Web-basierte Systeme

13: Web3

Wintersemester 2026

Rüdiger Kapitza

Lehrstuhl für Informatik 4
Systemsoftware

Friedrich-Alexander-Universität
Technische Fakultät

Vorläufiger Vorlesungsplan

16. Oktober	Einführung und Darstellung von Webseiten
22. Oktober	HTML und CSS
29. Oktober	Hypertext Transfer Protocol
5. November	
12. November	Browser Schnittstellen
19. November	Kommunikationsschnittstellen im Browser
26. November	WebAssembly
3. Dezember	Architektur moderner Browser
10. Dezember	Clientseitige Architekturmuster
17. Dezember	Serverseitige Implementierung von Web-basierten Systemen Vorbereitung Papieranalyse
7. Januar	Lastverteilung durch Zwischenspeicher
14. Januar	Papieranalyse
21. Januar	Aspekte von Web Sicherheit
28. Januar	**Web3**
5. Februar	Zusammenfassung und Ausblick

Zielsetzung der Lerneinheit

- Idee und erstes Verständnis von Blockchains und Web3
- Einblick in die Architektur des Internet-Computers (IC) als Beispiel für eine Web3-Infrastruktur
- Beispiel für die Programmierung eines Canisters (Smart Contracts des IC)

Blockchain/Web3

Web3

Warum der (*ursprüngliche*) Hype um Blockchains?

- Automatisiertes Vertrauen
- Autorität durch Technologie ersetzen
- Vermittler ausschalten
- Erhöhte Transparenz
- Neue Geschäftsmodelle

Bitcoin

Bitcoin

- Erste Kryptowährung
- Prägte das Konzept der Blockchain
- Dezentral, trustless, anonym
- Nicht unter Kontrolle einer einzelnen Entität (Satoshi Nakamoto?)
- Wurzeln in der „Cypherpunks“-Bewegung von 1990–1995

Status quo

Interagierende Märkte

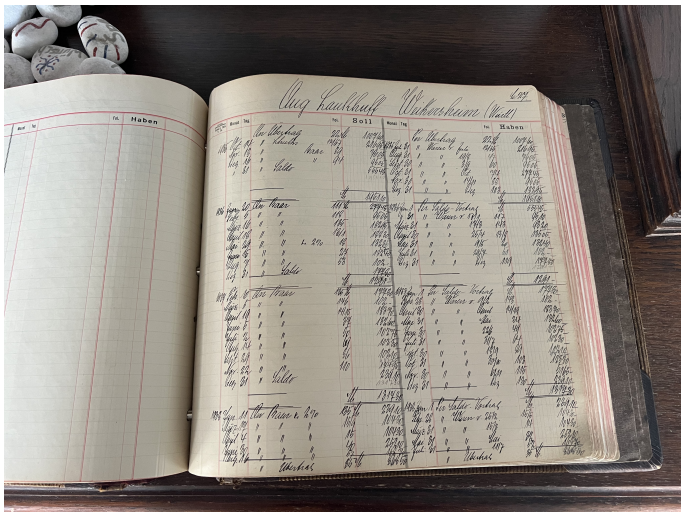
- Netzwerke verbinden Teilnehmer
  - Bspw.: Kunden, Lieferanten, Banken oder Verbraucher
- Märkte organisieren den Handel
  - öffentliche und private Märkte
- Vermögen wird durch den Handel von Vermögenswerten und Dienstleistungen zwischen den Teilnehmern generiert
  - physische Gegenstände (Haus, Auto ...), virtuelle Vermögenswerte (Anleihen, Patente ...), Dienstleistungen
- Transaktionen tauschen Vermögenswerte

Ledger

Ledger bzw. Hauptbuch

- Hauptbuch erfasst alle Geschäftsaktivitäten als Transaktionen
  - Datenbank
- Hauptbuch erfasst die Übertragung von Vermögenswerten zwischen den Teilnehmern
- Jeder Markt und jedes Netzwerk besitzt ein eigenes

