9. Blockchain / Distributed Ledger Technology

- Einführung
- Anwendungsfall Krypto-Währungen / Bitcoin
  - signierte Transaktionen
  - Blockkette / Ledger
  - Double Spending Anomalie
  - Konsensus-Verfahren: Proof of Work
- Proof of Stake
- Private DLTs
- Smart Contracts
- Blockchain-Anwendungen

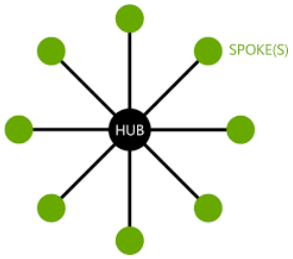
Motivation

- Datenbank mit Transaktionen
  - dauerhafte Speicherung von Daten
  - Fehlersicherheit / Korrektheit bei Systemausfällen
  - Konsistenz / Korrektheit bei konkurrierenden Zugriffen

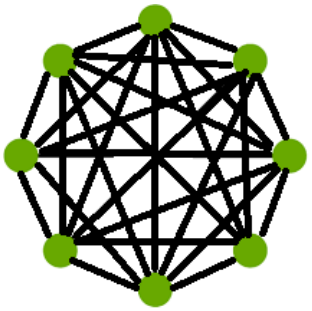
- Analogie Finanzwesen: Bau von Systemen
  - Repräsentation von Eigentümerschaft und -wechsel: Bankkonto, Finanzen, Grundstück, Reisebuchung, Warenlieferung …
  - Nachvollziehbarkeit aller Änderungen (Provenance)
  - Verhindern von doppelten Ausgaben („double spending“): das gleiche Produkt mehrfach verkaufen, das gleiche Geld zweimal ausgeben, …

Mögliche Lösungen

- Lösung 1: zentralisiertes System (Bank) mit Kontrolle über Zustände (Eigentum) und Zustandswechsel
  - Nutzung eines zentralisierten DBS mit Transaktionskontrolle
  - auch bei Einsatz von parallelen/verteilten DBS bleibt zentrale Kontrolle bzw. begrenzte Knotenautonomie (Verteilungstransparenz)



- Lösung 2: Blockchain-Systeme / verteilte Ledger-Systeme
  (*DLT: Distributed Ledger Technology*)
  - „Ledger“ (Buchführungssystem = Datenbank) zur Repräsentation des Zustandes
  - verteilte Kopien mit Append-Only Änderungen
  - globale Identität (Signatur) von Akteuren
  - Transaktion für Zustandsübergänge (Synchronisation gegen Double Spending)

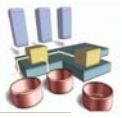


Zielsetzungen

- mit Blockchain/DLT sollen mögliche Probleme zentraler Systeme lösen
- keine Abhängigkeit von „trusted third parties“
  - auch kein Vertrauen gegenüber anderen Teilnehmern erforderlich
- gleichberechtigter Zugriff auf Daten für alle Teilnehmer
- Daten können nicht manipuliert /gelöscht werden
- besserer Schutz gegenüber Angriffen
  - kein Single Point of Failure
- hohe Skalierbarkeit
- Nutzung in zahlreichen Anwendungen

Blockchain: Begriff

- Blockchain = verteiltes System zur Verwaltung von Datensätzen mit dem Ziel, Konsens über den Zustand zu erzielen
- Eigenschaften
  - keine zentrale Instanz
  - Teilnehmer …
    - müssen andere Teilnehmer nicht kennen
    - müssen anderen Teilnehmern nicht vertrauen
    - können sich dennoch über einen Zustand einig sein
- Prinzip
  - Konsens über den initialen Zustand (z.B. leerer Zustand)
  - P2P-Netz aus Teilnehmern (Netzwerkknoten)
  - Transaktionen werden im Netzwerk angezeigt und weitergeleitet
  - Verhindern der Manipulation von Existenz oder Inhalt bereits ausgeführter Transaktionen



Krypto-Währungen

- digitale Zahlungsmittel
- Historie: Bitcoin

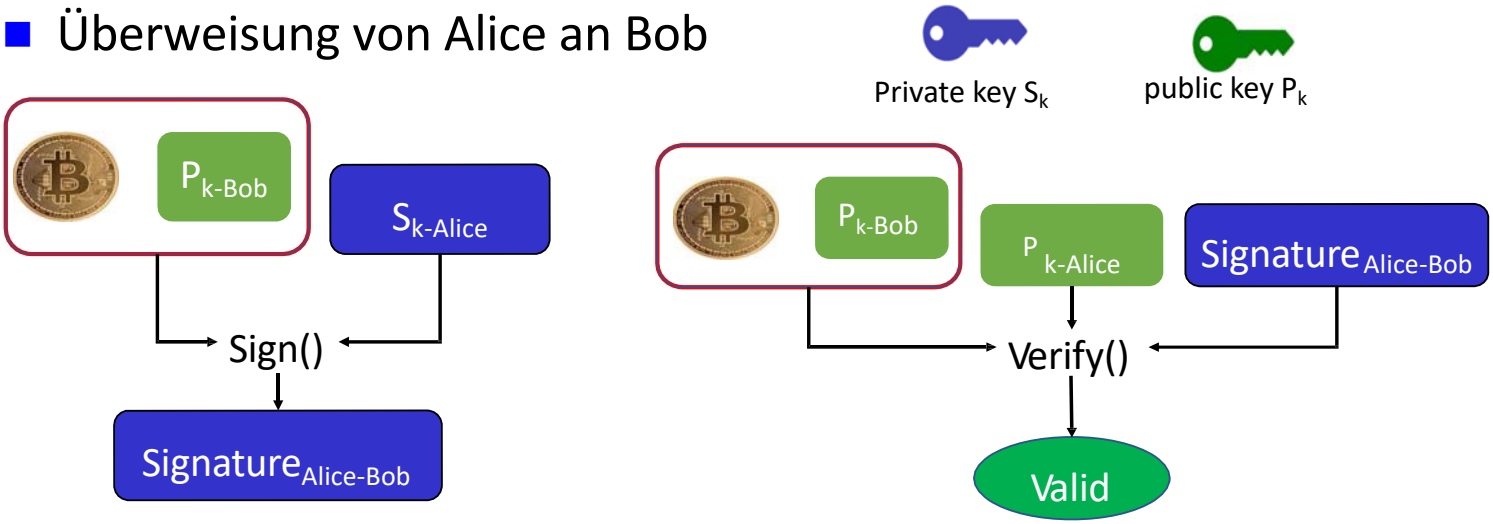
  - 2008: Artikel von „Satoshi Nakamoto" *Bitcoin – A Peer-to-Peer Electronic Cash System* https://bitcoin.org/bitcoin.pdf
  - 2009 Open-Source-Software, eigentlicher Start
  - starke Kursschwankungen
    - all-time high (Nov. 2021): 68,5 TE, Jan. 2022: 37 TE pro BTC (bitcoin)
- aktuell weit über 1000 Währungen
  - 80% aller Initial Coin Offerings mit betrügerischem Hintergrund (Wikipedia)
  - weniger als die Hälfte überlebt ersten vier Monate
  - Akzeptanz erfordert ausreichendes Vertrauen (durch sich gegenseitig kontrollierende Teilnehmer statt Zentralbank bzw. Staat)

