

# Informatik

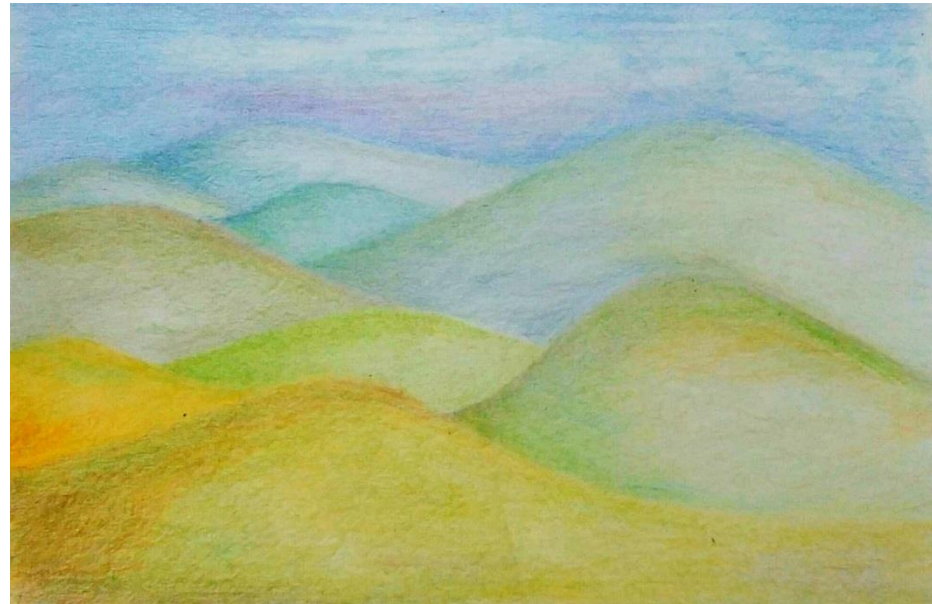
## Bitcoin & Blockchain

**Irene Rothe**

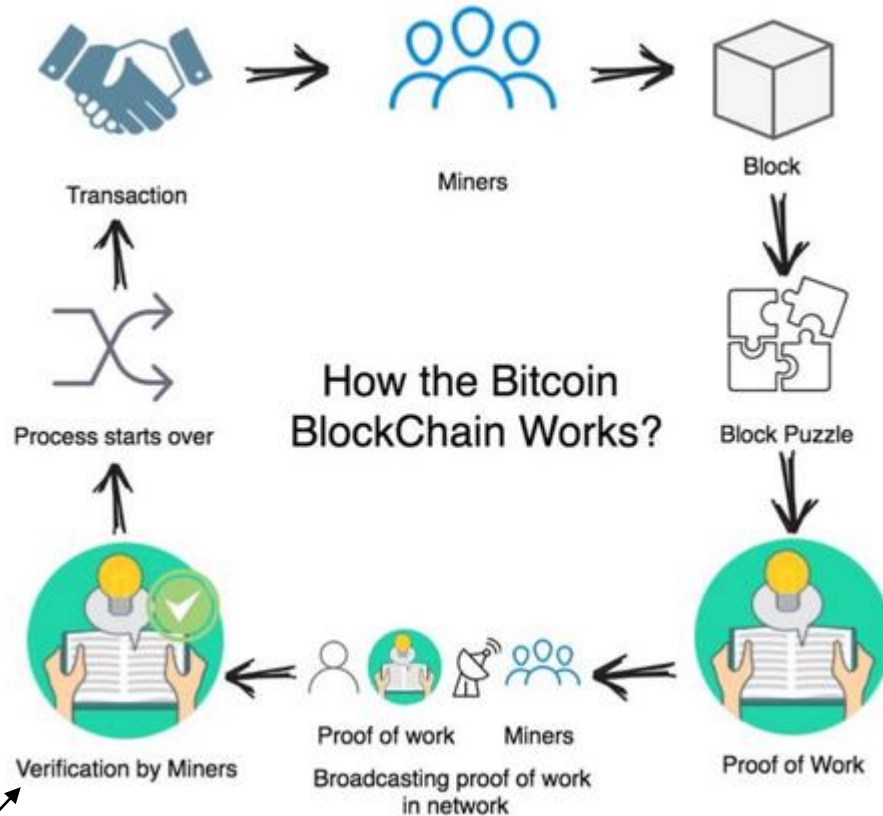
Zi. B 241

[irene.rothe@h-brs.de](mailto:irene.rothe@h-brs.de)

