



(10) **DE 20 2015 009 783 U1** 2020.03.19

(12) **Gebrauchsmusterschrift**

(21) Aktenzeichen: **20 2015 009 783.9**

(51) Int Cl.: **G06F 16/188** (2019.01)

(22) Anmeldetag: **18.02.2015**

(67) aus Patentanmeldung: **EP 15 75 2570.0**

(47) Eintragungstag: **12.02.2020**

(45) Bekanntmachungstag im Patentblatt: **19.03.2020**

(30) Unionspriorität:

201461941986 P	19.02.2014	US
201414518826	20.10.2014	US

(74) Name und Wohnsitz des Vertreters:

**Betten & Resch Patent- und Rechtsanwälte
PartGmbH, 80333 München, DE**

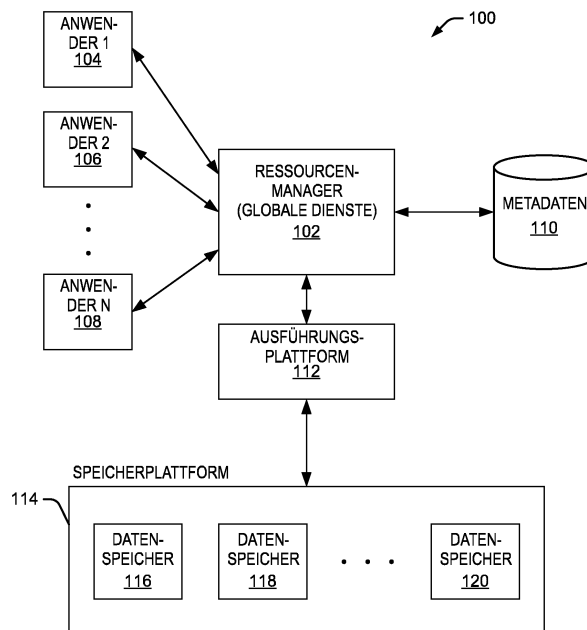
(73) Name und Wohnsitz des Inhabers:

Snowflake Inc., San Mateo, CA, US

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen.

(54) Bezeichnung: **Systeme zur Bereitstellung von Ressourcen**

(57) Hauptanspruch: Vorrichtung, umfassend:
eine Ausführungsplattform, die konfiguriert ist, um auf eine Vielzahl von entfernten Speichervorrichtungen zuzugreifen, wobei die Ausführungsplattform eine Vielzahl von virtuellen universellen Datenbanken bzw. Lagern enthält, wobei jedes virtuelle Lager einen Cache enthält, der konfiguriert ist, um aus den entfernten Speichervorrichtungen wiedergewonnene bzw. abgerufene Daten zu speichern, und einen Prozessor, der unabhängig von den entfernten Speichervorrichtungen ist; und
einen Ressourcenmanager, der mit der Ausführungsplattform gekoppelt ist und konfiguriert ist, um:
empfangene Datenverarbeitungsaufforderungen zu überwachen;
eine Ressourcennutzung in der Vielzahl von virtuellen Lagern zu überwachen;
zu bestimmen, ob zusätzliche virtuelle Lager nötig sind, basierend auf den Datenverarbeitungsaufforderungen und der Ressourcennutzung in der Vielzahl von Lagern; und
in Reaktion auf ein Bestimmen, dass ein zusätzliches virtuelles Lager nötig ist, ein neues virtuelles Lager bereitzustellen.



Beschreibung**QUERVERWEIS AUF
ZUGEHÖRIGE ANMELDUNGEN**

[0001] Diese Anmeldung beansprucht den Vorteil der provisorischen US-Anmeldung mit der seriellen Nr. 61/941,986 mit dem Titel „Apparatus and method for enterprise data warehouse data processing on cloud infrastructure“, eingereicht am 19. Februar 2014, deren Offenbarung hierin durch Bezugnahme ihrer Gesamtheit enthalten ist.

TECHNISCHES GEBIET

[0002] Die vorliegende Offenbarung betrifft Ressourcenmanagementsysteme und -verfahren, die eine Datenspeicherung und Computerressourcen managen.

[0003] Unter Schutz gestellt werden und Gegenstand des Gebrauchsmusters sind dabei, entsprechend den Vorschriften des Gebrauchsmustergesetzes, lediglich Vorrichtungen wie in den beigefügten Schutzansprüchen definiert, jedoch keine Verfahren. Soweit nachfolgend in der Beschreibung gegebenenfalls auf Verfahren Bezug genommen wird, dienen diese Bezugnahmen lediglich der beispielhaften Erläuterung der in den beigefügten Schutzansprüchen unter Schutz gestellten Vorrichtung oder Vorrichtungen.

HINTERGRUND

[0004] Viele existierende Datenspeicherungs- und -wiedergewinnungssysteme sind heutzutage verfügbar. Beispielsweise werden in einem System mit gemeinsam genutzter bzw. geteilter Platte auf einer gemeinsam genutzten bzw. geteilten Speichervorrichtung alle Daten gespeichert, die von allen Verarbeitungsknoten in einem Datencluster aus zugreifbar sind. Bei diesem Typ von System werden alle Datenänderungen zur geteilten Speichervorrichtung geschrieben, um sicherzustellen, dass alle Verarbeitungsknoten im Datencluster auf eine konsistente Version der Daten zugreifen. Wenn sich die Anzahl von Verarbeitungsknoten in einem System mit geteilter Platte erhöht, wird die geteilte Speichervorrichtung (und die Kommunikationsverbindungen zwischen den Verarbeitungsknoten und der geteilten Speichervorrichtung) ein Engpass, der Datenlese- und Datenschreiboperationen verlangsamt. Dieser Engpass wird mit dem Hinzufügen von mehr Verarbeitungsknoten weiter verschlimmert. Somit haben existierende Systeme mit geteilter Platte aufgrund dieses Engpassproblems eine beschränkte Skalierbarkeit.

[0005] Ein weiteres existierendes Datenspeicherungs- und -wiedergewinnungssystem wird „Shared-

Nothing-Architektur“ genannt. Bei dieser Architektur werden Daten über mehrere Verarbeitungsknoten so verteilt, dass jeder Knoten eine Untergruppe der Daten in der gesamten Datenbank speichert. Wenn ein neuer Verarbeitungsknoten hinzugefügt oder entfernt wird, muss die Shared-Nothing-Architektur Daten über die mehreren Verarbeitungsknoten neu anordnen. Diese Neuordnung von Daten kann zeitaufwendig und störend für Datenlese- und -schreiboperationen sein, die während der Datenneuordnung ausgeführt werden. Und die Affinität von Daten zu einem bestimmten Knoten kann „Hotspots“ auf dem Datencluster für populäre Daten erzeugen. Weiterhin erfordert diese Architektur deshalb, weil jeder Verarbeitungsknoten auch die Speicherfunktion durchführt, wenigstens einen Verarbeitungsknoten, um Daten zu speichern. Somit schlägt die Shared-Nothing-Architektur darin fehl, Daten zu speichern, wenn alle Verarbeitungsknoten entfernt sind. Zusätzlich ist ein Management von Daten in einer Shared-Nothing-Architektur aufgrund der Verteilung von Daten über viele unterschiedliche Verarbeitungsknoten komplex.

[0006] Die hierin beschriebenen Systeme und Verfahren stellen einen verbesserten Ansatz für eine Datenspeicherung und eine Datenwiedergewinnung zur Verfügung, der die oben identifizierten Beschränkungen existierender Systeme abmildert.

Figurenliste

[0007] Nicht beschränkende und nicht erschöpfende Ausführungsformen der vorliegenden Offenbarung werden unter Bezugnahme auf die folgenden Figuren beschrieben, wobei sich gleiche Bezugszeichen auf gleiche Teile in allen verschiedenen Figuren beziehen, solange nichts anderes spezifiziert ist.

Fig. 1 ist ein Blockdiagramm, das eine beispielhafte Ausführungsform der hierin beschriebenen Systeme und Verfahren darstellt.

Fig. 2 ist ein Blockdiagramm, das eine Ausführungsform eines Ressourcenmanagers darstellt.

Fig. 3 ist ein Blockdiagramm, das eine Ausführungsform einer Ausführungsplattform darstellt.

Fig. 4 ist ein Blockdiagramm, das eine beispielhafte Betriebsumgebung mit mehreren Anwendern darstellt, die über mehrere virtuelle Lager auf mehrere Datenbanken zugreifen.

Fig. 5 ist ein Blockdiagramm, das eine weitere beispielhafte Betriebsumgebung mit mehreren Anwendern darstellt, die über einen Lastausgleicher und mehrere in einer virtuellen Lagergruppe enthaltene virtuelle Lager auf mehrere Datenbanken zugreifen.

Fig. 6 ist ein Blockdiagramm, das eine weitere beispielhafte Betriebsumgebung darstellt, die

mehrere verteilte virtuelle Lager und virtuelle Lagergruppen hat.

Fig. 7 ist ein Ablaufdiagramm, das eine Ausführungsform eines Verfahrens zum Managen von Datenspeicherungs- und -wiedergewinnungsoperationen darstellt.

Fig. 8 ist ein Ablaufdiagramm, das eine Ausführungsform eines Verfahrens zum Bereitstellen von neuen virtuellen universellen Datenbanken bzw. Lagern darstellt.

Fig. 9A und **Fig. 9B** stellen ein Ablaufdiagramm einer Ausführungsform eines Verfahrens zum Skalieren von Ressourcen dar.

Fig. 10 ist ein Ablaufdiagramm, das eine Ausführungsform eines weiteren Verfahrens zum Skalieren von Ressourcen darstellt.

Fig. 11 ist ein Blockdiagramm, das eine beispielhafte Computervorrichtung darstellt.

DETAILLIERTE BESCHREIBUNG

[0008] Die hierin beschriebenen Systeme und Verfahren stellen eine neue Plattform zum Speichern und Wiedergewinnen von Daten ohne die Probleme zur Verfügung, denen man bei existierenden Systemen begegnet. Beispielsweise unterstützt diese neue Plattform das Hinzufügen von neuen Knoten ohne die Notwendigkeit für ein Neuordnen von Datendateien, wie es durch die Shared-Nothing-Architektur erforderlich ist. Zusätzlich können Knoten zu der Plattform ohne ein Erzeugen von Engpässen hinzugefügt werden, die bei dem System mit gemeinsam genutzter Platte üblich sind. Diese neue Plattform ist immer für Datenlese- und Datenschreiboperationen verfügbar, selbst wenn einige der Knoten zur Wartung offline sind oder ausgefallen sind. Die beschriebene Plattform trennt die Datenspeicherressourcen von den Computerressourcen, so dass Daten gespeichert werden können, ohne die Verwendung von bestimmten Computerressourcen zu erfordern. Dies ist eine Verbesserung gegenüber der Shared-Nothing-Architektur, die darin fehlschlägt, Daten zu speichern, wenn alle Computerressourcen entfernt sind. Daher fährt die neue Plattform damit fort, Daten zu speichern, selbst wenn die Rechenressourcen bzw. Computerressourcen nicht mehr verfügbar sind oder andere Aufgaben durchführen.

[0009] In der folgenden Beschreibung wird auf die beigefügten Zeichnungen Bezug genommen, die einen Teil davon bilden und wobei anhand einer Darstellung spezifische beispielhafte Ausführungsformen gezeigt sind, bei welchen die Offenbarung ausgeführt werden kann. Diese Ausführungsformen sind in ausreichendem Detail beschrieben, um Fachleuten auf dem Gebiet zu ermöglichen, die hierin offenbarten Konzepte auszuführen, und es ist zu verstehen, dass Modifikationen an den verschiedenen offenbar-

ten Ausführungsformen durchgeführt werden können und dass andere Ausführungsformen verwendet werden können, ohne vom Schutzzumfang der vorliegenden Offenbarung abzuweichen. Die folgende detaillierte Beschreibung ist daher nicht in einem beschränkenden Sinn zu nehmen.

[0010] Eine Bezugnahme in dieser gesamten Beschreibung auf „eine einzige Ausführungsform“, „eine Ausführungsform“, „ein einziges Beispiel“ oder „ein Beispiel“ bedeutet, dass ein bestimmtes Merkmal, eine bestimmte Struktur oder eine bestimmte Charakteristik, das oder die in Verbindung mit der Ausführungsform oder dem Beispiel beschrieben ist, in wenigstens einer Ausführungsform der vorliegenden Offenbarung enthalten ist. Somit beziehen sich nicht notwendigerweise alle Erscheinungsformen der Phrasen „bei einer einzigen Ausführungsform“, „bei einer Ausführungsform“, „ein einziges Beispiel“ oder „ein Beispiel“ an verschiedenen Stellen in dieser gesamten Beschreibung auf dieselbe Ausführungsform oder dasselbe Beispiel. Zusätzlich sollte erkannt werden, dass die hiermit zur Verfügung gestellten Figuren zu Erklärungs Zwecken für Personen dienen, die Fachleute auf dem Gebiet sind, und dass die Zeichnungen nicht notwendigerweise maßstäblich gezeichnet sind.

[0011] Ausführungsformen gemäß der vorliegenden Offenbarung können als eine Vorrichtung, ein Verfahren oder ein Computerprogrammprodukt verkörpert sein. Demgemäß kann die vorliegende Offenbarung die Form einer gänzlich hardwareumfassten Ausführungsform, einer gänzlich softwareumfassten Ausführungsform (Firmware, residente Software, eine Mikrocodes, etc. enthaltend) oder einer Ausführungsform, die Software- und Hardwareaspekte kombiniert, auf welche alle hierin allgemein als „Schaltung“, „Modul“ oder „System“ Bezug genommen werden kann, annehmen. Weiterhin können Ausführungsformen der vorliegenden Offenbarung die Form eines Computerprogrammprodukts annehmen, das in irgendeinem konkreten Ausdrucksmedium verkörpert ist, das einen computernutzbaren Programmcode im Medium verkörpert hat.

