(19) Deutsches Patent- und Markenamt

(10) **DE 11 2021 004 613 T5** 2023.07.13

(12) # Veröffentlichung

der internationalen Anmeldung mit der
(87) Veröffentlichungs-Nr.: **WO 2022/043778**
in der deutschen Übersetzung (Art. III § 8 Abs. 2 IntPatÜbkG)
(21) Deutsches Aktenzeichen: **11 2021 004 613.6**
(86) PCT-Aktenzeichen: **PCT/IB2021/056022**
(86) PCT-Anmeldetag: **06.07.2021**
(87) PCT-Veröffentlichungstag: **03.03.2022**
(43) Veröffentlichungstag der PCT Anmeldung in deutscher Übersetzung: **13.07.2023**

(51) Int Cl.: ***G06F 21/60*** (2013.01)
H04L 9/32 (2006.01)
H04L 9/06 (2006.01)

(30) Unionspriorität:
17/007,677 **31.08.2020** **US**

(71) Anmelder:
International Business Machines Corporation, Armonk, NY, US

(74) Vertreter:
LifeTech IP Spies & Behrndt Patentanwälte PartG mbB, 80687 München, DE

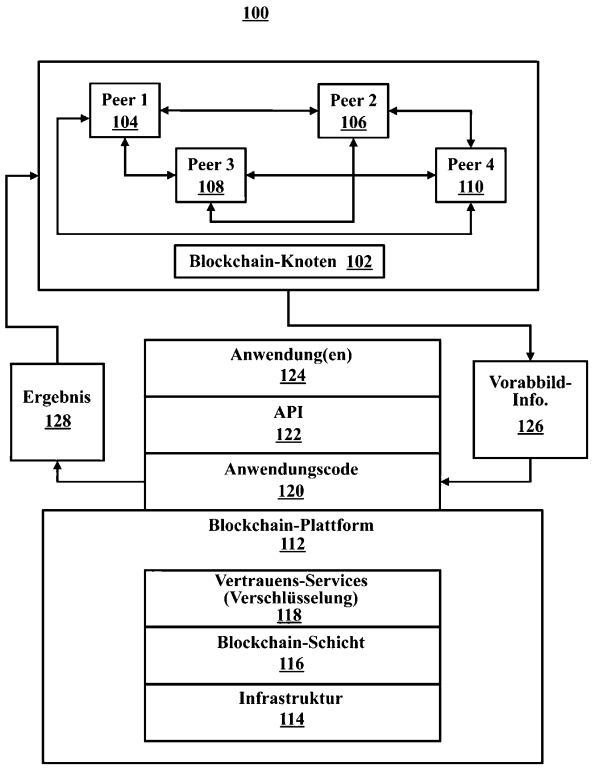
(72) Erfinder:
Manevich, Yacov, Haifa, IL; Barger, Artem, Haifa, IL; Meir, Hagar, Haifa, IL

Prüfungsantrag gemäß § 44 PatG ist gestellt.

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen.

(54) Bezeichnung: **REDIGIERBARE BLOCKCHAIN**

(57) Zusammenfassung: Bei einigen Ausführungsformen kann ein Prozessor einen Wert und einen Hash des Wertes in einem Ledger speichern, das einem Blockchain-Netzwerk zugehörig ist. Der Prozessor kann aus dem Hash des Wertes einen Block-Hash aufbauen. Der Prozessor kann eine Transaktion durch ein Erkennen validieren, dass der Hash des Wertes mit dem Hash des Wertes aus einer früheren Transaktion übereinstimmt. Der Prozessor kann eine Unversehrt des Block-Hash aufrechterhalten.
Bei einigen Ausführungsformen kann ein Prozessor eine Redigiertransaktion für ein Blockchain-Netzwerk bereitstellen. Der Prozessor kann die Redigiertransaktion einleiten. Die Redigiertransaktion kann einer validierten Transaktion zugehörig sein. Der Prozessor kann einen Hash-Wert erkennen, der innerhalb der Redigiertransaktion angegeben und der validierten Transaktion zugehörig ist. Der Prozessor kann einen Wert redigieren, der dem Hash-Wert zugehörig ist.



Beschreibung

HINTERGRUND

[0001] Die vorliegende Offenbarung betrifft allgemein das Gebiet des Datenschutzes und konkreter ein Redigieren von Informationen aus einem Blockchain-Netzwerk (Blockkettennetzwerk).

[0002] Gegenwärtig können sich Einzelpersonen gemäß der Datenschutzgrundverordnung (DSGVO) an ein/eine Unternehmen/Organisation/Körperschaft (Entität) mit der Bitte wenden, ihre personenbezogenen Daten innerhalb eines bestimmten Zeitrahmens aus allen Systemen der Entität zu löschen. Viele Entitäten nutzen jedoch möglicherweise eine bestimmte Art von Blockchain-Netzwerk innerhalb ihrer Systeme, und Blockchain-Netzwerke nutzen aufgrund ihrer Art unveränderbare Ledger, die Informationen dauerhaft speichern. Dementsprechend tritt ein Problem auf, wenn eine Einzelperson möchte, dass ihre personenbezogenen Daten in Systemen einer Entität redigiert bzw. daraus getilgt werden.

[0003] Ferner besteht gegenwärtig keine geeignete Möglichkeit zum Redigieren von Informationen aus einem Blockchain-Netzwerk, ohne den Block zu beschädigen, in dem die Informationen gespeichert sind. Herkömmlicherweise umfasst jeder Block in dem Blockchain-Netzwerk einen Hash eines vorhergehenden Blocks, über den die Blöcke miteinander verknüpft sind, um eine Kette (Chain) zu bilden. Infolgedessen wird der Hash des Blocks geändert, wenn Inhalte/Informationen eines Blocks geändert werden, somit besteht keine Möglichkeit, eine Gültigkeit des Blocks innerhalb der Chain zu attestieren. Somit ist ein Löschen von Daten direkt aus einem Ledger nicht möglich, das dem Blockchain-Netzwerk zugehörig ist, da sich dies auf die Fähigkeit auswirkt, Blöcke miteinander zu verketten. Daraus ergibt sich, dass ein dringender Bedarf bezüglich eines Mittels zum Redigieren von Informationen aus Blockchain-Netzwerken besteht, ohne die Fähigkeit zu beschädigen, Blöcke weiterhin miteinander zu verketten und die Gültigkeit eines Blocks mit redigierten Inhalten/Informationen aufrechtzuerhalten.

KURZDARSTELLUNG

[0004] Ausführungsformen der vorliegenden Offenbarung umfassen ein Verfahren, ein System und ein Computerprogrammprodukt zum Redigieren von Informationen aus einem Blockchain-Netzwerk. Ein Prozessor kann einen Wert und einen Hash des Wertes in einem Ledger speichern, das einem Blockchain-Netzwerk zugehörig ist. Der Prozessor kann aus dem Hash des Wertes einen Block-Hash aufbauen. Der Prozessor kann eine Transaktion durch ein Erkennen validieren, dass der Hash des Wertes mit dem Hash des Wertes aus einer früheren Transaktion übereinstimmt. Der Prozessor kann eine Unversehrtheit des Block-Hash aufrechterhalten.

[0005] Bei einigen Ausführungsformen kann der Prozessor den Wert redigieren. Das Redigieren des Wertes kann ein Setzen von Bits auf Null umfassen, die dem Wert zugehörig sind.

[0006] Bei einigen Ausführungsformen umfasst das Redigieren des Wertes ferner ein Aufrechterhalten des Hash des Wertes und des Block-Hash.

[0007] Bei einigen Ausführungsformen kann das Redigieren des Wertes ferner aufweisen, dass der Prozessor die Benutzerdaten durch den Hash des Wertes ersetzt. Der Prozessor kann ein entkoppeltes Vorabbild des Wertes aufrechterhalten.

[0008] Bei einigen Ausführungsformen zeigt der Hash des Wertes auf das entkoppelte Vorabbild.

[0009] Bei einigen Ausführungsformen kann der Prozessor die validierte Transaktion ohne den Wert aufrechterhalten.

[0010] Bei einigen Ausführungsformen kann der Prozessor den Wert empfangen. Der Prozessor kann erkennen, dass der Wert Benutzerdaten zugehörig ist. Der Prozessor kann den Hash des Wertes erzeugen.

[0011] Ausführungsformen der vorliegenden Offenbarung umfassen ein Verfahren und ein System zum Redigieren von Informationen aus einem Blockchain-Netzwerk. Ein Prozessor kann eine Redigiertransaktion für das Blockchain-Netzwerk bereitstellen. Der Prozessor kann die Redigiertransaktion einleiten. Die Redigiertransaktion kann einer validierten Transaktion zugehörig sein. Der Prozessor kann einen Hash-Wert erkennen, der innerhalb der Redigiertransaktion angegeben und der validierten Transaktion zugehörig ist. Der Prozessor kann einen Wert redigieren, der dem Hash-Wert zugehörig ist.

[0012] Bei einigen Ausführungsformen kann der Prozessor eine Transaktionshülle (transaction envelope) erzeugen. Die Transaktionshülle kann den Hash-Wert umfassen, der einen Zeiger auf ein Vorabbild enthält.

[0013] Bei einigen Ausführungsformen kann die Transaktionshülle ferner das Vorabbild umfassen, und das Vorabbild kann den Wert umfassen.

[0014] Bei einigen Ausführungsformen kann das Redigieren des Wertes ein Setzen von Bits auf Null umfassen, die dem Wert zugehörig sind. Das Setzen der dem Wert zugehörigen Bits auf Null kann ermöglichen, dass der Hash-Wert der gleiche bleibt.

[0015] Bei einigen Ausführungsformen kann der Prozessor die validierte Transaktion auf der Grundlage des Erkennens aufrechterhalten, dass der Hash-Wert nach der Redigiertransaktion der gleiche ist.

[0016] Mit der vorstehenden Kurzdarstellung ist nicht beabsichtigt, jede veranschaulichte Ausführungsform oder jede Realisierungsform der vorliegenden Offenbarung zu beschreiben.

Figurenliste

[0017] Die in der vorliegenden Offenbarung enthaltenen Zeichnungen sind in die Spezifikation einbezogen und bilden einen Teil davon. Sie veranschaulichen Ausführungsformen der vorliegenden Offenbarung und dienen zusammen mit der Beschreibung der Erläuterung der Grundgedanken der Offenbarung. Die Zeichnungen veranschaulichen lediglich bestimmte Ausführungsformen und schränken die Offenbarung nicht ein.

Fig. 1A veranschaulicht eine beispielhafte Blockchain-Architektur gemäß Ausführungsformen der vorliegenden Offenbarung.

Fig. 1B veranschaulicht einen Blockchain-Transaktionsablauf gemäß Ausführungsformen der vorliegenden Offenbarung.

Fig. 2 veranschaulicht eine herkömmliche Blockchain, die in eine redigierbare Blockchain überführt wird, gemäß Ausführungsformen der vorliegenden Offenbarung.

Fig. 3A veranschaulicht eine herkömmliche Transaktionseinleitung gemäß Ausführungsformen der vorliegenden Offenbarung.

Fig. 3B veranschaulicht eine aktualisierte Transaktionseinleitung gemäß Ausführungsformen der vorliegenden Offenbarung.

Fig. 4 veranschaulicht eine Transaktionsstruktur zum Ermöglichen des Redigierens von Informationen gemäß Ausführungsformen der vorliegenden Erfindung.

Fig. 5 veranschaulicht eine Validierung einer Transaktion durch einen Nachfahren-Peer gemäß Ausführungsformen der vorliegenden Offenbarung.

Fig. 6A veranschaulicht ein Flussdiagramm eines beispielhaften Verfahrens zum Aufrechterhalten der Unversehrtheit eines Block-Hash in einem Blockchain-Netzwerk gemäß Ausführungsformen der vorliegenden Offenbarung.

Fig. 6B veranschaulicht ein Flussdiagramm eines beispielhaften Verfahrens zum Redigieren von Informationen aus einem Blockchain-Netzwerk gemäß Ausführungsformen der vorliegenden Offenbarung.

Fig. 7A veranschaulicht eine Cloud-Computing-Umgebung gemäß Ausführungsformen der vorliegenden Offenbarung.

Fig. 7B veranschaulicht Abstraktionsmodellschichten gemäß Ausführungsformen der vorliegenden Offenbarung.

Fig. 8 veranschaulicht ein Übersichtsblockschema eines beispielhaften Computersystems, das beim Realisieren eines oder mehrerer der Verfahren, Werkzeuge und Module und beliebiger verwandter Funktionen, die hierin beschrieben sind, gemäß Ausführungsformen der vorliegenden Offenbarung verwendet werden kann.

[0018] Zwar können an den hierin beschriebenen Ausführungsformen verschiedene Modifikationen vorgenommen werden, und die Ausführungsformen können alternative Formen annehmen, deren Besonderheiten jedoch sind beispielhaft in den Zeichnungen gezeigt und werden ausführlich beschrieben. Es sollte jedoch klar sein, dass die bestimmten beschriebenen Ausführungsformen nicht in einem einschränkenden Sinne zu verstehen sind. Im Gegenteil, die Offenbarung soll alle Modifikationen, Äquivalente und Alternativen einschließen, die in den Grundgedanken und Schutzumfang der Erfindung fallen.

AUSFÜHRLICHE BESCHREIBUNG

[0019] Aspekte der vorliegenden Offenbarung betreffen allgemein das Gebiet des Datenschutzes und konkreter ein Redigieren von Informationen (z.B. Benutzerinformationen, personenbezogene Daten) aus einem Blockchain-Netzwerk. Viele Einzelpersonen möchten die Kontrolle über ihre Daten (z.B. personenbezogene, private oder anderweitige) behalten, und mit der bevorstehenden DSGVO haben Einzelpersonen jetzt das Recht auf Vergessenwerden. Mit einem derartigen Recht können sich Einzelpersonen jetzt an eine Entität (z.B. ein Unternehmen, eine Firma usw.) mit der Bitte wenden, ihre Daten innerhalb eines bestimmten Zeitrahmens aus den Systemen der Entität zu löschen. Viele Entitäten nutzen jedoch möglicherweise eine bestimmte Art von Blockchain-Netzwerk innerhalb ihrer Systeme, und Blockchain-Netzwerke nutzen aufgrund ihrer Aus unveränderbare Ledger, die Informationen dauerhaft speichern. Dementsprechend tritt ein Problem auf, wenn eine Einzelperson möchte, dass ihre personenbezogenen Daten in Systemen einer Entität redigiert bzw. daraus getilgt werden.

[0020] Gegenwärtig gibt es keine Optionen zum Redigieren von Informationen aus einem Blockchains-Netzwerk. Die naheliegendsten Alternativen sind private Datensammlungen, Verschlüsselung und Speicherung außerhalb der Chain mit Hashes in der Chain. Jede Alternative redigiert jedoch nicht

tatsächlich Informationen aus einem Blockchains-Netzwerk, und jede Alternative hat ihre eigenen Nachteile. Zum Beispiel private Datensammlungen: hängen von Block-To-Live-Richtlinien ab, die auf zeitliche Grundlage beruhen (z.B. Einstellen eines Ablaufzeitpunkts bei einem Block); haben Leistungseinbußen, da Peers/Knoten nach einem Festschreiben von Benutzerdaten abrufen müssen; und müssen im Vorfeld entworfen werden, sodass keinesfalls Benutzerdaten in einer Chain abgelegt werden, da eine öffentliche Chain nicht beseitigt werden kann, sobald sich Daten darauf befinden. Darüber hinaus erfordern Verschlüsselungsalternativen einen Schlüsselverwaltungsaufwand, wobei Vorwärtsgeheimhaltung in einem sich entwickelnden Zeitalter der Quantendatenverarbeitung unklar ist. Schließlich verursacht im Falle mehrerer Verwaltungsdomänen eine Speicherung außerhalb der Chain, wobei sich Hashes in der Chain befinden, Datenverteilungs- sowie Konsistenz- und Verfügbarkeitsschwierigkeiten. Das heißt, insgesamt betrachtet besteht das Hauptproblem bei jeder Alternative darin, dass sie Zusatzprodukte (Add-ons) erfordern, die ursprünglich nicht bereitgestellt wurden und bei einer neuen Blockchain jedes Mal realisiert werden müssen, während hierin eine inhärente Einbaulösung für (zugriffsrechtebehaftete) Blockchains mit einer Ausführen-Anordnen-Validieren-Architektur offenbart wird. Dementsprechend besteht ein echter Bedarf bezüglich eines Verfahrens, das hierin in dieser Offenbarung erörtert wird, das ein Entfernen von Informationen aus einer Blockchain ermöglicht (da es einen bestimmten Benutzer/eine bestimmte Einzelperson betrifft).

[0021] Es versteht sich ohne Weiteres, dass die vorliegenden Komponenten, die in diesem Dokument allgemein beschrieben und in den Figuren veranschaulicht sind, in einer großen Vielfalt unterschiedlicher Konfigurationen angeordnet und ausgestaltet sein können. Dementsprechend ist die folgende ausführliche Beschreibung der Ausführungsformen von mindestens einem aus einem Verfahren, einer Vorrichtung, einem nichtflüchtigen, durch einen Computer lesbaren Speichermedium und einem System, die in den beigefügten Figuren wiedergegeben sind, nicht als Einschränkung des beanspruchten Schutzumfangs der Anmeldung, sondern lediglich als repräsentativ für ausgewählte Ausführungsformen gedacht.

[0022] Die vorliegenden Merkmale, Strukturen oder Eigenschaften, die in dieser Beschreibung beschrieben sind, können bei einer oder mehreren Ausführungsformen in einer beliebigen geeigneten Weise kombiniert oder entfernt werden. Zum Beispiel bezieht sich die Verwendung der Formulierungen „beispielhafte Ausführungsformen“, „einige Ausführungsformen“ oder anderer ähnlicher Ausdrücke in der gesamten Beschreibung auf die Tatsache, dass ein bestimmtes Merkmal, eine bestimmte Struktur oder Eigenschaft, die in Verbindung mit der Ausführungsform beschrieben sind, bei mindestens einer Ausführungsform enthalten sein kann. Dementsprechend bezieht sich das Auftreten der Formulierungen „beispielhafte Ausführungsformen“, „einige Ausführungsformen“, „bei einigen Ausführungsformen“, „bei anderen Ausführungsformen“ oder anderer ähnlicher Ausdrücke in der gesamten Beschreibung nicht zwangsläufig in allen Fällen auf dieselbe Gruppe von Ausführungsformen, und die beschriebenen Merkmale, Strukturen oder Eigenschaften können bei einer oder mehreren Ausführungsformen in einer beliebigen geeigneten Weise kombiniert oder entfernt werden. Ferner kann in den FIGUREN eine beliebige Verbindung zwischen Elementen Einweg- und/oder Zweiweg-Datenübertragung zulassen, selbst wenn es sich bei der abgebildeten Verbindung um einen Einweg- oder Zweiweg-Pfeil handelt. Ebenso kann es sich bei einer beliebigen in den Zeichnungen bildlich dargestellten Einheit auch um eine andere Einheit handeln. Wenn zum Beispiel eine mobile Einheit gezeigt ist, die Daten sendet, könnte auch eine drahtgebundene Einheit verwendet werden, um die Daten zu senden.