Krypto-Währungen: Bausteine

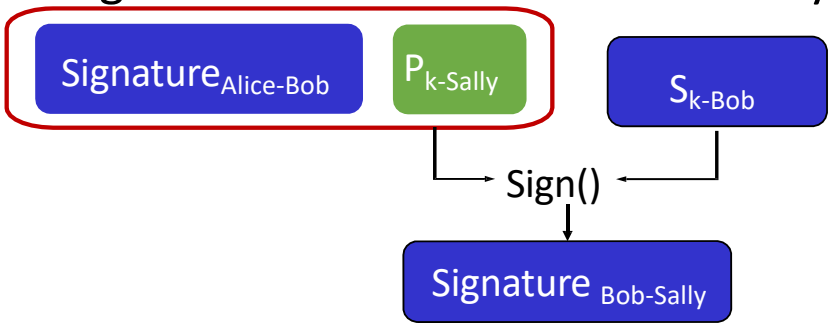
- Vernetzung der Teilnehmer: P2P-Netz statt zentrale Instanz
  - mehr als 15.000 Bitcoin-Knoten nach https://bitnodes.earn.com (>1.700 in D)
- kryptographische Signaturen: Public-Key-Kryptosystem
  - öffentlicher Schlüssel = Kontonummer
  - privater Schlüssel = Verfügungsgewalt über Konto
  - Überweisung: Betrag + öffentlicher Schlüssel des Empfängerkontos, signiert mit privatem Schlüssel des Senders (=Transaktion)
  - Überweisung wird im Netz verteilt und kann von allen überprüft werden
- Buchführung: Transaktionen werden im Ledger voll repliziert auf allen Knoten verwaltet
- Bitcoin-Transaktionen
  - keine expliziten Konten: Guthaben = eingegangene Gutschriften, die noch nicht weiter überwiesen wurden

Digitale Signaturen und Bitcoin

- Überweisung von Alice an Bob



- Weitergabe der Bitcoins von Bob an Sally



https://tinyurl.com/w585fve

Hashing H(x)

- Kombination von Signaturen und Public Keys über Hashing
  - Eingabe: String beliebiger Länge
  - Ausgabe fester Länge (z.B. 256 Bits)
  - effizient berechenbar
- Bitcoin nutzt SHA-256 (**S**ecure **H**ash **A**lgorithm)

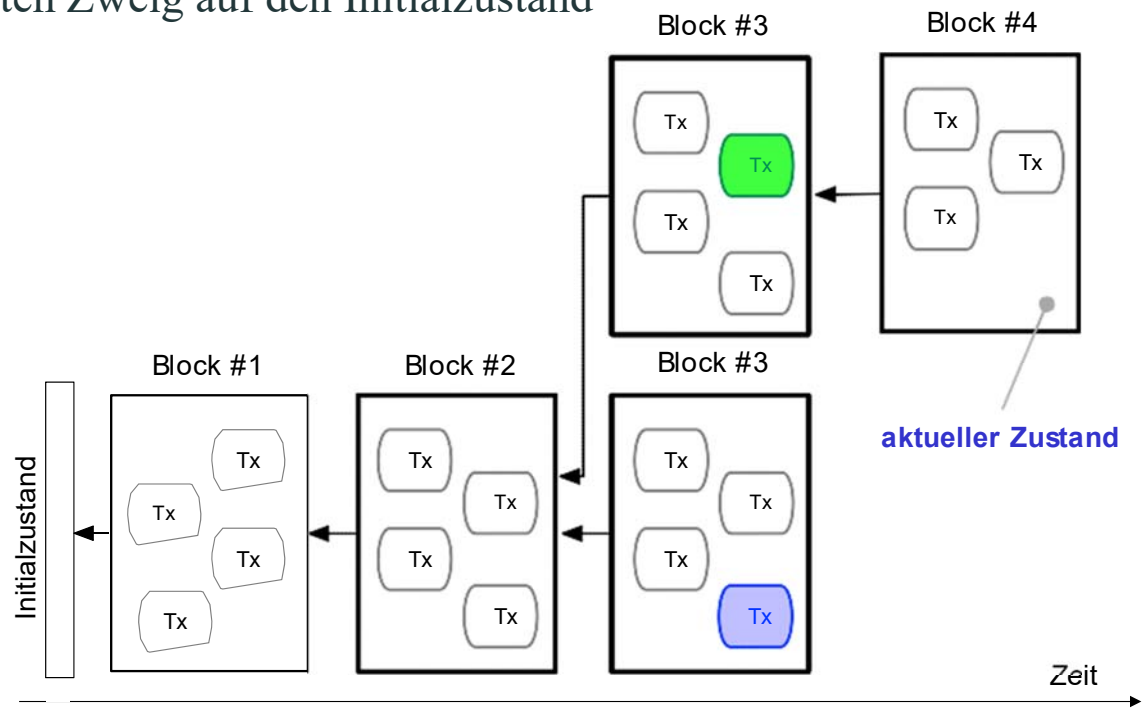
SHA256 ($Signature_{Alice\text{-}Bob}$ || $P_{k\text{-}Sally}$) =