[0012] Irgendeine Kombination von einem oder mehreren computernutzbaren oder computerlesbaren Medien kann verwendet werden. Beispielsweise kann ein computerlesbares Medium eines oder mehreres von einer tragbaren Computerdiskette, einer Festplatte, einer Direktzugriffsspeicher-(RAM-)Vorrichtung, einer Nurlesespeicher-(ROM-)Vorrichtung, einer löschbaren programmierbaren Nurlesespeicher-(EPROM- oder Flash-Speicher-)Vorrichtung, einem tragbaren Kompaktdisk-Nurlesespeicher (CDROM), einer optischen Speichervorrichtung und einer magnetischen Speichervorrichtung enthalten. Ein Computerprogrammcode zum Ausführen von Operationen der vorliegenden Offenbarung kann in

irgendeiner Kombination von einer oder mehreren Programmiersprachen geschrieben sein. Ein solcher Code kann von einem Quellcode zu einer computerlesbaren Assemblersprache oder einen Maschinencode, der für die Vorrichtung oder den Computer geeignet ist, auf der oder dem der Code ausgeführt werden wird, kompiliert werden.

[0013] Ausführungsformen können auch in Cloud-Computerumgebungen implementiert sein. In dieser Beschreibung und den folgenden Ansprüchen kann „Cloudcomputing“ als ein Modell zum Ermöglichen eines ubiquitären, angenehmen On-Demand-Netzwerkzugriffs auf einen gemeinsam genutzten bzw. geteilten Pool von konfigurierbaren Computerressourcen bzw. Rechnerressourcen (z.B. Netzwerken, Servern, Speichern, Anwendungen und Diensten) definiert sein, die über Virtualisierung schnell bereitgestellt und mit minimalem Managementaufwand oder minimaler Serviceprovider-Interaktion freigegeben und dann entsprechend skaliert werden können. Ein Cloud-Modell kann zusammengesetzt sein aus verschiedenen Eigenschaften (z.B. On-demand Self Service, Broad Network Access, Resource Pooling, Rapid Elasticity und Measured Services), Servicemodellen (z.B. Software as a Service („SaaS“), Platform as a Service („PaaS“) und Infrastructure as a Service („IaaS“)) und Bereitstellungsmodellen (Deployment models) (z.B. Private Cloud, Community Cloud, Public Cloud und Hybrid Cloud).

[0014] Die Ablaufdiagramme und Blockdiagramme in den beigefügten Figuren stellen die Architektur, die Funktionalität und die Operation von möglichen Implementierungen von Systemen, Verfahren und Computerprogrammprodukten gemäß verschiedenen Ausführungsformen der vorliegenden Offenbarung dar. Diesbezüglich kann jeder Block in den Ablaufdiagrammen oder Blockdiagrammen ein Modul, ein Segment oder einen Teilbereich eines Codes darstellen, das oder der eine oder mehrere ausführbare Anweisungen zum Implementieren der spezifizierten logischen Funktion(en) umfasst. Es wird auch bemerkt werden, dass jeder Block der Blockdiagramme und/oder Ablaufdiagramme und Kombinationen von Blöcken in den Blockdiagrammen und/oder Ablaufdiagrammen durch spezielle hardwarebasierte Systeme, die die spezifizierten Funktionen oder Handlungen durchführen, oder Kombinationen von spezieller Hardware und Computeranweisungen implementiert sein können. Diese Computerprogramm-anweisungen können auch in einem computerlesbaren Medium gespeichert sein, das einen Computer oder eine andere programmierbare Datenverarbeitungsvorrichtung anweisen kann, auf spezielle Weise zu funktionieren, so dass die in dem computerlesbaren Medium gespeicherten Anweisungen ein Herstellungsprodukt erzeugen, das Anweisungsmittel enthält, die die Funktion/Handlung implementieren, die

in dem Block oder den Blöcken des Ablaufdiagramms und/oder des Blockdiagramms spezifiziert ist.

[0015] Die hierin beschriebenen Systeme und Verfahren stellen ein flexibles und skalierbares Datenlager unter Verwendung einer neuen Datenverarbeitungsplattform zur Verfügung. Bei einigen Ausführungsformen setzen die beschriebenen Systeme und Verfahren eine Cloud-Infrastruktur ein, die cloudbasierte Speicherressourcen, Rechner- bzw. Computerressourcen und ähnliches unterstützt. Beispielhafte cloudbasierte Speicherressourcen bieten eine signifikante Speicherkapazität, die on-demand bzw. auf Anforderung bei niedrigen Kosten verfügbar ist. Weiterhin können diese cloudbasierten Speicherressourcen fehlertolerant und hochskalierbar sein, was bei privaten Datenspeichersystemen teuer zu erreichen sein kann. Beispielhafte cloudbasierte Computerressourcen sind on-demand verfügbar und für sie kann ein Preis basierend auf tatsächlichen Ausmaßen einer Nutzung der Ressourcen festgesetzt werden. Typischerweise wird die Cloud-Infrastruktur auf eine schnelle Weise dynamisch bereitgestellt, rekonfiguriert und außer Betrieb gesetzt.

[0016] Bei den beschriebenen Systemen und Verfahren verwendet ein Datenspeichersystem eine SQL-(strukturierte Abfragesprache)-basierte relationale Datenbank. Jedoch sind diese Systeme und Verfahren unter Verwendung von irgendeiner Datenspeicherarchitektur und unter Verwendung von irgendeiner Sprache, um Daten innerhalb der Datenspeicherungs- und -wiedergewinnungsplattform zu speichern und wiederzugewinnen, auf irgendeinen Typ von Datenbank und irgendeinen Typ von Datenspeicherungs- und -wiedergewinnungsplattform anwendbar. Die hierin beschriebenen Systeme und Verfahren stellen weiterhin ein mandantenfähiges System zur Verfügung, das eine Isolation von Computerressourcen und Daten zwischen unterschiedlichen Kunden/Klienten und zwischen unterschiedlichen Anwendern innerhalb desselben Kunden/Klienten unterstützt.

[0017] Fig. 1 ist ein Blockdiagramm, das eine beispielhafte Ausführungsform einer neuen Datenverarbeitungsplattform **100** darstellt. Wie es in Fig. 1 gezeigt ist, ist ein Ressourcenmanager **102** mit mehreren Anwendern **104**, **106** und **108** gekoppelt. Bei besonderen Implementierungen kann der Ressourcenmanager **102** irgendeine Anzahl von Anwendern unterstützen, die einen Zugriff auf die Datenverarbeitungsplattform **100** wünschen. Die Anwender **104-108** können beispielsweise Endanwender enthalten, die Datenspeicherungs- und -wiedergewinnungsanfragen zur Verfügung stellen, Systemadministratoren, die die hierin beschriebenen Systeme und Verfahren managen, und andere Komponenten/Vorrichtungen, die mit dem Ressourcenmanager **102** interagieren. Der Ressourcenmanager **102** stellt ver-

schiedene Dienste und Funktionen zur Verfügung, die den Betrieb von allen Systemen und Komponenten innerhalb der Datenverarbeitungsplattform **100** unterstützen. Der Ressourcenmanager **102**, wie er hierin genannt wird, kann auch „globales Dienstsysteem“ genannt werden, das verschiedene Funktionen durchführt, wie sie hierin diskutiert sind.

[0018] Der Ressourcenmanager **102** ist auch mit Metadaten **110** gekoppelt, die mit der Gesamtheit von Daten assoziiert sind, die über die gesamte Datenverarbeitungsplattform **100** hinweg gespeichert sind. Bei einigen Ausführungsformen enthalten die Metadaten **110** eine Zusammenfassung von Daten, die in entfernten Datenspeichersystemen gespeichert sind, so wie Daten, die von einem lokalen Cache verfügbar sind. Zusätzlich können die Metadaten **110** Information in Bezug darauf enthalten, wie Daten in den entfernten Datenspeichersystemen und den lokalen Caches organisiert sind. Die Metadaten **110** lassen zu, dass Systeme und Dienste bestimmen, ob auf ein Datenstück zugegriffen werden muss, ohne die aktuellen Daten aus einer Speichervorrichtung zu laden oder auf diese zuzugreifen.

[0019] Der Ressourcenmanager **102** ist weiterhin mit einer Ausführungsplattform **112** gekoppelt, die mehrere Computerressourcen zur Verfügung stellt, die verschiedene Datenspeicherungs- und Datenwiedergewinnungsaufgaben ausführen, wie es nachstehend noch ausführlicher diskutiert wird. Die Ausführungsplattform **112** ist mit mehreren Datenspeichervorrichtungen **116**, **118** und **120** gekoppelt, die Teil einer Speicherplattform **114** sind. Obwohl in **Fig. 1** drei Datenspeichervorrichtungen **116**, **118** und **120** gezeigt sind, kann die Ausführungsplattform **112** mit irgendeiner Anzahl von Datenspeichervorrichtungen kommunizieren. Bei einigen Ausführungsformen sind die Datenspeichervorrichtungen **116**, **118** und **120** cloudbasierte Speichervorrichtungen, die an einem oder mehreren geographischen Standorten angeordnet sind. Beispielsweise können die Datenspeichervorrichtungen **116**, **118** und **120** Teil einer Infrastruktur eine öffentlichen Cloud (Public Cloud) oder einer Infrastruktur einer privaten Cloud (Private Cloud) sein. Die Datenspeichervorrichtungen **116**, **118** und **120** können Festplattenlaufwerke (HDDs), Festkörperlaufwerke (SSDs), Speichercluster, Amazon S3™-Speichersysteme oder irgendeine andere Datenspeichertechnologie sein. Zusätzlich kann die Speicherplattform **114** verteilte Dateiensysteme (wie beispielsweise verteilte Hadoop-Datensysteme (HDFS)), Objektspeichersysteme und ähnliches enthalten.

[0020] Bei bestimmten Ausführungsformen sind die Kommunikationsverbindungen zwischen dem Ressourcenmanager **102** und den Anwendern **104-108**, den Metadaten **110** und der Ausführungsplattform **112** über ein oder mehrere Datenkommunikations-

netzwerke implementiert. Gleichmaßen sind die Kommunikationsverbindungen zwischen der Ausführungsplattform **112** und den Datenspeichervorrichtungen **116-120** in der Speicherplattform **114** über ein oder mehrere Datenkommunikationsnetzwerke implementiert. Diese Datenkommunikationsnetzwerke können irgendein Kommunikationsprotokoll und irgendeinen Typ von Kommunikationsmedium verwenden. Bei einigen Ausführungsformen sind die Datenkommunikationsnetzwerke eine Kombination aus zwei oder mehr Datenkommunikationsnetzwerken (oder Unternetzwerken), die miteinander gekoppelt sind. Bei alternativen Ausführungsformen sind diese Kommunikationsverbindungen unter Verwendung von irgendeinem Typ von Kommunikationsmedium und irgendeinem Kommunikationsprotokoll implementiert.

[0021] Wie es in **Fig. 1** gezeigt ist, sind die Datenspeichervorrichtungen **116**, **118** und **120** von den Computerressourcen entkoppelt, die mit der Ausführungsplattform **112** assoziiert sind. Diese Architektur unterstützt dynamische Änderungen an der Datenverarbeitungsplattform **100** basierend auf den sich ändernden Datenspeicherungs-/wiedergewinnungsnotwendigkeiten sowie den Änderungsnotwendigkeiten der Anwender und Systeme, die auf die Datenverarbeitungsplattform **100** zugreifen. Die Unterstützung von dynamischen Änderungen lässt zu, dass sich die Datenverarbeitungsplattform **100** in Reaktion auf sich ändernde Anforderungen auf den Systemen und Komponenten innerhalb der Datenverarbeitungsplattform **100** schnell skaliert. Die Entkopplung der Computerressourcen von den Datenspeichervorrichtungen unterstützt die Speicherung großer Mengen von Daten, ohne eine entsprechende große Menge von Computerressourcen zu erfordern. Gleichmaßen unterstützt diese Entkopplung von Ressourcen eine signifikante Erhöhung bezüglich der zu einer bestimmten Zeit verwendeten Computerressourcen, ohne eine entsprechende Erhöhung bezüglich der verfügbaren Datenspeicherressourcen zu erfordern.

[0022] Der Ressourcenmanager **102**, die Metadaten **110**, die Ausführungsplattform **112** und die Speicherplattform **114** sind in **Fig. 1** als individuelle Komponenten gezeigt. Jedoch kann jedes von dem Ressourcenmanager **102**, den Metadaten **110**, der Ausführungsplattform **112** und der Speicherplattform **114** als ein verteiltes System (z.B. über mehrere Systeme/Plattformen an mehreren geographischen Standorten verteilt) implementiert sein. Zusätzlich kann jedes von dem Ressourcenmanager **102**, den Metadaten **110**, der Ausführungsplattform **112** und der Speicherplattform **114** in Abhängigkeit von Änderungen bei den von den Anwendern **104-108** empfangenen Anfragen und den sich ändernden Notwendigkeiten der Datenverarbeitungsplattform **100** (unabhängig voneinander) vergrößert oder verkleinert werden. So-

mit ist die Datenverarbeitungsplattform **100** bei den beschriebenen Ausführungsformen dynamisch und unterstützt reguläre Änderungen, um die aktuellen bzw. tatsächlichen Datenverarbeitungsnotwendigkeiten zu erfüllen.