[0023] Zwar wird bei der Beschreibung von Ausführungsformen möglicherweise der Begriff „Nachricht“ verwendet, die Anmeldung kann darüber hinaus jedoch auf viele Arten von Netzwerken und Daten angewendet werden. Zwar sind möglicherweise bei beispielhaften Ausführungsformen bestimmte Arten von Verbindungen, Nachrichten und Signalisierung bildlich dargestellt, die Anmeldung ist des Weiteren jedoch nicht auf eine bestimmte Art von Verbindung, Nachricht und Signalisierung beschränkt.

[0024] Hierin ausführlich beschrieben werden ein Verfahren, ein System und ein Computerprogrammprodukt, die angepasste Transaktionshüllen nutzen, die das Redigieren von Informationen aus einem Blockchain-Netzwerk ermöglichen.

[0025] Bei einigen Ausführungsformen nutzen das Verfahren, das System und/oder das Computerprogrammprodukt eine dezentrale Datenbank (wie z.B. eine Blockchain), bei der es sich um ein verteiltes Speichersystem handelt, das mehrere Knoten umfasst, die untereinander Daten austauschen. Die dezentrale Datenbank kann eine unveränderbare Nur-anhängen-Datenstruktur umfassen, die einem verteilten Ledger ähnelt, das in der Lage ist, Datensätze zwischen Parteien zu aufrechtzuerhalten, die sich gegenseitig nicht vertrauen. Diese nicht vertrauenswürdigen Parteien werden hierin als „Peers“ oder „Peer-Knoten“ bezeichnet. Jeder Peer führt ein Exemplar der Datenbankdatensätze, und kein einziger Peer kann Datenbankdatensätze verändern, ohne dass unter den verteilten Peers ein Konsens erzielt wurde. Zum Beispiel können die Peers ein

Konsensprotokoll ausführen, um Blockchain-Speichertransaktionen zu validieren, die Speichertransaktionen in Blöcken zu gruppieren und eine blockübergreifende Hash-Chain aufzubauen. Dieser Prozess bildet das Ledger, indem die Speichertransaktionen gegebenenfalls angeordnet werden, um Konsistenz zu erreichen.

[0026] Bei verschiedenen Ausführungsformen kann eine zugriffsrechtebehaftete und/oder zugriffsrechtelose Blockchain verwendet werden. An einer öffentlichen bzw. zugriffsrechtelosen Blockchain kann jeder ohne eine bestimmte Identität (z.B. durch Bewahren der Anonymität) teilnehmen. Öffentliche Blockchains können native Kryptowährung einschließen und einen Konsens verwenden, der auf verschiedenen Protokollen wie z.B. dem Arbeitsnachweis (Proof of Work) beruht. Andererseits stellt eine zugriffsrechtebehaftete Blockchain-Datenbank sichere Interaktionen in einer Gruppe von Entitäten bereit, die ein gemeinsames Ziel haben, aber einander nicht vollständig vertrauen, wie z.B. Unternehmen, die Finanzmittel, Waren, (private) Informationen und dergleichen austauschen.

[0027] Ferner können das Verfahren, das System und/oder das Computerprogrammprodukt bei einer bestimmten Ausführungsform eine Blockchain nutzen, die mit programmierbarer Zufallslogik arbeitet, die auf ein dezentrales Speicherschema zugeschnitten und als „Smart Contracts“ oder „Chaincodes“ bezeichnet wird. Bei einigen Ausführungsformen können das Verfahren, das System und/oder das Computerprogrammprodukt ferner Smart Contracts nutzen, bei denen es sich um vertrauenswürdige verteilte Anwendungen handelt, die manipulationssichere Eigenschaften der Blockchain-Datenbank und eine zugrunde liegende Vereinbarung zwischen Knoten effizient nutzen, die als „Bewilligung“ oder „Bewilligungsrichtlinie“ bezeichnet wird. Blockchain-Transaktionen im Zusammenhang mit dieser Anmeldung können bewilligt werden, bevor sie in der Blockchain festgeschrieben werden, während Transaktionen außer Acht gelassen werden, die nicht bewilligt sind.

[0028] Eine Bewilligungsrichtlinie ermöglicht es Chaincode, Bewilliger für eine Transaktion in Form eines Satzes von Peer-Knoten anzugeben, die zur Bewilligung nötig sind. Wenn ein Client die Transaktion an die in der Bewilligungsrichtlinie angegebenen Peers sendet, wird die Transaktion ausgeführt, um die Transaktion zu validieren. Bei einigen Ausführungsformen wird die Transaktion zunächst simuliert, z.B. bewilligt, um eine Zustimmung von den Peers zu erhalten, sodass die Bewilligungsrichtlinie eingehalten wird. Daher gelangt die Transaktion in eine Anordnungsphase, in der ein Konsensprotokoll verwendet wird, um eine angeordnete Abfolge von bewilligten Transaktionen zu erzeugen, die in Blöcken gruppiert sind. Ferner wird die Transaktion bei einigen Ausführungsformen validiert, nachdem die Transaktion in einen Block/in Blöcke geht und die Peers erreicht.

[0029] Bei einer bestimmten Ausführungsform können das Verfahren, das System und/oder das Computerprogrammprodukt Knoten nutzen, bei denen es sich um die Datenübertragungs-Entitäten des Blockchain-Systems handelt. Ein „Knoten“ kann eine logische Funktion in dem Sinne durchführen, dass mehrere Knoten unterschiedlicher Arten auf demselben physischen Server ausgeführt werden können. Knoten sind in Vertrauensdomänen gruppiert und logischen Einheiten zugehörig, die sie auf verschiedene Weise steuern. Knoten können unterschiedliche Arten aufweisen, z.B. einen Client oder einen übermittelnden Client-Knoten, der einen Transaktionsaufruf an einen Bewilliger (z.B. Peer) übermittelt und Transaktionsvorschläge an einen Anordnungsservice (z.B. an einen Anordnungsknoten) übermittelt.

[0030] Bei einer weiteren Art von Knoten handelt es sich um einen Peer-Knoten, der durch einen Client übermittelte Transaktionen empfangen, die Transaktionen festschreiben und einen Zustand (State) und ein Exemplar des Ledger von Blockchain-Transaktionen verwalten kann. Peers können außerdem die Rolle eines Bewilligers haben, obwohl dies nicht zwangsläufig der Fall sein muss. Bei einem Anordnungsserviceknoten oder Anordner handelt es sich um einen Knoten, der einen Datenübertragungsservice für alle Knoten ausführt und der eine Übergabegarantie wie z.B. eine Rundsendung an jeden der Peer-Knoten in dem System beim Festschreiben/-Bestätigen von Transaktionen und Verändern eines World-State der Blockchain realisiert, bei dem es sich um einen weiteren Namen für die anfängliche Blockchain-Transaktion handelt, die normalerweise Steuerungs- und Einrichtungsinformationen enthält.

[0031] Bei einer bestimmten Ausführungsform können das Verfahren, das System und/oder das Computerprogrammprodukt ein Ledger nutzen, bei dem es sich um einen in einer Abfolge angeordneten manipulationsbeständigen Datensatz aller Zustandsübergänge einer Blockchain handelt. Zustandsübergänge können aus Chaincode-Aufrufen (z.B. Transaktionen) entstehen, die durch teilnehmende Parteien (z.B. Client-Knoten, Anordnungsknoten, Bewilligerknoten, Peer-Knoten usw.) übermittelt wurden. Jede teilnehmende Partei (wie z.B. ein Peer-Knoten) kann ein Exemplar des Ledger führen. Eine Transaktion kann zu einem Satz von Asset-Schlüsselwertpaaren führen, die in dem Ledger als ein oder mehrere Operanden wie z.B. Erstellungen, Aktualisierungen, Löschungen und dergleichen festgeschrieben werden. Das Ledger enthält eine Blockchain (die auch als „Chain“ bezeichnet wird), die verwendet wird, um einen unveränderlichen, in einer Abfolge angeordneten Datensatz in Blöcken zu spei-

chern. Das Ledger enthält außerdem eine Zustandsdatenbank, die einen aktuellen Zustand (State) der Blockchain führt.

[0032] Bei einer bestimmten Ausführungsform können das Verfahren, das System und/oder das Computerprogrammprodukt, die hierin beschrieben sind, eine Chain nutzen, bei der es sich um ein Transaktionsprotokoll handelt, das als Hash-verknüpfte Blöcke strukturiert ist und jeder Block eine Abfolge von N Transaktionen enthält, wobei N gleich oder größer als eins ist. Die Block-Kopfdaten enthalten einen Hash der Transaktionen des Blocks sowie einen Hash der Kopfdaten des vorhergehenden Blocks. Auf diese Weise können alle Transaktionen des Ledger in einer Abfolge angeordnet und kryptografisch miteinander verknüpft werden. Dementsprechend ist es nicht möglich, die Ledger-Daten zu manipulieren, ohne die Hash-Verknüpfungen aufzubrechen. Ein Hash des zuletzt hinzugefügten Blockchain-Blocks gibt jede Transaktion der Chain wieder, die vor dieser stattgefunden hat, wodurch es möglich wird zu gewährleisten, dass sich alle Peer-Knoten in einem konsistenten und vertrauenswürdigen Zustand befinden. Die Chain kann in einem Peer-Knotendateisystem (z.B. lokaler Speicher, angeschlossener Speicher, Cloud usw.) gespeichert werden, das die Nur-anhängen-Art der Blockchain-Betriebslast wirksam unterstützt.

[0033] Der aktuelle Zustand des unveränderbaren Ledger gibt die neuesten Werte aller Schlüssel wieder, die im Chain-Transaktionsprotokoll enthalten sind. Da der aktuelle Zustand die neuesten Schlüsselwerte wiedergibt, die einem Kanal bekannt sind, wird er manchmal als „World-State“ bezeichnet. Chaincode-Aufrufe führen Transaktionen in Bezug auf die Daten des aktuellen Zustands des Ledger aus. Um diese Chaincode-Interaktionen effizient ablaufen zu lassen, können die neuesten Werte der Schlüssel in einer Zustandsdatenbank gespeichert werden. Bei der Zustandsdatenbank kann es sich einfach um eine indizierte Ansicht des Transaktionsprotokolls der Chain handeln, sie kann daher zu einem beliebigen Zeitpunkt aus der Chain neu erzeugt werden. Die Zustandsdatenbank kann nach dem Start von Peer-Knoten und vor dem Akzeptieren von Transaktionen automatisch wiederhergestellt (oder bei Bedarf erzeugt) werden.

[0034] Zu einigen Vorteilen der hierin beschriebenen und bildlich dargestellten Lösungen gehören ein Verfahren, ein System und ein Computerprogrammprodukt zum Redigieren von (personenbezogenen) Informationen aus einem Blockchain-Netzwerk. Die beispielhaften Ausführungsformen stellen eine Lösung zum Schützen von Informationen in einer Blockchain bei gleichzeitigem Aufrechterhalten der Widerstandsfähigkeit der Chain bereit (z.B. verursacht das Redigieren von Informationen keine Außerkraftsetzung einer nachfolgenden Transaktion).

[0035] Es wird angemerkt, dass sich die Blockchain von einer herkömmlichen Datenbank darin unterscheidet, dass es sich bei Blockchain nicht um einen zentralen Speicher handelt, sondern vielmehr um einen dezentralen, unveränderbaren und sicheren Speicher, bei dem Knoten Änderungen an Datensätzen im Speicher gemeinsam nutzen können. Zu einigen Eigenschaften, die für eine Blockchain charakteristisch sind und eine Realisierung der Blockchain erleichtern, gehören, ohne auf diese beschränkt zu sein, ein unveränderbares Ledger, Smart Contracts, Sicherheit, Privatsphäre, Dezentralisierung, Konsens, Bewilligung, Zugriffsfähigkeit und dergleichen, die im Folgenden ausführlicher beschrieben werden. Gemäß verschiedenen Aspekten wird das hierin beschriebene System aufgrund Verantwortlichkeit, Sicherheit, Privatsphäre, zulässiger Dezentralisierung, Verfügbarkeit von Smart Contracts, Bewilligungen und Zugriffsfähigkeit realisiert, die für Blockchain charakteristisch und eindeutig sind.

[0036] Insbesondere sind die Daten des Blockchain-Ledger herkömmlicherweise unveränderbar und stellen ein effizientes Verfahren zum Erkennen von Unstimmigkeiten in einem Blockchain-Netzwerk bereit; die vorliegende Offenbarung stellt ein Verfahren, ein System und ein Computerprogrammprodukt zum Abmildern der Auswirkungen von Unstimmigkeiten bereit, die zu einer nachfolgenden Außerkraftsetzung von Transaktionen (z.B. Aufzeichnen von Daten) führen würden. Außerdem stellt eine Verwendung von Verschlüsselung in der Blockchain Sicherheit bereit und baut Vertrauen auf. Der Smart Contract verwaltet den Zustand des Assets, um einen Lebenszyklus zu Ende zu führen. Die beispielhaften Blockchains sind in Bezug auf Zugriffsrechte dezentralisiert. Somit kann jeder Endbenutzer sein eigenes Ledger-Exemplar haben, auf das er zugreift. Mehrere Organisationen (und Peers) können in das Blockchain-Netzwerk aufgenommen werden. Die wichtigsten Organisationen können als Bewilligungs-Peers dienen, um die Ergebnisse der Ausführung von Smart Contracts, Lesesatz und Schreibsatz zu validieren.

[0037] Einer der Vorteile der beispielhaften Ausführungsformen besteht darin, dass er die Funktionalität eines Datenverarbeitungssystems verbessert, indem ein Verfahren zum Redigieren von (personenbezogenen/privaten/usw.) Daten aus einem Blockchain-Netzwerk realisiert wird (z.B. mit Hilfe des Redigierens von Informationen aus einem Vorabbild in der Transaktionshülle). Über das hierin beschriebene Blockchain-System kann ein Datenverarbeitungssystem (oder ein Prozessor in dem Datenverarbeitungssystem) unter Nutzung von Blockchain-Netz-

werken Funktionalität zur Redaktionsverarbeitung durchführen, indem Zugriff auf Fähigkeiten wie z.B. verteiltes Ledger, Peers, Verschlüsselungstechnologien, MSP, Ereignisbehandlung usw. bereitgestellt wird. Außerdem ermöglicht die Blockchain ein Erstellen eines Geschäftsnetzwerks und ein Aufnehmen beliebiger Benutzer oder Organisationen zur Teilnahme. Die Blockchain an sich ist nicht nur eine Datenbank. Die Blockchain enthält Fähigkeiten zum Erstellen eines Netzwerks aus Benutzern und zum Aufnehmen/Entfernen von Organisationen zum Zusammenarbeiten und Ausführen von Serviceprozessen in Form von Smart Contracts (die Redaktionsprozessen und/oder bestimmten zu redigieren den Informationen zugehörig sein können).

[0038] Die beispielhaften Ausführungsformen stellen gegenüber einer herkömmlichen Datenbank zahlreiche Vorteile bereit. Zum Beispiel stellen die Ausführungsformen über die Blockchain Verantwortlichkeit, Sicherheit, Privatsphäre, zulässige Dezentralisierung, Verfügbarkeit von Smart Contracts, Bewilligungen und Zugriffsfähigkeit bereit, die für die Blockchain charakteristisch und eindeutig sind.

[0039] Mittlerweile könnte eine herkömmliche Datenbank nicht verwendet werden, um die beispielhaften Ausführungsformen zu realisieren, da sie nicht alle Parteien in das Geschäftsnetzwerk bringt, keine vertrauenswürdige Zusammenarbeit herstellt und keine effiziente Speicherung digitaler Assets bereitstellt. Die herkömmliche Datenbank stellt keine manipulationssichere Speicherung bereit und stellt keine Aufbewahrung der gespeicherten digitalen Assets bereit. Somit können die hierin beschriebenen vorgeschlagenen Ausführungsformen, die Blockchain-Netzwerke nutzen, nicht in der herkömmlichen Datenbank realisiert werden.

[0040] Wenn eine herkömmliche Datenbank verwendet werden müsste, um die beispielhaften Ausführungsformen zu realisieren, würden mittlerweile für die beispielhaften Ausführungsformen unnötige Nachteile wie z.B. mangelnde Suchfähigkeit, Fehlen von Sicherheit und niedrige Geschwindigkeit von Transaktionen entstehen (indem z.B. versucht wird, Unstimmigkeiten redigierter Informationen zu erkennen). Dementsprechend stellen die beispielhaften Ausführungsformen eine konkrete Lösung für ein Problem in der Technik/auf dem Gebiet der Informationsredaktion in einer Blockchain bereit.

[0041] Nunmehr unter Bezugnahme auf **Fig. 1A** ist dort eine beispielhafte Blockchain-Architektur 100 gemäß Ausführungsformen der vorliegenden Offenbarung veranschaulicht. Bei einigen Ausführungsformen kann die Blockchain-Architektur 100 bestimmte Blockchain-Elemente enthalten, zum Beispiel eine Gruppe von Blockchain-Knoten 102. Die Blockchain-Knoten 102 können einen oder mehrere Blockchain-Knoten umfassen, z.B. Peers 104 bis 110 (diese vier Knoten sind lediglich beispielhaft abgebildet). Diese Knoten nehmen an einer Reihe von Aktivitäten wie z.B. an einem Hinzufügungs- und Validierungsprozess (Konsens) in Bezug auf Blockchain-Transaktionen teil. Einer oder mehrere der Peers 104 bis 110 können auf der Grundlage einer Bewilligungsrichtlinie Transaktionen bewilligen und/oder empfehlen und können einen Anordnungsservice für alle Blockchain-Knoten 102 in der Blockchain Architektur 100 bereitstellen. Ein Blockchain-Knoten kann eine Blockchain-Berechtigungsprüfung einleiten und versuchen, in ein unveränderbares Blockchain-Ledger zu schreiben, das in einer Blockchain-Schicht 116 gespeichert ist, von dem ein Exemplar auch in der unterlagerten physischen Infrastruktur 114 gespeichert sein kann. Die Blockchain-Konfiguration kann eine oder mehrere Anwendungen 124 enthalten, die mit Anwendungsprogrammierschnittstellen (Application Programming Interfaces, APIs) 122 verknüpft sind, um auf gespeicherten Programm-/Anwendungscode 120 (z.B. Chaincode, Smart Contracts usw.) zuzugreifen und diesen auszuführen, der gemäß einer von Teilnehmern gesuchten maßgeschneiderten Konfiguration erstellt werden kann, und kann ihren eigenen Zustand führen, ihre eigenen Assets steuern und externe Informationen empfangen. Dies kann als Transaktion bereitgestellt und über ein Anhängen an das verteilte Ledger bei allen Blockchain-Knoten 104 bis 110 installiert werden.

[0042] Die Blockchain-Grundlage oder -Plattform 112 kann verschiedene Schichten von Blockchain-Daten, Services (z.B. kryptografische Vertrauensservices, virtuelle Ausführungsumgebung usw.) und eine unterlagerte physische Computerinfrastruktur enthalten, die verwendet werden kann, um neue Transaktionen zu empfangen und zu speichern und Zugriff für Auditoren bereitzustellen, die versuchen, auf Dateneinträge zuzugreifen. Die Blockchain-Schicht 116 kann eine Schnittstelle zeigen, die Zugriff auf die virtuelle Ausführungsumgebung bereitstellt, die notwendig ist, um den Programmcode zu verarbeiten und die physische Infrastruktur 114 einzubeziehen. Kryptografische Vertrauensservices 118 können verwendet werden, um Transaktionen wie z.B. Asset-Austauschtransaktionen zu überprüfen und Informationen privat zu halten.