256-Bit (32-Byte) eindeutiger String

- Eigenschaften:
  - *kollisionsfrei*: keine zwei x, y so dass H(x) = H(y)
  - *sicher:* unmöglich x aus H(x) abzuleiten (one-way hash function)

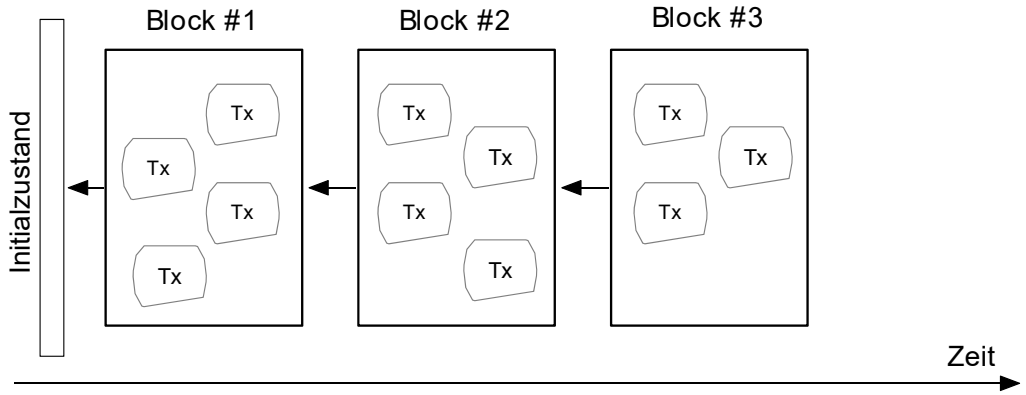
Blockchain-Elemente: Ledger

- Ledger = Blockkette (enthält Transaktions-Log)
  - jeder Knoten im Netzwerk verwaltet eigene Kopie des Ledgers
  - aktueller Zustand = Anwendung aller Transaktionen der Blöcke im längsten Zweig auf den Initialzustand



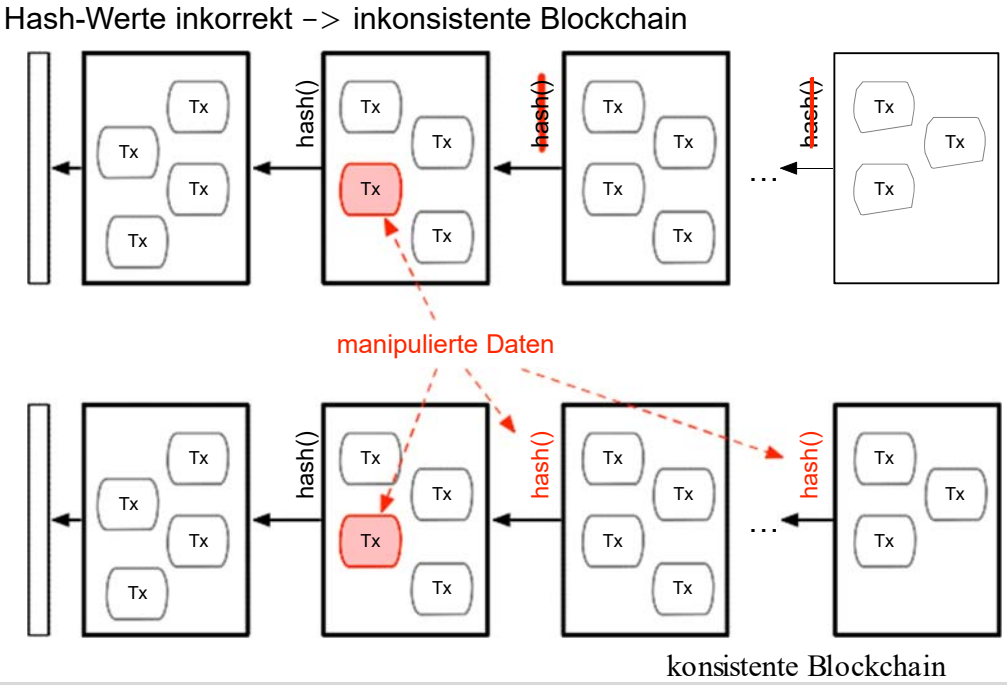
Blockchain-Elemente: Blöcke

- Lösung:
  - Zusammenfassung von Transaktionen zu Blöcken (Block-)
  - Blöcke werden verkettet (-chain), d.h. ein Block basiert auf seinen Vorgänger
    - Block enthält kryptographisch sicheren Hashwert seines Vorgängerblocks
- Blockinhalt: Transaktionsdaten, Zeitstempel, Hash des Vorgängerblocks (unveränderlich)
- jeder Teilnehmer kann jederzeit neuen Block erstellen



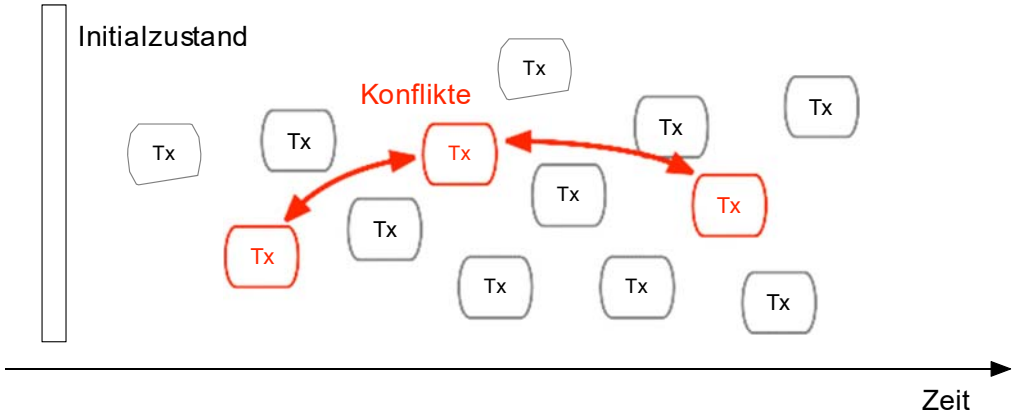
Blockchain: Manipulationssicherheit

- Verkettung der Blöcke durch Hash-Zeiger
  - Manipulation eines Blockinhaltes soll erkannt werden können
  - alleine nicht ausreichend: Ersetzen einer Teilkette durch eine manipulierte Teilkette muss extrem schwer gemacht werden