[0023] Während eines typischen Betriebs verarbeitet die Datenverarbeitungsplattform **100** mehrere Anfragen (oder Aufforderungen), die von irgendeinem der Anwender **104-108** empfangen sind. Diese Anfragen werden durch den Ressourcenmanager **102** gemanagt, um zu bestimmen, wann und wie die Anfragen auszuführen sind. Beispielsweise kann der Ressourcenmanager **102** bestimmen, welche Daten nötig sind, um die Anfrage zu verarbeiten, und weiterhin bestimmen, welche Knoten innerhalb der Ausführungsplattform **112** am besten geeignet sind, um die Anfrage zu verarbeiten. Einige Knoten können bereits die Daten im Cache gespeichert haben, die dazu nötig sind, die Anfrage zu verarbeiten, und sind daher gute Kandidaten zum Verarbeiten der Anfrage. Die Metadaten **110** unterstützen den Ressourcenmanager **102** beim Bestimmen, welche Knoten in der Ausführungsplattform **112** bereits wenigstens einen Teilbereich der Daten im Cache haben, die dazu nötig sind, die Anfrage zu verarbeiten. Ein oder mehrere Knoten in der Ausführungsplattform **112** verarbeitet oder verarbeiten die Anfrage unter Verwendung von Daten, die durch die Knoten im Cache gespeichert sind, und, wenn es nötig ist, von Daten, die aus der Speicherplattform **114** wiedergewonnen sind. Es ist erwünscht, so viele Daten wie möglich aus Caches innerhalb der Ausführungsplattform **112** wiederzugewinnen, weil die Wiedergewinnungsgeschwindigkeit typischerweise viel schneller als ein Wiedergewinnen von Daten aus der Speicherplattform **114** ist.

[0024] Wie es in **Fig. 1** gezeigt ist, trennt die Datenverarbeitungsplattform **100** die Ausführungsplattform **112** von der Speicherplattform **114**. Bei dieser Anordnung arbeiten die Verarbeitungsressourcen und Cacheressourcen in der Ausführungsplattform **112** unabhängig von den Datenspeicherressourcen **116-120** in der Speicherplattform **114**. Somit sind die Computerressourcen und Cacheressourcen nicht auf spezifische Datenspeicherressourcen **116-120** beschränkt. Stattdessen können alle Computerressourcen und alle Cacheressourcen Daten aus irgendeiner der Datenspeicherressourcen in der Speicherplattform **114** wiedergewinnen und Daten dorthin speichern. Zusätzlich unterstützt die Datenverarbeitungsplattform **100** das Hinzufügen von neuen Computerressourcen und Cacheressourcen zu der Ausführungsplattform **112**, ohne irgendwelche Änderungen an der Speicherplattform **114** zu erfordern. Gleichermaßen unterstützt die Verarbeitungsplattform **100** das Hinzufügen von Datenspeicherressourcen zu der Speicherplattform **114**, ohne irgendwelche Änderungen an Knoten in der Ausführungsplattform **112** zu erfordern.

[0025] **Fig. 2** ist ein Blockdiagramm, das eine Ausführungsform des Ressourcenmanagers **102** darstellt. Wie es in **Fig. 2** gezeigt ist, enthält der Ressourcenmanager **102** einen Zugriffsmanager **202** und einen Schlüsselmanager **204**, die mit einer Datenspeichervorrichtung **206** gekoppelt sind. Der Zugriffsmanager **202** handhabt Authentifizierungs- und Autorisierungsaufgaben für die hierin beschriebenen Systeme. Der Schlüsselmanager **204** managt ein Speichern und eine Authentifizierung von Schlüsseln, die während Authentifizierungs- und Autorisierungsaufgaben verwendet werden. Beispielsweise managen der Zugriffsmanager **202** und der Schlüsselmanager **204** die Schlüssel, die dazu verwendet werden, auf Daten zuzugreifen, die in entfernten Speichervorrichtungen (z.B. Datenspeichervorrichtungen in der Speicherplattform **114**) gespeichert sind. Wie es hierin verwendet ist, können die entfernten Speichervorrichtungen auch „persistente Speichervorrichtungen“ genannt werden. Ein Aufforderungsverarbeitungsdienst **208** managt empfangene Datenspeicheraufforderungen und Datenwiedergewinnungsaufforderungen (z.B. Datenbankanfragen). Beispielsweise kann der Aufforderungsverarbeitungsdienst **208** die Daten bestimmen, die dazu nötig sind, die empfangene Datenspeicheraufforderung oder Datenwiedergewinnungsaufforderung zu verarbeiten. Die nötigen Daten können in einem Cache innerhalb der Ausführungsplattform **112** (wie es nachstehend noch ausführlicher diskutiert wird) oder in einer Datenspeichervorrichtung in der Speicherplattform **114** gespeichert werden. Der Aufforderungsverarbeitungsdienst **208** kann unter Verwendung eines „Aufforderungsverarbeitungsmoduls“ implementiert sein. Ein Managementkonsolendienst **210** unterstützt einen Zugriff auf verschiedene Systeme und Prozesse durch Administratoren und andere Systemmanager. Zusätzlich kann der Managementkonsolendienst **210** Aufforderungen von den Anwendern **104-108** empfangen, um Anfragen auszugeben und um die Arbeitsbelastung auf dem System zu überwachen. Der Managementkonsolendienst **210** kann unter Verwendung eines „Managementkonsolenmoduls“ implementiert sein. Bei einigen Ausführungsformen kann ein bestimmter Anwender eine Aufforderung ausgeben, die Arbeitsbelastung zu überwachen, die seine spezifische Anfrage auf dem System platziert.

[0026] Der Ressourcenmanager **102** enthält auch einen SQL-Compiler **212**, einen SQL-Optimierer **214** und einen SQL-Ausführer **210**. Der SQL-Compiler **212** parst SQL-Anfragen und erzeugt den Ausführungscode für die Anfragen. Der SQL-Optimierer **214** bestimmt das beste Verfahren zum Ausführen von Anfragen basierend auf den Daten, die verarbeitet werden müssen. Der SQL-Optimierer **214** handhabt auch verschiedene Datenbeschneidungsoperationen bzw. Daten-Pruning-Operationen und andere Datenoptimierungstechniken, um die Geschwindigkeit und die Effizienz zum Ausführen der SQL-Anfrage zu ver-

bessern. Der SQL-Ausführer **216** führt den Anfragecode für durch den Ressourcenmanager **102** empfangene Anfragen aus.

[0027] Ein Anfrageplaner und -koordinator **218** sendet empfangene Anfragen zu den geeigneten Diensten oder Systemen zur Kompilation, Optimierung und Abfertigung zu der Ausführungsplattform **112**. Beispielsweise können Anfragen priorisiert und in dieser priorisierten Reihenfolge verarbeitet werden. Bei einigen Ausführungsformen identifiziert der Anfrageplaner und -koordinator **218** bestimmte Knoten in der Ausführungsplattform **112** oder ordnet sie zu, um bestimmte Anfragen zu verarbeiten. Ein Manager für ein virtuelles Lager **220** managt die Operation bzw. den Betrieb von mehreren virtuellen Lagern, die in der Ausführungsplattform **112** implementiert sind. Wie es nachstehend diskutiert ist, enthält jedes virtuelle Lager mehrere Ausführungsknoten, die jeweils einen Cache und einen Prozessor enthalten.

[0028] Zusätzlich enthält der Ressourcenmanager **102** eine Konfigurations- und Metadatenmanager **222**, der die Information in Bezug auf die Daten managt, die in den entfernten Datenspeichervorrichtungen und in den lokalen Caches (d.h. den Caches in der Ausführungsplattform **112**) gespeichert sind. Wie es nachstehend noch ausführlicher diskutiert wird, verwendet der Konfigurations- und Metadatenmanager **222** die Metadaten, um zu bestimmen, auf welche Datendateien zugegriffen werden muss, um Daten zur Verarbeitung einer bestimmten Anfrage wiederzugewinnen. Ein Überwachungs- und Arbeitsbelastungsanalysator **224** überblickt die durch den Ressourcenmanager **102** durchgeführten Prozesse und managt die Verteilung von Aufgaben (z.B. Arbeitsbelastung) quer über die virtuellen Lager und Ausführungsknoten in der Ausführungsplattform **112**. Der Überwachungs- und Arbeitsbelastungsanalysator **224** verteilt Aufgaben auch neu, wie es nötig ist, basierend auf sich ändernden Arbeitsbelastungen in der gesamten Datenverarbeitungsplattform **100**. Der Konfigurations- und Metadatenmanager **222** und der Überwachungs- und Arbeitsbelastungsanalysator **224** sind mit einer Datenspeichervorrichtung **226** gekoppelt. Die Datenspeichervorrichtungen **206** und **226** in **Fig. 2** stellen irgendeine Datenspeichervorrichtung innerhalb der Datenverarbeitungsplattform **100** dar. Beispielsweise können die Datenspeichervorrichtungen **206** und **226** Caches in der Ausführungsplattform **112**, Speichervorrichtungen in der Speicherplattform **114** oder irgendeine andere Speichervorrichtung darstellen.

[0029] Der Ressourcenmanager **102** enthält auch ein Transaktionsmanagement- und Zugriffssteuermodul **228**, das die verschiedenen Aufgaben und andere mit der Verarbeitung von Datenspeicheraufforderungen und Datenzugriffsaufforderungen assoziierte Aktivitäten managt. Beispielsweise stellt das

Transaktionsmanagement- und Zugriffssteuermodul **228** einen konsistenten und synchronisierten Zugriff auf Daten durch mehrere Anwender oder Systeme zur Verfügung. Da mehrere Anwender/Systeme auf dieselben Daten gleichzeitig zugreifen können, müssen Änderungen an den Daten synchronisiert werden, um sicherzustellen, dass jeder Anwender/jedes System mit der aktuellen Version der Daten arbeitet. Das Transaktionsmanagement- und Zugriffssteuermodul **228** stellt eine Steuerung von verschiedenen Datenverarbeitungsaktivitäten bei einer einzelnen zentralisierten Stelle im Ressourcenmanager **102** zur Verfügung. Bei einigen Ausführungsformen interagiert das Transaktionsmanagement- und Zugriffssteuermodul **228** mit dem SQL-Ausführer **216**, um das Management von verschiedenen Aufgaben zu unterstützen, die durch den SQL-Ausführer **216** ausgeführt werden.

[0030] **Fig. 3** ist ein Blockdiagramm, das eine Ausführungsform einer Ausführungsplattform **112** darstellt. Wie es in **Fig. 3** gezeigt ist, enthält die Ausführungsplattform **112** mehrere virtuelle Lager **302**, **304** und **306**. Jedes virtuelle Lager enthält mehrere Ausführungsknoten, die jeweils einen Datencache und einen Prozessor enthalten. Die virtuellen Lager **302**, **304** und **306** können mehrere Anfragen (und andere Aufgaben) durch Verwenden der mehreren Ausführungsknoten parallel ausführen. Wie es hierin diskutiert ist, kann die Ausführungsplattform **112** basierend auf den aktuellen Verarbeitungsnotwendigkeiten der Systeme und Anwender in Echtzeit neue virtuelle Lager hinzufügen und existierende virtuelle Lager fallenlassen. Diese Flexibilität lässt zu, dass die Ausführungsplattform **112** große Mengen von Computerressourcen schnell nutzt, wenn es nötig ist, ohne dazu gezwungen zu sein, ein Bezahlen für diese Computerressourcen fortzusetzen, wenn sie nicht länger benötigt werden. Alle virtuellen Lager können auch auf Daten von irgendeiner Datenspeichervorrichtung (z.B. irgendeiner Speichervorrichtung in der Speicherplattform **114**) zugreifen.

[0031] Obwohl jedes virtuelle Lager **302-306**, die in **Fig. 3** gezeigt sind, drei Ausführungsknoten enthält, kann ein bestimmtes virtuelles Lager irgendeine Anzahl von Ausführungsknoten enthalten. Weiterhin ist die Anzahl von Ausführungsknoten in einem virtuellen Lager dynamisch, so dass neue Ausführungsknoten erzeugt werden, wenn ein zusätzlicher Bedarf vorhanden ist, und existierende Ausführungsknoten gelöscht werden, wenn sie nicht länger nötig sind.

[0032] Jedes virtuelle Lager **302-306** kann auf irgendeine der Datenspeichervorrichtungen **116-120** zugreifen, die in **Fig. 1** gezeigt sind. Somit sind die virtuellen Lager **302-306** nicht notwendigerweise einer spezifischen Datenspeichervorrichtung **116-120** zugeordnet und können stattdessen auf Daten von irgendeiner der Datenspeichervorrichtungen **116-120**

zugreifen. Gleichmaßen kann jeder der in **Fig. 3** gezeigten Ausführungsknoten auf Daten von irgendeiner der Datenspeichervorrichtungen **116-120** zugreifen. Bei einigen Ausführungsformen kann ein bestimmtes virtuelles Lager oder ein bestimmter Ausführungsknoten temporär einer spezifischen Datenspeichervorrichtung zugeordnet sein, aber das virtuelle Lager oder der Ausführungsknoten kann später auf Daten von irgendeiner anderen Datenspeichervorrichtung zugreifen.

[0033] Bei dem Beispiel der **Fig. 3** enthält das virtuelle Lager **302** drei Ausführungsknoten **308**, **310** und **312**. Der Ausführungsknoten **308** enthält einen Cache **314** und einen Prozessor **316**. Der Ausführungsknoten **310** enthält einen Cache **318** und einen Prozessor **320**. Der Ausführungsknoten **312** enthält einen Cache **322** und einen Prozessor **324**. Jeder Ausführungsknoten **308-312** ist mit einer Verarbeitung von einem oder mehreren von einer oder mehreren von Datenspeicher- und/oder Datenwiedergewinnungsaufgaben assoziiert. Zum Beispiel kann ein bestimmtes virtuelles Lager mit einem bestimmten Anwender oder Kunden assoziierte Datenspeicher- und Datenwiedergewinnungsaufgaben handhaben. Bei anderen Implementierungen kann ein bestimmtes virtuelles Lager mit einem bestimmten Datenspeichersystem oder einer bestimmten Kategorie von Daten assoziierte Datenspeicher- und Datenwiedergewinnungsaufgaben handhaben.