[0043] Die Blockchain-Architektur 100 von **Fig. 1A** kann Programm-/ Anwendungscode 120 über eine oder mehrere durch die Blockchain-Plattform 112 gezeigte Schnittstellen und bereitgestellte Services verarbeiten und ausführen. Der Anwendungscode 120 kann Blockchain-Assets steuern. Zum Beispiel kann der Anwendungscode 120 Daten speichern und übertragen und kann durch die Peers 104 bis 110 in Form eines Smart Contract und zugehörigen Chaincodes mit Bedingungen oder anderen Codee-

lementen ausgeführt werden, die seiner Ausführung unterliegen. Als nicht einschränkendes Beispiel können Smart Contracts erstellt werden, um die Übertragung von Ressourcen und/oder die Erzeugung von Ressourcen auszuführen, usw. Die Smart Contracts selbst können verwendet werden, um Regeln im Zusammenhang mit Berechtigung, Zugriffsanforderungen (z.B. eines Datenspeichers, eines Datenspeichers außerhalb der Chain, einer Vorabbilddatenbank usw.) und/oder Verwendung des Ledger zu erkennen. Zum Beispiel können die Vorabbildinformationen 126 durch eine oder mehrere Verarbeitungsentitäten (z.B. virtuelle Maschinen) verarbeitet werden, die in der Blockchain-Schicht 116 enthalten sind. Das Ergebnis 128 kann eine Mehrzahl von verknüpften, gemeinsam genutzten Dokumenten umfassen (z.B. mit jedem verknüpften, gemeinsam genutzten Dokument, das die Ausgabe eines Smart Contract in Bezug auf die Vorabbildinformationen 126 aufzeichnet, die als entweder zugelassene oder verweigerte Validierung erkannt wurden, usw.). Bei einigen Ausführungsformen kann die physische Infrastruktur 114 genutzt werden, um beliebige der hierin beschriebenen Daten oder Informationen abzurufen.

[0044] Ein Smart Contract kann über eine Anwendung und eine Programmiersprache einer höheren Ebene erstellt und anschließend in einen Block in der Blockchain geschrieben werden. Der Smart Contract kann ausführbaren Code enthalten, der bei einer Blockchain (z.B. bei einem verteilten Netzwerk aus Blockchain-Peers) registriert, gespeichert und/oder repliziert ist. Bei einer Transaktion handelt es sich um eine Ausführung des Smart-Contract-Codes, der als Reaktion auf Bedingungen durchgeführt werden kann, die dem erfüllten Smart Contract zugehörig sind. Das Ausführen des Smart Contract kann (eine) vertrauenswürdige Veränderung(en) an einem Zustand eines digitalen Blockchain-Ledger auslösen. Die Veränderung(en) an dem Blockchain-Ledger, die durch die Ausführung des Smart Contract verursacht wurde(n), kann bzw. können über ein oder mehrere Konsensprotokolle automatisch über das gesamte verteilte Netzwerk aus Blockchain-Peers hinweg repliziert werden.

[0045] Der Smart Contract kann Daten im Format von Schlüsselwertpaaren in die Blockchain schreiben. Des Weiteren kann der Smart-Contract-Code die in einer Blockchain gespeicherten Werte lesen und sie bei Anwendungsoperationen verwenden. Der Smart-Contract-Code kann die Ausgabe verschiedener Logikoperationen in die Blockchain schreiben. Der Code kann verwendet werden, um eine zeitweilige Datenstruktur in einer virtuellen Maschine oder in einer anderen Datenverarbeitungsplattform zu erstellen. In die Blockchain geschriebene Daten können öffentlich und/oder verschlüsselt sein und als privat geführt werden. Die durch den Smart Contract verwendeten/erzeugten Daten werden durch die bereitgestellte Ausführungsumgebung im Speicher gehalten und anschließend gelöscht, sobald die für die Blockchain benötigten Daten erkannt wurden.

[0046] Ein Chaincode kann die Codeinterpretation eines Smart Contract mit zusätzlichen Merkmalen enthalten. Wie hierin beschrieben kann es sich bei dem Chaincode um Programmcode handeln, der auf einem Datenverarbeitungsnetzwerk bereitgestellt ist, in dem er ausgeführt und während eines Konsensprozesses durch Chain-Validation gemeinsam validiert wird. Der Chaincode empfängt einen Hash und ruft aus der Blockchain einen Hash ab, der der Datenvorlage zugehörig ist, die durch Verwendung eines zuvor erstellten Merkmalextraktors erstellt wurde. Wenn die Hashes der Hash-Kennung und der Hash übereinstimmen, der aus den gespeicherten Kennungsvorlagendaten erstellt wurde, sendet der Chaincode einen Berechtigungsschlüssel an den angeforderten Service. Der Chaincode kann Daten in die Blockchain schreiben, die den kryptografischen Einzelheiten zugehörig sind (z.B. somit ein Beitrag bestätigen, eine Unstimmigkeit bei einem Beitrag erkennen usw.).

[0047] Fig. 1B veranschaulicht ein Beispiel eines Blockchain-Transaktionsablaufs 150 zwischen Knoten der Blockchain gemäß beispielhaften Ausführungsformen. Unter Bezugnahme auf **Fig. 1B** kann der Transaktionsablauf einen Transaktionsvorschlag 191 enthalten, der durch einen Anwendungs-Client-Knoten 160 an einen Bewilligungs-Peer-Knoten 181 gesendet wird (z.B. kann es sich bei einigen Ausführungsformen bei dem Transaktionsvorschlag 191 um eine Anforderung handeln, die eine Kennung enthält, die einem Datenspeicher/einer Datenbank außerhalb der Chain zugehörig ist). Der Bewilligungs-Peer 181 kann die Client-Signatur überprüfen und eine Chaincode-Funktion ausführen, um die Transaktion auszulösen. Die Ausgabe kann die Chaincode-Ergebnisse, einen Satz von Schlüssel-/Wert-Versionen, die im Chaincode gelesen wurden (Lesesatz), und einen Satz von Schlüsseln/Werten enthalten, die im Chaincode geschrieben wurden (Schreibsatz). Die Vorschlagsantwort 192 wird zusammen mit einer Bewilligungssignatur, sofern diese genehmigt wurde, zurück an den Client 160 gesendet. Der Client 160 setzt die Bewilligungen zu einer Transaktionsnutzlast 193 zusammen und sendet diese per Rundsendung an einen Anordnungsserviceknoten 184. Der Anordnungsserviceknoten 184 stellt dann allen Peers 181 bis 183 auf einem Kanal geordnete Transaktionen als Blöcke bereit. Vor der Festschreibung in die Blockchain kann jeder Peer 181 bis 183 die Transaktion validieren. Zum Beispiel können die Peers die Bewilligungsrichtlinie kontrollieren, um sicherzustellen, dass der korrekte Anteil der angegebenen Peers die Ergebnisse signiert und die Signa-

turen gegen die Transaktionsdaten 193 auf Berechtigung geprüft haben (z.B. validieren alle oder eine Schwellenwertanzahl von Peers, dass die Anforderung die Kennung und/oder den symmetrischen Schlüssel enthält, die/der das Auffinden eines Datenspeicherverbindungsobjekts und/oder den Zugriff auf einen Datenspeicher außerhalb der Blockchain ermöglicht).

[0048] Unter erneuter Bezugnahme auf **Fig. 1B** leitet der Client-Knoten 160 die Transaktion 191 durch Aufbauen und Senden eine Anforderung an den Peer-Knoten 181 ein, bei dem es sich um einen Bewilliger handelt. Der Client 160 kann eine Anwendung enthalten, die ein unterstütztes Software Development Kit (SDK) nutzt, das eine verfügbare API nutzt, um einen Transaktionsvorschlag 191 zu erzeugen. Bei dem Vorschlag handelt es sich um eine Anforderung zum Aufrufen einer Chaincode-Funktion, sodass die Daten in das Ledger geschrieben (z.B. Schreiben neuer Schlüsselwertpaare für die Assets) oder daraus gelesen werden können. Das SDK kann das Paket des Transaktionsvorschlags 191 zu einem Format mit einer ordnungsgemäßen Architektur (z.B. Protokollpuffer über einen Prozedurfernaufruf (Remote Procedure Call, RPC)) verringern und die Kryptografie-Berechtigungsnachweise des Clients nehmen, um eine eindeutige Signatur für den Transaktionsvorschlag 191 zu erzeugen.

[0049] Als Reaktion kann der Bewilligungs-Peer-Knoten 181 überprüfen, dass (a) der Transaktionsvorschlag 191 richtig gebildet ist, (b) die Transaktion nicht bereits in der Vergangenheit übermittelt wurde (Schutz vor Attacken durch Nachrichtenaufzeichnung und -wiederholung (replay-attacks)), (c) die Signatur gültig ist und (d) dass der Übermittler (bei diesem Beispiel Client 160) ordnungsgemäß berechtigt ist, die vorgeschlagene Operation auf diesem Kanal durchzuführen. Der Bewilligungs-Peer-Knoten 181 kann die Eingaben des Transaktionsvorschlags 191 als Argumente zu der aufgerufenen Chaincode-Funktion nehmen. Der Chaincode wird dann im Vergleich mit einer aktuellen Zustandsdatenbank ausgeführt, um Transaktionsergebnisse zu erzeugen, die einen Antwortwert, einen Lesesatz und einen Schreibsatz enthalten. Zu diesem Zeitpunkt werden jedoch keine Aktualisierungen an dem Ledger vorgenommen. Bei einigen Ausführungsformen wird der Satz von Werten zusammen mit der Signatur des Bewilligungs-Peer-Knotens 181 als Vorschlagsantwort 192 an das SDK des Clients 160 zurückgegeben, das die Nutzlast für die Anwendung zum Verbrauch analysiert.

[0050] Als Reaktion untersucht/überprüft die Anwendung des Clients 160 die Signaturen der Bewilligungs-Peers und vergleicht die Vorschlagsantworten, um zu ermitteln, ob es sich bei der Vorschlagsantwort um dieselbe handelt. Wenn der Chaincode nur das Ledger abgefragt hat, würde die Anwendung die Abfrageantwort untersuchen und die Transaktion in der Regel nicht an den Anordnungsknotenservice 184 übermitteln. Wenn die Client-Anwendung beabsichtigt, die Transaktion an den Anordnungsknotenservice 184 zu übermitteln, um das Ledger zu aktualisieren, ermittelt die Client-Anwendung, ob die angegebene Bewilligungsrichtlinie vor dem Übermitteln erfüllt wurde (z.B. wurde eine Anforderung akzeptiert). Hierbei kann der Client nur eine von mehreren Parteien in die Transaktion einbeziehen. In diesem Fall kann jeder Client seinen eigenen Bewilligungsknoten haben, und jeder Bewilligungsknoten muss die Transaktion bewilligen. Die Architektur ist so gestaltet, dass die Bewilligungsrichtlinie nach wie vor durch Peers durchgesetzt und in der Validierungsphase der Festschreibung aufrechterhalten wird, selbst wenn eine Anwendung entscheidet, Antworten nicht zu untersuchen oder eine nicht bewilligte Transaktion weiterleitet.

[0051] Nach erfolgreicher Untersuchung setzt der Client 160 im Transaktionsnutzlastschritt 193 Bewilligungen zu einer Transaktion zusammen und sendet den Transaktionsvorschlag 191 und die Antwort innerhalb einer Transaktionsnachricht per Rundsendung an den Anordnungsknoten 184. Die Transaktion kann die Lese-/Schreibe-Sätze, die Signaturen der Bewilligungs-Peers und eine Kanal-ID enthalten. Der Anordnungsknoten 184 muss nicht den gesamten Inhalt einer Transaktion untersuchen, um seine Operation durchzuführen, stattdessen kann der Anordnungsknoten 184 möglicherweise einfach Transaktionen von allen Kanälen im Netzwerk empfangen, sie chronologisch nach Kanal anordnen und Blöcke von Transaktionen pro Kanal erstellen.

[0052] Die Blöcke der Transaktion werden aus dem Anordnungsknoten 184 an alle Peer-Knoten 181 bis 183 auf dem Kanal übergeben. Die Transaktionen 194 innerhalb des Blocks werden validiert, um sicherzustellen, dass beliebige Bewilligungsrichtlinien ausnahmslos erfüllt sind, und um sicherzustellen, dass am Ledger-Zustand bei Lesesatz-Variablen keine Änderungen vorgenommen wurden, seitdem der Lesesatz durch die Transaktionsausführung erzeugt wurde. Transaktionen in dem Block sind als „gültig“ oder „ungültig“ markiert. Des Weiteren hängt jeder Peer-Knoten 181 bis 183 in den Schritten 195 den Block an die Chain des Kanals an, und für jede gültige Transaktion werden die Schreibsätze in die aktuelle Zustandsdatenbank festgeschrieben. Ein Ereignis wird ausgegeben, um die Client-Anwendung zu benachrichtigen, dass die Transaktion (der Aufruf) unveränderbar an die Chain angehängt wurde, sowie um mitzuteilen, ob die Transaktion für gültig oder ungültig erklärt wurde (z.B. ob die Anforderung zulässig ist oder verweigert wurde, Zugriff auf einen Datenspeicher außerhalb der Chain).

[0053] Nunmehr unter Bezugnahme auf **Fig. 2** ist dort eine herkömmliche Blockchain 202A gemäß Ausführungsformen der vorliegenden Offenbarung veranschaulicht, die in eine redigierbare Blockchain 202B überführt wird. Es wird angemerkt, dass in den beigefügten Zeichnungen gleiche Bezugsnummern zur Bezeichnung gleicher Teile verwendet werden. Wie bildlich dargestellt umfasst die herkömmliche Blockchain 202A einen Block-Hash 204, bei dem es sich um einen Hash von Block-Kopfdaten (Block-Header) handelt und der als Referenznummer für einen Block (nicht bildlich dargestellt) in der herkömmlichen Blockchain 202A fungiert. Der Block-Hash 204 ist herkömmlicherweise einem ersten Wert 206 und/oder einem zweiten Wert 208 zugehörig (referenziert, zeigt auf diese usw.). Bei einigen Ausführungsformen kann es sich bei dem ersten Wert 206 und/oder dem zweiten Wert 208 um Werte handeln, die Transaktionen zugehörig sind, in die Informationen einbezogen sind. In einem derartigen Fall sind die Informationen unveränderbar in der herkömmlichen Blockchain 202A gespeichert.

[0054] Um die Redaktion des ersten Wertes 206 und/oder des zweiten Wertes 208 zu ermöglichen, stellt die vorliegende Offenbarung bereit, dass die herkömmliche Blockchain 202A in die redigierbare Blockchain 202B überführt wird, indem Hash-Werte des ersten Wertes 206 und/oder des zweiten Wertes 208 gebildet werden. Das Bilden der Hash-Werte des ersten Wertes 206 und/oder des zweiten Wertes 208 erzeugt den jeweiligen ersten Hash 210 und/oder den zweiten Hash 212, der nun dem Block-Hash 204 zugehörig ist. In einem derartigen Fall und wie nachstehend noch ausführlicher zu erörtern ist, können der erste Wert 206 und/oder der zweite Wert 208 nun redigiert werden, ohne den Block-Hash 204 oder den durch den Block-Hash 204 referenzierten Block außer Kraft zu setzen, da es sich bei dem ersten Hash 210 und/oder dem zweiten Hash 212 effektiv um den ersten Wert 206 und den zweiten Wert 208 in der herkömmlichen Blockchain 202A handelt, die sich nicht ändern. Es wird angemerkt, dass es sich bei der herkömmlichen Blockchain 202A und der redigierbaren Blockchain 202B um dieselbe Blockchain handelt, jedoch mit einer strukturellen Änderung.

[0055] Nunmehr unter Bezugnahme auf **Fig. 3A** ist dort eine herkömmliche Transaktionseinleitung 300A gemäß Ausführungsformen der vorliegenden Offenbarung veranschaulicht. Wie bildlich dargestellt umfasst die herkömmliche Transaktion 300A eine Hülle 302, die eine Nutzlast 304 und eine Signatur 314 umfasst, die die Nutzlast 304 betrifft. Die Nutzlast 304 umfasst Kopfdaten 306, die Zielinformationen umfassen (die z.B. kennzeichnen, wohin Assets der Transaktion gesendet werden sollen usw.). Die Nutzlast 304 umfasst ferner eine Transaktion 308, die Aktionen 310 umfasst. Die Transaktionsaktionen 310 enthalten Anweisungen, die als Transaktionsaktion 312 bildlich dargestellt sind, es wird jedoch angemerkt, dass mehr als eine Anweisung in der Transaktion 308 enthalten sein kann (z.B. Informationen aus Datenbank x abrufen und Informationen in Datenbank y speichern usw.). Die Transaktionsaktion 312 kann Informationen/Daten speichern, die einem Benutzer zugehörig sind, und/oder den Block-Hash 204 enthalten, wie in **Fig. 2** bildlich dargestellt.

[0056] Bei einigen Ausführungsformen belegt die Signatur 314 das Zustimmen zu Ergebnissen der Ausführung der Aktionen 310 der Transaktion 308 (z.B. Belegen des Abrufens von Informationen aus der Datenbank X usw.), wodurch die Festschreibung in einer Blockchain ermöglicht wird. Bei einigen Ausführungsformen wird, wenn die Signatur 314 nicht empfangen wird, die Transaktion 308 außer Kraft gesetzt und/oder zurückgewiesen, was der Fall sein kann, wenn Informationen in den Transaktionsaktionen 312 nicht übereinstimmen. Wenn zum Beispiel eine Transaktion mit einem Benutzernamen in der Nutzlast 304 validiert ist und sie in der Blockchain festgeschrieben ist, hat jeder Peer in der Blockchain jetzt den Benutzernamen in seinem Exemplar des Blockchain-Ledger. Wenn dann der Benutzername aus der Transaktion entfernt wird und ein neuer Peer der Blockchain beitritt, erkennen die Peers in der Blockchain vor dem Entfernen des Benutzernamens, dass die Transaktion nicht mehr abgeglichen ist, und die Transaktion wird zurückgewiesen.

[0057] Nunmehr unter Bezugnahme auf **Fig. 3B** ist dort eine aktualisierte Transaktionseinleitung 300B gemäß Ausführungsformen der vorliegenden Offenbarung veranschaulicht. Es wird angemerkt, dass die aktualisierte Transaktionseinleitung 300B dieselben Komponenten enthält, die in der herkömmlichen Transaktionseinleitung 300A enthalten sind, aber ferner Hashes 316 enthält, die Vorabbildern 318 zugehörig sind und auf diese zeigen (z.B. einen Zeiger enthalten), die von der Nutzlast 304 entkoppelt sind. Bei den Vorabbildern 318 handelt es sich um beliebige Informationen/Werte, die für die Transaktion 308 verwendet werden müssen, aber da sich die Hashes 316 in der Transaktionsaktion 312 befinden, werden die Werte der Hashes 316 verwendet, und die Vorabbilder 318 werden nie gegenüber der Blockchain exponiert (z.B. werden Benutzerdaten nie in der Validierungslogik verwendet). Ferner können die Vorabbilder 318 aus der Hülle 302 gelöscht/redigiert werden und setzen die Transaktion nicht außer Kraft, da die Hashes 316 die Werte ersetzen und die Transaktionsvalidierung gegenüber dem Inhalt der Werte indifferent ist, die den Hashes 316 zugehörig sind. Es wird angemerkt, dass das, was hierin beschrieben ist, funktionell erreichbar ist, da während der Validierung einer Transaktion die Validierung nur Versionen eines Schlüssels/von Schlüsseln

(z.B. eines Schlüssels/von Schlüsseln, die den Hashes 316 zugehörig sind) und die Einhaltung der Bewilligungsrichtlinie betrifft, wodurch die Löschung der Vorabbilder 318 ermöglicht wird.