Instagram: irenerothesdesign

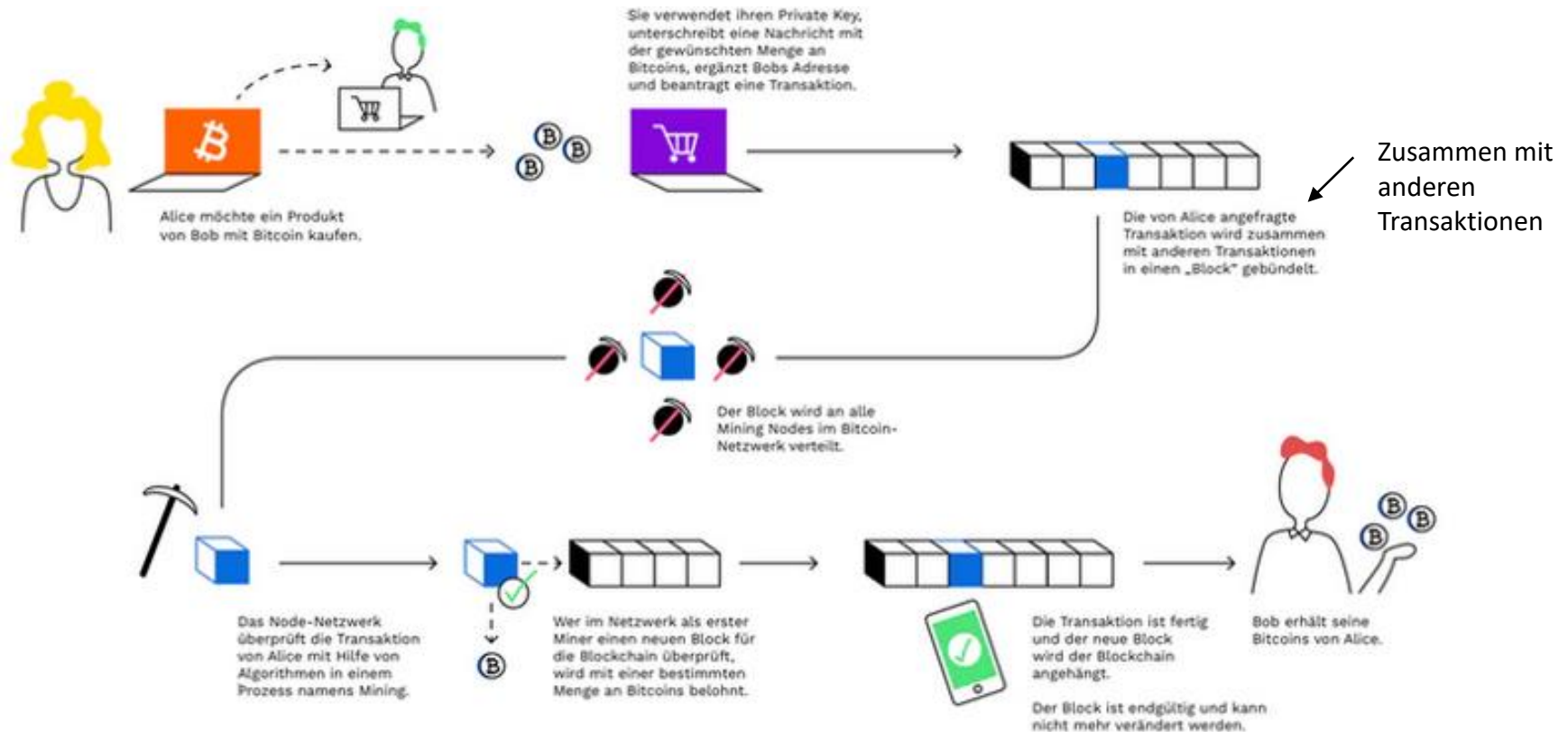


# Ablauf



Bedeutet, dass Blockpuzzle, proof of work und broadcasting fertig sind und die Validierung des neuen Blockes abgeschlossen ist