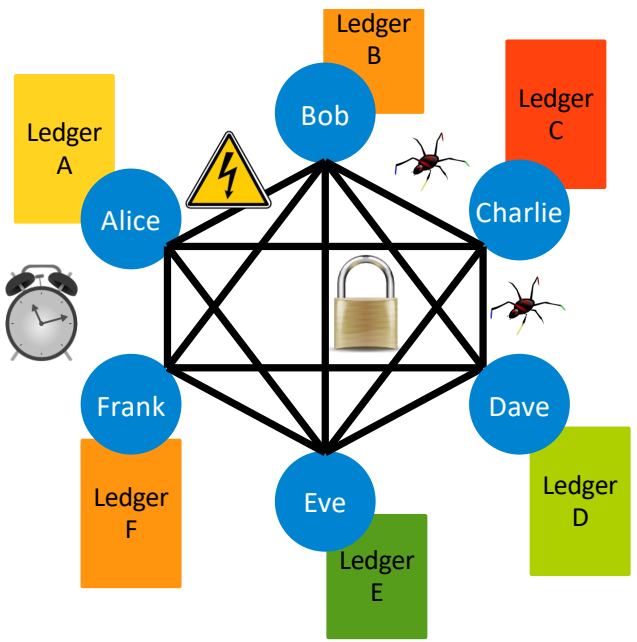
- Problem: (Zu) viele Hauptbücher
  - Jeder Markt hat sein Hauptbuch
  - Jede Organisation hat ihr eigenes Hauptbuch



Ledger

Warum können unabhängige Hauptbücher ein Problem darstellen?

- Jede Partei führt ihr eigenes Hauptbuch
- Probleme, Vorfälle, Fehler $\rightarrow$ Hauptbücher weichen häufig voneinander ab
- Abstimmung und Bereinigung von Inkonsistenzen ist teuer

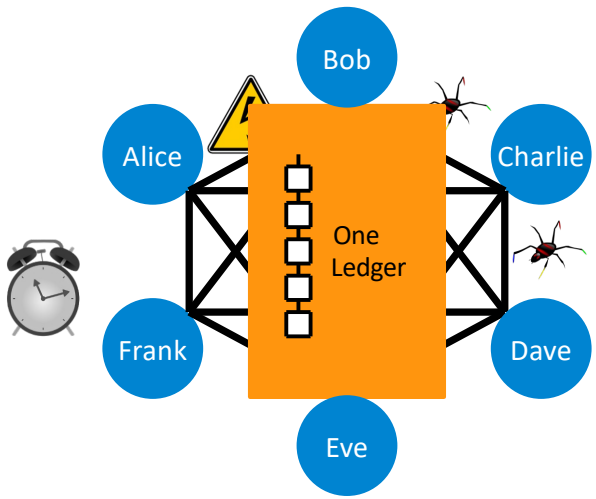


Ledger

Blockchain repräsentiert *ein* virtuelles Hauptbuch

- Virtuelles **vertrauenswürdiges** Hauptbuch
- Agiert als zentraler Vermittler
- Blockchain erfasst den globalen Zustand für alle
- Repliziert und durch Konsens erzeugt/fortgeschrieben

- Vertrauen in das Hauptbuch ist gegeben durch
  - Kryptografische Verfahren
  - Verteilte Validierung



Blockchain

Vier Säulen charakterisieren eine Blockchain

- **Repliziertes Hauptbuch**
  - Historie enthält alle Transaktionen
  - Historie ist unveränderlich
  - Verteilt und repliziert
- **Konsens**
  - Dezentrales Protokoll
  - Gemeinsame Kontrolle, die Störungen toleriert
  - Transaktionen werden validiert
- **Kryptografie**
  - Integrität des Hauptbuchs
  - Authentizität von Transaktionen
  - Vertraulichkeit von Transaktionen
  - Identität der Teilnehmer
- **Geschäftslogik**
  - In das Hauptbuch eingebettete Logik (einfach oder komplex)
  - Wird zusammen mit Transaktionen ausgeführt

Blockchain vereinfacht komplexe Transaktionen

Beispiele für Anwendungsfelder

- Verwaltung von finanziellen Vermögenswerte
  - Schnellere Abwicklungszeiten
  - Erhöhte Kreditverfügbarkeit
  - Transparenz und Überprüfbarkeit
  - Keine Abgleichungskosten
- Verwaltung von Grundbuchdaten und Immobilien
  - Digital, aber fälschungssicher
  - Weniger Streitigkeiten
  - Transparenz und Überprüfbarkeit
  - Geringere Transfergebühren bei Verkauf
- Logistik
  - Echtzeit-Transparenz
  - Verbesserte Effizienz
  - Transparenz und Überprüfbarkeit
  - Geringere Kosten