[0058] Nunmehr unter Bezugnahme auf **Fig. 4** ist dort eine Transaktionsstruktur 400 zum Ermöglichen des Redigierens von Informationen gemäß Ausführungsformen der vorliegenden Erfindung veranschaulicht. Bei einigen Ausführungsformen kann es sich bei der Transaktionsstruktur 400 um die Struktur handeln, die durch die Aktionen 310 der **Fig. 3A** und **Fig. 3B** verwendet wird. Bei einigen Ausführungsformen umfasst die Transaktionsstruktur 400 eine Transaktionsaktion 452, bei der es sich um die Transaktionsaktion 312 der **Fig. 3A** und **Fig. 3B** handeln könnte. Wie bildlich dargestellt umfasst die Transaktionsaktion 452 eine Chaincode-Aktionsnutzlast 454, die eine Chaincode-Bewilligungsaktion 456 umfasst. Die Chaincode-Bewilligungsaktion 456 umfasst Bewilligungen 458, bei denen es sich um Signaturen/Validierungen/Genehmigungen/usw.
handeln kann, die angeben, dass eine der Transaktionsaktion 452 zugehörige Transaktion fortfahren und/oder in die Blockchain festgeschrieben werden kann.

[0059] Wie ferner bildlich dargestellt umfasst die Chaincode-Bewilligungsaktion 456 eine Vorschlagsantwortnutzlast 460, die einen Vorschlags-Hash 462 enthält, der den Informationen zugehörig sein kann, die zum Hinzufügen zu der Blockchain vorgeschlagen werden. Die Vorschlagsantwortnutzlast 460 kann ferner eine Chaincode-Aktion 464 enthalten, die im Einzelnen angeben kann, wie/wo die Informationen bezüglich der Transaktion in der Blockchain gespeichert werden sollten. Bei einigen Ausführungsformen umfasst die Chaincode-Aktion 464 Transaktionssimulationsergebnisse (Tx-Simulationsergebnisse) 466, die einen Transaktions-Lese-/Schreibsatz 468 enthalten. Der Transaktions-Lese-/Schreibsatz 468 enthält einen Nächster-Schritt-Lese-/Schreibsatz (Ns-Rw-Satz) 468, der einen Hash eines (a) Wertes 472 enthält, der auf den Wert 484 zeigt und diesem zugehörig ist, bei dem es sich um Informationen/Daten/Inhalte handeln kann.

[0060] Bei einigen Ausführungsformen umfasst die Chaincode-Aktion 464 ferner ein Chaincode-Ereignis 474, das einen Hash einer (a) ersten Nutzlast 476 umfasst, der auf eine erste Nutzlast 482 zeigt und dieser zugehörig ist, die sensible Informationen enthalten kann. Bei einigen Ausführungsformen umfasst die Chaincode-Aktion 464 ferner eine Antwort 478, die einen Hash einer (a) zweiten Nutzlast 480 umfasst, die auf eine zweite Nutzlast 486 zeigt und dieser zugehörig ist, bei der es sich ferner um Informationen handeln kann, dass ein Benutzer möglicherweise redigierbar sein möchte. Bei einigen Ausführungsformen können der Hash des Wertes 472, der Hash der ersten Nutzlast 476 und der Hash der zweiten Nutzlast 480 alle auf dieselben Informationen (z.B. den Wert 484, die erste Nutzlast 482 oder die zweite Nutzlast 486) zeigen und diesen zugeordnet sein.

[0061] Es wird angemerkt, dass der Wert 484, die erste Nutzlast 482 und die zweite Nutzlast 486 in einem entkoppelten Bereich der Transaktionsstruktur 400 in den Vorabbildern 488 gespeichert sind, in dem der Wert 484, die erste Nutzlast 482 und die zweite Nutzlast 486 in vollem Umfang angezeigt werden können, die aber während der Transaktion, die der Transaktionsstruktur 400 zugehörig ist, gegenüber keiner Blockchain exponiert sind.

[0062] Nunmehr unter Bezugnahme auf **Fig. 5** ist dort eine Validierung 500 einer Transaktion durch einen Nachfahren-Peer 518 gemäß Ausführungsformen der vorliegenden Offenbarung veranschaulicht. Wie bildlich dargestellt enthält ein Vorfahren-Peer 502 einen Block 504, der Transaktionen (Tx) 506A bis C (z.B. validierte Transaktionen) enthält, bei denen der Block 504 auf demselben Vorfahren-Peer 502 über eine Blockchain in Datenaustausch mit dem Vorfahren-Peer 502 gespeichert wird, der die Transaktionen 506A bis C in die Blockchain festschreibt. Bei einigen Ausführungsformen umfasst jede der Transaktionen 506A bis C jeweils Nutzlasten 508A bis C und Signaturen 510A bis C, wobei Transaktion eins 506A Vorabbilder 512 umfasst, Transaktion zwei 506B Vorabbilder 514 umfasst und Transaktion drei 506C Vorabbilder 516 umfasst. Bei einigen Ausführungsformen kann es sich bei jedem der Vorabbilder 512 bis 516 tatsächlich um Hashes der Vorabbilder 512 bis 516 handeln, oder sie können über Hashes, die auf die Vorabbilder 512 bis 516 zeigen, in die Transaktionen 506A bis C abgerufen werden, aber aus Gründen der Übersichtlichkeit sind die Hashes nicht bildlich dargestellt.

[0063] Bei einigen Ausführungsformen kann der Nachfahren-Peer 518 in die Blockchain aufgenommen werden und ein Exemplar des Blockchain-Ledger empfangen, unter anderen den Block 504, oder es kann sich bei einigen Ausführungsformen bei dem Nachfahren-Peer 518 um den Vorfahren-Peer 502 handeln, und der Block 504 kann nach einer vierten (nachfolgenden) Transaktion aktualisiert werden. In beiden Fällen werden die Vorabbilder 514 durch Nullen 520 ersetzt (zum Beispiel werden die den Vorabbildern 514 zugehörigen Werte über eine Redigiertransaktion der Vorabbilder 514 zu Nullen), wodurch sie als in den bzw. in dem Block 504 redigiert erscheinen. Das Ersetzen der Vorabbilder 514 durch die Nullen 520 ermöglicht, dass die Transaktion zwei 506B gültig bleibt. Das heißt, der Verlass auf die Hashes während des Berechnens einer Hash-Chain und das Nichtverlassen auf die Vorabbilder 514

ermöglichen eine fortgesetzte Validierung der Transaktion zwei 506B und der Hash-Chain (nach Redaktion).

[0064] Wie in dieser Offenbarung bereitgestellt und in **Fig. 5** bildlich dargestellt können Informationen aus der Blockchain redigiert werden, ohne eine Transaktion außer Kraft zu setzen und die Unversehrtheit des Block-Hash aufrechtzuerhalten (was z.B. das fortgesetzte Verketten/Verknüpfen von Blöcken ermöglicht), wodurch nach wie vor ermöglicht wird, dass die Blockchain das Vorhandensein einer Transaktion unveränderbar bestätigt. Darüber hinaus wird angemerkt, dass eine Transaktion mit einem nicht übereinstimmenden Hash/Vorabbild alle Schlüssel in ihrem Schreibsatz als zerstört markiert und ein anschließendes (z.B. Transaktion) Lesen eines zerstörten Schlüssels während der Chaincode-Ausführung zu einer sofortigen Beendigung der Transaktion führt. Normalerweise validieren Anordner Hashes und weisen Transaktionen mit nicht übereinstimmenden Hashes/Vorabbildern zurück, was erforderlich ist, um eine Verzweigung zwischen einer Transaktion mit nicht übereinstimmenden Hashes/Vorabbildern (z.B. Werten) und derselben Transaktion nach Redaktion zu verhindern. Somit stellen die hierin offenbarten Ausführungsformen eine Möglichkeit zur Redaktion bereit, bei der beliebige Schlüssel beschädigt werden oder die Verzweigungen in derselben Transaktion verursacht.

[0065] Nunmehr unter Bezugnahme auf **Fig. 6A** ist dort ein Flussdiagramm eines beispielhaften Verfahrens 600 zum Aufrechterhalten der Unversehrtheit des Block-Hash gemäß Ausführungsformen der vorliegenden Offenbarung veranschaulicht. Bei einigen Ausführungsformen kann das Verfahren 600 durch einen Prozessor (z.B. Peer/Knoten in einer Blockchain, in einem System usw.) durchgeführt werden. Bei einer Ausführungsform beginnt das Verfahren 600 bei einer Operation 602, bei der der Prozessor einen Wert und einen Hash des Wertes in einem Ledger speichert, das dem Blockchain-Netzwerk zugehörig ist.

[0066] Bei einigen Ausführungsformen geht das Verfahren 600 zu Operation 604 weiter, bei der der Prozessor aus dem Hash des Wertes einen Block-Hash aufbaut. Bei einigen Ausführungsformen geht das Verfahren 600 zu Operation 606 weiter. Bei Operation 606 validiert der Prozessor eine Transaktion durch Erkennen, dass der Hash des Wertes mit dem Hash des Wertes aus einer früheren Transaktion (oder bei einigen Ausführungsformen aus dem Aufnehmen eines neuen Peer) übereinstimmt. Bei einigen Ausführungsformen geht das Verfahren 600 zu Operation 608 weiter, bei der der Prozessor die Unversehrtheit des Block-Hash aufrechterhält (z.B. liegt keine Verzweigung zwischen einer Transaktion mit nicht übereinstimmenden Hashes derselben Transaktion nach Redaktion vor, daher werden die Transaktion nicht außer Kraft gesetzt und das fortgesetzte Verketten/verknüpfen von Blöcken ermöglicht). Bei einigen Ausführungsformen kann das Verfahren 600 enden.

[0067] Bei einigen nachstehend erörterten Ausführungsformen liegen eine oder mehrere Operationen des Verfahrens 600 vor, die der Kürze wegen nicht abgebildet sind. Dementsprechend kann bei einigen Ausführungsformen das Verfahren 600 ferner umfassen, dass der Prozessor den Wert redigiert. Bei einer derartigen Ausführungsform kann das Redigieren der Transaktion ein Setzen von Bits auf Null umfassen, die dem Wert zugehörig sind. Bei einigen Ausführungsformen kann das Verfahren 600 nach einer Redaktion des Wertes und/oder einer Reaktion von Informationen (z.B. Benutzerdaten, personenbezogene Informationen usw.) durchgeführt werden, die dem Wert zugehörig sind.

[0068] Bei einigen Ausführungsformen kann das Redigieren des Wertes ferner ein Aufrechterhalten des Hash des Wertes und des Block-Hash umfassen. Bei einigen Ausführungsformen kann das Redigieren des Wertes ferner aufweisen, dass der Prozessor den Wert durch den Hash des Wertes ersetzt. Der Prozessor kann ferner ein entkoppeltes Vorabbild des Wertes aufrechterhalten. Bei einigen Ausführungsformen kann der Hash des Wertes auf das entkoppelte Vorabbild zeigen.

[0069] Bei einigen Ausführungsformen kann das Verfahren 600 ferner aufweisen, dass der Prozessor die validierte Transaktion ohne den Wert aufrechterhält (z.B. bleibt die Transaktion gültig, da der Hash des Wertes in dem Ledger aufgezeichnet wird und unveränderlich bleibt und sich nicht ändert, selbst wenn der Wert redigiert wird). Bei einigen Ausführungsformen kann das Verfahren 600 ferner aufweisen, dass der Prozessor den Wert empfängt. Der Prozessor kann erkennen, dass der Wert Benutzerdaten zugehörig ist. Der Prozessor kann dann den Hash des Wertes erzeugen (z.B. erkennt der Prozessor, dass der Wert einer bestimmten Art von Daten/-Informationen zugehörig ist, und der Hash wird dann für den Wert und die vorgeschlagene Blockchain-Struktur erzeugt, wie in der gesamten vorliegenden Offenbarung erörtert und bildlich dargestellt).

[0070] Nunmehr unter Bezugnahme auf **Fig. 6B** ist dort ein Flussdiagramm eines beispielhaften Verfahrens 650 zum Redigieren von Informationen aus einem Blockchain-Netzwerk gemäß Ausführungsformen der vorliegenden Offenbarung veranschaulicht. Bei einigen Ausführungsformen kann das Verfahren 650 durch einen Prozessor (z.B. Peer/Knoten in einer Blockchain, in einem System usw.) durchgeführt werden. Bei einigen Ausführungsformen beginnt das Verfahren 650 bei Operation 652, bei

der der Prozessor eine Redigiertransaktion bezüglich des Blockchain-Netzwerks bereitstellt.

[0071] Bei einigen Ausführungsformen geht das Verfahren 650 zu Operation 654 weiter, bei der der Prozessor eine Redigiertransaktion einleitet. Die Redigiertransaktion kann einer validierten Transaktion zugehörig sein. Bei einigen Ausführungsformen geht das Verfahren 650 zu Operation 656 weiter. Bei Operation 656 erkennt der Prozessor einen Hash-Wert, der innerhalb der Redigiertransaktion angegeben und der validierten Transaktion zugehörig ist. Bei einigen Ausführungsformen geht das Verfahren 650 zu Operation 658 weiter, bei der der Prozessor einen Wert redigiert, der dem Hash-Wert zugehörig ist. Bei einigen Ausführungsformen kann das Verfahren 650 nach der Operation 658 enden.

[0072] Bei einigen nachstehend erörterten Ausführungsformen liegen eine oder mehrere Operationen des Verfahrens 650 vor, die der Kürze wegen nicht abgebildet sind. Dementsprechend kann bei einigen Ausführungsformen das Verfahren 650 ferner umfassen, dass der Prozessor eine Transaktionshülle erzeugt. Bei einigen Ausführungsformen kann die Transaktionshülle den Hash-Wert umfassen, der einen Zeiger auf ein Vorabbild enthält. Bei einigen Ausführungsformen kann die Transaktionshülle ferner das Vorabbild umfassen, und das Vorabbild kann den Wert umfassen.

[0073] Bei einigen kann das Redigieren des Wertes das Setzen von Bits auf Null umfassen, die dem Wert zugehörig sind. In einem derartigen Fall kann das Setzen der dem Wert zugehörigen Bits auf Null ermöglichen, dass der Hash-Wert der gleiche (z.B. unverändert) bleibt. Bei einigen Ausführungsformen kann der Verfahren 650 ferner umfassen, dass der Prozessor die validierte Transaktion auf der Grundlage des Erkennens aufrechterhält, dass der Hash-Wert nach der Redigiertransaktion der gleiche ist.

[0074] Es versteht sich, dass diese Offenbarung zwar eine ausführliche Beschreibung des Cloud-Computing enthält, die Realisierung der hierin vorgestellten Lehren jedoch nicht auf eine Cloud-Computing-Umgebung beschränkt ist. Vielmehr können Ausführungsformen der vorliegenden Offenbarung in Verbindung mit einer beliebigen anderen Art von Datenverarbeitungsumgebung realisiert werden, die gegenwärtig bekannt ist oder in Zukunft entwickelt wird.

[0075] Bei Cloud-Computing handelt es sich um ein Modell zur Bereitstellung von Diensten, um einen praktischen und bei Bedarf verfügbaren Netzwerkzugriff auf ein gemeinsam genutztes Reservoir konfigurierbarer Datenverarbeitungsressourcen (z.B. Netzwerke, Netzwerkbandbreite, Server, Verarbeitung, Hauptspeicher, Speicher, Anwendungen, virtuelle Maschinen und Dienste) zu ermöglichen, die bei minimalem Verwaltungsaufwand oder minimaler Interaktion mit einem Anbieter des Dienstes schnell bereitgestellt und freigegeben werden können. Dieses Cloud-Modell kann mindestens fünf Eigenschaften, mindestens drei Dienstmodelle und mindestens vier Bereitstellungsmodelle umfassen.

Die Eigenschaften sind folgende:

[0076] On-demand Self Service (Selbstzuweisung bei Bedarf): Ein Cloud-Kunde kann sich einseitig Datenverarbeitungsfunktionen wie zum Beispiel Serverzeit und Netzwerkspeicher dem Bedarf entsprechend automatisch bereitstellen ohne dass eine Interaktion von Menschen mit dem Anbieter des Dienstes erforderlich ist.

[0077] Broad Network Access (umfassender Netzwerkzugriff): Es stehen Funktionen über ein Netzwerk zur Verfügung, auf die der Zugriff über Standardmechanismen erfolgt, die die Verwendung durch heterogene Plattformen Thin- oder Thick-Client-Plattformen (z.B. Mobiltelefone, Notebook-Computer und PDAs) unterstützen.

[0078] Resource Pooling (Ressourcenbündelung): Die Datenverarbeitungsressourcen des Anbieters werden gebündelt, um mehrere Kunden unter Verwendung eines Mehrfachnutzermodells mit unterschiedlichen physischen und virtuellen Ressourcen zu bedienen, die entsprechend dem Bedarf dynamisch zugewiesen und neu zugewiesen werden. Es besteht eine gefühlte Abschnittunabhängigkeit in der Weise, dass der Kunde im Allgemeinen keine Kontrolle oder Kenntnis über den exakten Abschnitt der bereitgestellten Ressourcen hat, aber möglicherweise in der Lage ist, den Abschnitt auf einer höheren Abstraktionsebene (z.B. Land, Bundesstaat oder Datenzentrum) anzugeben.

[0079] Rapid Elasticity (rasche Elastizität): Funktionen können rasch und elastisch bereitgestellt werden, in einigen Fällen automatisch, um den Funktionsumfang schnell nach oben anzupassen, und schnell freigegeben werden, um den Funktionsumfang schnell nach unten anzupassen. Für den Kunden entsteht oftmals der Eindruck, dass die zum Bereitstellen verfügbaren Funktionen unbegrenzt sind und jederzeit in jeder beliebigen Menge gekauft werden können.

[0080] Measured Service (bemessener Dienst): Cloud-Systeme steuern und optimieren automatisch die Ressourcenverwendung durch Nutzung einer Bemessungsfunktion auf einer bestimmten Abstraktionsebene, die für die Art des Dienstes geeignet ist (z.B. Speicher, Verarbeitung, Bandbreite und aktive Benutzerkonten). Die Ressourcennutzung kann überwacht, gesteuert und gemeldet werden, sodass

Transparenz sowohl für den Anbieter als auch den Kunden des genutzten Dienstes besteht.

Die Dienstmodelle sind folgende:

[0081] Software as a Service (SaaS) (Software als Dienst): Die dem Kunden bereitgestellte Funktion besteht darin, die auf einer Cloud-Infrastruktur ausgeführten Anwendungen des Anbieters zu nutzen. Die Anwendungen sind von verschiedenen Client-Einheiten aus über eine Thin-Client-Schnittstelle wie z.B. über einen Web-Browser (z.B. auf dem Web beruhende eMail) zugänglich. Der Kunde verwaltet oder steuert die zugrunde liegende Cloud-Infrastruktur, unter anderem Netzwerke, Server, Betriebssysteme, Speicherplatz oder sogar einzelne Anwendungsfähigkeiten, nicht, abgesehen von der möglichen Ausnahme eingeschränkter benutzerspezifischer Konfigurationseinstellungen von Anwendungen.