[0034] Gleich dem oben diskutierten virtuellen Lager **302** enthält das virtuelle Lager **304** drei Ausführungsknoten **326**, **328** und **330**. Der Ausführungsknoten **326** enthält einen Cache **332** und einen Prozessor **334**. Der Ausführungsknoten **328** enthält einen Cache **336** und einen Prozessor **338**. Der Ausführungsknoten **330** enthält einen Cache **340** und einen Prozessor **342**. Zusätzlich enthält das virtuelle Lager **306** drei Ausführungsknoten **344**, **346** und **348**. Der Ausführungsknoten **344** enthält einen Cache **350** und einen Prozessor **352**. Der Ausführungsknoten **346** enthält einen Cache **354** und einen Prozessor **356**. Der Ausführungsknoten **348** enthält einen Cache **358** und einen Prozessor **360**.

[0035] Bei einigen Ausführungsformen sind die in **Fig. 3** gezeigten Ausführungsknoten zustandslos in Bezug auf die Daten und sind die Ausführungsknoten als Cache speichernd bzw. zwischenspeichernd. Beispielsweise speichern diese Ausführungsknoten Zustandsinformation über den Ausführungsknoten oder die Daten, die gerade durch einen bestimmten Ausführungsknoten im Cache gespeichert sind, nicht oder behalten diese nicht auf andere Weise bei. Somit kann in dem Fall eines Ausführungsknotenfehlers bzw. -ausfalls der Knoten mit Fehler bzw. ausgefallene Knoten transparent durch einen anderen Knoten ersetzt werden. Da es keine mit dem fehlgeschlagenen Ausführungsknoten assoziierte Zustandsinfor-

mation gibt, kann der neue (Austausch- bzw. Ersatz-)Ausführungsknoten auf einfache Weise den fehlgeschlagenen Knoten ersetzen, ohne sich um eine Wiedererstellung eines bestimmten Zustands zu kümmern.

[0036] Obwohl die in **Fig. 3** gezeigten Ausführungsknoten jeweils einen Datencache und einen Prozessor enthalten, können alternative Ausführungsformen Ausführungsknoten enthalten, die irgendeine Anzahl von Prozessoren und irgendeine Anzahl von Caches enthalten. Zusätzlich können die Caches bezüglich einer Größe zwischen den unterschiedlichen Ausführungsknoten variieren. Die in **Fig. 3** gezeigten Caches speichern in dem lokalen Ausführungsknoten Daten, die aus einer oder mehreren Datenspeichervorrichtungen in der Speicherplattform **114** wiedergewonnen wurden (**Fig. 1**). Somit reduzieren oder eliminieren die Caches die Engpassprobleme, die bei Plattformen auftreten, die Daten aus entfernten Speichersystemen konsistent wiedergewinnen. Anstelle eines wiederholten Zugreifens auf Daten von den entfernten Speichervorrichtungen greifen die hierin beschriebenen Systeme und Verfahren auf Daten aus den Caches in den Ausführungsknoten zu, was signifikant schneller ist und das oben diskutierte Engpassproblem vermeidet. Bei einigen Ausführungsformen sind die Caches unter Verwendung von Hochgeschwindigkeitsspeichervorrichtungen implementiert, die einen schnellen Zugriff auf die im Cache gespeicherten Daten zur Verfügung stellen. Jeder Cache kann Daten von irgendeiner der Speichervorrichtungen in der Speicherplattform **114** speichern.

[0037] Weiterhin können die Cacheressourcen und die Computerressourcen zwischen unterschiedlichen Ausführungsknoten variieren. Beispielsweise kann ein Ausführungsknoten signifikante Computerressourcen und minimale Cacheressourcen enthalten, was den Ausführungsknoten nützlich für Aufgaben macht, die signifikante Computerressourcen erfordern. Ein anderer Ausführungsknoten kann signifikante Cacheressourcen und minimale Computerressourcen enthalten, was diesen Ausführungsknoten nützlich für Aufgaben macht, die eine Speicherung von großen Mengen von Daten in Caches erfordern. Noch ein weiterer Ausführungsknoten kann Cacheressourcen enthalten, die schnellere Eingabe-Ausgabe-Operationen zur Verfügung stellen, was nützlich für Aufgaben ist, die ein schnelles Scannen von großen Mengen von Daten erfordern. Bei einigen Ausführungsformen werden die mit einem bestimmten Ausführungsknoten assoziierten Cacheressourcen und Computerressourcen dann, wenn der Ausführungsknoten erzeugt wird, basierend auf den erwarteten durch den Ausführungsknoten durchzuführenden Aufgaben bestimmt.

[0038] Zusätzlich können sich die mit einem bestimmten Ausführungsknoten assoziierten Cacheressourcen und Computerressourcen im Verlaufe der Zeit basierend auf durch den Ausführungsknoten durchgeführten sich ändernden Aufgaben ändern. Beispielsweise können einem bestimmten Ausführungsknoten mehr Verarbeitungsressourcen zugeordnet werden, wenn die durch den Ausführungsknoten durchgeführten Aufgaben mehr prozessorintensiv werden. Gleichmaßen können einem Ausführungsknoten mehr Cacheressourcen zugeordnet werden, wenn die durch den Ausführungsknoten durchgeführten Aufgaben eine größere Cachekapazität erfordern.

[0039] Obwohl virtuelle Lager **302-306** mit derselben Ausführungsplattform **112** assoziiert sind, können die virtuellen Lager unter Verwendung von mehreren Computersystemen an mehreren geographischen Standorten implementiert sein. Beispielsweise kann das virtuelle Lager **302** durch ein Computersystem an einem ersten geographischen Standort implementiert sein, während die virtuellen Lager **304** und **306** durch ein weiteres Computersystem an einem zweiten geographischen Standort implementiert sind. Bei einigen Ausführungsformen sind diese unterschiedlichen Computersysteme cloudbasierte Computersysteme, die durch eine oder mehrere unterschiedliche Einheiten gewartet werden.

[0040] Zusätzlich ist jedes virtuelle Lager in **Fig. 3** derart gezeigt, dass es mehrere Ausführungsknoten hat. Die mit jedem virtuellen Lager assoziierten mehreren Ausführungsknoten können unter Verwendung von mehreren Computersystemen an mehreren geographischen Standorten implementiert sein. Beispielsweise implementiert ein besonderer Fall eines virtuellen Lagers **302** Ausführungsknoten **308** und **310** auf einer Computerplattform an einem bestimmten geographischen Standort und implementiert den Ausführungsknoten **312** auf einer anderen Computerplattform an einem anderen geographischen Standort. Ein Auswählen von bestimmten Computersystemen, um einen Ausführungsknoten zu implementieren, kann von verschiedenen Faktoren abhängen, wie beispielsweise dem Ausmaß von Ressourcen, die für einen bestimmten Ausführungsknoten nötig sind (z.B. Verarbeitungsressourcenerfordernissen und Cacheerfordernissen), den Ressourcen, die bei bestimmten Computersystemen verfügbar sind, Kommunikationsfähigkeiten von Netzwerken innerhalb eines geographischen Standorts oder zwischen geographischen Standorten und welche Computersysteme bereits andere Ausführungsknoten in dem virtuellen Lager implementieren.

[0041] Die Ausführungsplattform **112** ist auch fehler-tolerant. Wenn beispielsweise ein virtuelles Lager einen Fehler hat bzw. ausfällt, wird dieses virtuelle La-

ger schnell durch ein anderes virtuelles Lager an einem anderen geographischen Standort ersetzt.

[0042] Eine bestimmte Ausführungsplattform **112** kann irgendeine Anzahl von virtuellen Lagern **302-306** enthalten. Zusätzlich ist die Anzahl von virtuellen Lagern in einer bestimmten Ausführungsplattform dynamisch, so dass neue virtuelle Lager erzeugt werden, wenn zusätzliche Verarbeitungs- und/oder Cacheressourcen nötig sind. Gleichmaßen können existierende virtuelle Lager gelöscht werden, wenn die mit dem virtuellen Lager assoziierten Ressourcen nicht länger nötig sind.

[0043] Bei einigen Ausführungsformen können die virtuellen Lager **302**, **304** und **306** an denselben Daten in der Speicherplattform **114** arbeiten, aber jedes virtuelle Lager hat seine eigenen Ausführungsknoten mit unabhängigen Verarbeitungs- und Cacheressourcen. Diese Konfiguration lässt zu, dass Aufforderungen an unterschiedliche virtuelle Lager unabhängig und ohne Beeinträchtigung zwischen den Aufforderungen verarbeitet werden. Diese unabhängige Verarbeitung, kombiniert mit der Fähigkeit, virtuelle Lager dynamisch hinzuzufügen und zu entfernen, unterstützt das Hinzufügen von neuer Verarbeitungskapazität für neue Anwender, ohne die Leistungsfähigkeit zu beeinträchtigen, die durch die existierenden Anwender beobachtet wird.

[0044] **Fig. 4** ist ein Blockdiagramm, das eine beispielhafte Betriebsumgebung **400** mit mehreren Anwendern darstellt, die über mehrere virtuelle Lager auf mehrere Datenbanken zugreifen. In der Umgebung **400** greifen mehrere Anwender **402**, **404** und **406** über mehrere virtuelle Lager **408**, **410** und **412** auf mehrere Datenbanken **414**, **416**, **418**, **420**, **422** und **424** zu. Obwohl es in **Fig. 4** nicht gezeigt ist, können die Anwender **402**, **404** und **406** auf die virtuellen Lager **408**, **410** und **412** über den Ressourcenmanager **102** (**Fig. 1**) zugreifen. Bei bestimmten Ausführungsformen sind Datenbanken **414-424** in der Speicherplattform **114** (**Fig. 1**) enthalten und durch irgendein in der Ausführungsplattform **112** implementiertes virtuelles Lager zugreifbar. Bei einigen Ausführungsformen greifen die Anwender **402-406** auf eines der virtuellen Lager **408-412** unter Verwendung eines Datenkommunikationsnetzwerks, wie beispielsweise das Internet, zu. Bei einigen Implementierungen spezifiziert jeder Anwender **402-406** ein bestimmtes virtuelles Lager **408-412**, um zu einer spezifischen Zeit mit ihm zu arbeiten. Bei dem Beispiel der **Fig. 4** interagiert der Anwender **402** mit dem virtuellen Lager **408**, interagiert der Anwender **404** mit dem virtuellen Lager **410** und interagiert der Anwender **406** mit dem virtuellen Lager **412**. Somit reicht der Anwender **402** Datenwiedergewinnungs- und Datenspeicheraufforderungen über das virtuelle Lager **408** ein. Gleichmaßen reichen die Anwender **404** und **406** Daten-

wiedergewinnungs- und Datenspeicheraufforderungen jeweils über die virtuellen Lager **410** und **412** ein.

[0045] Jedes virtuelle Lager **408-412** ist konfiguriert, um mit einer Untergruppe von allen Datenbanken **414-424** zu kommunizieren. Beispielsweise ist in der Umgebung **400** das virtuelle Lager **408** konfiguriert, um mit den Datenbanken **414**, **416** und **422** zu kommunizieren. Gleichermaßen ist das virtuelle Lager **410** konfiguriert, um mit den Datenbanken **416**, **418**, **420** und **424** zu kommunizieren. Und das virtuelle Lager **412** ist konfiguriert, um mit den Datenbanken **416**, **422** und **424** zu kommunizieren. Bei alternativen Ausführungsformen kommuniziert oder kommunizieren eines oder mehrere der virtuellen Lager **408-412** mit allen von den Datenbanken **414-424**. Die in **Fig. 4** gezeigte Anordnung lässt zu, dass einzelne Anwender alle Datenwiedergewinnungs- und Datenspeicheraufforderungen über ein einziges virtuelles Lager senden. Dieses virtuelle Lager verarbeitet die Datenwiedergewinnungs- und Datenspeicheraufgaben unter Verwendung von im Cache gespeicherten Daten innerhalb von einem der Ausführungsknoten in dem virtuellen Lager oder gewinnt die nötigen Daten aus einer geeigneten Datenbank wieder (und speichert sie im Cache). Das Abbilden zwischen den virtuellen Lagern ist ein logisches Abbilden, kein hardwaremäßiges Abbilden. Dieses logische Abbilden basiert auf Zugriffssteuerparametern in Bezug auf Sicherheits- und Ressourcenzugriffsmangementeinstellungen. Die logischen Abbildungen werden auf einfache Weise geändert, ohne eine Rekonfiguration der virtuellen Lager oder der Speicherressourcen zu erfordern.

[0046] Obwohl die Umgebung **400** virtuelle Lager **408-412** zeigt, die konfiguriert sind, um mit spezifischen Untergruppen von Datenbanken **414-424** zu kommunizieren, ist diese Konfiguration dynamisch. Zum Beispiel kann das virtuelle Lager **408** rekonfiguriert werden, um mit einer anderen Untergruppe von Datenbanken **414-424** zu kommunizieren, basierend auf sich ändernden Aufgaben, um durch das virtuelle Lager **408** durchgeführt zu werden. Beispielsweise dann, wenn das virtuelle Lager **408** Aufforderungen empfängt, auf Daten von der Datenbank **418** zuzugreifen, kann das virtuelle Lager **408** rekonfiguriert werden, um auch mit der Datenbank **418** zu kommunizieren. Wenn es das virtuelle Lager **408** zu einer späteren Zeit nicht länger nötig hat, auf Daten von der Datenbank **418** zuzugreifen, kann das virtuelle Lager **408** rekonfiguriert werden, um die Kommunikation mit der Datenbank **418** zu löschen.