# Ablauf einer Transaktion



<https://www.bitpanda.com/academy/de/lektionen/was-ist-bitcoin-mining-und-wie-funktioniert-es/>

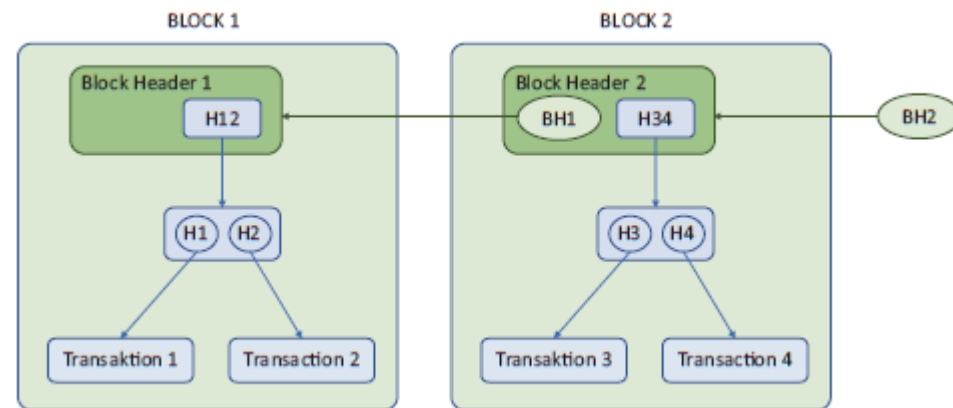
# Beantwortung meiner eigenen Fragen

- Wird das Ergebnis des Blockpuzzles benutzt, z.B. als Adresse für die validierte Transaktion?  
→ die eindeutige Identifikationskennzeichnung für den neuen Block ist der aufwändig errechnete Hash-Wert, wofür der erste Miner (ders geschafft hat, einen Nonce (number only used once) für einen geforderten Ergebnis-Hash-Wert mit so und so vielen (16) Nullen am Anfang zu finden) einen Lohn bekommt  
Beispiel: die größte mineralogische Sammlung (Wella) hat der bekommen, der als erstes ein tolles Haus dafür fertig hatte (also nicht günstig ist diesmal ausschlaggebend, sondern als erster)
- Was genau bedeutet Transaktion validieren? Hashwert berechnen durch Suche nach Nonce+Überprüfen durch Konsens (proof of work)+broadcasting in network
- Was bedeutet redundante Datenhaltung? Haben alle alles?
- Was ist die Nonce: Ist das die Zahl, die durch Ausprobieren gefunden werden muss von den Miners? A nonce is an abbreviation for "number only used once," which is a number added to the block transaction information in a blockchain that, when hashed, meets the difficulty level restrictions (dh die 16 Nullen am Anfang). The nonce is the number that blockchain miners are solving for/that entitles the miner the transaction fee.



