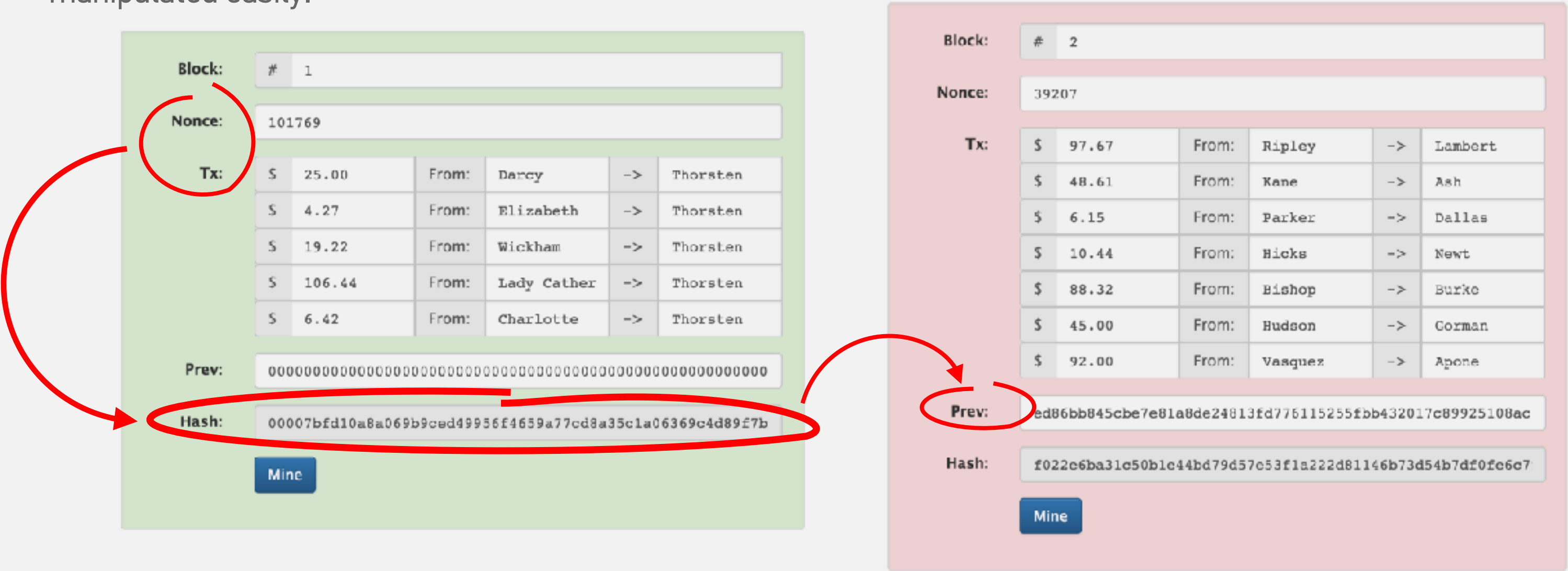
Chaining blocks together

This allows to store an unlimited number of Transactions within a chain structure.



Data consistency

Once validated and chained, transactions can not be manipulated easily.



Data redundancy

Each user (peer) works with her own copy of the full chain.

Peer A

Block: # 1

Nonce: 139358

Tx

\$ 25.00	From: Samy	->	Mingling
\$ 4.27	From: Elizabeth	->	Jane
\$ 35.02	From: William	->	Lydia
\$ 300.44	From: Gault Galtier	->	Cullins
\$ 6.42	From: Charlotte	->	Elizabeth

Prev: D00000CCCC099880099880033300000CCCC099880033400333000000

Hash: D00000C725908880d8556c13d322887c145d71615d03e6d73c7288133

Mine

Block: # 2

Nonce: 29287

Tx

\$ 91.47	From: Mingling	->	Lambert
\$ 48.01	From: Jane	->	Josh
\$ 6.15	From: Esther	->	Bellon
\$ 10.88	From: Eliska	->	Bera
\$ 88.38	From: Eliska	->	Berke
\$ 45.03	From: Norman	->	Norman
\$ 92.03	From: Nanquet	->	Apone