[0047] **Fig. 5** ist ein Blockdiagramm, das eine weitere beispielhafte Betriebsumgebung **500** mit mehreren Anwendern darstellt, die auf mehrere Datenbanken über einen Lastausgleicher und in einer virtuellen Lagergruppe enthaltene mehrere virtuelle Lager zugreifen. Die Umgebung **500** ist ähnlich der Umge-

bung **400** (**Fig. 4**), enthält aber zusätzlich einen Ressourcenmanager **508** für virtuelle Lager und mehrere virtuelle Lager **510**, **512** und **514**, die in einer virtuellen Lagergruppe **516** angeordnet sind. Der Ressourcenmanager **508** für virtuelle Lager kann im Ressourcenmanager **102** enthalten sein. Insbesondere greifen mehrere Anwender **502**, **504** und **506** auf mehrere Datenbanken **518**, **520**, **522**, **524**, **526** und **528** über den Ressourcenmanager **508** für virtuelle Lager und die virtuelle Lagergruppe **516** zu. Bei einigen Ausführungsformen greifen die Anwender **502-506** auf den Ressourcenmanager **508** für virtuelle Lager unter Verwendung eines Datenkommunikationsnetzwerks, wie beispielsweise das Internet, zu. Obwohl es in **Fig. 5** nicht gezeigt ist, können die Anwender **502**, **504** und **506** auf den Ressourcenmanager **508** für virtuelle Lager über den Ressourcenmanager **102** (**Fig. 1**) zugreifen. Bei einigen Ausführungsformen ist der Ressourcenmanager **508** für virtuelle Lager innerhalb des Ressourcenmanagers **102** implementiert.

[0048] Die Anwender **502-506** können Datenwiedergewinnungs- und Datenspeicheraufforderungen zu dem Ressourcenmanager **508** für virtuelle Lager einreichen, der die Datenwiedergewinnungs- und Datenspeicheraufforderungen zu einem geeigneten virtuellen Lager **510-514** in der virtuellen Lagergruppe **516** weiterleitet. Bei einigen Implementierungen stellt der Ressourcenmanager **508** für virtuelle Lager eine dynamische Zuordnung der Anwender **502-506** zu den virtuellen Lagern **510-514** zur Verfügung. Wenn sie eine Datenwiedergewinnungs- oder Datenspeicheraufforderung einreichen, können die Anwender **502-506** die virtuelle Lagergruppe **516** spezifizieren, um die Aufforderung zu verarbeiten, ohne das bestimmte virtuelle Lager **510-514** zu spezifizieren, das die Aufforderung verarbeiten wird. Diese Anordnung lässt zu, dass der Ressourcenmanager **508** für virtuelle Lager mehrere Aufforderungen basierend auf einer Effizienz, verfügbaren Ressourcen und der Verfügbarkeit von in Cache gespeicherten Daten innerhalb der virtuellen Lager **510-514** quer über die virtuellen Lager **510-514** verteilt. Wenn er bestimmt, wie Datenverarbeitungsaufforderungen weiterzuleiten sind, berücksichtigt der Ressourcenmanager **508** für virtuelle Lager verfügbare Ressourcen, aktuelle Ressourcenbelastungen, eine Anzahl von aktuellen Anwendern und ähnliches.

[0049] Bei einigen Ausführungsformen erzeugen Fehlertoleranzsysteme ein neues virtuelles Lager in Reaktion auf einen Fehler bzw. Ausfall eines virtuellen Lagers. Das neue virtuelle Lager kann in derselben virtuellen Lagergruppe sein oder kann in einer anderen virtuellen Lagergruppe an einem anderen geographischen Standort erzeugt werden.

[0050] Jedes virtuelle Lager **510-514** ist konfiguriert, um mit einer Untergruppe von allen Datenbanken

518-528 zu kommunizieren. Zum Beispiel ist das virtuelle Lager **510** in der Umgebung **500** konfiguriert, um mit den Datenbanken **518**, **520** und **526** zu kommunizieren. Gleichmaßen ist das virtuelle Lager **512** konfiguriert, um mit den Datenbanken **520**, **522**, **524** und **528** zu kommunizieren. Und das virtuelle Lager **514** ist konfiguriert, um mit den Datenbanken **520**, **526** und **528** zu kommunizieren. Bei alternativen Ausführungsformen können die virtuellen Lager **510-514** mit irgendwelchen (oder allen) der Datenbanken **518-528** kommunizieren.

[0051] Obwohl die Umgebung **500** eine virtuelle Lagergruppe **516** zeigt, können alternative Ausführungsformen irgendeine Anzahl von virtuellen Lagergruppen enthalten, wobei jede mit irgendeiner Anzahl von virtuellen Lagern assoziiert ist. Zum Beispiel können unterschiedliche virtuelle Lager für jeden Kunden oder eine Gruppe von Anwendern erzeugt werden. Zusätzlich können unterschiedliche Lager für unterschiedliche Einheiten oder irgendeine andere Gruppe, die auf unterschiedliche Datensätze zugreift, erzeugt werden. Mehrere virtuelle Lagergruppen können unterschiedliche Größen und Konfigurationen haben. Die Anzahl von virtuellen Lagergruppen in einer bestimmten Umgebung ist dynamisch und kann sich basierend auf den sich ändernden Notwendigkeiten der Anwender und anderer Systeme in der Umgebung ändern.

[0052] Fig. 6 ist ein Blockdiagramm, das eine weitere beispielhaften Betriebsumgebung **600** zeigt, die mehrere verteilte virtuelle Lager und virtuelle Lagergruppen hat. Die Umgebung **600** enthält den Ressourcenmanager **102**, der mit den virtuellen Lagergruppen **604** und **606** über ein Datenkommunikationsnetzwerk **602** kommuniziert. Die Lagergruppe **604** enthält zwei virtuelle Lager **608** und **610** und die Lagergruppe **606** enthält weitere zwei virtuelle Lager **614** und **616**. Der Ressourcenmanager **102** kommuniziert auch mit dem virtuellen Lager **612** (das nicht Teil einer virtuellen Lagergruppe ist) über das Datenkommunikationsnetzwerk **602**.

[0053] Die virtuellen Lagergruppen **604** und **606** sowie das virtuelle Lager **612** kommunizieren mit Datenbanken **620**, **622** und **624** über ein Datenkommunikationsnetzwerk **618**. Bei einigen Ausführungsformen sind die Datenkommunikationsnetzwerke **602** und **618** dasselbe Netzwerk. Die Umgebung **600** lässt zu, dass der Ressourcenmanager **102** Anwenderdatenspeicherungs- und -wiedergewinnungsaufforderungen quer über die mehreren virtuellen Lager **608-616** koordiniert, um Daten in den Datenbanken **620-624** zu speichern und wiederzugewinnen. Die virtuellen Lagergruppen **604** und **606** können in demselben geographischen Bereich angeordnet sein oder können geographisch getrennt sein. Zusätzlich können die virtuellen Lagergruppen **604** und

606 durch dieselbe Einheit oder durch unterschiedliche Einheiten implementiert sein.

[0054] Die hierin beschriebenen Systeme und Verfahren lassen zu, dass Daten, die gespeichert werden sollen und auf die zugegriffen werden soll als Service, getrennt von Computer- (oder Verarbeitungs-) Ressourcen sind. Selbst wenn keine Computerressourcen von der Ausführungsplattform zugeteilt worden sind, sind Daten für ein virtuelles Lager verfügbar, ohne ein erneutes Laden der Daten von einer entfernten Datenquelle zu erfordern. Somit sind Daten unabhängig von der Zuteilung von mit den Daten assoziierten Computerressourcen verfügbar. Die beschriebenen Systeme und Verfahren sind nützlich mit irgendeinem Typ von Daten. Bei bestimmten Ausführungsformen werden Daten in einem strukturierten, optimierten Format gespeichert. Die Entkopplung des Datenspeichers/Zugriffsdienstes von den Computerdiensten vereinfacht auch das gemeinsame Nutzen bzw. Teilen von Daten unter unterschiedlichen Anwendern und Gruppen. Wie es hierin diskutiert ist, kann jedes virtuelle Lager auf irgendwelche Daten zugreifen, zu welchen es Zugriffserlaubnisse hat, selbst zur gleichen Zeit, zu welcher andere virtuellen Lager auf dieselben Daten zugreifen. Diese Architektur unterstützt laufende Anfragen ohne irgendwelche im lokalen Cache gespeicherten tatsächlichen Daten. Die hierin beschriebenen Systeme und Verfahren sind auf eine Weise, die für den Anwender des Systems transparent ist, zu einer transparenten dynamischen Datenbewegung fähig, welche Daten von einer entfernten Speichervorrichtung zu einem lokalen Cache bewegt, wie es nötig ist. Weiterhin unterstützt diese Architektur ein gemeinsames Nutzen bzw. Teilen von Daten ohne eine vorherige Datenbewegung, da aufgrund der Entkopplung des Datenspeicherdienstes vom Computerdienst irgendein virtuelles Lager auf irgendwelche Daten zugreifen kann.

[0055] Fig. 7 ist ein Ablaufdiagramm, das eine Ausführungsform eines Verfahrens **700** zum Managen von Datenspeicherungs- und -wiedergewinnungsoperationen darstellt. Zu Anfang empfängt das Verfahren **700** bei **702** eine Anweisung, eine Aufforderung oder eine Anfrage von einem Anwender. Eine Anweisung ist irgendeine Aufforderung oder ein Befehl, um eine datenbezogene Operation durchzuführen. Beispielhafte Anweisungen enthalten Datenwiedergewinnungsaufforderungen, Datenspeicherungsaufforderungen, Datenübertragungsaufforderungen, Datenanfragen und ähnliches. Bei einigen Ausführungsformen ist die Anweisung als eine SQL-Anweisung implementiert. Ein Ressourcenmanager erzeugt einen Anfragekoordinator bei **704**, um die empfangene Anweisung zu managen. Beispielsweise managt der Anfragekoordinator die verschiedenen Aufgaben, die dazu nötig sind, die empfangene Anweisung zu verarbeiten, einschließlich eines Interagierens mit einer Ausführungsplattform und einer

oder mehreren Datenspeichervorrichtungen. Bei einigen Ausführungsformen ist der Anfragekoordinator eine temporäre Routine, die spezifisch erzeugt ist, um die empfangene Anweisung zu managen.

[0056] Das Verfahren **700** fährt fort, indem der Ressourcenmanager bei **706** mehrere Aufgaben bestimmt, die dazu nötig sind, die empfangene Anweisung zu verarbeiten. Die mehreren Aufgaben können beispielsweise ein Zugreifen auf Daten aus einem Cache in einem Ausführungsknoten, ein Wiedergewinnen von Daten aus einer entfernten Speichervorrichtung, ein Updaten von Daten in einem Cache, ein Speichern von Daten in einer entfernten Speichervorrichtung und ähnliches enthalten. Der Ressourcenmanager verteilt bei **708** auch die mehreren Aufgaben auf Ausführungsknoten in der Ausführungsplattform. Wie es hierin diskutiert ist, sind die Ausführungsknoten in der Ausführungsplattform innerhalb von virtuellen Lagern implementiert. Bei **710** führt jeder Ausführungsknoten eine zugeordnete Aufgabe durch und bringt ein Aufgabenergebnis zum Ressourcenmanager zurück. Bei einigen Ausführungsformen bringen die Ausführungsknoten die Aufgabenergebnisse zum Anfragekoordinator zurück. Bei **712** empfängt der Ressourcenmanager die mehreren Aufgabenergebnisse und erzeugt ein Anweisungsergebnis und kommuniziert bei **714** das Anweisungsergebnis zum Anwender. Bei einigen Ausführungsformen wird der Anfragekoordinator gelöscht, nachdem das Anweisungsergebnis zum Anwender kommuniziert ist.

[0057] Fig. 8 ist ein Ablaufdiagramm, das eine Ausführungsform eines Verfahrens **800** zum Bereitstellen von neuen virtuellen Lagern darstellt. Anfangs überwacht ein Ressourcenmanager (z.B. der Ressourcenmanager **102**) bei **802** von mehreren Anwendern empfangene Datenverarbeitungsaufforderungen. Wie es hierin diskutiert ist, können diese Datenverarbeitungsaufforderungen Datensuchanfragen, Datenleseaufforderungen, Datenschreibaufforderungen und ähnliches enthalten. Der Ressourcenmanager überwacht bei **804** auch eine gegenwärtige bzw. aktuelle Ressourcennutzung, Anfragereaktionsraten, eine Anzahl von Anwendern, die mit einem jeweiligen virtuellen Lager interagieren, und andere Leistungskennzahlen für die existierenden virtuellen Lager. Die aktuelle Ressourcennutzung enthält zum Beispiel eine Cachenutzung und eine Prozessornutzung in einem oder mehreren Ausführungsknoten in den virtuellen Lagern. Die aktuelle Ressourcennutzung kann als Prozentsatz einer maximalen Nutzung ausgedrückt werden (z.B. 40% Nutzung eines Caches und 60% Nutzung eines Prozessors). Die Anfragereaktionsraten können zum Beispiel als eine Durchschnittszeit ausgedrückt werden, die erforderlich ist, um auf Anwenderanfragen zu antworten bzw. zu reagieren. Bei einigen Ausführungsformen überwacht der Ressourcenmanager auch eine Anzahl von Anfragen, die aktuell laufen bzw. in Be-

trieb sind, einen Prozentsatz einer maximalen Last bzw. Auslastung, ob irgendeine Anfrageverarbeitung verzögert werden muss, und ähnliches.