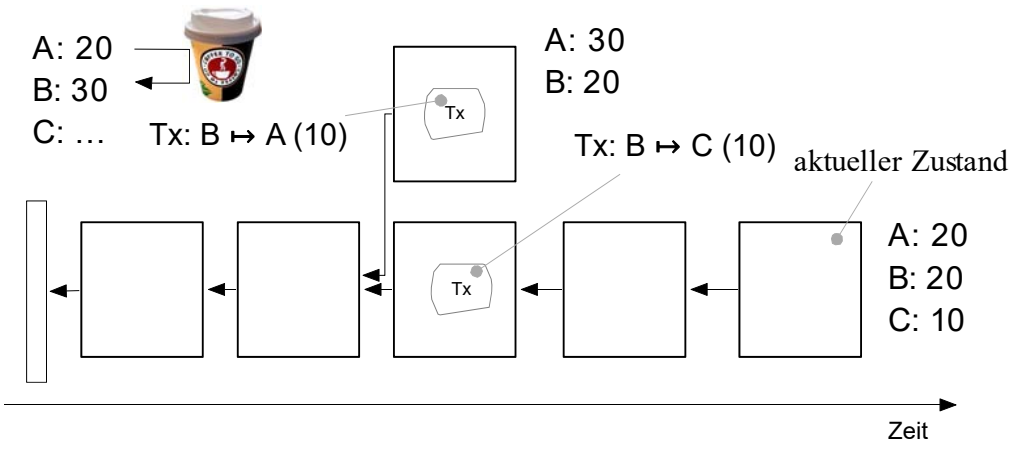
Probleme verteilter Transaktionen

- (kurzzeitig) unterschiedliche Zustände der Knoten (Konsistenzproblem)
- Reihenfolge der Transaktionen
- Versuche doppelter Ausgaben
- Konflikte zwischen Transaktionen bzw. Abhängigkeit von in Konflikt stehenden Transaktionen



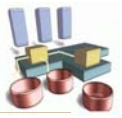
Double Spending in einer Blockchain

- Beispiel
  - B kauft bei A einen Becher Kaffee und zahlt dafür 10 Einheiten
  - B erzeugt weitere Transaktion, in der diese 10 Einheiten an C überwiesen werden sowie schnell hintereinander neue Blöcke
  - B behält Kaffee, A bekommt das Geld nicht



Mining / Konsens-Findung

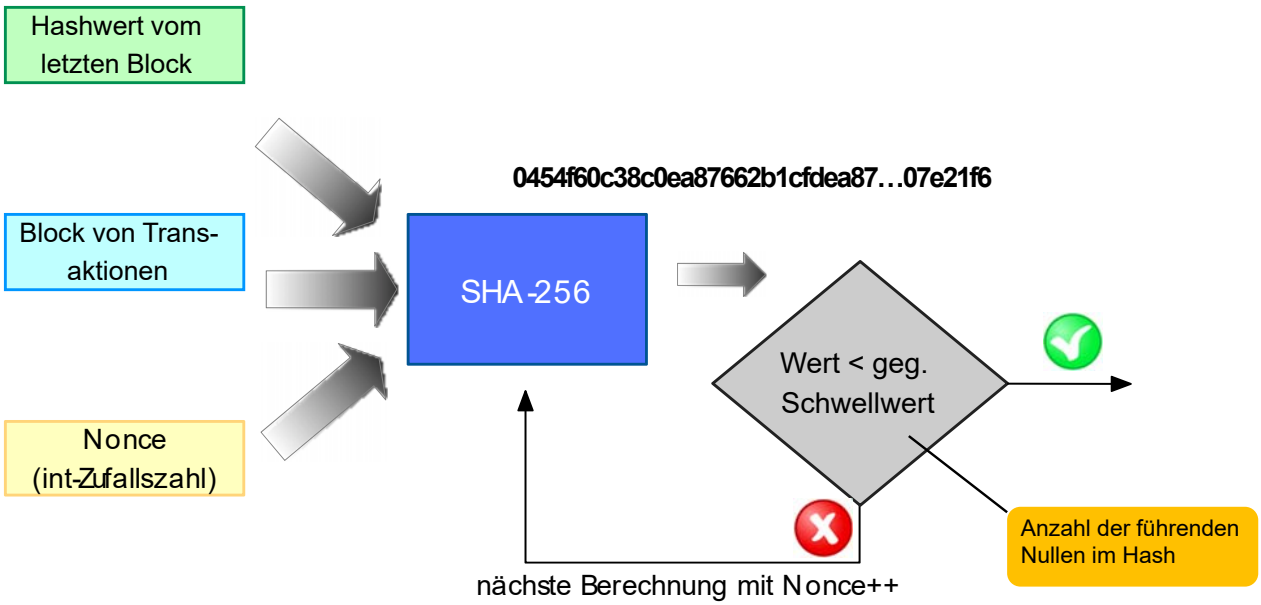
- Miner sind Nodes, die das Konsens-Protokoll ausführen.
  - werden dafür belohnt, da dies Dienstleistung für die Nutzer (Peers) ist
  - Miner empfangen neue Transaktionen von Nutzern und bündeln sie in einem neuen Block. Neue Blöcke werden per Broadcast im Netz verteilt.
  - Da Miner parallel (konkurrent) arbeiten, können gleichzeitig verschiedene neue Blöcke im Netz kursieren. (temporäre Inkonsistenz)
- Eignung bekannter Konsensverfahren wie Paxos …?
  - keine Behandlung byzantinischer Fehler (bösartiges Verhalten von Teilnehmern/Knoten, z.B. Austausch gefälschter Nachrichten)
  - Knoten müssen bekannt und immer verfügbar sein
- anderer Ansatz notwendig ->Proof of Work (PoW)