Prev: C09980529930086d885c4b9b22bee47f15d71676dc0e0f8e7b8993

Hash: C09978be133437044e34e5253ee2c6fb158f1374039f71d5485e1a5d

Mine

Block: # 3

Nonce: 13804

Tx

\$ 10.00	From: Ming	->	Norman
\$ 5.00	From: William	->	Jackson
\$ 20.00	From: Lorne	->	Grace

Prev: 103417044c34e5253ee2c6fb158f1374039f71d5485e1a5d88e0

Hash: D000e2e95854e45b4f2bee7356a7dc3b7e1c4087c6e848b529a2Bee0

Mine

Block: # 4

Nonce: 20008

Tx

\$ 67.19	From: Nina	->	Lisa
\$ 867.90	From: Captain Lee	->	Strasser
\$ 276.15	From: Victor Jean	->	Lisa
\$ 57.13	From: Rick	->	Sam
\$ 119.69	From: Captain Lee	->	Jan Brandel

Prev: CCCC0e93258a89b4E2bee7036e10c3b7c1a8397D6c84E030e2C0e0

Hash: CCCC0e1019a15458982575078888887B65236a42892880c53e08f8d17

Mine

Block: # 5

Nonce: 33033

Tx

\$ 18.12	From: Norman Lorne	->	Emmett Lee
\$ 2,760.38	From: Lord Glenda	->	John Money
\$ 813.70	From: Nathaniel D	->	Alan Rodney

Prev: 0003a33019ad88f64c85f609f88888b68286e82452c8848fe3a5a8d8f

Hash: 0003aaaD00c5cb5102ee5b3331624be3fe6776cf8e75bd43d4276e03

Mine

Peer A

[illegible][illegible]

Block	# 5				
Miner	10003				
Tx	5 14.12	From	James Lamm	->	James Lamm
	5 2,760.28	From	Lord Glenda	->	John Henry
	5 415.70	From	Rachaela D	->	Alan Pottery
Proof	00000302088f1c14-01409388804c268d240c-1849f3a5d6d1				
Hash	00000ad0e08c122ee33313240876770c78443762f6d03				
	More				

Each user (peer) works with her own copy of the full chain.

Peer C

Block:	#	2																																			
Header:	25020																																				
TX:	<table><tr><td>5</td><td>11.02</td><td>From: winging</td><td>-></td><td>Lamwatt</td></tr><tr><td>4</td><td>10.51</td><td>From: Bata</td><td>-></td><td>Jah</td></tr><tr><td>5</td><td>4.15</td><td>From: Eulach</td><td>-></td><td>Belina</td></tr><tr><td>5</td><td>10.00</td><td>From: Eulach</td><td>-></td><td>Bew.</td></tr><tr><td>5</td><td>00.02</td><td>From: Eulach</td><td>-></td><td>Buck</td></tr><tr><td>5</td><td>05.00</td><td>From: oceanon</td><td>-></td><td>oceanon</td></tr><tr><td>5</td><td>02.05</td><td>From: Vagabond</td><td>-></td><td>Apona</td></tr></table>		5	11.02	From: winging	->	Lamwatt	4	10.51	From: Bata	->	Jah	5	4.15	From: Eulach	->	Belina	5	10.00	From: Eulach	->	Bew.	5	00.02	From: Eulach	->	Buck	5	05.00	From: oceanon	->	oceanon	5	02.05	From: Vagabond	->	Apona
5	11.02	From: winging	->	Lamwatt																																	
4	10.51	From: Bata	->	Jah																																	
5	4.15	From: Eulach	->	Belina																																	
5	10.00	From: Eulach	->	Bew.																																	
5	00.02	From: Eulach	->	Buck																																	
5	05.00	From: oceanon	->	oceanon																																	
5	02.05	From: Vagabond	->	Apona																																	
prev:	0009e3995d995d9ed6ee0369322e046718705476706e036798392																																				
Next:	0009f80c130817086e1c4c2532e046718705476706e036798392f4080a1a6f																																				
	More																																				

Beck:	6 4			
Phone:	21488			
Tz:	6.2.19	From:	stara	-> 1 ka
	847.81	From:	Capitán Los	-> Stranzer
	274.15	From:	Victor Los	-> Los
	37.13	From:	Bluh	-> Ben
	119.49	From:	Capitán Los	-> Osa Brander
Prev:	CCCCC93733a8b42d280703647e10b7c1a3870b0a5520a20a0e			
Next:	CCCCC93733a8b42d280703647e10b7c1a3870b0a5520a20a0e			

[illegible]

Peer D

Block:

Message:

Tx:

5	11.02	From:	utopia	->	LANDART
5	19.01	From:	Baba	->	kah
5	1.02	From:	...	->	...