[0058] Das Verfahren **800** fährt bei **806** fort, wenn der Ressourcenmanager aktuellen und zukünftigen Bedarf an Ressourcen bestimmt. Zum Beispiel kann der Ressourcenmanager anhängige Datenverarbeitungsaufforderungen sowie erwartete Aufforderungen in der nahen Zukunft identifizieren. Die erwarteten Aufforderungen können basierend auf vorherigen Mustern von zuvor empfangenen Datenverarbeitungsaufforderungen von bestimmten Anwendern zu bestimmten Zeiten bestimmt werden. Zusätzlich kann der Ressourcenmanager eine Vorankündigung eines Datenverarbeitungsprojekts empfangen und kann die Ressourcen bestimmen, die nötig sind, um dieses Projekt zu handhaben. Der Ressourcenmanager bestimmt dann bei **808**, ob ein oder mehrere zusätzliche virtuelle Lager nötig sind, basierend auf den aktuellen Datenverarbeitungsaufforderungen, der aktuellen Ressourcennutzung, Anfragereaktionsraten und anderen Leistungskennzahlen, die mit den existierenden virtuellen Lagern assoziiert sind. Wenn ein zusätzliches virtuelles Lager nötig ist, stellt der Ressourcenmanager bei **810** ein neues virtuelles Lager bereit. Wenn der Ressourcenmanager zum Beispiel Kenntnis von einem bevorstehenden Datenverarbeitungsprojekt hat, das mehr Ressourcen erfordert wird als aktuell verfügbar sind, kann der Ressourcenmanager entscheiden, ein oder mehrere neue virtuelle Lager bereitzustellen, um das bevorstehende Datenverarbeitungsprojekt zu handhaben. Das neue virtuelle Lager wird schnell bereitgestellt, so dass das neue virtuelle Lager dazu bereit ist, die Datenverarbeitungsaufforderung sofort zur Startzeit des Projekts zu handhaben.

[0059] Zusätzlich zum Hinzufügen neuer virtueller Lager kann das Verfahren **800** bei **812** bestimmen, ob ein oder mehrere virtuelle Lager zu deaktivieren ist oder sind. Wenn irgendwelche virtuellen Lager nicht mehr nötig sind, deaktiviert der Ressourcenmanager bei **814** ein oder mehrere virtuelle Lager. Wenn zum Beispiel ein bestimmtes virtuelles Lager für ein spezifisches Datenverarbeitungsprojekt erzeugt wurde, kann dieses virtuelle Lager deaktiviert werden, wenn das spezifische Projekt beendet ist. Zusätzlich können dann, wenn viele virtuelle Lager mit niedriger Ressourcennutzung arbeiten bzw. in Betrieb sind, einige der existierenden virtuellen Lager deaktiviert werden, ohne die Leistungsfähigkeit der übrigen virtuellen Lager zu verschlechtern. Bei einigen Ausführungsformen wird ein virtuelles Lager für eine spezifische Zeitperiode erzeugt. Nachdem diese Zeitperiode verstrichen ist, kann das virtuelle Lager deaktiviert werden. Bei anderen Ausführungsformen identifiziert der Ressourcenmanager virtuelle Lager, die für einen bestimmten Zeitraum nicht in Betrieb gewe-

sen sind, und deaktiviert diese virtuellen Lager automatisch.

[0060] In einigen Situationen können bestimmte Anwender (oder Systemadministratoren) eine erhöhte Leistungsfähigkeit (z.B. eine erhöhte Anfragereaktionszeit) wünschen. In diesen Situationen können zusätzliche virtuelle Lager hinzugefügt werden, um diese erhöhte Leistungsfähigkeit zu unterstützen. Bei anderen Implementierungen sagt der Ressourcenmanager einen bevorstehenden Bedarf an Ressourcen basierend auf geplanten (aber noch nicht ausgeführten) Anfragen vorher. Wenn die geplanten Anfragen die Leistungsfähigkeit des Systems signifikant verschlechtern werden, kann der Ressourcenmanager vor einem Ausführen dieser Anfragen mehr Ressourcen hinzufügen, um dadurch eine gesamte Systemleistungsfähigkeit beizubehalten. Nachdem diese Anfragen ausgeführt sind, können diese Ressourcen durch den Ressourcenmanager deaktiviert werden.

[0061] Bei einigen Ausführungsformen sagt der Ressourcenmanager eine Zeit vorher, die erforderlich ist, um eine bestimmte Anfrage (oder Gruppe von Anfragen) auszuführen. Basierend auf einer aktuellen Anfrageverarbeitungsleistungsfähigkeit (z.B. Anfrageverarbeitungsverzögerungen, Systemnutzung, etc.) bestimmt der Ressourcenmanager, ob zusätzliche Ressourcen für diese bestimmte Anfrage oder Gruppe von Anfragen nötig sind. Wenn zum Beispiel die aktuelle Anfrageverarbeitungsverzögerung einen Schwellenwert übersteigt, kann der Ressourcenmanager einen oder mehrere neue Ausführungsknoten erzeugen, um zusätzliche Ressourcen zum Verarbeiten der bestimmten Anfrage oder Gruppe von Anfragen zur Verfügung zu stellen. Nachdem eine Verarbeitung der Anfrage oder Gruppe von Anfragen beendet ist, kann der Ressourcenmanager den (die) neuen Ausführungsknoten deaktivieren, wenn er (sie) zum Verarbeiten anderer Anfragen nicht mehr nötig ist (sind).

[0062] Bei einigen Ausführungsformen kann ein bestimmter Anwender bestimmte Leistungsebenen beim Verarbeiten der die Anfragen des Anwenders erfordern. Zum Beispiel kann der Anwender eine Anfragereaktion innerhalb einer bestimmten Zeitperiode, wie beispielsweise 5 Sekunden, erfordern. Bei diesen Ausführungsformen kann der Ressourcenmanager vor einem Ausführen der Anfragen des Anwenders zusätzliche Ressourcen zuteilen, um sicherzustellen, dass die Leistungsebenen des Anwenders erreicht werden.

[0063] Die **Fig. 9A** und **Fig. 9B** stellen ein Ablaufdiagramm einer Ausführungsform eines Verfahrens **900** zum Skalieren von Ressourcen dar. Anfangs überwacht der Ressourcenmanager bei **902** von mehreren Anwendern empfangene Aufforderungen. Zusätzlich überwacht der Ressourcenmanager bei **904**

eine aktuelle Datenzuteilung in mehreren Speichervorrichtungen (z.B. mehreren entfernten Speichervorrichtungen) und überwacht er eine aktuelle Nutzung von mehreren Prozessoren. Das Verfahren **900** fährt bei **906** fort, wenn der Ressourcenmanager basierend auf den Datenverarbeitungsanforderungen und der aktuellen Datenzuteilung in den mehreren Speichervorrichtungen bestimmt, ob zusätzliche Datenspeicherkapazität nötig ist. Der Ressourcenmanager teilt bei **908** zusätzliche Datenspeicherressourcen zu, um die mehreren Anwender zu unterstützen, wenn er bestimmt, dass zusätzliche Datenkapazität nötig ist. Diese zusätzliche Datenkapazität kann für irgendeine Anzahl der mehreren Anwender zugreifbar sein, wie es durch den Ressourcenmanager bestimmt ist. Bei einigen Ausführungsformen wird auf das Hinzufügen von mehr Datenkapazität als „horizontales Skalieren“ Bezug genommen.

[0064] Der Ressourcenmanager bestimmt bei **910** auch, ob einiges der zugeteilten Datenkapazität nicht mehr nötig ist. Wenn der Ressourcenmanager bestimmt, dass einiges der zugeteilten Datenkapazität nicht mehr nötig ist, setzt der Ressourcenmanager bei **912** einiges der Datenkapazität frei. Zum Beispiel wird die freigesetzte Datenkapazität zu einem Pool von verfügbarer Datenkapazität zurückgebracht, die zur Verwendung durch andere Systeme oder Dienste verfügbar wird. Wenn zu einer späteren Zeit zusätzliche Datenkapazität nötig ist, kann der Ressourcenmanager auf Datenressourcen von dem verfügbaren Pool zugreifen.

[0065] Das Verfahren fährt bei **914** fort, wenn der Ressourcenmanager basierend auf den Datenverarbeitungsanforderungen und der aktuellen Nutzung der mehreren Prozessoren bestimmt, ob zusätzliche Verarbeitungsressourcen nötig sind. Der Ressourcenmanager teilt bei **916** zusätzliche Verarbeitungsressourcen zu, um die mehreren Anwender zu unterstützen, wenn er bestimmt, dass zusätzliche Verarbeitungsressourcen nötig sind. Die zusätzlichen Verarbeitungsressourcen können durch irgendeine Anzahl der mehreren Anwender zugreifbar sein, wie es durch den Ressourcenmanager angewiesen ist. Wenn zum Beispiel ein bestimmter Anwender eine große Anzahl von Datenanfragen überreicht hat, kann ein Teilbereich der zusätzlichen Verarbeitungsressourcen diesem Anwender zugeordnet werden, um beim Verarbeiten der Datenanfragen zu helfen. Bei einigen Ausführungsformen wird auf das Hinzufügen von mehr Verarbeitungsressourcen als „vertikales Skalieren“ Bezug genommen.

[0066] Der Ressourcenmanager bestimmt bei **918** auch, ob einige der aktuell zugeteilten Verarbeitungsressourcen nicht mehr nötig sind. Der Ressourcenmanager setzt bei **920** einige der Verarbeitungsressourcen frei, die nicht mehr nötig sind, wenn er bestimmt, dass einige der aktuell zugeteilten Verar-

beitungsressourcen nicht mehr nötig sind. Bei einigen Ausführungsformen werden die freigesetzten Verarbeitungsressourcen zu einem Pool von verfügbaren Verarbeitungsressourcen zurückgebracht, die zur Verwendung durch andere Systeme oder Dienste verfügbar sind. Wenn zu einer späteren Zeit zusätzliche Verarbeitungsressourcen nötig sind, kann der Ressourcenmanager auf Verarbeitungsressourcen von dem verfügbaren Pool zugreifen.

[0067] Wie es hierin beschrieben ist, unterstützt die Verarbeitungsplattform **100** die dynamische Aktivierung und Deaktivierung von verschiedenen Ressourcen, wie beispielsweise eine Datenspeicherkapazität, Verarbeitungsressourcen, Cacheressourcen und ähnliches. Die einzelne Verarbeitungsplattform **100** kann basierend auf den aktuellen Datenspeicher- und Verarbeitungserfordernissen der anhängigen und der erwarteten Datenverarbeitungsaufforderungen auf eine Anforderung hin dynamisch geändert werden. Wenn sich die Datenspeicher- und Verarbeitungserfordernisse ändern, stellt sie Datenverarbeitungsplattform **100** automatisch dahingehend ein, eine einheitliche Ebene einer Datenverarbeitungsleistungsfähigkeit im Wesentlichen beizubehalten.

[0068] Zusätzlich erlaubt die beschriebene Datenverarbeitungsplattform **100** Änderungen an der Datenspeicherkapazität und den Verarbeitungsressourcen unabhängig. Zum Beispiel kann die Datenspeicherkapazität modifiziert werden, ohne irgendwelche Änderungen an den existierenden Verarbeitungsressourcen durchzuführen. Gleichmaßen können die Verarbeitungsressourcen modifiziert werden, ohne irgendwelche Änderungen an der existierenden Datenspeicherkapazität durchzuführen.

[0069] **Fig. 10** ist ein Ablaufdiagramm, das eine Ausführungsform eines weiteren Verfahrens zum Skalieren von Ressourcen darstellt. Anfangs überwacht ein Ressourcenmanager (z.B. der Ressourcenmanager **102**) bei **1002** von mehreren Anwendern empfangene Aufforderungen. Wie es hierin diskutiert ist, können diese Aufforderungen Datensuchanfragen, Datenleseanforderungen, Datenschreibeanforderungen und ähnliches enthalten. Der Ressourcenmanager identifiziert bei **1004** auch eine Anzahl von aktuellen Anwendern. Diese Anzahl ändert sich ständig, wenn neue Anwender auf das System zugreifen und existierende Anwender sich vom System trennen. Zusätzlich identifiziert der Ressourcenmanager bei **1006** Aktivitätsebenen der aktuellen Anwender. Die Aktivitätsebene enthält zum Beispiel die Anzahl von Anfragen, die über eine Zeitperiode überreicht sind, die Anzahl von anhängigen Anfragen und ähnliches. Diese Aktivitätsebene kann für jeden Anwender bestimmt werden oder kann als eine durchschnittliche Aktivität für alle der aktuellen Anwender (oder Gruppen von aktuellen Anwendern) dargestellt werden.

[0070] Basierend auf der Anzahl von aktuellen Anwendern und der Aktivitätsebene der aktuellen Anwender bestimmt der Ressourcenmanager bei **1008** Datenspeicherressourcen und Verarbeitungsressourcen, die nötig sind, um die Aktivitätsebene der aktuellen Anwender zu unterstützen. Da sich die Anzahl von Anwendern ständig ändert und sich die Anwender-Aktivitätsebenen häufig ändern können, bestimmt der Ressourcenmanager kontinuierlich, ob die aktuellen Ressourcen die aktuellen Anwender adäquat unterstützen. Wenn bei **1010** zusätzliche Ressourcen nötig sind, stellt der Ressourcenmanager bei **1012** ein oder mehrere virtuelle Lager bereit, um die aktuellen Anwender zu unterstützen. Gleichmaßen kann, wenn die Anzahl von Anwendern und/oder die Aktivitätsebene niedriger wird, der Ressourcenmanager ein oder mehrere virtuellen Lager deaktivieren, wenn es oder sie nicht mehr nötig ist oder sind, um die aktuellen Anwender zu unterstützen.