Proof of Work

- um einen neuen Block zu signieren, muss eine absichtlich sehr rechenaufwändige Aufgabe gelöst werden.
- Großteil des Netzwerks muss an der längsten Block-Kette mitgearbeitet haben →Peers übernehmen die längste Kette.
- PoW Anforderungen:
  - aufwändige Berechnung (nur mit Brute Force) aber einfache und schnelle Validierung
  - muss abhängig vom zu erzeugenden Block sein (Vermeidung von Vorabberechnungsangriffen)
  - variabler Schwierigkeitsgrad
  - Anreizsystem: Mining selbst sollte lohnenswert sein: Belohnungstransaktion (Reward Transaction)
  - Teil des neuen Blocks (*Coinbase* in Bitcoin)

PoW: Hashcash von Bitcoin



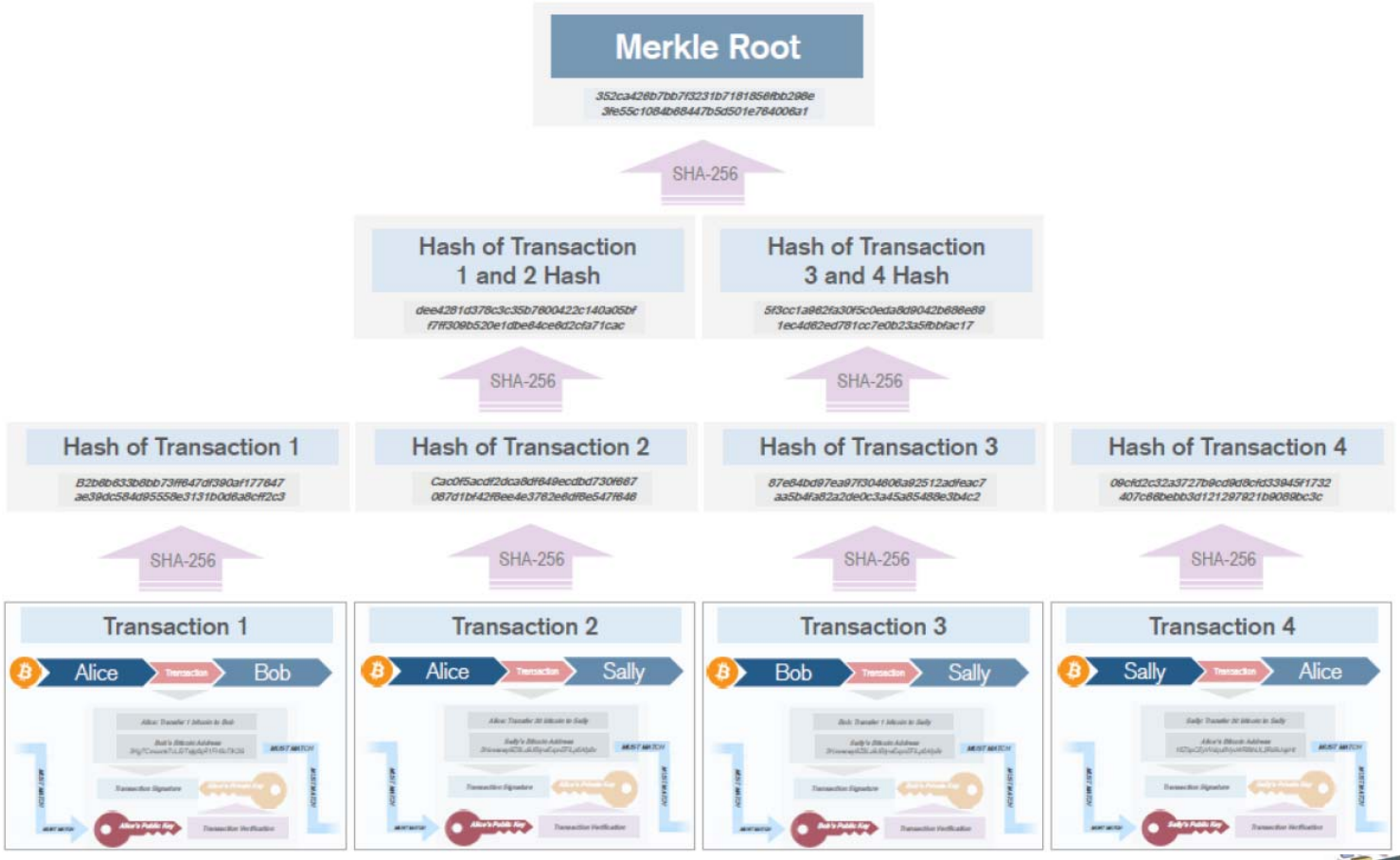
PoW: Ablauf

- wenn Knoten Aufgabe gelöst hat (Mining abgeschlossen):
  - füge Block von Transaktionen der Blockchain hinzu
  - Multicast (Flooding) der Lösung an andere Netzwerkknoten
  - Netzwerkknoten validieren und akzeptieren Lösung
- eingehende Blöcke werden nur akzeptiert, wenn Sie längste Kette korrekt erweitern
- in welchem Zweig der Blockchain sollte ein Miner arbeiten?
  - für Belohnung: Zweig muss Teil des aktuellen Zustands sein (=längster Zweig)
  - keine Koordination notwendig!
  - für von Mehrheit akzeptierte Blöcke erhält Miner geschürfte Bitcoins + Gebühren der enthaltenen Transaktionen

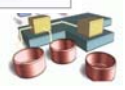
Bitcoin: Transaktionen und Blöcke

- Transaktionen
  - Inhalt: Senderadresse, Empfängeradresse, Betrag, Signatur
  - selbstgewählte Transaktionsgebühren
  - mit privatem Schlüssel des Senders signiert
  - im Netzwerk validiert und verbreitet
- Blöcke
  - feste Größe (z.B. 1 MB)
  - 1. Block = Genesis-Block
  - neue Blöcke werden durch Mining erzeugt (Übernahme von noch unbestätigten Transaktionen aus Mempool)
  - erste Transaktion eines Blocks (coinbase) enthält Überweisung neu erzeugter Bitcoins für Mining + Transaktionsgebühren (Reward)
  - Hashwert = paarweises Hashing der Transaktionen in *Merkle-Baum*, Hashwert des Wurzelknotens (Root-Hash) als Prüfsumme des Blocks
- Mining: PoW wie beschrieben (Nonce-Variation, Flooding)

Merkle-Baum (Bsp.)