Week:	6 4				
Name:	JACK				
Tx:	5	62.19	From:	Mike	-> 1:00
	5	63.95	From:	Captain Lee	-> Strauss
	6	65.10	From:	Mike	-> 1:00

Black	#	5
Name	11011	
Tic	\$	11.12
	Face	David Laro
		-5
	Count	1000
	\$	2,700.28
	Face	Lord Glando
		-5
	Count	John Henry

[illegible]

This increases the consistency guarantee.

[illegible][illegible]

Block: # 1	Block: # 2	Block: # 3	Block: # 4	Block: # 5
Name: 130358	Name: 29287	Name: 13004	Name: 21668	Name: 11031
Tx: 5 25.10 From: Jerry -> string	Tx: 5 47.47 From: string -> Lawrence	Tx: 5 11.10 From: emup -> jammem	Tx: 5 42.19 From: Hira -> 1.44	Tx: 5 14.12 From: joshua1234 -> emmett1234
6 4.27 From: Elanbeth -> Jane	5 44.51 From: Vera -> jeb	5 5.00 From: daffman -> Jackson	5 147.01 From: Captain Lee -> Ptrauser	5 20,760.28 From: Lord Gladio -> John Henry
6 66.66 From: Elanbeth -> Jane	6 4.14 From: Vera -> jeb	6 66.66 From: emup -> jammem	6 66.66 From: Hira -> 1.44	6 66.66 From: joshua1234 -> emmett1234

Controlling asset circulation with coinbase transactions

Initial asset creation

Creation and dissemination of is central to any consensus system.

Block:	#	1		
Nonce:	16651			
Coinbase:	\$	100.00	->	Anders
Tx:				
Prev:	00			
Hash:	0000438d7625b86a6f366545b1929975a0d3f51f8847e5			
<div>Mine</div>				

Block:	#	2				
Nonce:	215459					
Coinbase:	\$	100.00	->	Anders		
Tx:	\$	10.00	From:	Anders	->	Sophia
	\$	20.00	From:	Anders	->	Lucas
	\$	15.00	From:	Anders	->	Emily
	\$	15.00	From:	Anders	->	Nadison
Prev:	0000438d7625b86a6f366545b1929975a0d3ff1f8847e5					
Hash:	0000baeab68c2a60f9a6fa56355438d97c672a15494fce					

Block:	#
Nonce:	146
Coinbase:	\$
Tx:	\$
	\$
	\$
Prev:	000
Hash:	000
	Min

Authorising transactions

Public / private key pairs

A public key is data that is computed from a private key.

There is no way to derive from a public key what the private key is that relates to it.

Private Key

30911981

Random

Public Key

04d6e78f227811c065e02f746f5905afb2db809d691cdd

Signing a messages with a private key

Signing data with a private key yields a unique signature that can be verified against the public key of the signee.

Message

Hallo Leipzig

Private Key

30911981

Sign

Message Signature

3046022100d93586571ec12aa6d59dee74f6343cedceea

Verifying a message against a public key

Verifying a signature validates that the person who signed the data in question had access to the private key behind the public key.

Message

Hallo Leipzig

Public Key

04d6e78f227811c065e02f746f5905afb2db809d691cdd

Signature

3046022100d93586571ec12aa6d59dee74f6343cedceea

Verify

Signing a transaction

In a block chain context, every transactions must be signed by the actor who initiates the transaction.

Message

\$	20.00	From:	04d6e78f227811c	->	04cc955bf8e359c
----	-------	-------	-----------------	----	-----------------

Private Key

30911981

Sign

Message Signature

304402202ae02c3e0c04aa430a9d9a6fd0c12c7e2337b710e2238d67331693b48

Verifying a transaction

Verifying signed transactional data against the public key of the sender, validates that the sender did initiate said transaction with his or her private key.