Merkmale eines sinnvollen Blockchain-Szenarios

- Aufgabe/Problem aber keine (zentrale) vertrauenswürdige Partei verfügbar um sie zu lösen
- Protokoll zwischen mehreren Knoten um eine verteilte Aufgabe zu lösen
  - Knoten erzielen gemeinsam Konsens
- Schlüsselaspekte des Szenarios
  - Daten werden verarbeitet und gespeichert
  - Mehrere Knoten schreiben
  - Nicht alle schreibenden Knoten sind vertrauenswürdig
  - Vorgänge sind (einigermaßen) überprüfbar
- Wenn alle schreibenden Knoten bekannt sind $\rightarrow$ *permissioned* oder Konsortium-Blockchain
- Andernfalls sind die schreibenden Knoten nicht bekannt $\rightarrow$ *permissionless* oder öffentliche Blockchain

Warum gerade jetzt Blockchain?

- Kryptografie ist seit Jahrzehnten eine Schlüsseltechnologie in der Finanzwelt
  - Zahlungsnetzwerke, Geldautomatensicherheit, Smartcards, Online-Banking ...
- Vertrauensmodell des Finanzgeschäfts hat sich nicht geändert
- Für den Austausch zwischen nicht vertrauenswürdigen Partnern wird ein vertrauenswürdiger Vermittler benötigt
- Heute sichert Kryptografie hauptsächlich 1:1 Interaktionen
- Bitcoin entstand 2009
  - Verkörpert nur die Kryptografie der 1990er Jahre und früher
  - Erste prominente Verwendung der Kryptografie für ein neues Vertrauensmodell (= Vertrauen in keine Entität)
- Das Versprechen von Blockchain – Vertrauen reduzieren und durch Technologie ersetzen
  - Fortschrittliche kryptografische Techniken nutzen

Was ist eine Blockchain?

Zustandsmaschine

- Geben eine Funktionalität F
  - Operation o wandelt einen Zustand s in einen neuen Zustand s' um und kann eine Antwort r erzeugen
  - $(s', r) \leftarrow F(s, o)$
- Gültigkeitsbedingung
  - Die Operation muss gültig sein, basierende auf dem aktuellen Zustand, gemäß einem Prädikat $P()$
  - $P(s, o) = TRUE$

Was ist eine Blockchain?