# Blockchain und Bitcoin

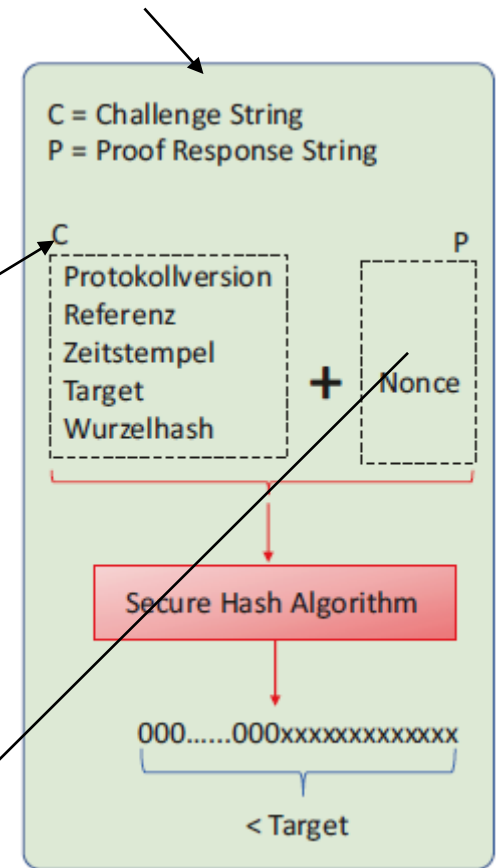
- Eine **Blockchain** ist eine Kombination aus drei Komponenten:
  1. einem Signatur-Verfahren
  2. einer dezentralen Datenbank (also keine zentrale Geldbank wie Sparkasse oder so), sowie
  3. einem Konsensfindungs-System.
- Alle Transaktionen sind in verketteten Blöcken dokumentiert
- Neue Transaktionen werden in neuen Blöcken hinzugefügt
- Informationen werden redundant auf allen Knoten der Kette gehalten (???kapiert ich nicht wie, so viel Platz hat doch keiner???)
- Beispiel Buchhaltung: Blockchain = Hauptbuch
- Jeder Teilnehmer der Bitcoin-Datenbank kann nachvollziehen, welche Transaktionen getätigt wurden
- Transaktionen werden von Adresse zu Adresse abgehandelt. Die Personen hinter den Adressen sind anonym.



# Miner

- ... kann jeder sein, der die *quelloffene* Bitcoin Software installiert und seine *Rechnerkapazität* zur Verfügung stellt
- ... fassen mehrere Transaktionen eines bestimmten Zeitraums zusammen und validieren sie
- Neue angestoßene Transaktionen werden in einem neuen Block dokumentiert und an das Ende der Kette angehängt
- Bevor der Block der Kette hinzugefügt werden kann, müssen die darin enthaltenen Informationen vom Netzwerk überprüft werden. Dies geschieht durch Erstellen eines sogenannten Hashs, eine 256-Bit-Zahl (SHA256), die die Daten im Block eindeutig identifiziert.
- *Um diesen Hash zu erstellen*, müssen Knoten (Miners) im Netzwerk ein komplexes „mathematisches Puzzle“ lösen.
- Sobald ein Miner das Puzzle gelöst hat (also P=Nonce gefunden hat, damit die geforderte Anzahl von Nullen am Anfang des Hashs entstehen, prüfen alle anderen Knoten im Netzwerk, ob die Berechnungen korrekt sind (was sehr einfach (effizient) geht, sie nehmen die Daten aus C+den bekannten Algorithmus+den gefunden Hash vom Gewinnerminer) → peer-to-peer Konsens
- Wer das Puzzle als erster raus hat, erhält Bezahlung (=finanzieller Anreiz für Rechenleistung, gleichzeitig also Schaffung neuer Bitcoins).
- Mit jedem neuem Block aktualisiert sich die Kette auf jedem Knoten im Blockchain-Netz (Frage: also liegt in jedem Knoten im ganzen Netz jeweils die ganze Blockchain? Ist das nicht recht viel an Daten?)
- Gibt es mehrere Blöcke, die in die Hauptkette eingebunden werden sollen, wird die längste genommen, weil da schon die meiste Rechenleistung reingegangen ist (Blöcke mit einem oder keinen Nachfolger, gelten generell als unverbindlich).

Hashen des Blockheaders



Nonce ist extra Zufallszahl, die gefunden werden muss von den Miners durch Ausprobieren, damit mit dem vorgegebenen Hash-Algorithmus aus dem Datenheader des neuen Blocks die 16 Nullen am Anfang des Hashwertes entstehen