Message

\$

20.00

From:

04d6e78f227811c

->

04cc955bf8e359c

Signature

304402202ae02c3e0c04aa430a9d9a6fd0c12c7e2337b710e2238d67331693b48

Verify

Block chain transactions

Transactions in a block chain consist of public keys for sender & recipient and have to be signed by the sender.

This protects each transaction independent from the mining / block validation process.

Block:

#2

Nonce:

7054

Coinbase:

\$100.00->04fe1be031bc7a54d900ff

Tx:

\$10.00From:04fe1be031b->04cc17dc129

Sig:3046022100bcfe74e2ee8972367dda52a8f90008800ad10fb

\$20.01From:04fe1be031b->04997ac426a

Sig:304502210089cbf8f4bc854fb010c3bb7747f8c4c010fd029

\$15.00From:04fe1be031b->042222d7af3

Sig:3045022036cfd31dbdc400993a612bf9ba9c897a75b4578c8

\$15.00From:04fe1be031b->041c377677b

Sig:3045022036cfd31dbdc400993a612bf9ba9c897a75b4578c8

Prev:

00006908f507a101e89544498978e9bd2e35462b91d86ef1351068f

Hash:

0000b1df512e06962d2d177bb6c0fd0e5513d1cdeffdc52a79cc49f

Takeaways

1. **Immutability:** History can not be changed.
2. **Openness:** Everyone and everything with a key pair can participate.

**A fault tolerant, highly redundant
transaction store...**

**...to establish & maintain irrevocable consensus
among market participants.**

A value machine

Blockchain technology can be utilised to facilitate and enshrine any kind of transaction between market participants.

Backup

Blockchain-Akteure & Adressen.

Jedes Objekt, das eine Transaktion in der Blockchain vornehmen kann, hat eine eindeutige Adresse.

Rückschlüsse, ob es sich dabei um einen Marktakeur, SmartContract, oder Token handelt, sind nicht möglich.

Adressen sind (unveränderlich). Ein Smart Contract mit einer bestimmten Adresse kann nicht mehr nachträglich in seiner Funktion verändert werden.

Transfers of value must adhere to rules to be effective

A transfer of value requires irrevocable and verifiable proof of the transfer itself and of its necessary preconditions.

Transaction rules may be enshrined in (Smart) Contracts. Contracts effect the transfer of value.

```
1 pragma solidity 0.4.18;
2 contract SimpleMultiSig {
3
4     uint public nonce;           // [only] mutable state
5     uint public threshold;       // immutable state
6     mapping (address => bool) isOwner; // immutable state
7     address[] public ownersArr;  // immutable state
8
9     function SimpleMultiSig(uint threshold_, address[] owners_) public {
10         require(owners_.length <= 10 && threshold_ <= owners_.length && threshold_ != 0);
11
12         address lastAdd = address(0);
13         for (uint i=0; i<owners_.length; i++) {
14             require(owners_[i] > lastAdd);
15             isOwner[owners_[i]] = true;
16             lastAdd = owners_[i];
17         }
18         ownersArr = owners_;
19         threshold = threshold_;
20     }
21
22     // Note that address recovered from signatures must be strictly increasing
23     function execute(uint8[] sigV, bytes32[] sigR, bytes32[] sigS, address destination, uint
value, bytes data) public {
24         require(sigR.length == threshold);
25         require(sigR.length == sigS.length && sigR.length == sigV.length);
26
27         // Follows ERC191 signature scheme: https://github.com/ethereum/EIPs/issues/191
28         bytes32 txHash = keccak256(byte(0x19), byte(0), address(this), destination, value, data,
nonce);
29
30         address lastAdd = address(0); // cannot have address(0) as an owner
31         for (uint i = 0; i < threshold; i++) {
32             address recovered = ecrecover(txHash, sigV[i], sigR[i], sigS[i]);
33             require(recovered > lastAdd && isOwner[recovered]);
34             lastAdd = recovered;
35         }
36
37         // If we make it here all signatures are accounted for
38         nonce = nonce + 1;
39         require(destination.call.value(value)(data));
40     }
41
42     function () public payable {}
43 }
```