[0082] Platform as a Service (PaaS) (Plattform als Dienst): Die dem Kunden bereitgestellte Funktion besteht darin, auf der Cloud-Infrastruktur vom Kunden erzeugte oder erworbene Anwendungen bereitzustellen, die unter Verwendung von Programmiersprachen und Programmierwerkzeugen erzeugt wurden, die durch den Anbieter unterstützt werden. Der Kunde verwaltet oder steuert die zugrunde liegende Cloud-Infrastruktur, unter anderem Netzwerke, Server, Betriebssysteme oder Speicherplatz, nicht, hat aber die Kontrolle über die bereitgestellten Anwendungen und möglicherweise über Konfigurationen der Hosting-Umgebung der Anwendungen.

[0083] Infrastructure as a Service (IaaS) (Infrastruktur als Dienst): Die dem Kunden bereitgestellte Funktion besteht darin, Verarbeitung, Speicherplatz, Netzwerke und andere grundlegende Datenverarbeitungsressourcen bereitzustellen, wobei der Kunde beliebige Software bereitstellen und ausführen kann, zu der Betriebssysteme und Anwendungen gehören können. Der Kunde verwaltet oder steuert die zugrunde liegende Cloud-Infrastruktur nicht, hat aber die Kontrolle über Betriebssysteme, Speicherplatz sowie bereitgestellte Anwendungen und möglicherweise eine eingeschränkte Kontrolle über ausgewählte Vernetzungskomponenten (z.B. Host-Firewalls).

Die Bereitstellungsmodelle sind folgende:

[0084] Private Cloud (private Cloud): Die Cloud-Infrastruktur wird ausschließlich für eine Organisation betrieben. Sie kann durch die Organisation oder einen Dritten verwaltet werden und in den Räumen der Organisation oder außerhalb davon vorhanden sein.

[0085] Community Cloud (Community-Cloud): Die Cloud-Infrastruktur wird von mehreren Organisationen genutzt und unterstützt eine bestimmte Benutzergemeinschaft, die gemeinsame Interessen hat (z.B. Gesichtspunkte im Zusammenhang mit einer Aufgabe, mit Sicherheitsanforderungen, Richtlinien und mit der Einhaltung von Gesetzen und Vorschriften). Sie kann durch die Organisationen oder einen Dritten verwaltet werden und in den Räumen der Organisation oder außerhalb davon vorhanden sein.

[0086] Public Cloud (öffentliche Cloud): Die Cloud-Infrastruktur wird der allgemeinen Öffentlichkeit oder einer großen Gruppe in einem Industriezweig zur Verfügung gestellt und ist Eigentum einer Organisation, die Cloud-Dienste verkauft.

[0087] Hybrid cloud (Hybrid-Cloud): Bei der Cloud-Infrastruktur handelt es sich um eine Mischung aus zwei oder mehreren Clouds (Private Cloud, Community Cloud oder Public Cloud), die eigenständige Entitäten bleiben, aber über eine standardisierte oder proprietäre Technologie miteinander verbunden sind, die die Portierbarkeit von Daten und Anwendungen ermöglicht (z.B. Cloud-Zielgruppenverteilung (Cloud Bursting) zum Lastausgleich zwischen Clouds).

[0088] Eine Cloud-Computing-Umgebung ist dienstorientiert, wobei der Schwerpunkt auf Zustandsunabhängigkeit, geringer Kopplung, Modularität und semantischer Interoperabilität liegt. Im Mittelpunkt des Cloud-Computing steht eine Infrastruktur, die ein Netzwerk aus untereinander verbundenen Knoten umfasst.

[0089] In **Fig. 7A** ist eine Cloud-Cloud-Computing-Umgebung 710 veranschaulicht. Wie gezeigt umfasst die Cloud-Computing-Umgebung 710 einen oder mehrere Cloud-Computing-Knoten 700, mit denen von Cloud-Kunden verwendete lokale Datenverarbeitungseinheiten wie z.B. ein Personal Digital Assistent (PDA) oder ein Mobiltelefon 700A, ein Desktop-Computer 700B, ein Notebook-Computer 700C und/oder ein Automobil-Computersystem 700N Daten austauschen können. Die Knoten 700 können untereinander Daten austauschen. Sie können physisch oder virtuell in einem oder mehreren Netzwerken wie zum Beispiel in einer hierin oben beschriebenen Private Cloud, Community Cloud, Public Cloud oder Hybrid Cloud oder in einer Kombination davon gruppiert sein (nicht gezeigt).

[0090] Dies ermöglicht der Cloud-Computing-Umgebung 710, Infrastruktur, Plattformen und/oder Software als Dienste zu bieten, für die ein Cloud-Kunde keine Ressourcen auf einer lokalen Datenverarbeitungseinheit zu verwalten braucht. Es versteht sich, dass die Arten von in **Fig. 7A** gezeigten Datenverarbeitungseinheiten 700A bis N lediglich veran-

schaulichend sein sollen und dass die Datenverarbeitungsknoten 700 und die Cloud-Computing-Umgebung 710 über eine beliebige Art von Netzwerk und/oder über eine beliebige Art von Verbindung, die über ein Netzwerk aufgerufen werden kann (z.B. unter Verwendung eines Webbrowsers), mit einer beliebigen Art von computergestützter Einheit Daten austauschen können.

[0091] In **Fig. 7B** ist ein Satz funktionaler Abstraktionsschichten veranschaulicht, die durch die Cloud-Computing-Umgebung 710 (**Fig. 7A)** bereitgestellt werden. Es sollte von vornherein klar sein, dass die in **Fig. 7B** gezeigten Komponenten, Schichten und Funktionen lediglich veranschaulichend sein sollen und Ausführungsformen der Offenbarung nicht darauf beschränkt sind. Wie nachstehend abgebildet werden die folgenden Schichten und entsprechenden Funktionen bereitgestellt.

[0092] Eine Hardware- und Softwareschicht 715 umfasst Hardware- und Softwarekomponenten. Zu Beispielen von Hardware-Komponenten gehören: Großrechner 702; auf der RISC-Architektur (RISC = Reduced Instruction Set Computer) beruhende Server 704; Server 706; Blade-Server 708; Speichereinheiten 711; und Netzwerke und Vernetzungskomponenten 712. Bei einigen Ausführungsformen umfassen Softwarekomponenten eine Netzwerk-Anwendungsserversoftware 714 und eine Datenbanksoftware 716.

[0093] Eine Virtualisierungsschicht 720 stellt eine Abstraktionsschicht bereit, von der aus die folgenden Beispiele von virtuellen Einheiten bereitgestellt sein können: virtuelle Server 722; virtueller Speicher 724; virtuelle Netzwerke 726, unter anderem virtuelle private Netzwerke; virtuelle Anwendungen und Betriebssysteme 728; und virtuelle Clients 730.

[0094] Bei einem Beispiel kann eine Verwaltungsschicht 740 die nachstehend beschriebenen Funktionen bereitstellen. Eine Ressourcenbereitstellung 742 stellt eine dynamische Beschaffung von Datenverarbeitungsressourcen sowie anderen Ressourcen bereit, die genutzt werden, um Aufgaben innerhalb der Cloud-Computing-Umgebung durchzuführen. Eine Gebührenerfassung und Preisberechnung 744 stellt eine Kostenverfolgung bereit, während Ressourcen innerhalb der Cloud-Computing-Umgebung genutzt werden, sowie eine Abrechnung und Fakturierung der Inanspruchnahme dieser Ressourcen. Bei einem Beispiel können diese Ressourcen Anwendungssoftwarelizenzen umfassen. Eine Sicherheit stellt eine Identitätsüberprüfung für Cloud-Kunden und Aufgaben sowie einen Schutz für Daten und andere Ressourcen bereit. Ein Benutzerportal 746 stellt Kunden und Systemadministratoren einen Zugang zur Cloud-Computing-Umgebung bereit. Eine Dienstgüteverwaltung (Service Level Management) 748 stellt eine Zuordnung und Verwaltung von Cloud-Computing-Ressourcen bereit, sodass die jeweils erforderliche Dienstgüte erreicht wird. Eine Planung und Erfüllung von Dienstgütevereinbarungen 750 (Service Level Agreement (SLA), Servicevertrag) stellt die Vorausplanung für und die Beschaffung von Cloud-Computing-Ressourcen bereit, für die auf der Grundlage eines SLA zukünftige Anforderungen erwartet werden.

[0095] Eine Betriebslastschicht 760 stellt Beispiele von Funktionalitäten bereit, für die die Cloud-Computing-Umgebung genutzt werden kann. Zu Beispielen von Betriebslasten und Funktionen, die von dieser Schicht aus bereitgestellt werden können, gehören: Zuordnung und Navigation 762; Softwareentwicklung und -verwaltung 764 während des Lebenszyklus; Bereitstellung 766 von Schulungen in virtuellen Schulungsräumen; Verarbeitung 768 von Datenanalysen; Transaktionsverarbeitung 770; und Redigiertransaktionsverarbeitung 772.

[0096] In **Fig. 8** ist ein Übersichtsblockschema eines beispielhaften Computersystems 801 veranschaulicht, das beim Realisieren eines oder mehrerer der hierin beschriebenen Verfahren, Werkzeuge und Module und beliebiger verwandter Funktionen (z.B. unter Verwendung einer oder mehrerer Prozessorschaltungen oder Computerprozessoren des Computers) gemäß Ausführungsformen der vorliegenden Offenbarung verwendet werden kann. Bei einigen Ausführungsformen können die Hauptkomponenten des Computersystems 801 eine oder mehrere CPUs 802, ein Hauptspeicher-Teilsystem 804, eine Endgeräteschnittstelle 812, eine Speicherschnittstelle 816, eine E/A-Einheitenschnittstelle (Eingabe/Ausgabe-Einheitenschnittstelle) 814 und eine Netzwerkschnittstelle 818 aufweisen, die alle zu Datenaustauschzwecken direkt oder indirekt zum Datenaustausch zwischen Komponenten über einen Hauptspeicherbus 803, einen E/A-Bus 808 und eine E/A-Busschnittstelleneinheit 810 verbunden sein können.

[0097] Das Computersystem 801 kann eine oder mehrere programmierbare Universal-Zentraleinheiten (CPUs) 802A, 802B, 802C und 802D enthalten, die hierin allgemein als „CPU 802“ bezeichnet werden. Bei einigen Ausführungsformen kann das Computersystem 801 mehrere Prozessoren enthalten, die für ein relativ großes System typisch sind; bei anderen Ausführungsformen kann es sich bei dem Computersystem 801 jedoch um nur einen Prozessor mit nur einer CPU handeln. Jede CPU 802 kann Anweisungen ausführen, die im Hauptspeicher-Teilsystem 804 gespeichert sind, und kann eine oder mehrere Ebenen von Onboard-Cache enthalten.

[0098] Der Systemhauptspeicher 804 kann durch ein Computersystem lesbare Medien in Form von

flüchtigem Speicher wie z.B. Direktzugriffsspeicher (Random Access Memory, RAM) 822 oder Cache-Speicher 824 enthalten. Das Computersystem 801 kann ferner andere wechselbare/nichtwechselbare, flüchtige/nichtflüchtige Computersystemspeichermedien enthalten. Lediglich beispielhaft kann ein Speichersystem 826 zum Lesen von einem und Schreiben auf ein nichtwechselbares, nichtflüchtiges magnetisches Medium wie z.B. eine „Festplatte“ bereitgestellt sein. Obwohl nicht gezeigt, können ein Magnetplattenlaufwerk zum Lesen von einer und Schreiben auf eine wechselbare nichtflüchtige Magnetplatte (z.B. eine „Diskette“) oder optisches Plattenlaufwerk zum Lesen von einer oder Schreiben auf eine wechselbare nichtflüchtige optische Platte wie z.B. eine CD-ROM, eine DVD-ROM oder ein anderes optisches Medium bereitgestellt sein. Darüber hinaus kann der Hauptspeicher 804 Flash-Speicher enthalten, z.B. einen Flash-Speicherstick oder ein Flash-Laufwerk. Hauptspeichereinheiten können durch eine oder mehrere Datenmedienschnittstellen mit dem Hauptspeicherbus 803 verbunden sein. Der Hauptspeicher 804 kann mindestens ein Programmprodukt enthalten, das einen Satz (z.B. mindestens einen) von Programmmodulen umfasst, die so konfiguriert sind, dass die Funktionen verschiedener Ausführungsformen ausgeführt werden.

[0099] Ein oder mehrere Programme/Dienstprogramme 828, von denen jedes mindestens einen Satz von Programmmodulen 830 enthält, können im Hauptspeicher 804 gespeichert sein. Die Programme/Dienstprogramme 828 können einen Hypervisor (der auch als „Virtual Machine Monitor“ bezeichnet wird), ein oder mehrere Betriebssysteme, ein oder mehrere Anwendungsprogramme, andere Programmmodule und Programmdaten enthalten. Jedes der Betriebssysteme, ein oder mehrere Anwendungsprogramme, andere Programmmodule und Programmdaten oder eine bestimmte Kombination davon kann bzw. können eine Realisierungsform einer Vernetzungsumgebung umfassen. Programme 828 und/oder Programmmodule 830 führen allgemein die Funktionen oder Methodiken verschiedener Ausführungsformen durch.

[0100] Zwar ist der Hauptspeicherbus 803 in **Fig. 8** als einzelne Busstruktur gezeigt, die einen direkten Datenübertragungspfad zwischen den CPUs 802, dem Hauptspeicher-Teilsystem 804 und der E/A-Busschnittstelle 810 bereitstellt, der Hauptspeicherbus 803 kann jedoch bei einigen Ausführungsformen mehrere unterschiedliche Busse oder Datenübertragungspfade enthalten, die in einer beliebigen von verschiedenen Formen wie z.B. als Punkt-zu-Punkt-Verknüpfungen in hierarchischen, sternförmigen oder netzförmigen Konfigurationen, als mehrere hierarchische Busse, parallele und redundante Pfade oder in einer beliebigen anderen geeigneten Art von Konfiguration angeordnet sein können. Des Weiteren sind zwar die E/A-Busschnittstelle 810 und der E/A-Bus 808 als einzelne jeweilige Einheiten gezeigt, das Computersystem 801 kann jedoch bei einigen Ausführungsformen mehrere E/A-Busschnittstelleneinheiten 810, mehrere E/A-Busse 808 oder beides enthalten. Ferner sind zwar sind mehrere E/A-Schnittstelleneinheiten gezeigt, die den E/A-Bus 808 von verschiedenen Datenübertragungspfaden trennen, die zu den verschiedenen E/A-Einheiten führen, bei anderen Ausführungsformen können jedoch einige oder alle der E/A-Einheiten direkt mit einem oder mehreren System-E/A-Bussen verbunden sein.

[0101] Bei einigen Ausführungsformen kann es sich bei dem Computersystem 801 um ein Großrechner-Mehrbenutzercomputersystem, ein Einzelbenutzersystem oder einen Server-Computer oder eine ähnliche Einheit handeln, das/der eine Benutzeroberfläche mit geringem Funktionsumfang oder keine Benutzeroberfläche enthält, aber Anforderungen von anderen Computersystemen (Clients) empfängt. Ferner kann das Computersystem 801 bei einigen Ausführungsformen als Desktop-Computer, transportabler Computer, Laptop- oder Notebook-Computer, Tablet-Computer, Pocket-Computer, Telefon, Smartphone, Netzwerk-Switches oder -Router oder als beliebige andere geeignete Art von elektronischer Einheit realisiert sein.

[0102] Es wird angemerkt, dass **Fig. 8** die repräsentativen Hauptkomponenten eines beispielhaften Computersystems 801 darstellen soll. Bei einigen Ausführungsformen können einzelne Komponenten jedoch eine höhere oder niedrigere Komplexität haben als in **Fig. 8** wiedergegeben, andere Komponenten als die in **Fig. 8** gezeigten oder zusätzliche Komponenten können vorhanden sein, und die Konfiguration derartiger Komponenten kann variieren.

[0103] Wie hierin ausführlicher erörtert ist es vorstellbar, dass einige oder alle Operationen einiger Ausführungsformen hierin beschriebener Verfahren in alternativen Reihenfolgen durchgeführt werden können oder überhaupt nicht durchgeführt werden können; des Weiteren können mehrere Operationen gleichzeitig oder als interner Teil eines größeren Prozesses stattfinden.

[0104] Bei der vorliegenden Offenbarung kann es sich um ein System, ein Verfahren und/oder ein Computerprogrammprodukt in einem beliebigen möglichen Integrationsgrad technischer Einzelheiten handeln. Das Computerprogrammprodukt kann (ein) durch einen Computer lesbare(s) Speichermedium (oder -medien) enthalten, auf dem/denen durch einen Computer lesbare Programmanweisungen gespeichert ist/sind, um einen Prozessor zu veranlassen, Aspekte der vorliegenden Offenbarung auszuführen.

[0105] Bei dem durch einen Computer lesbaren Speichermedium kann es sich um eine materielle Einheit handeln, auf der Anweisungen zur Verwendung durch eine Einheit zur Ausführung von Anweisungen aufbewahrt und gespeichert sein können. Bei dem durch einen Computer lesbaren Speichermedium kann es sich zum Beispiel, ohne auf diese beschränkt zu sein, um eine elektronische Speichereinheit, eine magnetische Speichereinheit, eine optische Speichereinheit, eine elektromagnetische Speichereinheit, eine Halbleiterspeichereinheit oder eine beliebige geeignete Kombination des Vorstehenden handeln. Eine nicht erschöpfende Liste genauerer Beispiele des durch einen Computer lesbaren Speichermediums umfasst Folgendes: eine transportable Computerdiskette, eine Festplatte, einen Direktzugriffsspeicher (RAM), einen Nur-Lese-Speicher (ROM), einen löschbaren programmierbaren Nur-Lese-Speicher (EPROM oder Flash-Speicher), einen statischen Direktzugriffsspeicher (SRAM), einen transportablen Nur-Lese-Speicher in Form einer Compact Disc (CD-ROM), eine Digital Versatile Disc (DVD), einen Speicherstick, eine Diskette, eine mechanisch codierte Einheit wie zum Beispiel Lochkarten oder erhöhte Strukturen in einer Rille mit darauf aufgezeichneten Anweisungen oder beliebige geeignete Kombinationen des Vorstehenden. Ein durch einen Computer lesbares Speichermedium im hierin verwendeten Sinne ist nicht so auszulegen, dass es sich dabei um flüchtige Signale an sich handelt, beispielsweise um Funkwellen oder sich frei ausbreitende elektromagnetische Wellen, um elektromagnetische Wellen, die sich durch einen Hohlleiter oder andere Übertragungsmedien ausbreiten (z.B. ein Lichtwellenleiterkabel durchlaufende Lichtimpulse) oder um elektrische Signale, die über ein Kabel übertragen werden.

[0106] Hierin beschriebene, durch einen Computer lesbare Programmanweisungen können über ein Netzwerk, zum Beispiel das Internet, ein lokales Netzwerk ein Weitverkehrsnetzwerk und/oder ein kabelloses Netzwerk von einem durch einen Computer lesbaren Speichermedium auf betreffende Datenverarbeitungs-/Verarbeitungseinheiten oder auf einen externen Computer oder eine externe Speichereinheit heruntergeladen werden. Das Netzwerk kann Kupferübertragungskabel, Lichtwellenübertragungsleiter, kabellose Übertragung, Router, Firewalls, Switches, Gateway-Computer und/oder Edge-Server aufweisen. Eine Netzwerkadapterkarte oder eine Netzwerkschnittstelle in der Datenverarbeitungs-/Verarbeitungseinheit empfängt durch einen Computer lesbare Programmanweisungen aus dem Netzwerk und leitet die durch einen Computer lesbaren Programmanweisungen zur Speicherung in einem durch einen Computer lesbaren Speichermedium innerhalb der jeweiligen Datenverarbeitungs-/Verarbeitungseinheit weiter.