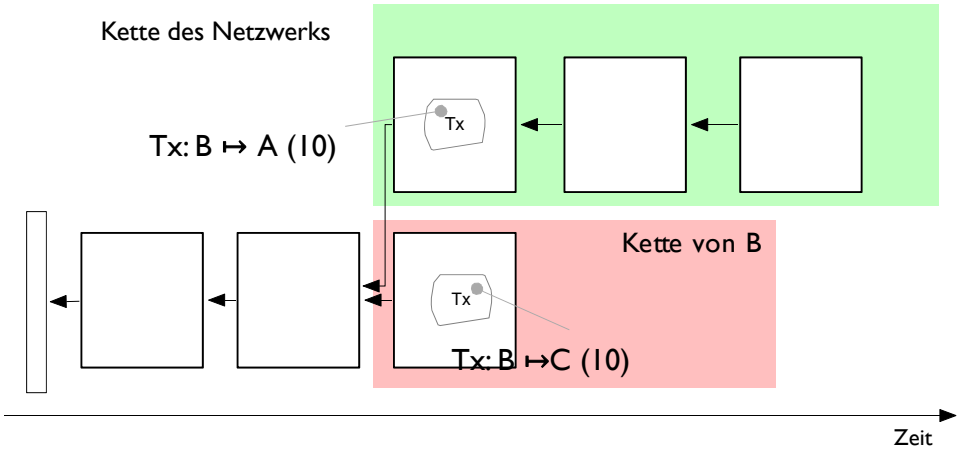


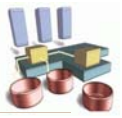
https://www.p-ic.at/uploads/simplex/images/BMVIT Dossier Blockchain 2017 FINAL.pdf



Double Spending bei PoW

- Beispiel:
  - B erzeugt wieder 2 Transaktionen in verschiedenen Blöcken, nur die erfolgreich validierte wird veröffentlicht
  - Zuordnung weiterer Transaktionen zum Originalblock, nur B arbeitet auf seinem Block
  - Netzwerk hat mehr Rechenleistung und kann schneller neue Blöcke erzeugen -> Kette von B bleibt kürzer