[0071] Bei einigen Implementierungen wird dieselbe Datei durch mehrere Ausführungsknoten zur selben Zeit in einem Cache gespeichert. Dieses mehrfache Speichern im Cache von Dateien hilft bei einer Lastausgleichung (z.B. einer Ausgleichung von Datenverarbeitungsaufgaben) über mehrere Ausführungsknoten hinweg. Zusätzlich hilft ein Speichern im Cache von einer Datei in mehreren Ausführungsknoten beim Vermeiden von potentiellen Engpässen, wenn signifikante Mengen von Daten versuchen, durch dieselbe Kommunikationsverbindung zu gelangen. Diese Implementierung unterstützt auch die Parallelverarbeitung derselben Daten durch unterschiedliche Ausführungsknoten.

[0072] Die hierin beschriebenen Systeme und Verfahren ziehen einen Vorteil aus den Vorteilen von sowohl Systemen mit gemeinsam genutzter Platte als auch der Shared-Nothing-Architektur. Die beschriebene Plattform zum Speichern und Wiedergewinnen von Daten ist skalierbar wie die Shared-Nothing-Architektur, wenn einmal Daten lokal in einem Cache gespeichert sind. Sie hat auch alle Vorteile einer Architektur einer gemeinsam genutzten Platte, wo Verarbeitungsknoten hinzugefügt und entfernt werden können, ohne irgendwelche Beschränkungen (z.B. für 0 bis N) und ohne irgendeine explizite Umstrukturierung von Daten zu erfordern.

[0073] Die **Fig. 11** ist ein Blockdiagramm, das eine beispielhafte Computervorrichtung **1100** darstellt. Bei einigen Ausführungsformen wird die Computervorrichtung **1100** verwendet, um eines oder mehreres von den Systemen und Komponenten zu implementieren, die hierin diskutiert sind. Zum Beispiel kann die Computervorrichtung **1100** zulassen, dass ein Anwender oder ein Administrator auf den Ressourcenmanager **102** zugreift. Weiterhin kann die Computervorrichtung **1100** mit irgendetwas von den Systemen und den Komponenten interagieren, die hier-

in beschrieben sind. Demgemäß kann die Computervorrichtung **1100** verwendet werden, um verschiedene Prozeduren und Aufgaben durchzuführen, wie beispielsweise diejenigen, die hierin diskutiert sind. Die Computervorrichtung **1100** kann als Server, Client oder irgendeine andere Computereinheit fungieren. Die Computervorrichtung **1100** kann irgendeine von einer breiten Vielfalt von Computervorrichtungen sein, wie beispielsweise ein Desktopcomputer, ein Notebookcomputer, ein Servercomputer, ein in der Hand gehaltener Computer bzw. Handheld-Computer, ein Tablet und ähnliches.

[0074] Die Computervorrichtung **1100** enthält einen oder mehrere Prozessoren **1102**, eine oder mehrere Speichervorrichtungen **1104**, eine oder mehrere Schnittstellen **1106**, eine oder mehrere Massenspeichervorrichtungen **1108** und eine oder mehrere Eingabe/Ausgabe-(I/O)-Vorrichtungen **1110**, von welchen alle mit einem Bus **1112** gekoppelt sind. Der Prozessor (die Prozessoren) **1102** enthält (enthalten) einen oder mehrere Prozessoren oder Steuerungen, die Anweisungen ausführen, die in der Speichervorrichtung (den Speichervorrichtungen) **1104** und/oder der Massenspeichervorrichtung (den Massenspeichervorrichtungen) **1108** gespeichert sind. Der Prozessor (die Prozessoren) **1102** kann (können) auch verschiedene Typen von computerlesbaren Medien, wie beispielsweise einen Cachespeicher, enthalten.

[0075] Die Speichervorrichtung(en) **1104** enthält (enthalten) verschiedene computerlesbare Medien, wie beispielsweise flüchtige Speicher (z.B. einen Direktzugriffsspeicher (RAM)) und/oder einen nichtflüchtigen Speicher (z.B. einen Nurlesespeicher (ROM)). Die Speichervorrichtung(en) **1104** kann (können) auch einen wiederbeschreibbaren ROM, wie beispielsweise einen Flash-Speicher, enthalten.

[0076] Die Massenspeichervorrichtung(en) **1108** enthält (enthalten) verschiedene computerlesbare Medien, wie beispielsweise Magnetbänder, Magnetplatten, optische Platten, einen Festkörperspeicher (z.B. einen Flash-Speicher), und so weiter. Verschiedene Laufwerke können auch in der (den) Massenspeichervorrichtung(en) **1108** enthalten sein, um ein Lesen von und/oder ein Schreiben zu den verschiedenen computerlesbaren Medien zu ermöglichen. Die Massenspeichervorrichtung(en) **1108** enthält (enthalten) entfernbare Medien und/oder nicht entfernbare Medien.

[0077] Die I/O-Vorrichtung(en) **1110** enthält (enthalten) verschiedene Vorrichtungen, die zulassen, dass Daten und/oder andere Information zu der Computervorrichtung **1100** eingegeben oder aus dieser ausgelesen werden. Eine beispielhafte I/O-Vorrichtung (beispielhafte I/O-Vorrichtungen) **1110** enthält (enthalten) Cursor-Steuervorrichtungen, Tastaturen,

Keypads bzw. kleine Tastaturen bzw. Folientastaturen, Mikrofone, Monitore oder andere Anzeigevorrichtungen, Lautsprecher, Drucker, Netzwerkschnittstellenkarten, Modems, eine Optik, CCDs oder andere Bilderfassungsvorrichtungen und ähnliches.

[0078] Die Schnittstelle(n) **1106** enthält (enthalten) verschiedene Schnittstellen, die zulassen, dass die Computervorrichtung **1100** mit anderen Systemen, Vorrichtungen oder Computerumgebungen interagiert. Eine beispielhafte Schnittstelle (beispielhafte Schnittstellen) **1106** enthält (enthalten) irgendeine Anzahl von unterschiedlichen Netzwerkschnittstellen, wie beispielsweise Schnittstellen zu lokalen Netzen (LAN), Weitverkehrsnetzen (WAN), drahtlosen Netzen und dem Internet.

[0079] Der Bus **1112** lässt zu, dass der Prozessor (die Prozessoren) **1102**, die Speichervorrichtung (en) **1104**, die Schnittstelle(n) **1106**, die Massenspeichervorrichtung(en) **1108** und die I/O-Vorrichtung (en) **1110** miteinander kommunizieren, sowie anderen Vorrichtungen oder Komponenten, die mit dem Bus **1112** gekoppelt sind. Der Bus **1112** stellt eine oder mehrere von einigen Typen von Busstrukturen dar, wie beispielsweise einen Systembus, einen PCI-Bus, einen IEEE-1394-Bus, einen USB-Bus und so weiter.

[0080] Zu Darstellungszwecken sind Programme und andere ausführbare Programmkomponenten hierin als diskrete Blöcke gezeigt, obwohl es verstanden wird, dass solche Programme und Komponenten zu unterschiedlichen Zeiten in unterschiedlichen Speicherkomponenten der Computervorrichtung **1100** sitzen können und durch den Prozessor (die Prozessoren) **1102** ausgeführt werden. Alternativ können die Systeme und Prozeduren, die hierin beschrieben sind, in Hardware oder einer Kombination aus Hardware, Software und/oder Firmware implementiert sein. Beispielsweise kann oder können eine oder mehrere anwendungsspezifische integrierte Schaltkreise (ASICs) programmiert sein, um eines oder mehreres der hierin beschriebenen Systeme und Prozeduren auszuführen.

[0081] Obwohl die vorliegende Offenbarung in Bezug auf bestimmte bevorzugte Ausführungsformen beschrieben ist, werden andere Ausführungsformen für Fachleute auf dem Gebiet offensichtlich werden, denen der Vorteil dieser Offenbarung gegeben ist, einschließlich Ausführungsformen, die nicht alle der hierin dargelegten Vorteile und Merkmale zur Verfügung stellen, welche auch innerhalb des Schutzbereichs dieser Offenbarung sind. Es ist zu verstehen, dass andere Ausführungsformen verwendet werden können, ohne vom Schutzbereich der vorliegenden Offenbarung abzuweichen.

[0082] Es sind beispielhafte Systeme und Verfahren zum Bereitstellen von Ressourcen beschrieben worden. Bei einer Implementierung greift eine Ausführungsplattform auf mehrere entfernte Speichervorrichtungen zu. Die Ausführungsplattform enthält mehrere virtuelle universelle Datenbanken bzw. Lager, von welchen jede bzw. jedes einen Cache enthält, um aus den entfernten Speichervorrichtung wiedergewonnene bzw. abgerufene Daten zu speichern, und einen Prozessor, der unabhängig von den entfernten Speichervorrichtungen ist. Ein Ressourcenmanager ist mit der Ausführungsplattform gekoppelt und überwacht empfangene Datenverarbeitungsaufforderungen und eine Ressourcennutzung. Der Ressourcenmanager bestimmt basierend auf den Datenverarbeitungsaufforderungen und der Ressourcennutzung auch, ob zusätzliche virtuelle Lager nötig sind. Wenn zusätzliche virtuelle Lager nötig sind, stellt der Ressourcenmanager ein neues virtuelles Lager bereit.

ZITATE ENTHALTEN IN DER BESCHREIBUNG

Diese Liste der vom Anmelder aufgeführten Dokumente wurde automatisiert erzeugt und ist ausschließlich zur besseren Information des Lesers aufgenommen. Die Liste ist nicht Bestandteil der deutschen Patent- bzw. Gebrauchsmusteranmeldung. Das DPMA übernimmt keinerlei Haftung für etwaige Fehler oder Auslassungen.

Zitierte Patentliteratur

- US 61/941986 [0001]

Schutzansprüche**1. Vorrichtung, umfassend:**

eine Ausführungsplattform, die konfiguriert ist, um auf eine Vielzahl von entfernten Speichervorrichtungen zuzugreifen, wobei die Ausführungsplattform eine Vielzahl von virtuellen universellen Datenbanken bzw. Lagern enthält, wobei jedes virtuelle Lager einen Cache enthält, der konfiguriert ist, um aus den entfernten Speichervorrichtungen wiedergewonnene bzw. abgerufene Daten zu speichern, und einen Prozessor, der unabhängig von den entfernten Speichervorrichtungen ist; und

einen Ressourcenmanager, der mit der Ausführungsplattform gekoppelt ist und konfiguriert ist, um:

- empfangene Datenverarbeitungsaufforderungen zu überwachen;
- eine Ressourcennutzung in der Vielzahl von virtuellen Lagern zu überwachen;
- zu bestimmen, ob zusätzliche virtuelle Lager nötig sind, basierend auf den Datenverarbeitungsaufforderungen und der Ressourcennutzung in der Vielzahl von Lagern; und
- in Reaktion auf ein Bestimmen, dass ein zusätzliches virtuelles Lager nötig ist, ein neues virtuelles Lager bereitzustellen.

2. Vorrichtung nach Anspruch 1, wobei der Ressourcenmanager weiterhin konfiguriert ist, um Anfragereaktionsraten zu überwachen.

3. Vorrichtung nach Anspruch 1, wobei der Ressourcenmanager weiterhin konfiguriert ist, um eine aktuelle Ressourcennutzung zu überwachen, die mit der Vielzahl von virtuellen Lagern assoziiert ist, und wobei ein Bestimmen, ob zusätzliche virtuelle Lager nötig sind, weiterhin auf der aktuellen Ressourcennutzung basiert.

4. Vorrichtung nach Anspruch 1, wobei der Ressourcenmanager weiterhin konfiguriert ist, um wenigstens eines von einer Anzahl von Anfragen, die gleichzeitig laufen bzw. in Betrieb sind, einem Prozentsatz einer maximalen Last und ob irgendeine Anfrageverarbeitung verzögert ist zu überwachen.

5. Vorrichtung nach Anspruch 1, wobei der Ressourcenmanager weiterhin konfiguriert ist, um das neue virtuelle Lager in Reaktion auf ein Bestimmen, dass das neue virtuelle Lager nicht mehr nötig ist, zu deaktivieren.

6. Vorrichtung nach Anspruch 1, wobei der Ressourcenmanager weiterhin konfiguriert ist, um:

- eine Nachricht über ein bevorstehendes Datenverarbeitungsprojekt zu empfangen; und
- in Reaktion auf ein Empfangen der Nachricht über das bevorstehende Datenverarbeitungsprojekt ein neues virtuelles Lager zu erzeugen, das mit dem

bevorstehenden Datenverarbeitungsprojekt assoziiert ist.

7. Vorrichtung nach Anspruch 6, wobei der Ressourcenmanager weiterhin konfiguriert ist, um das neue virtuelle Lager auf ein Beenden des Datenverarbeitungsprojekt hin zu deaktivieren.

8. Vorrichtung nach Anspruch 6, wobei das bevorstehende Datenverarbeitungsprojekt eine vorbestimmte Zeitdauer hat und wobei das neue virtuelle Lager nach der vorbestimmten Zeitdauer deaktiviert wird.

9. Vorrichtung nach Anspruch 1, wobei der Ressourcenmanager weiterhin konfiguriert ist, um:

- eine Aufforderung zu empfangen, um eine Vielzahl von Datenverarbeitungsaktivitäten durchzuführen; und
- in Reaktion auf ein Empfangen der Aufforderung, die Vielzahl von Datenverarbeitungsaktivitäten durchzuführen, automatisch ein neues virtuelles Lager zu erzeugen, das mit der empfangenen Aufforderung assoziiert ist.