[0107] Bei durch einen Computer lesbaren Programmanweisungen zum Ausführen von Operationen der vorliegenden Offenbarung kann es sich um Assembler-Anweisungen, ISA-Anweisungen (ISA = Instruction-Set-Architecture), Maschinenanweisungen, maschinenabhängige Anweisungen, Mikrocode, Firmware-Anweisungen, zustandssetzende Daten oder entweder Quellcode oder Objektcode handeln, die in einer beliebigen Kombination aus einer oder mehreren Programmiersprachen geschrieben sind, unter anderem objektorientierte Programmiersprachen wie z.B. Smalltalk, C++ oder dergleichen sowie prozedurale Programmiersprachen wie z.B. die Programmiersprache „C“ oder ähnliche Programmiersprachen. Die durch einen Computer lesbaren Programmanweisungen können vollständig auf dem Computer des Benutzers, teilweise auf dem Computer des Benutzers, als eigenständiges Softwarepaket, teilweise auf dem Computer des Benutzers und teilweise auf einem entfernt angeordneten Computer oder vollständig auf dem entfernt angeordneten Computer oder Server ausgeführt werden. Bei dem letztgenannten Szenario kann der entfernt angeordnete Computer mit dem Computer des Benutzers über eine beliebige Art von Netzwerk verbunden sein, unter anderem über ein lokales Netzwerk (Local Area Network, LAN) oder über ein Weitverkehrsnetzwerk (Wide Area Network, WAN), oder die Verbindung kann zu einem externen Computer hergestellt werden (beispielsweise über das Internet unter Nutzung eines Internetdienstanbieters (Internet Service Provider)). Bei einigen Ausführungsformen können elektronische Schaltungen, zu denen zum Beispiel programmierbare Logikschaltungen, vor Ort programmierbare Schaltungen (Field-Programmable Gate Arrays, FPGA) oder programmierbare Logik-Arrays (PLA) gehören, die durch einen Computer lesbaren Programmanweisungen ausführen, indem Zustandsinformationen der durch einen Computer lesbaren Programmanweisungen genutzt werden, um die elektronische Schaltung zu personalisieren, sodass Aspekte der vorliegenden Offenbarung durchgeführt werden.

[0108] Hierin sind Aspekte der vorliegenden Offenbarung unter Bezugnahme auf Flussdiagramme und/oder Blockschemata von Verfahren, Vorrichtungen (Systemen) und Computerprogrammprodukten gemäß Ausführungsformen der Offenbarung beschrieben. Es wird klar sein, dass jeder Block der Flussdiagramme und/oder der Blockschemata und Kombinationen von Blöcken in den Flussdiagrammen und/oder Blockschemata mit Hilfe von durch einen Computer lesbaren Programmanweisungen realisiert werden kann bzw. können.

[0109] Diese Computerprogrammanweisungen können einem Prozessor eines Computers oder anderer programmierbarer Datenverarbeitungsvorrichtungen bereitgestellt werden, um eine Maschine

zu erzeugen, sodass die Anweisungen, die über den Prozessor des Computers oder anderer programmierbarer Datenverarbeitungsvorrichtungen ausgeführt werden, Mittel schaffen, um die in einem Block bzw. in den Blöcken des Flussdiagramms bzw. der Flussdiagramme und/oder des Blockschemas bzw. der Blockschemata angegebenen Funktionen/Aktionen zu realisieren. Diese durch einen Computer lesbaren Programmanweisungen können ebenfalls in einem durch einen Computer lesbaren Medium gespeichert sein, das einen Computer, andere programmierbare Datenverarbeitungsvorrichtungen oder andere Einheiten anweisen kann, in einer bestimmten Weise zu funktionieren, sodass das durch einen Computer lesbare Medium mit darauf gespeicherten Anweisungen ein Erzeugnis aufweist, das Anweisungen enthält, die die in einem Block bzw. in den Blöcken der Flussdiagramme und/oder der Blockschemata angegebene Funktion/Aktion realisieren.

[0110] Die durch einen Computer lesbaren Programmanweisungen können auch in einen Computer, in andere programmierbare Datenverarbeitungsvorrichtungen oder in andere Einheiten geladen werden, um zu bewirken, dass auf dem Computer, auf anderen programmierbaren Vorrichtungen oder anderen Einheiten eine Reihe von Operationen ausgeführt wird, um einen mittels Computer realisierten Prozess zu schaffen, sodass die Anweisungen, die auf dem Computer, auf anderen programmierbaren Vorrichtungen oder Einheiten ausgeführt werden, die in einem Block bzw. in den Blöcken der Flussdiagramme und/oder der Blockschemata angegebenen Funktionen/Aktionen realisieren.

[0111] Die Flussdiagramme und Blockschemata in den Figuren veranschaulichen die Architektur, Funktionalität und Wirkungsweise möglicher Realisierungsformen von Systemen, Verfahren, und Computerprogrammprodukten gemäß den verschiedenen Ausführungsformen der vorliegenden Offenbarung. In diesem Zusammenhang kann jeder Block in den Flussdiagrammen bzw. in den Blockschemata ein Modul, ein Segment oder einen Abschnitt von Anweisungen darstellen, das bzw. der eine oder mehrere ausführbare Anweisungen zum Realisieren der angegebenen Logikfunktion bzw. Logikfunktionen aufweist. Bei einigen alternativen Realisierungsformen können die in dem Block angegebenen Funktionen in einer anderen als in der Reihenfolge ausgeführt werden, die in den Figuren angegeben ist. Zum Beispiel können zwei hintereinander gezeigte Blöcke tatsächlich als ein Schritt ausgeführt, gleichzeitig, Wesentlichen gleichzeitig, in einer teilweise oder vollständig zeitlich überlappenden Weise ausgeführt werden, oder die Blöcke können je nach der mit den Blöcken verbundenen Funktionalität manchmal in umgekehrter Reihenfolge ausgeführt werden. Darüber hinaus ist anzumerken, dass jeder Block der Blockschemata und/oder Flussdiagrammdarstellungen sowie Kombinationen von Blöcken in den Blockschemata und/oder Flussdiagrammdarstellungen mit Hilfe zweckgebundener hardwaregestützter Systeme zum Ausführen der angegebenen Funktionen bzw. Aktionen oder mit Hilfe von Kombinationen aus zweckgebundener Hardware und zweckgebundenen Computeranweisungen realisiert werden kann bzw. können.

[0112] Die Beschreibungen der verschiedenen Ausführungsformen der vorliegenden Offenbarung sollen der Veranschaulichung dienen, sind jedoch nicht als vollständig oder auf die offenbarten Ausführungsformen beschränkt gedacht. Für Fachleute sind viele Modifikationen und Variationen denkbar, ohne dass diese eine Abweichung vom Schutzumfang und Grundgedanken der beschriebenen Ausführungsformen darstellen würden. Die hierin verwendete Terminologie wurde gewählt, um die Grundgedanken der Ausführungsformen, die praktische Anwendung bzw. die technische Verbesserung gegenüber den auf dem Markt vorgefundenen Technologien zu erläutern bzw. anderen mit entsprechenden Fachkenntnissen das Verständnis der hierin offenbarten Ausführungsformen zu ermöglichen.

[0113] Zwar wurde die vorliegende Offenbarung im Hinblick auf bestimmte Ausführungsformen beschrieben, es wird jedoch erwartet, dass Abwandlungen und Modifikationen daran für den Fachmann offensichtlich werden. Es ist daher beabsichtigt, dass die folgenden Ansprüche so zu interpretieren sind, dass alle derartigen Abwandlungen und Modifikationen als unter den tatsächlichen Grundgedanken und Schutzumfang der Offenbarung fallend zu betrachten sind.

Patentansprüche

1. Mittels Computer realisiertes Verfahren, wobei das Verfahren aufweist:
Speichern eines Wertes und eines Hash des Wertes in einem Ledger, das einem Blockchain-Netzwerk zugehörig ist;
Aufbauen eines Block-Hash aus dem Hash des Wertes;
Validieren einer Transaktion durch Erkennen, dass der Hash des Wertes mit dem Hash des Wertes aus einer früheren Transaktion übereinstimmt; und
Aufrechterhalten einer Unversehrtheit des Block-Hash.

2. Verfahren nach Anspruch 1, das ferner aufweist:
Redigieren des Wertes, wobei das Redigieren ein Setzen von Bits auf Null aufweist, die dem Wert zugehörig sind.

3. Verfahren nach Anspruch 2, wobei das Redigieren des Wertes ferner ein Aufrechterhalten des Hash des Wertes und des Block-Hash aufweist.

4. Verfahren nach Anspruch 3, wobei das Redigieren des Wertes ferner aufweist:
Ersetzen des Wertes durch den Hash des Wertes; und
Aufrechterhalten eines entkoppelten Vorabbilds des Wertes.

5. Verfahren nach Anspruch 4, wobei der Hash des Wertes auf das entkoppelte Vorabbild zeigt.

6. Verfahren nach Anspruch 1, das ferner aufweist:
Aufrechterhalten der validierten Transaktion ohne den Wert.

7. Verfahren nach Anspruch 1, das ferner aufweist:
Empfangen des Wertes;
Erkennen, dass der Wert Benutzerdaten zugehörig ist; und
Erzeugen des Hash des Wertes.

8. System, wobei das System aufweist:
einen Hauptspeicher; und
einen Prozessor in Datenaustausch mit dem Hauptspeicher, wobei der Prozessor so konfiguriert ist, dass Operationen durchgeführt werden, die aufweisen:
Speichern eines Wertes und eines Hash des Wertes in einem Ledger, das einem Blockchain-Netzwerk zugehörig ist;
Aufbauen eines Block-Hash aus dem Hash des Wertes;
Validieren einer Transaktion durch Erkennen, dass der Hash des Wertes mit dem Hash des Wertes aus einer früheren Transaktion übereinstimmt; und
Aufrechterhalten der Unversehrtheit des Block-Hash.

9. System nach Anspruch 8, wobei die Operationen ferner aufweisen:
Redigieren des Wertes, wobei das Redigieren das Setzen von Bits auf Null aufweist, die dem Wert zugehörig sind.

10. System nach Anspruch 9, wobei das Redigieren des Wertes ferner das Aufrechterhalten des Hash des Wertes und des Block-Hash aufweist.

11. System nach Anspruch 10, wobei das Redigieren des Wertes ferner aufweist:
Ersetzen des Wertes durch den Hash des Wertes; und
Aufrechterhalten eines entkoppelten Vorabbilds des Wertes.

12. System nach Anspruch 11, wobei der Hash des Wertes auf das entkoppelte Vorabbild zeigt.

13. Computerprogrammprodukt, wobei das Computerprogrammprodukt ein durch einen Computer lesbares Speichermedium aufweist, das darauf verkörperte Programmanweisungen enthält, wobei die Programmanweisungen durch einen Prozessor ausführbar sind, um zu bewirken, dass die Prozessoren eine Funktion ausführen, wobei die Funktion aufweist:
Speichern eines Wertes und eines Hash des Wertes in einem Ledger, das einem Blockchain-Netzwerk zugehörig ist;
Aufbauen eines Block-Hash aus dem Hash des Wertes;
Validieren einer Transaktion durch Erkennen, dass der Hash des Wertes mit dem Hash des Wertes aus einer früheren Transaktion übereinstimmt; und
Aufrechterhalten der Unversehrtheit des Block-Hash.

14. Computerprogrammprodukt nach Anspruch 13, wobei die Funktionen ferner aufweisen:
Redigieren des Wertes, wobei das Redigieren das Setzen von Bits auf Null aufweist, die dem Wert zugehörig sind.

15. Computerprogrammprodukt nach Anspruch 14, wobei das Redigieren des Wertes ferner das Aufrechterhalten des Hash des Wertes und des Block-Hash aufweist.

16. Computerprogrammprodukt nach Anspruch 15, wobei Redigieren der Transaktion ferner aufweist:
Ersetzen des Wertes durch den Hash des Wertes, wobei der Hash des Wertes auf das entkoppelte Vorabbild zeigt; und
Aufrechterhalten eines entkoppelten Vorabbilds des Wertes.

17. Verfahren zum Redigieren von Informationen aus einem Blockchain-Netzwerk, wobei das Verfahren aufweist:
Bereitstellen einer Redigiertransaktion für das Blockchain-Netzwerk;
Einleiten der Redigiertransaktion, wobei die Redigiertransaktion einer validierten Transaktion zugehörig ist;
Erkennen eines Hash-Wertes, der innerhalb der Redigiertransaktion angegeben und der validierten Transaktion zugehörig ist; und
Redigieren eines Wertes, der dem Hash-Wert zugehörig ist.

18. Verfahren nach Anspruch 17, das ferner aufweist:
Erzeugen einer Transaktionshülle, wobei die Trans-

aktionshülle den Hash-Wert aufweist, der einen Zeiger auf ein Vorabbild aufweist.

19. Verfahren nach Anspruch 18, wobei die Transaktionshülle ferner ein Vorabbild aufweist und wobei das Vorabbild den Wert aufweist.

20. Verfahren nach Anspruch 17, wobei das Redigieren des Wertes das Setzen von dem Wert zugehörigen Bits auf Null umfasst und wobei das Setzen der dem Wert zugehörigen Bits auf Null ermöglicht, dass der Hash-Wert der gleiche bleibt.

21. Verfahren nach Anspruch 20, das ferner aufweist:
Aufrechterhalten der validierten Transaktion auf der Grundlage des Erkennens, dass der Hash-Wert nach der Redigiertransaktion der gleiche ist.

22. System zum Redigieren von Informationen aus einem Blockchain-Netzwerk, wobei das System aufweist:
einen Hauptspeicher; und
einen Prozessor in Datenaustausch mit dem Hauptspeicher, wobei der Prozessor so konfiguriert ist, dass Operationen durchgeführt werden, die aufweisen:
Bereitstellen einer Redigiertransaktion für das Blockchain-Netzwerk;
Einleiten der Redigiertransaktion, wobei die Redigiertransaktion einer validierten Transaktion zugehörig ist;
Erkennen eines Hash-Wertes, der innerhalb der Redigiertransaktion angegeben und der validierten Transaktion zugehörig ist; und
Redigieren eines Wertes, der dem Hash-Wert zugehörig ist.

23. System nach Anspruch 22, bei dem die Operationen ferner aufweisen:
Erzeugen einer Transaktionshülle, wobei die Transaktionshülle den Hash-Wert umfasst, der einen Zeiger auf ein Vorabbild aufweist, wobei die Transaktionshülle ferner das Vorabbild aufweist und wobei das Vorabbild den Wert aufweist.

24. System nach Anspruch 22, wobei das Redigieren des Wertes das Setzen von dem Wert zugehörigen Bits auf Null umfasst und wobei das Setzen der dem Wert zugehörigen Bits auf Null ermöglicht, dass der Hash-Wert der gleiche bleibt.

25. System nach Anspruch 24, das ferner aufweist:
Aufrechterhalten der validierten Transaktion auf der Grundlage des Erkennens, dass der Hash-Wert nach der Redigiertransaktion der gleiche ist.