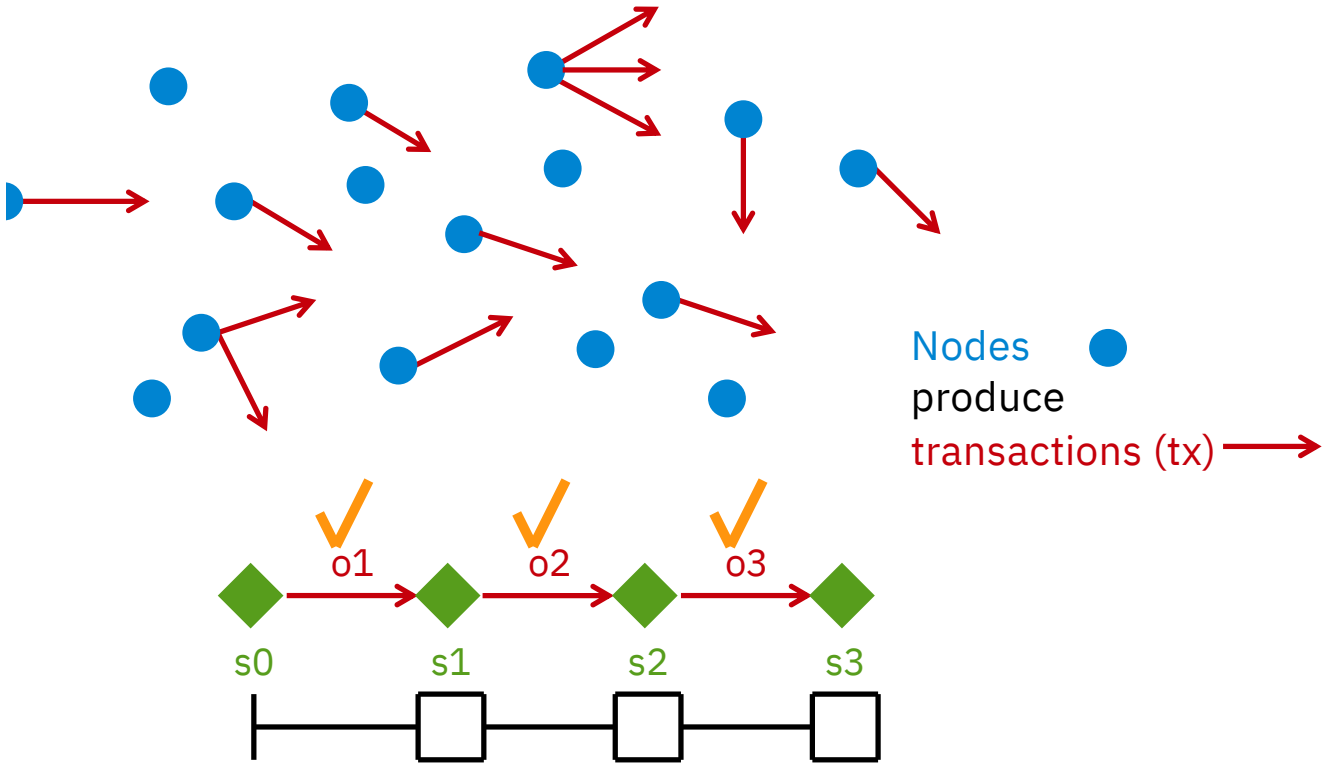
Zustandsmaschine

- Die Blockchain kann nur ergänzt werden
- Bei jedem Vorgang wird ein „Block“ gültiger Transaktionen (tx) der Blockchain angehängt
  - Inhalt einer Blockchain kann lückenlos überprüft werden
  - Log-Einträge bilden eine Hash-Kette
  - $h_t \leftarrow hash([tx_1, tx_2, ...]||h_{t-1}||t)$

Beispiel – die „Bitcoin“-Zustandsmaschine

- Bitcoins sind fälschungssichere Bitstrings
  - „Mined“ bzw. erzeugt durch das Protokoll
- Signaturschlüssel (ECDSA) besitzen und übertragen Bitcoins
  - Eigentümer sind pseudonym, z. B. `3JDs4hAZeKE7vER2YvmH4yTMDEfoA1trnC`
- Jede Transaktion überträgt einen Bitcoin (Bruchteil) vom aktuellen zum nächsten Eigentümer
  - „Dieser Bitcoin gehört jetzt 3JDs …“ – signiert mit dem Schlüssel des aktuellen Eigentümers
  - Der Münzfluss ist durch das Design verknüpfbar und nicht anonym, wenn er mit der realen Welt verbunden ist
- Die Validierung basiert auf der globalen Historie vergangener Transaktionen
  - Der Unterzeichner hat den Bitcoin zuvor erhalten
  - Der Unterzeichner hat den Bitcoin noch nicht ausgegeben

Konsensprotokoll erzeugt die Blockchain



- Das Konsensprotokoll ordnet die Transaktionen und fügt sie in das Hauptbuch ein

Merkmale eines Blockchain-Protokolls

- Nur „gültige“ Vorgänge (Transaktionen) werden „ausgeführt“
- Transaktionen können einfach sein
  - Bitcoin-Transaktionen sind eine Eigentumserklärung für Coins, digital signiert „Dieser Bitcoin gehört jetzt **K2**“, signiert von **K1**
- Transaktionen können aus beliebigem Code bestehen (intelligente Verträge)
  - Verkörperung von Logik, die auf Ereignisse (in der Blockchain) reagiert und als Reaktion darauf bspw. Vermögenswerte überträgt
  - Auktionen, Wahlen, Investitionsentscheidungen, Erpressung …

Wie kann man eine Einigung erzielen?

- Demokratie – Wählen Sie! …aber wer zählt die Stimmen?
- Knoten können nur Nachrichten untereinander austauschen
- Kein gemeinsames Wissen
- Wie kann man trotz Fehlern, Verzögerungen, Netzwerkangriffen und Betrug einen Konsens erzielen?
  - Zentrale Frage verteilter Algorithmen bzw. Protokolle

Wie kann man trotz Fehlern einen Konsens erzielen?

Ausfalltolerante Einigung

- Demokratie – Wählen Sie!
- Die Mehrheit hat von Natur aus Recht
  - Vorausgesetzt, die Wähler haben eine Identität
- Quorumsystem
  - N Knoten, F Fehler $\rightarrow N > 2 * F$
- Netzwerk nicht zuverlässig und asynchron
  - Warten Sie nicht auf die letzten F Stimmen
- Toleriert fehlerhafte (abgestürzte) Knoten

Wie kann man trotz Angriffen einen Konsens erzielen?