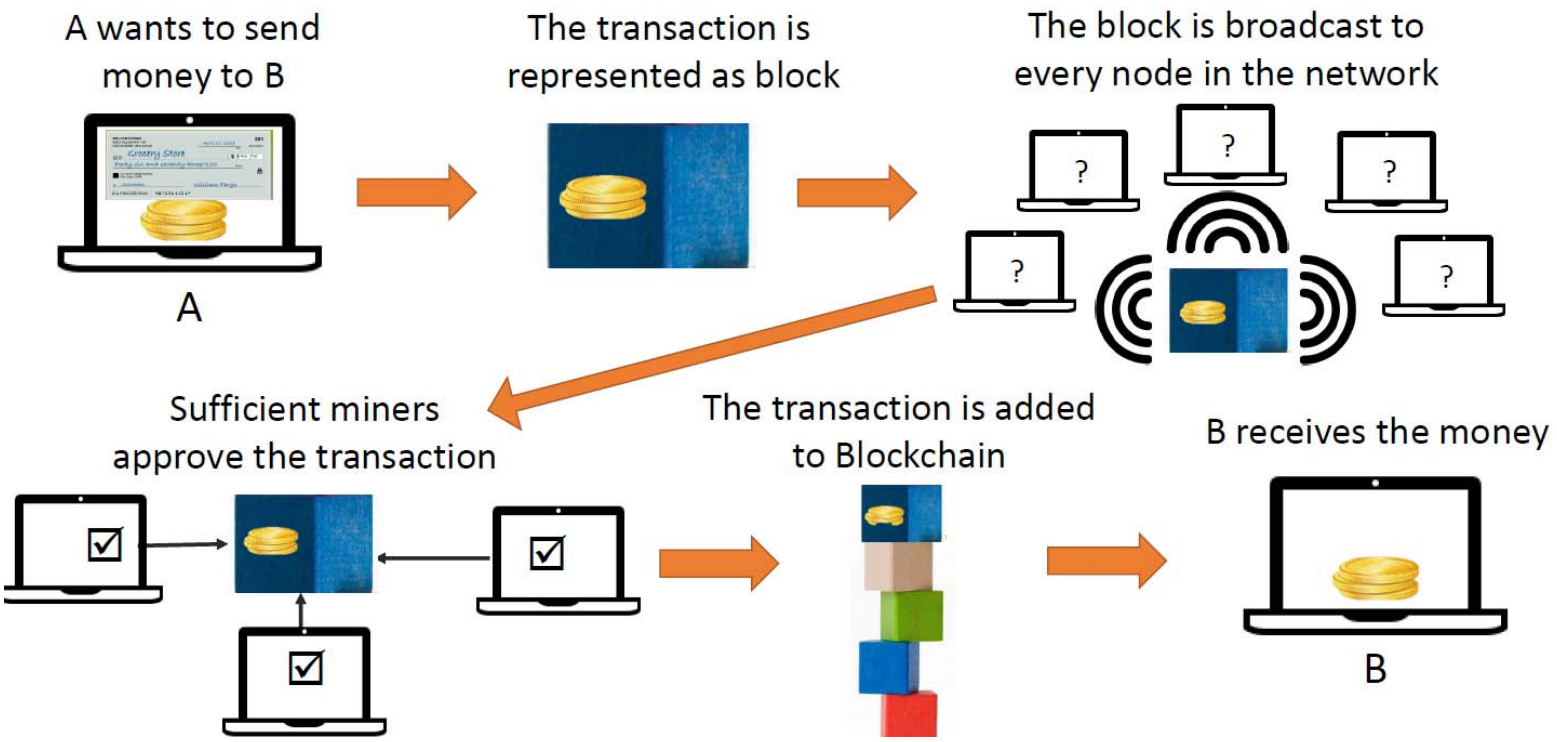




Netzwerkangriffe: 51%-Angriff

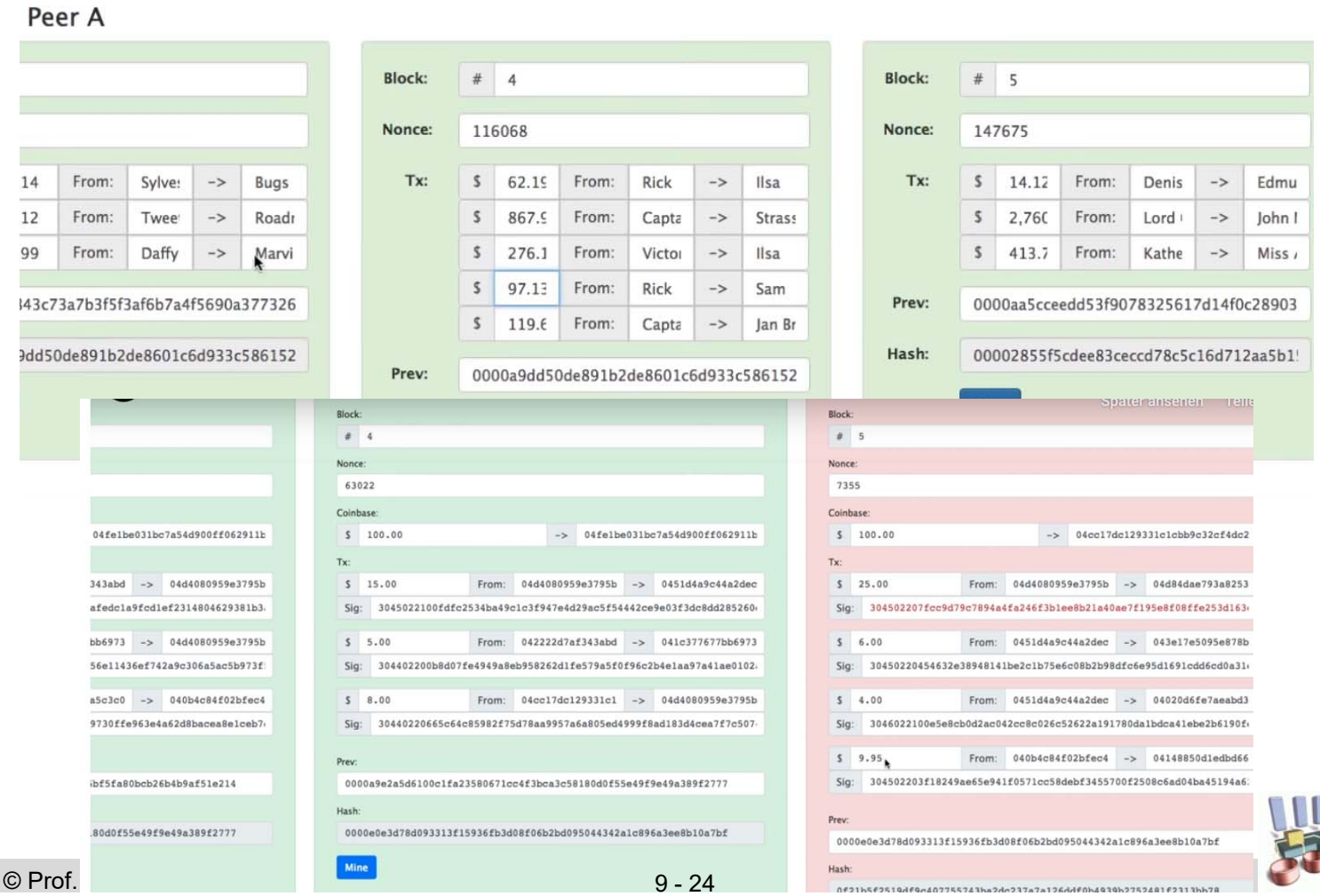
- Mehrheitsangriff: Angreifer kontrolliert über die Hälfte der Rechenleistung des Netzwerks
  - ermöglicht Double Spends, Rückgängigmachen von Transaktionen
  - Prinzip: eigene Blöcke schneller anlegen als der Rest des Netzes und nachträglich gültig machen
- nach Wikipedia:
  - 2014: Mining Pool GHash überschreitet kurzzeitig 50%-Marke
  - Attacken auf Bitcoin Gold (2018) und Ethereum Classic (2019)
- Gegenmaßnahmen
  - 51%-Angriff ist ein auffälliges Ereignis -> z.B. Hard Forks in Bitcoin
    - ersten Block einer verdächtigen Kette ungültig erklären
  - eingebaute Anreizmechanismen:
    - hohe Miningkosten im Falle einer Abwehr verloren
    - Senden anderer Transaktionen kann nicht verhindert werden

Gesamtablauf



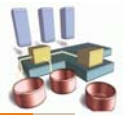
© Mooly Sagiv

Demo https://andersbrownworth.com/blockchain/



Bitcoin: Fakten (Wikipedia)

- Größe Blockchain 310 GB (Nov 2020)
  - seit 2020: 6,25 neu erzeugte Bitcoins pro Block; halbiert sich alle 4 Jahre
  - Transaktionskosten: 1.000 Satoshi (= 10 µBTC)
- max. 7 Transaktionen pro Sekunde (schlechte Skalierbarkeit)
- extremer Ressourcen/Energie-Bedarf
  - Schätzung: 120 Terawattstunden pro Jahr (0,5 % des Weltenergiebedarfs von 21 Billionen KWh)
  - pro Transaktion: 1200 kWh (2021); Kreditkartentransaktion: 1,5 Wh
  - 2017: ca. 75% aller Bitcoins in China geschürft (2017) – durch Kohlestrom aus der Inneren Mongolei; dabei CO2-Ausstoß von 8 bis 13 Tonnen pro Bitcoin