Es folgen 12 Seiten Zeichnungen

Anhängende Zeichnungen

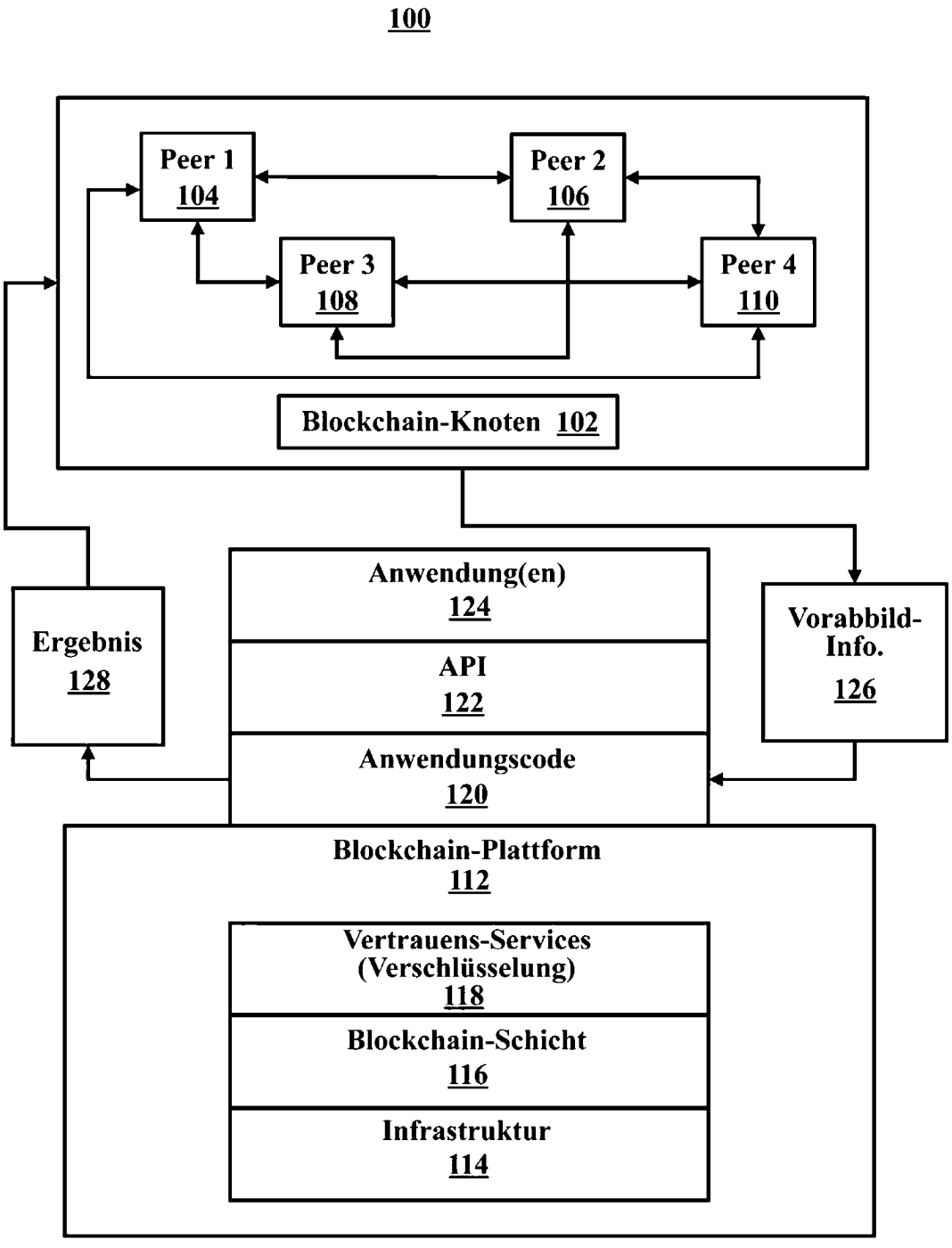


FIG. 1A

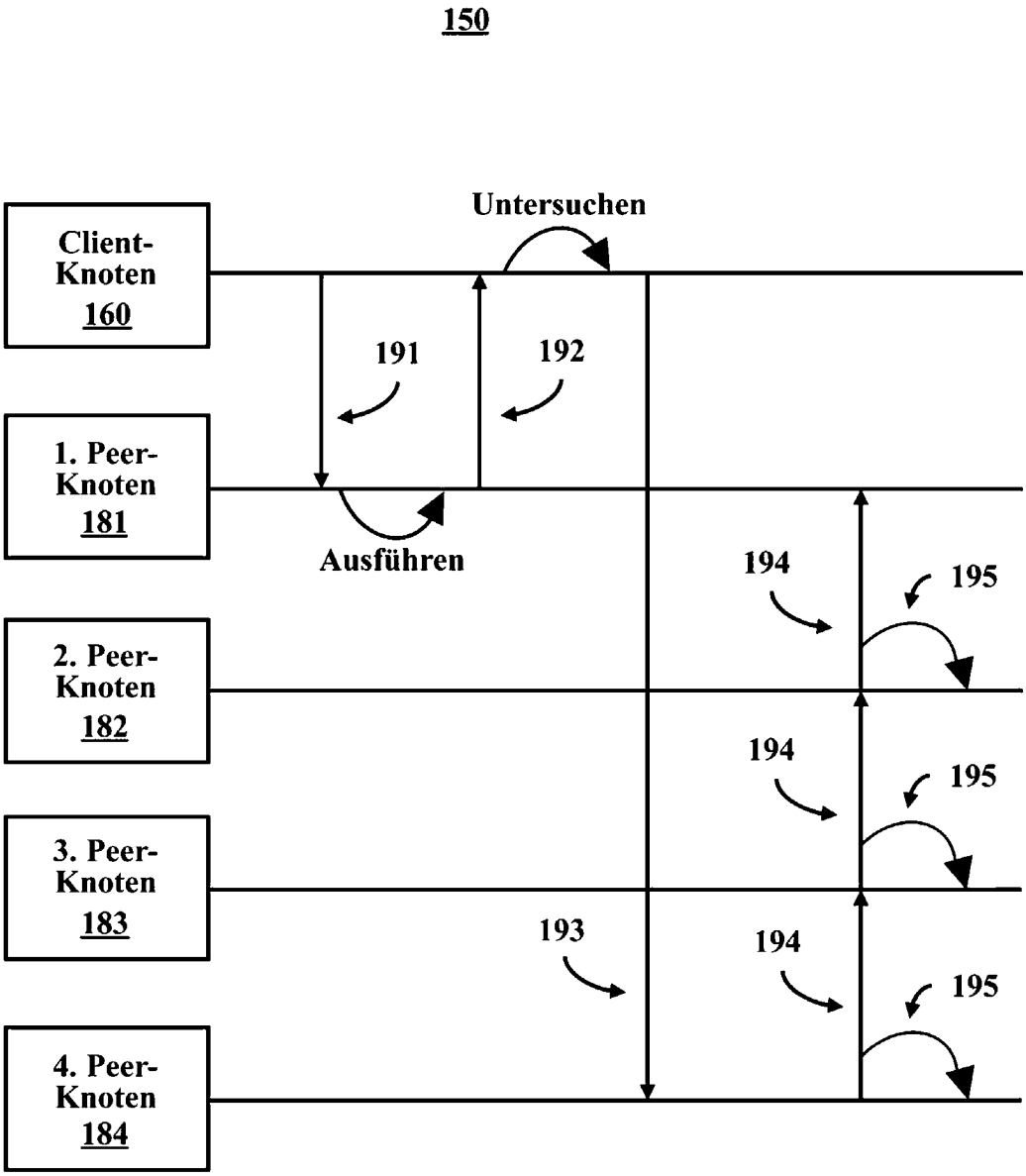

150
Client-Knoten 160
1. Peer-Knoten 181
2. Peer-Knoten 182
3. Peer-Knoten 183
4. Peer-Knoten 184
Untersuchen
191
192
Ausführen
193
194
195
194
195
194
195

FIG. 1B

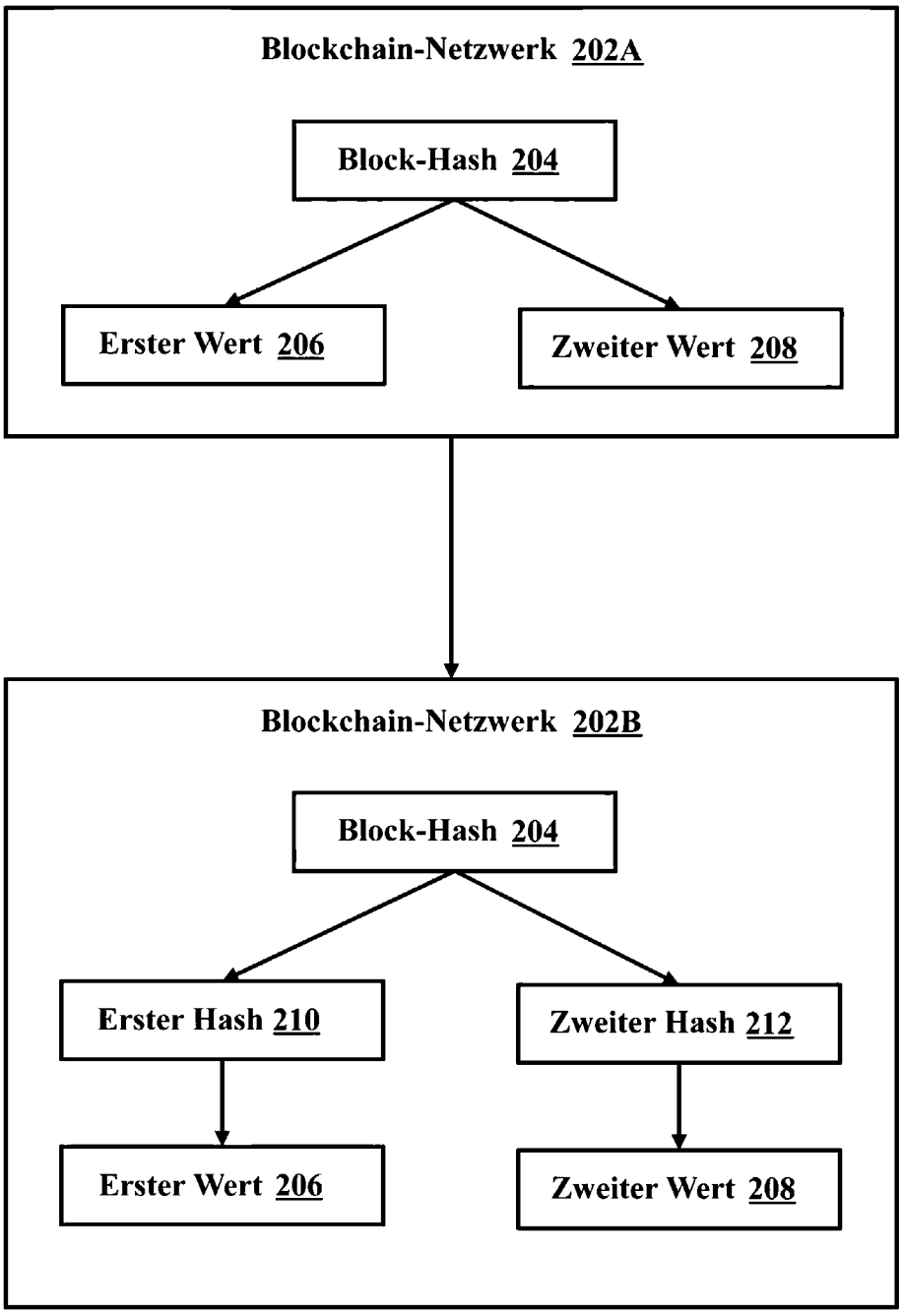

Blockchain-Netzwerk 202A
Block-Hash 204
Erster Wert 206
Zweiter Wert 208
Blockchain-Netzwerk 202B
Block-Hash 204
Erster Hash 210
Zweiter Hash 212
Erster Wert 206
Zweiter Wert 208

FIG. 2

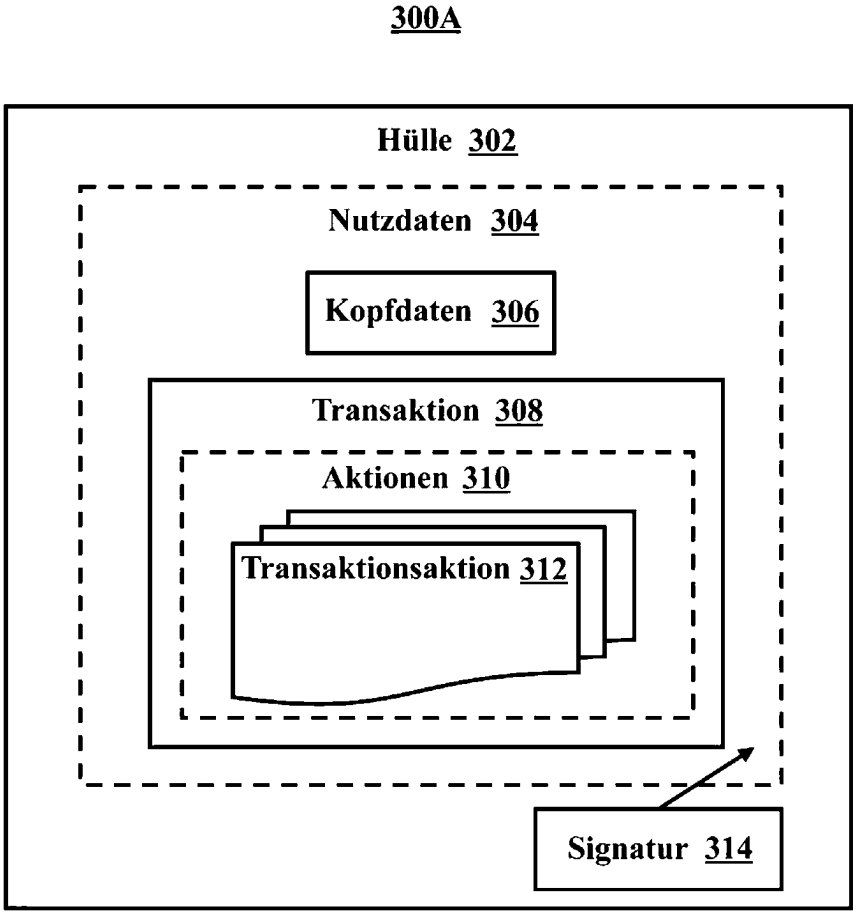

300A
Hülle 302
Nutzdaten 304
Kopfdaten 306
Transaktion 308
Aktionen 310
Transaktionsaktion 312
Signatur 314

FIG. 3A

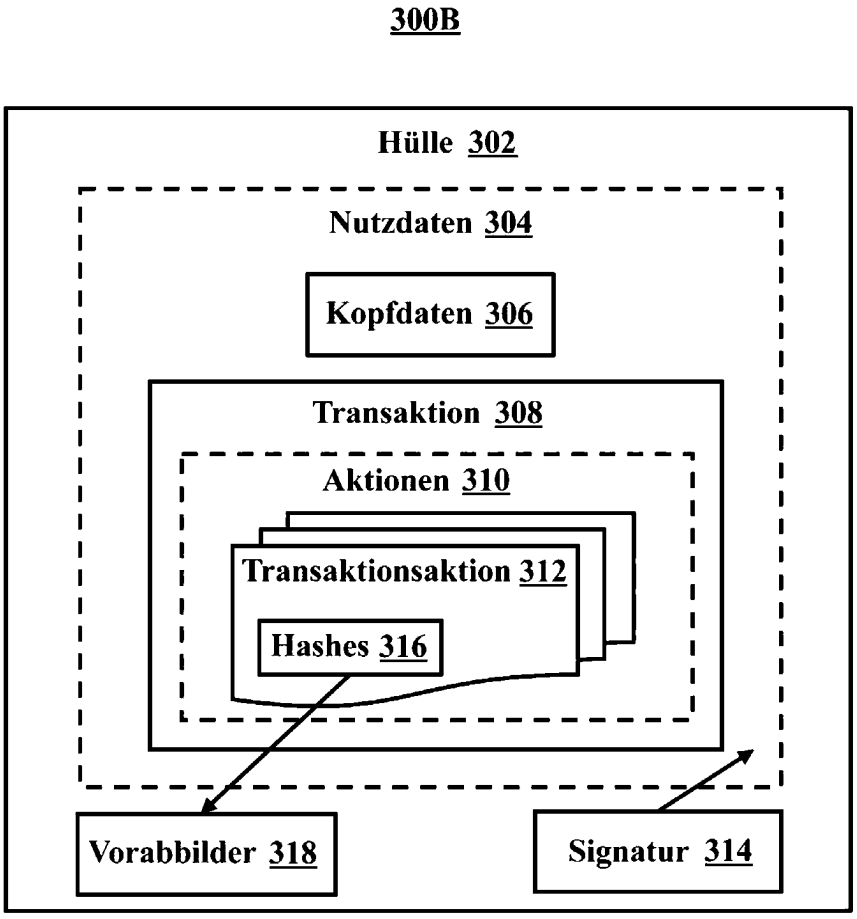

300B
Hülle 302
Nutzdaten 304
Kopfdaten 306
Transaktion 308
Aktionen 310
Transaktionsaktion 312
Hashes 316
Vorabbilder 318
Signatur 314

FIG. 3B

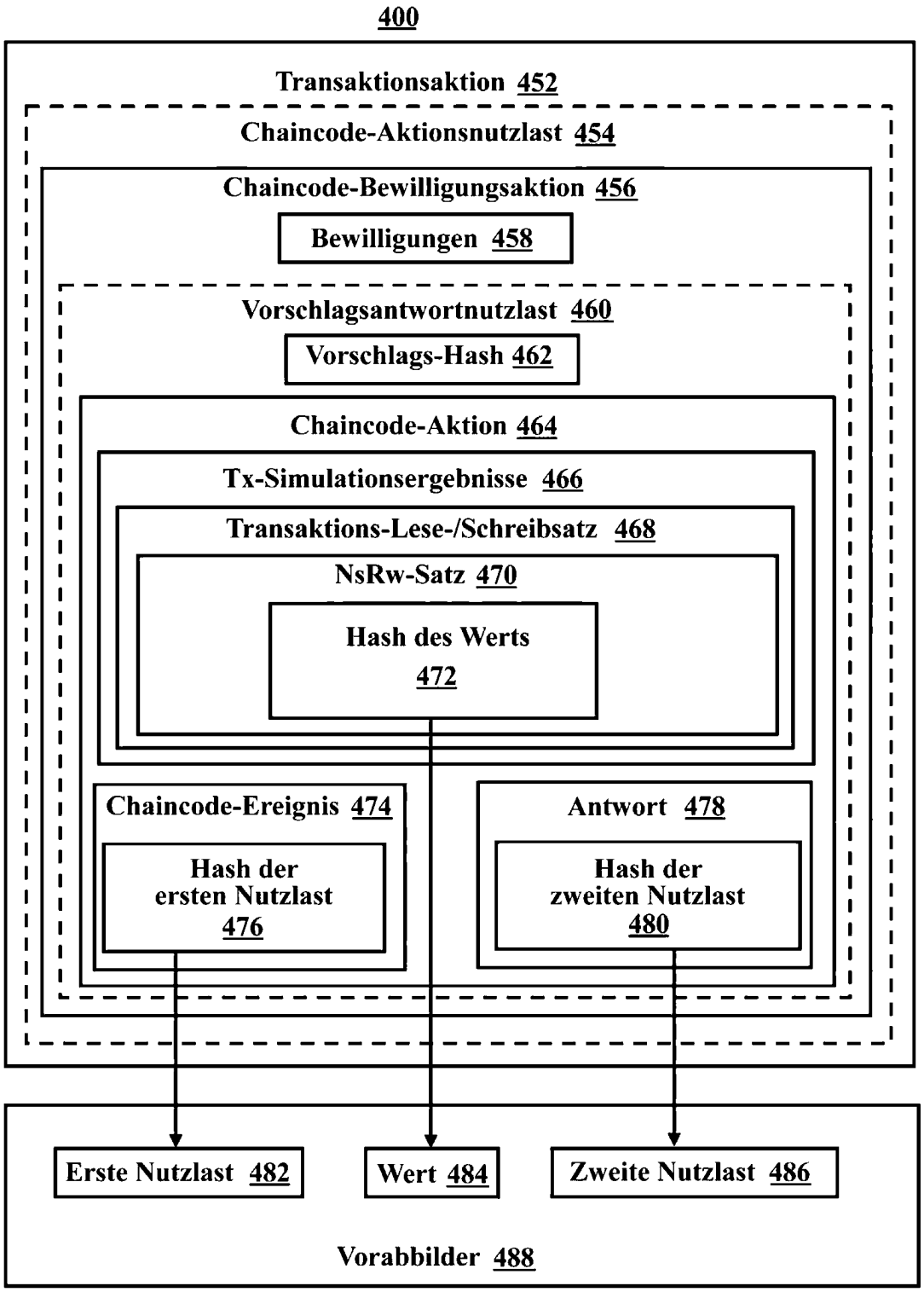

400
Transaktionsaktion 452
Chaincode-Aktionsnutzlast 454
Chaincode-Bewilligungsaktion 456
Bewilligungen 458
Vorschlagsantwortnutzlast 460
Vorschlags-Hash 462
Chaincode-Aktion 464
Tx-Simulationsergebnisse 466
Transaktions-Lese-/Schreibsatz 468
NsRw-Satz 470
Hash des Werts 472
Chaincode-Ereignis 474
Hash der ersten Nutzlast 476
Antwort 478
Hash der zweiten Nutzlast 480
Erste Nutzlast 482
Wert 484
Zweite Nutzlast 486
Vorabbilder 488