Byzantinische Fehlertoleranz

- Demokratie – Mit unehrlichen Wählern!
- Qualifizierte 2/3-Mehrheit erforderlich
- Byzantinisches Quorum-System
  - N Knoten, F Fehler $\rightarrow N > 3 * F$
- Netzwerk nicht zuverlässig und asynchron
  - Nicht auf letzte F Stimmen warten
- Toleriert böswillige oder „byzantinische“ Knoten

Arten der Einigung in Blockchains

Vorgehensweise bei der Einigung hängt vom Systemmodel ab

- Dezentralisiert / permissionless / Nakamoto-Konsens
  - Bspw. Bitcoin, Ethereum, ...
- Konsortium / mit Genehmigung / BFT-Konsens
  - BFT-Konsens (Byzantinische Fehlertoleranz), Quoren
  - Flexible Quoren: Ripple und Stellar
- Es gibt noch weitere Klassen auf anderen Ebene bspw. proof-of-stake

Alternative Sicht: Was ist eine Blockchain?

Idealisierte Abstraktion einer (Turing-vollständigen) Blockchain

- Weltweit verteilter Computer
- Es gibt keinen zentralen Besitzer oder Betreiber
- Jeder kann Anwendungen betreiben und nutzen
- Unterstützt den Besitz und die Vermarktung von Daten
- Setzt Eigentumsrechte durch

Evolution zu Web3

- Möglichst nahtlose Interaktion zwischen dem Web und Web3
  - Smart Contracts können HTTP-Inhalte bereitstellen und ermöglichen neue APIs/Webdiensten anzubieten
- Interoperable Anwendungen
  - Bereitstellung von Anwendungen, die direkt mit externen Netzwerken und Diensten kommunizieren können
- Datenspeicherung
  - Gesamter Anwendungen und ihrer Daten
- Dezentrale Finanzmanagement
  - Automatisierte Konvertierung von digitalen Währungen
  - Aufbau von dezentrale Börsen

Internet Computer

Internet Computer als Beispiel für Web3

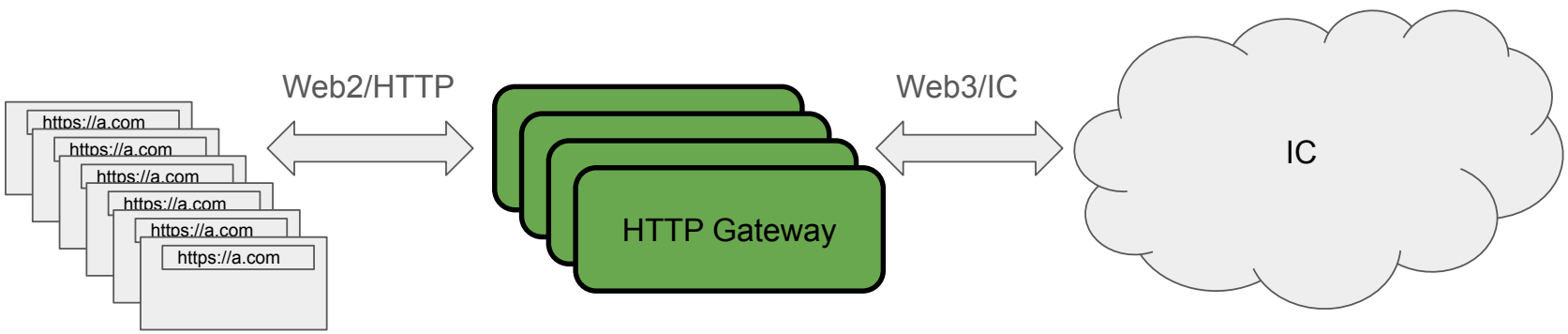
Überblick zu zentralen Elementen der Infrastruktur

- **Internet Computer Übersicht**
- Abortable Broadcast
- Internet Computer Consensus
- Internet Computer State Machine Replication

Verbindung zwischen Web und IC

Protokollumsetzung

- Damit Browser transparent mit dem IC kommunizieren können, muss eine Protokollkonvertierung stattfinden
- Wird über HTTP-Gateways erreicht, hat aber Sicherheitsimplikationen



Programmierung eines Smart Contracts

- Interaktive Programmierung im Browser mittels ICP.Ninja