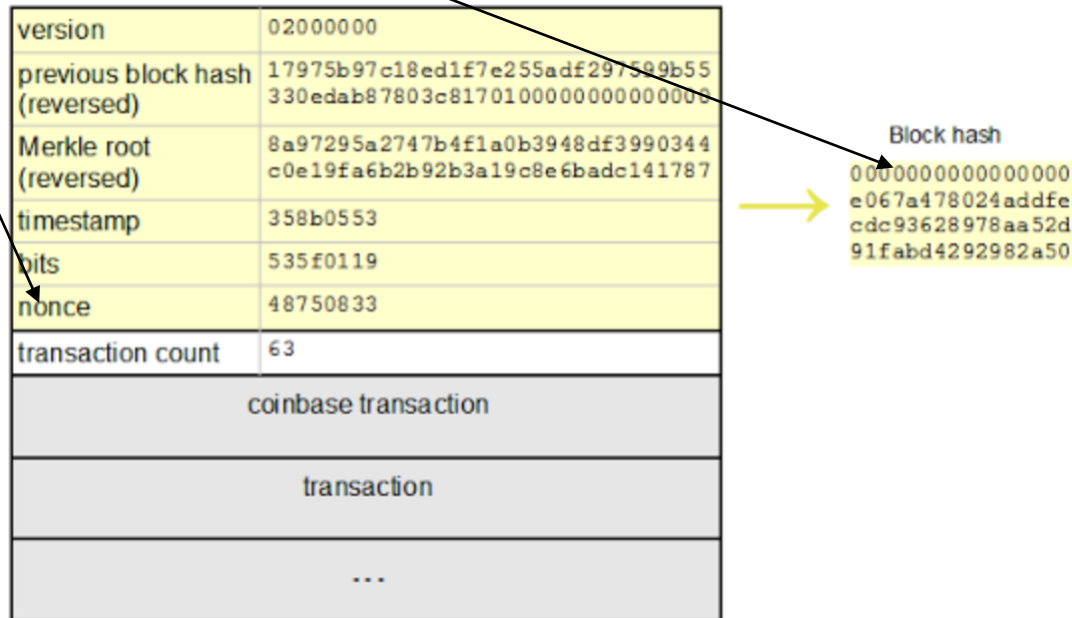
# Miner – noch mal genauer

- ... are looking for the correct nonce that would solve the puzzle
- This is the only field that is permitted to be changed in each trail
- Ziel: 16 Nullen am Anfang

version	02000000
previous block hash (reversed)	17975b97c18ed1f7e255adf297509b55330edab87803c817010000000000000000
Merkle root (reversed)	8a97295a2747b4f1a0b3948df3990344c0e19fa6b2b92b3a19c8e6badc141787
timestamp	358b0553
bits	535f0119
nonce	48750833
transaction count	63
coinbase transaction	
transaction	
...	

Block hash

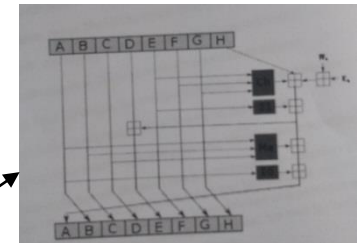
0000000000000000  
e067a478024addfe  
cdc93628978aa52d  
91fabd4292982a50



Beispiel aus dem täglichen Leben?