FIG. 4

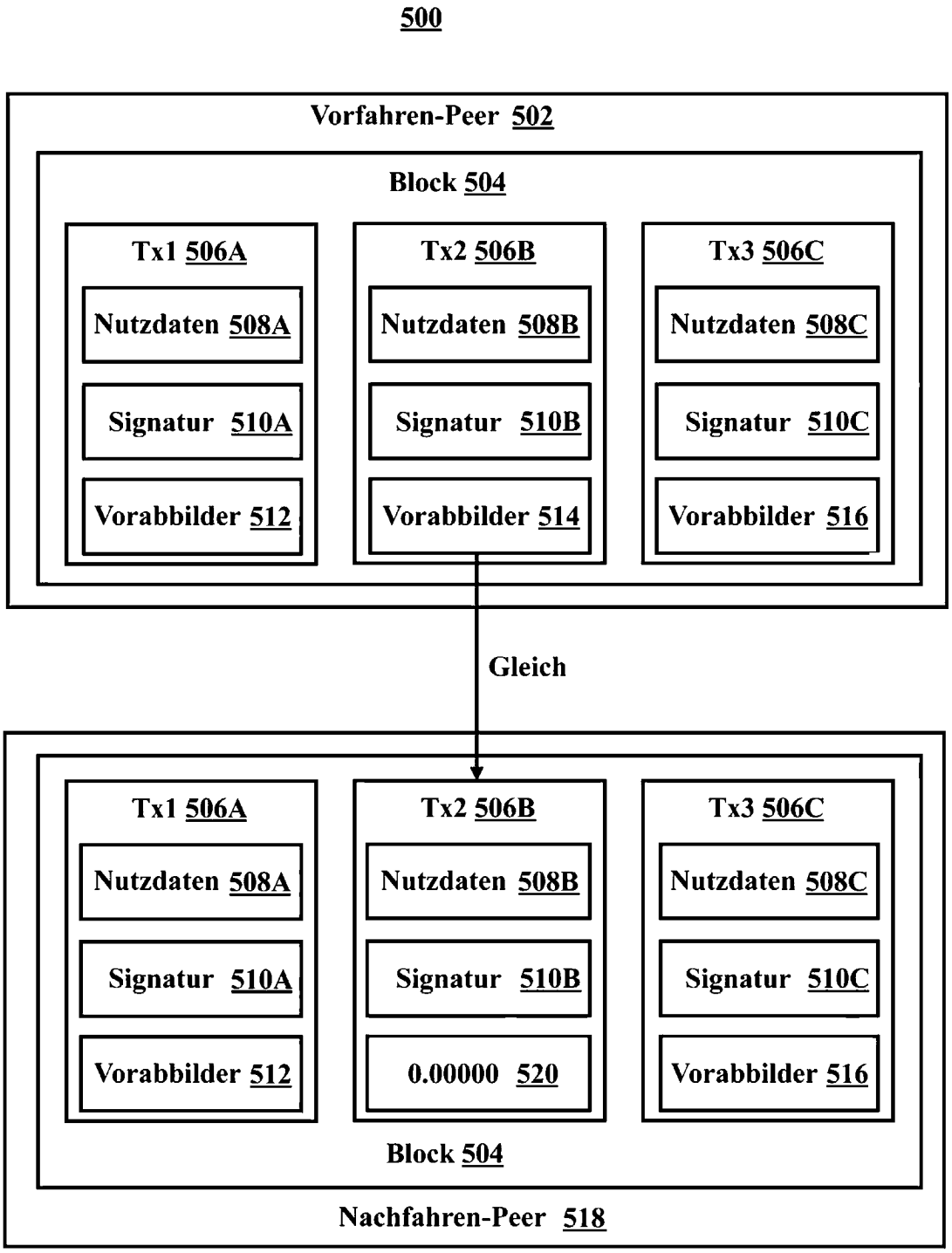

500
Vorfahren-Peer 502
Block 504
Tx1 506A
Nutzdaten 508A
Signatur 510A
Vorabbilder 512
Tx2 506B
Nutzdaten 508B
Signatur 510B
Vorabbilder 514
Tx3 506C
Nutzdaten 508C
Signatur 510C
Vorabbilder 516
Gleich
Tx1 506A
Nutzdaten 508A
Signatur 510A
Vorabbilder 512
Tx2 506B
Nutzdaten 508B
Signatur 510B
0.00000 520
Tx3 506C
Nutzdaten 508C
Signatur 510C
Vorabbilder 516
Block 504
Nachfahren-Peer 518

FIG. 5

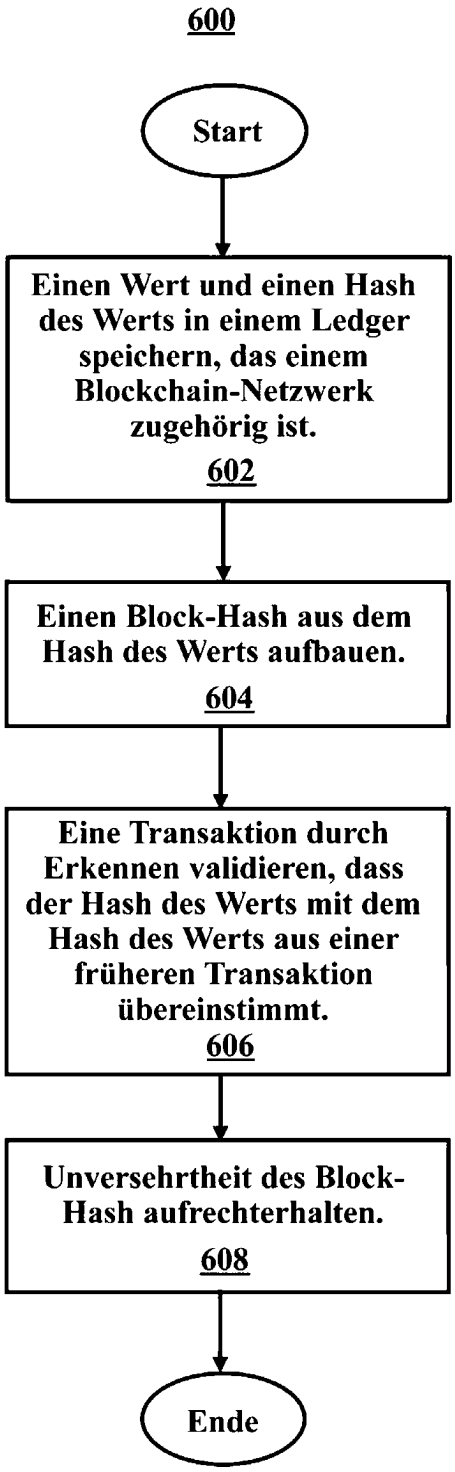

600
Start
Einen Wert und einen Hash des Werts in einem Ledger speichern, das einem Blockchain-Netzwerk zugehörig ist.
602
Einen Block-Hash aus dem Hash des Werts aufbauen.
604
Eine Transaktion durch Erkennen validieren, dass der Hash des Werts mit dem Hash des Werts aus einer früheren Transaktion übereinstimmt.
606
Unversehrtheit des Block-Hash aufrechterhalten.
608
Ende

FIG. 6A

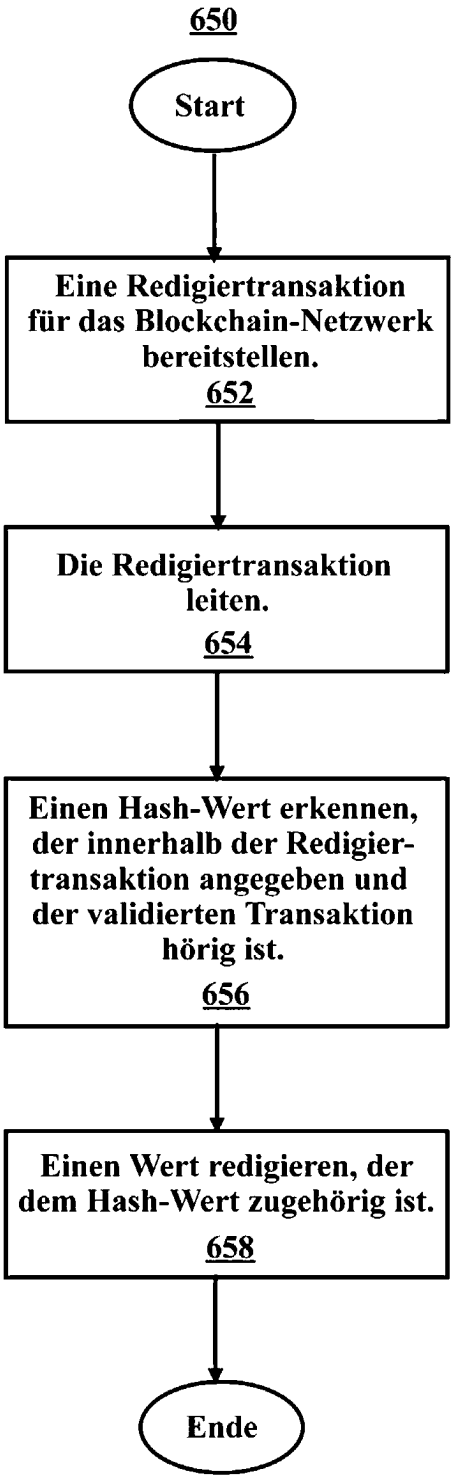

650
Start
Eine Redigiertransaktion für das Blockchain-Netzwerk bereitstellen.
652
Die Redigiertransaktion leiten.
654
Einen Hash-Wert erkennen, der innerhalb der Redigier-transaktion angegeben und der validierten Transaktion hörig ist.
656
Einen Wert redigieren, der dem Hash-Wert zugehörig ist.
658
Ende

FIG. 6B

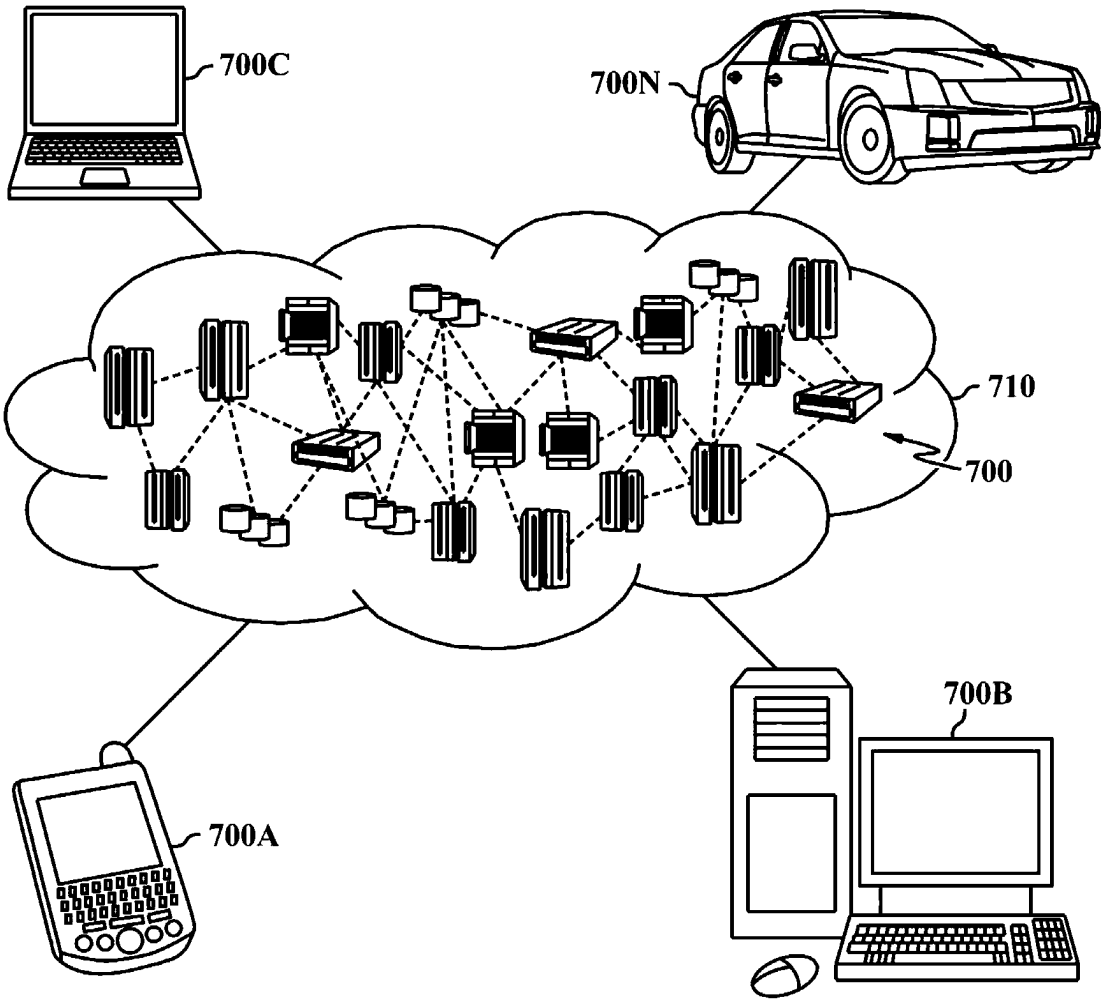


FIG. 7A

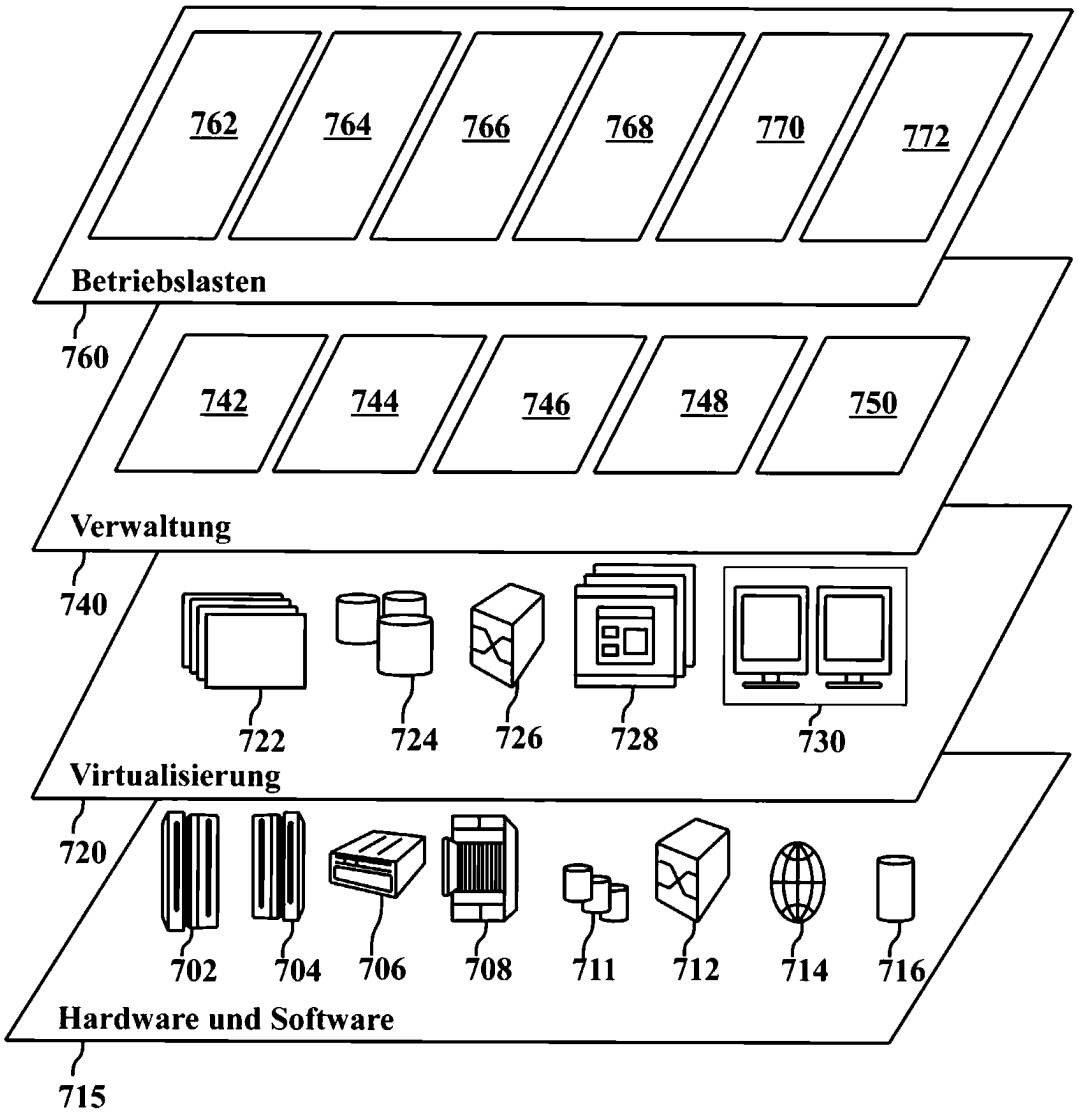

762
764
766
768
770
772
Betriebslasten
760
742
744
746
748
750
Verwaltung
740
722
724
726
728
730
Virtualisierung
720
702
704
706
708
711
712
714
716
Hardware und Software
715

FIG. 7B

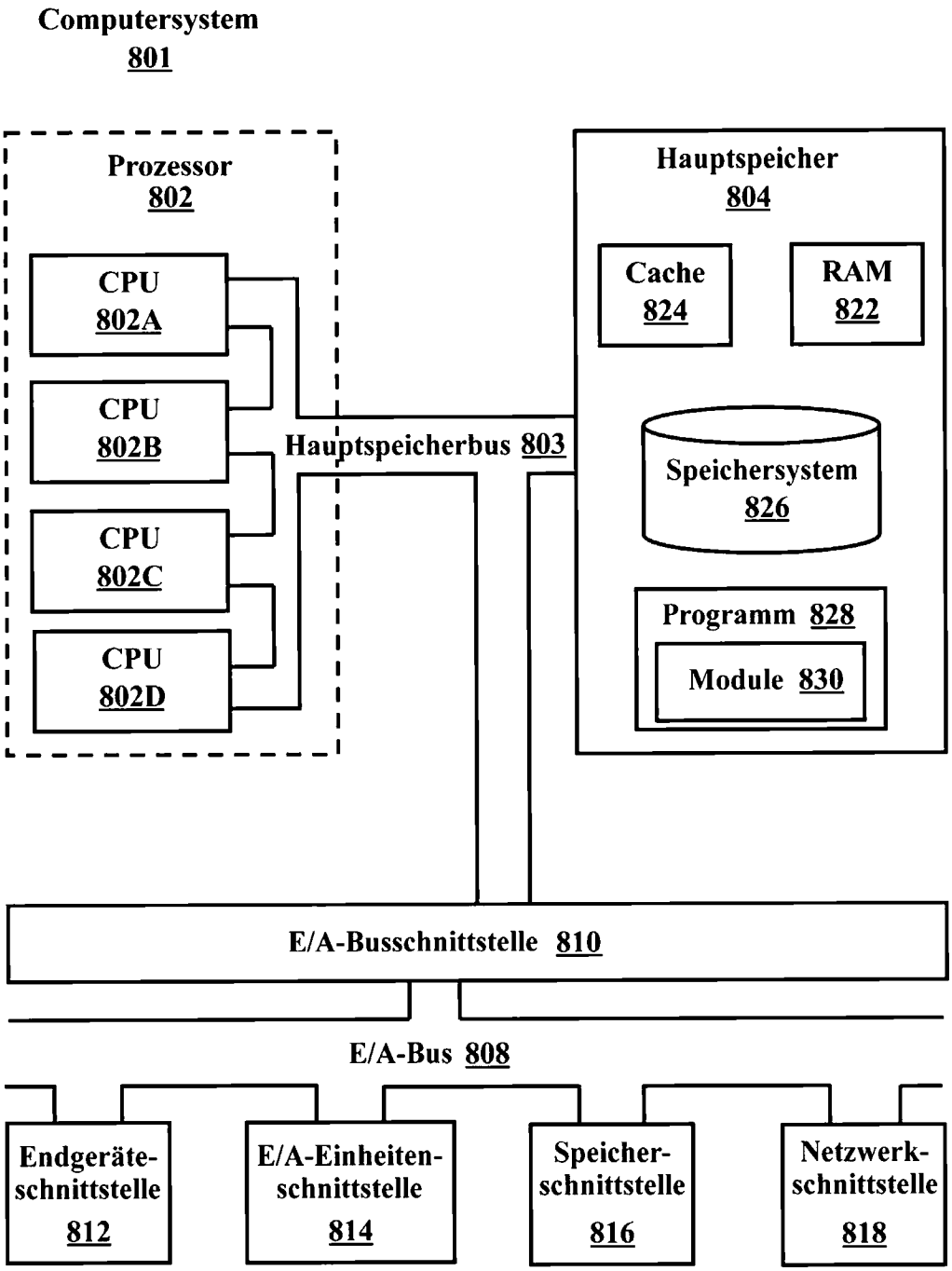


FIG. 8