Proof of Stake

- Alternative zu ineffizientem Proof of Work (PoW) in öffentlichen Blockchains
- Proof of Stake
  - Miner hinterlegen einen großen Betrag an Krypto-Tokens in einem Smart Contract.
  - falls ihnen böswilliges Verhalten nachgewiesen werden kann, so verlieren sie ihr Guthaben.
  - welcher der Miner einen neuen Block erstellen darf, wird immer wieder zufällig bestimmt (Miner-Gewicht nach Vermögen und Teilnahmedauer)
  - 99% weniger Energieverbrauch als PoW
- Krypto-Währung Ethereum wird 2022 auf PoS umstellen.

Blockchain/DLT-Typen

- öffentliche vs. private Blockchains
- öffentliche Blockchains (z.B. für Kryptowährungen)
  - lassen beliebige Teilnehmer zu
  - maximale Transparenz, gesamter Ledger öffentlich
  - Pseudonymität der Nutzer suggeriert Privacy, aber oft Tracking-Angriffe möglich
  - sehr hoher Ressourcenbedarf durch Proof of Work
- private Blockchains (z.B. für Unternehmensanwendungen)
  - durch Eigentümer oder Konsortium kontrollierter Teilnehmerkreis (*permissioned* blockchains)
  - effizientere und kostengünstigere Realisierung von Transaktionen
  - typische Anwendung: Prozesse zwischen großen Organisationen abbilden, z.B für Lieferketten (supply chains)
  - Beispielrealisierung: Hyperledger (www.hyperledger.org)

Warum private Blockchains (DLTs)?

- Unternehmen wollen ihre Daten oft nicht veröffentlichen, sondern den Zugriff kontrollieren
- gemeinsame Geschäftsdaten können nicht einseitig verändert werden und werden doch automatisch synchronisiert
- Unveränderlichkeit vergangener Transaktionen bietet Grundlage für Audits
- Unterstützung komplexer Prozessmodelle
- manche Teilnehmer können besondere Rechte haben, z.B. der Betreiber einer DLT-basierten Plattform

Konsens in privaten DLTs

- Netzwerkteilnehmer kennen einander, dadurch Vertrauen. (im Betrugsfall klagen sie nicht gegen Unbekannt)
- deswegen Tradeoff möglich:
  - weniger Sicherheit gegen böswilliges Verhalten.
  - aber bessere Performance, da kein teures Proof of Work
- Beispiel RAFT-Konsensalgorithmus
  - Verwendet von Hyperledger Fabric.
  - Miner heißen Orderer
  - Orderer wählen einen Anführer und replizieren seine Entscheidung.
  - Fällt der Anführer aus, wird ein neuer gewählt
  - jede an einem Fabric-Netz teilnehmende Organisation betreibt Orderer und sorgt für seine Integrität.

Smart Contracts

- wichtige Nutzungsform in öffentlichen und privaten DLTs
  - z.B. in Etherum (programmiert mit Solidity)
- Protokoll zur Repräsentation, Verifikation und Ausführung von Verträgen
  - Programme werden in Blockchain gespeichert und bei Eintreten bestimmten Bedingungen automatisch ausgeführt (Änderung der Daten in Blockchain)
  - ermöglicht Spezifikation von Prozessen zwischen mehreren Organisationen, die nicht mehr einseitig geändert werden können
- Beispiel:

DLT: Anwendungen

- Krypto-Währungen (Bitcoin, Ether etc.)
- viele dezentrale, mit Smart Contracts realisierte, Anwendungen
- E-Voting-Systeme
- virtuelle Organisationen
- Crowdfunding
- Auditing: Aufzeichnung sicherheitskritischer Operationen
  - Zugriff auf bzw. Veränderung von Ressourcen (z.B. Daten, Dokumente)
  - Zugriff auf Gesundheitsakten, …
- dezentrale Energieversorgung und –Abrechnung
- Lieferketten: Dokumentation der Teilschritte

  …
- generell: Regeln eines gemeinsamen Prozesses automatisch Durchsetzen ohne auf zentrale Instanz zu vertrauen

Anwendungsbeispiel: MedRec

- Projekt des MIT
- System zur Verwaltung von medizinischen Daten
  - vor allem Patientendaten: MRT-Bilder, Befunde, Rezepte ...
  - Eigentümerschaft und Zugriffsrechte in privater Blockchain
  - tatsächliche Daten in herkömmlichen Datenbanken von Gesundheitsdienstleistern
  - zusätzliche öffentliche Blockchain (Ethereum) enthält:
    - pseudonimisierte Beziehungen zwischen Patienten und Gesundheitsdienstleistern
    - Pointer auf Datensätze in den externen Datenbanken
    - Hashes der Datensätze für garantierte Datenintegrität
- Vorteil: bei Besuch eines neuen Arztes kann der Patient diesem Zugriff auf seine Bestandsdaten geben.
- Ziel: Zulassung von Data Mining auf anonymisierten Patientendaten (noch offene Forschungsfragen)

Zusammenfassung

- Blockchains/Distributed Ledgers: neues Paradigma für verteilte Daten- und Transaktionsverwaltung
- Popularität durch Kryptowährungen wie Bitcoin & Ether
- Trend: private Blockchains für Unternehmensanwendungen mit besserer Leistungsfähigkeit und geringerem Ressourcenbedarf
- wesentliche Vorteile:
  - gleichberechtigte Datennutzung, keine Abhängigkeit von zentralen Institutionen, keine Veränderung bereits erfolgter Transaktionen …
- technische Realisierung
  - Blockbildung und Verkettung durch Hashing verschlüsselter und signierter Transaktionen
  - vollständige Replikation der Blockchain
  - Validierung durch Mining und Konsensbildung
  - Smart Contracts: neue Transaktionen nur gemäß vordefinierter Regeln