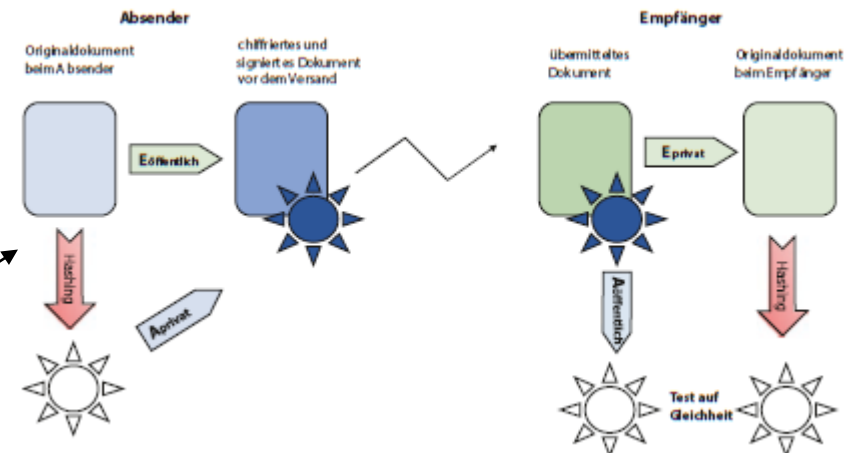
# Hash-Algorithmus

- ... ist Hash (Fleischwolfgehacktes-ursprüngliche Form nicht mehr sichtbar und darauf schließbar)-Funktion (Einwegfunktion: unumkehrbar): Eingabe beliebiger Länge wird auf feste Länge **effizient** abgebildet = Hash
- injektive Abbildung
- Wichtig: Effizienz der Berechnung und Lawineneffekt (kleinste Änderung in Eingabe erzeugt totale Änderung im Hash)
- Beispiele für Hash-Funktionen:
  - Sehr einfach: Quersumme von der Eingabe
  - SHA 256: Aufteilung der Eingabe in Blöcke gleicher Länge, Padding der Blöcke, stellenweise modulo 2 rechnen von Block 1 mit Block 2, Ergebnis mit Block 3, usw .... dies wird 64 Runden oder 80 Runden wiederholt
  - Oder mit Primzahl und exklusives ODER und inklusives ODER



## Anwendungen:

- über Hash-Funktionen werden Adressen an Variablen im Speicher vergeben
- Passwörter verschlüsseln (zum Abgleich wird eingegebenes Passwort gehasht und mit dem als Hash-Wert abgelegten Wort in der Datenbank verglichen)
- Überprüfung von sicher unveränderten Daten (Integritätsschutz)
- Suche in Datenbanken über kompakte Hash-Tabelle





# Bitcoin-Konten

- ... werden verwaltet mit virtuellem Wallet
- Wallets generieren Schlüsselpaare (privat und public)
- Der public Schlüssel wird in Adresse umgewandelt = sichtbare Kontonummer
- Private Schlüssel kann auch als digitale Unterschrift genutzt werden

# Neuer Block in Blockchain

Bitcoin-Transaktion beinhaltet

- die öffentliche Adresse des Empfänger-Kontos
- den Überweisungsbetrag
- die öffentliche Adresse des Sender-Kontos
- den private Key zu dieser öffentlichen Adresse, um Transaktionen ausgehend von diesem Konto zu signieren

Neuer Block besteht aus folgenden Daten:

- Liste vieler Transaktionen
- Nonce (die die Miners suchen, um dafür Bitcoins zu erhalten)
- Referenz des Blockvorgängers (track blockchain über blockchain.info)
- SHA256 Hashwert

Nach Validation des neuen Blocks wird die Blockchain wieder auf mehrere Speicherorte (Knoten) hochgeladen. Wird eine Transaktion entfernt oder verändert, stimmen die Hashwerte der folgenden Transaktionen nicht mehr.

Es ist eigentlich nicht möglich, dass unterschiedliche Werte gleich Hash-Werte ergeben.

Beispiel aus Wikipedia:

```
Input: Previous tx: f5d8ee39a430901c91a5917b9f2dc19d6d1a0e9cea205b009ca73dd04470b9a6 Index: 0
scriptSig: 304502206e21798a42fae0e854281abd38bacd1aeed3ee3738d9e1446618c4571d10
90db022100e2ac980643b0b82c0e88ffdfec6b64e3e6ba35e7ba5fdd7d5d6cc8d25c6b241501 Output: Value:
2500000000 scriptPubKey: OP_DUP OP_HASH160 404371705fa9bd789a2fcd52d2c580b65d35549d OP_EQUALVERIFY
OP_CHECKSIG
```

**Bemerkung:** Nach Anhängen eines neuen Blocks dauert es locker 10 Minuten bis nächster Block angehängt werden kann, weil so lange die Miners an dem Nonce für den speziellen Hashwert rumrechnen



# Eigenschaften

- Kein double spending möglich
- Kein trust party nötig
- Teilnehmer anonym
- Fully peer-to-peer
- New coins are made from hashcash via style proof-of-work
- Programs=Data structures+algorithms → Blockchain=Datenstruktur + Konsens

Probleme:

- Hoher Energieverbrauch
- Keine Identitätsprüfung möglich

# Others

- Zur Zeit (Juni 2021) ist ein Bitcoin 30 T Euro wert
- Unter Pseudonym **Satoshi Nakamoto** ist der Erfinder nur bekannt, 2008 kam er mit der Idee durch ein White Paper raus, 2009 Implementierung
- Es gibt nur endlich viele Bitcoins: **21 Millionen** (man kann Bitcoins für immer verlieren, also gibt es wahrscheinlich noch weniger), Miner erhalten fürs Blöcke bilden diese Bitcoins, der Betrag halbiert sich alle 4 Jahre (mit anderen Worten: die Hashrate (Maßeinheit für die Rechenleistung eines Miners) wird immer höher=das zu lösende mathematische Problem immer schwerer (z.B. werden immer mehr Nullen gefordert am Anfang des Hashwert als Kennzeichnung eines neuen Blockes)) :
  - die ersten 4 Jahre: 10.500.000 Coins
  - die nächsten 4 Jahre: 5.250.000 Coins
  - nächste 4 Jahre: 2.625.000 Coins
  - die nächsten 4 Jahre: 1.312.500 Coins
  - ...
- Man nimmt an, dass es bis 2140 dauert, bis alle Bitcoins geschürft sind
- Danach werden eventuell Transaktionen kostenlos angeboten
- Umtausch in Euro? Ähnlich wie Tulpen ehemals. Wenn man jemanden findet, der welche gegen Euros verkauft, dann fein. (Thomas Straubhaar: Der Bitcoin zeigt einzig und allein, dass offenbar die Skala der menschlichen Dummheit genauso wenige Grenzen kennt wie der Bitcoin-Kurs.)

# Begriffe

- **Blockchain** (Gruppierung von Transaktionen mit Zeitstempel und Fingerprint des Vorgängers, Block-Header wird gehasht, um den Proof-of-work zu erzeugen und damit die Transaktion zu validieren.): Kette an Daten, die mit Hashes rückversichert ist. Über das Internet sind die Blöcke verteilt, mehrfach. Gültige Blöcke werden der Haupt-Blockchain durch Netzwerkkonsens hinzugefügt.
- **Bitcoin-Adresse**: etwas ähnlich zu einem public key (String nach Basis 58 kodierter Hash-key)
- **Proof of work**: ein Stück Daten (die gehashte eindeutige Identifikationskennzeichnung des neuen Blockes), dessen Auffindung (nur durch Ausprobieren verschiedener Noncen) rechnerisch sehr anstrengend ist, wird von mehreren überprüft, ob es stimmt; also die Lösung des gegebenen Puzzles wird überprüft
- **Wallet**: Mit Schlüsseln aus dem Wallet werden Transaktionen signiert
- **Bitcoin** hat feste Obergrenze: 21 Millionen, ist deflationär, fürs Gegenprüfen einer Transaktion werden Bitcoins neu erschaffen
- **Miner** (Entstehung von Bitcoins) lösen mathematische Aufgaben (dauer mindestens 10 min, in den nächsten Jahren werden die Probleme immer komplizierter und es dauert immer länger, neue Bitcoins zu erzeugen), Validieren so (Vorgang von Hashen des neuen Datenblockes+Überprüfen im Konsens (peer to peer)) Transaktionen und erzeugen zusätzlich Bitcoins als Belohnung für Ihre Arbeit (Rechenleistung). Sie sind im Netz verteilt. Nur einer schafft die Aufgabe als erster, der bekommt die Bitcoins.
- **Nonce**: number only used once

# Literatur

- Sandra Rueß, 2018
- [https://de.wikipedia.org/wiki/Kryptographische\\_Hashfunktion#Klassifizierung](https://de.wikipedia.org/wiki/Kryptographische_Hashfunktion#Klassifizierung)
- Fill, Hans-Georg und Maier, Andreas: Blockchain kompakt, Springer 2020
- <https://www.vpnmentor.com/blog/hash-puzzle-bitcoin/>
- <https://www.youtube.com/watch?v=bBC-nXj3Ng4>

## Links:

- SHA-Maschine: [sha1-online.com](http://sha1-online.com).
- Transaction history: [blockchain.info](https://blockchain.info)