10. Computerprogramm, das Anweisung umfasst, die dann, wenn sie durch einen oder mehrere Prozessoren ausgeführt werden, veranlassen, das der eine oder die mehreren Prozessoren Operationen durchführt oder durchführen, die folgendes umfassen:

- Überwachen empfangener Datenverarbeitungsaufforderungen, wobei die Datenverarbeitungsaufforderungen mit Daten assoziiert sind, die in einer Vielzahl von entfernten Speichervorrichtungen gespeichert sind;
- Überwachen einer Ressourcennutzung in einer Vielzahl von virtuellen universellen Datenbanken bzw. Lagern, wobei jedes virtuelle Lager einen Cache enthält, der konfiguriert ist, um aus den entfernten Speichervorrichtungen wiedergewonnene bzw. abgerufene Daten zu speichern, und einen Prozessor, der unabhängig von den entfernten Speichervorrichtungen ist;
- Bestimmen, ob zusätzliche Lager nötig sind, basierend auf den Datenverarbeitungsaufforderungen und der Ressourcennutzung in der Vielzahl von virtuellen Lagern; und
- Erzeugen eines neuen virtuellen Lagers in Reaktion auf ein Bestimmen, dass ein zusätzliches virtuelles Lager nötig ist.

11. Computerprogramm nach Anspruch 10, wobei die Operationen weiterhin ein Überwachen von Anfragereaktionsraten umfassen, wobei ein Bestimmen, ob zusätzliche virtuelle Lager nötig sind, weiterhin auf den Anfragereaktionsraten basiert.

12. Computerprogramm nach Anspruch 10, wobei die Operationen weiterhin ein Überwachen einer Anzahl von Anwendern umfassen, die mit jedem der

Vielzahl von virtuellen Lagern assoziiert sind, wobei ein Bestimmen, ob zusätzliche virtuelle Lager nötig sind, weiterhin auf der Anzahl von Anwendern basiert, die mit jedem der Vielzahl von virtuellen Lagern assoziiert sind.

13. Computerprogramm nach Anspruch 10, wobei die Operationen weiterhin folgendes umfassen:
Bestimmen, ob das neue virtuelle Lager nicht mehr nötig ist; und
Deaktivieren des neuen virtuellen Lagers in Reaktion auf ein Bestimmen, dass das neue virtuelle Lager nicht mehr nötig ist.

14. Computerprogramm nach Anspruch 10, wobei die Operationen weiterhin ein Überwachen einer Anzahl von Anfragen umfassen, die gleichzeitig laufen bzw. in Betrieb sind, wobei ein Bestimmen, ob zusätzliche virtuelle Lager nötig sind, weiterhin auf der Anzahl von Anfragen basiert, die gleichzeitig laufen bzw. in Betrieb sind.

15. Computerprogramm nach Anspruch 10, wobei die Operationen weiterhin ein Bestimmen umfassen, ob eine Ausführung von irgendwelchen Anfragen aktuell verzögert wird, wobei ein Bestimmen, ob zusätzliche virtuelle Lager nötig sind, weiterhin darauf basiert, ob irgendwelche Anfragen aktuell verzögert werden.

16. Computerprogramm nach Anspruch 10, wobei die Operationen weiterhin folgendes umfassen:
Empfangen einer Nachricht über ein bevorstehendes Datenverarbeitungsprojekt;
Erzeugen eines neuen virtuellen Lagers, das mit dem bevorstehenden Datenverarbeitungsprojekt assoziiert ist; und
Deaktivieren des neuen virtuellen Lagers auf ein Beenden des Datenverarbeitungsprojekts hin.

17. Computerprogramm nach Anspruch 10, wobei die Operationen weiterhin folgendes umfassen:
Empfangen einer Nachricht über ein bevorstehendes Datenverarbeitungsprojekt, wobei das bevorstehende Datenverarbeitungsprojekt eine vorbestimmte Zeitperiode hat;
Erzeugen eines neuen virtuellen Lagers, das mit dem bevorstehenden Datenverarbeitungsprojekt assoziiert ist; und
Deaktivieren des neuen virtuellen Lagers nach der vorbestimmten Zeitperiode.

18. Vorrichtung, umfassend:
Mittel zum Zugreifen auf eine Vielzahl von entfernten Speichervorrichtungen, wobei die Mittel zum Zugreifen eine Vielzahl von virtuellen universellen Datenbanken bzw. Lagern enthalten, wobei jedes virtuelle Lager eine Cache enthält, der konfiguriert ist, um von den entfernten Speichervorrichtungen wiedergewonnene bzw. abgerufene Daten zu speichern, und einen

Prozessor, der unabhängig von den entfernten Speichervorrichtungen ist; und
Mittel zum Managen von Ressourcen, welche Mittel mit den Mitteln zum Zugreifen gekoppelt sind und konfiguriert sind, um:
empfangene Datenverarbeitungsaufforderungen zu überwachen;
eine Ressourcennutzung in der Vielzahl von virtuellen Lagern zu überwachen;
zu bestimmen, ob zusätzliche virtuelle Lager nötig sind, basierend auf den Datenverarbeitungsaufforderungen und der Ressourcennutzung in der Vielzahl von Lagern; und
in Reaktion auf ein Bestimmen, dass ein zusätzliches virtuelles Lager nötig ist, ein neues virtuelles Lager bereitzustellen.

19. Vorrichtung nach Anspruch 18, wobei die Mittel zum Managen von Ressourcen weiterhin konfiguriert sind, um das neue virtuelle Lager auf eine Bestimmung hin zu deaktivieren, dass das neue virtuelle Lager nicht mehr nötig ist.

20. Vorrichtung nach Anspruch 18, wobei die Mittel zum Managen von Ressourcen weiterhin konfiguriert sind, um:
eine Zeit vorherzusagen, die erforderlich ist, um eine Anfrage auszuführen; und
ein neues virtuelles Lager bereitzustellen, wenn die Zeit, die erforderlich ist, um die Anfrage auszuführen, einen Schwellenwert übersteigt.

Es folgen 12 Seiten Zeichnungen

Anhängende Zeichnungen

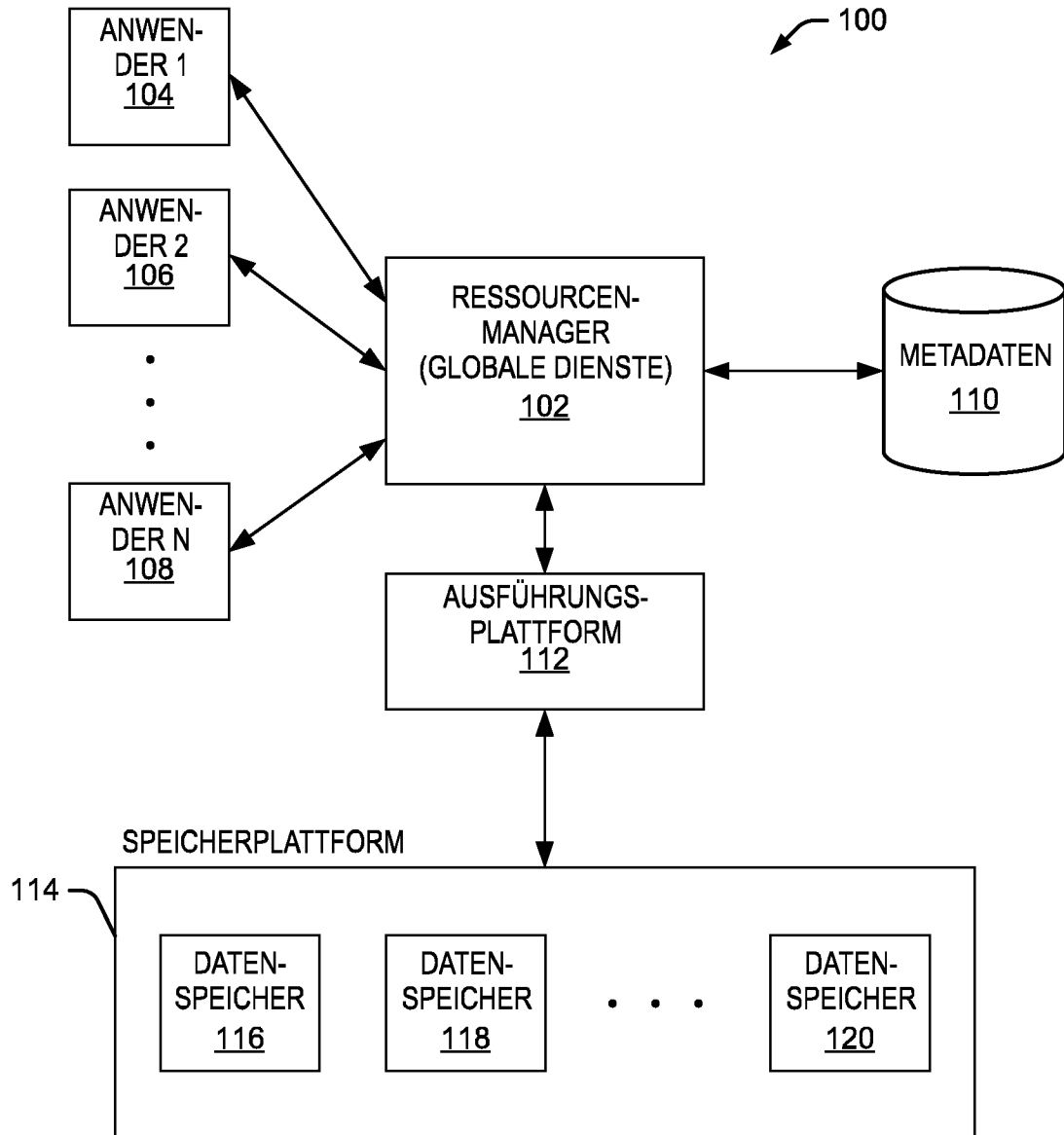


FIG. 1

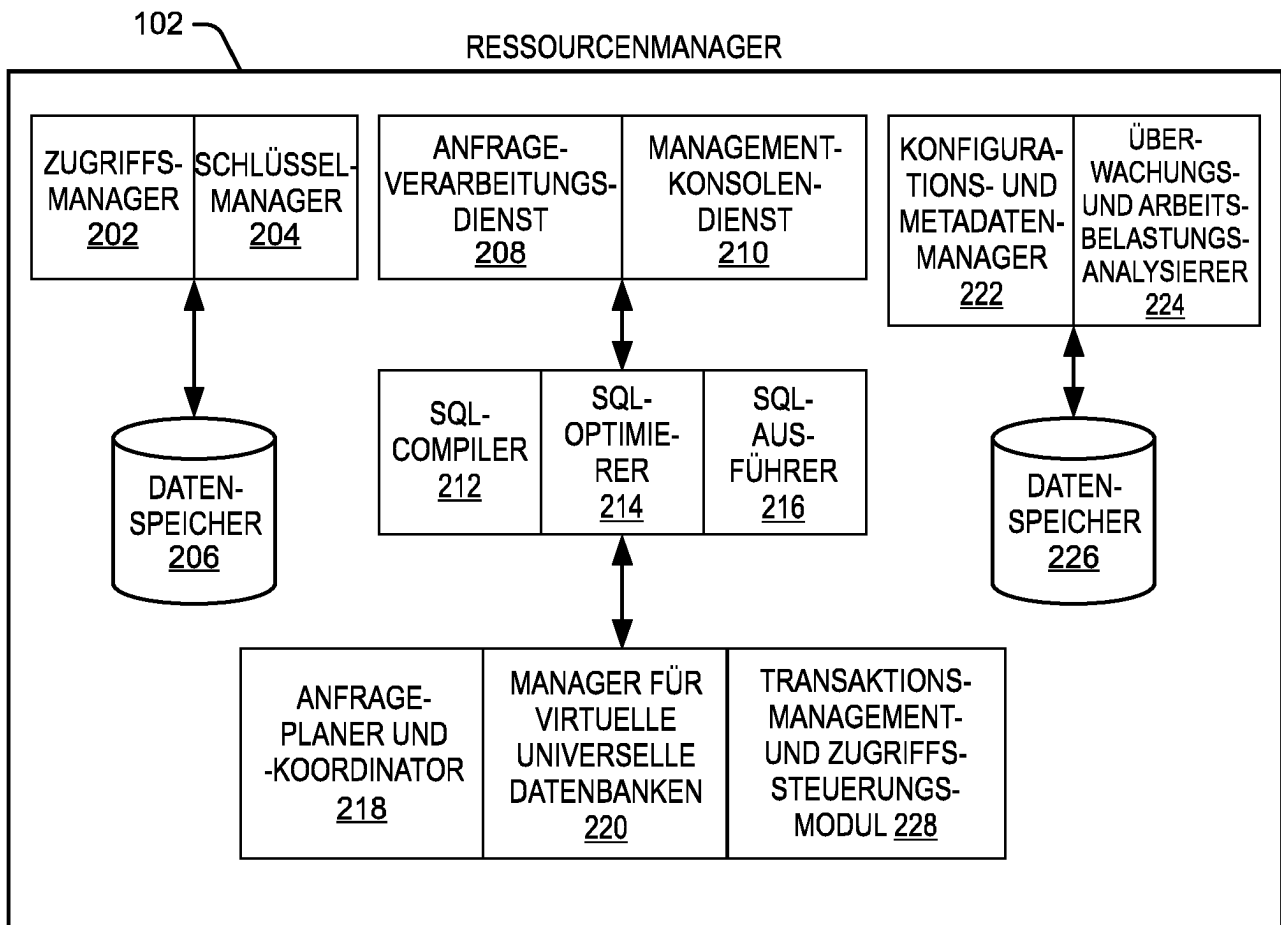


FIG. 2

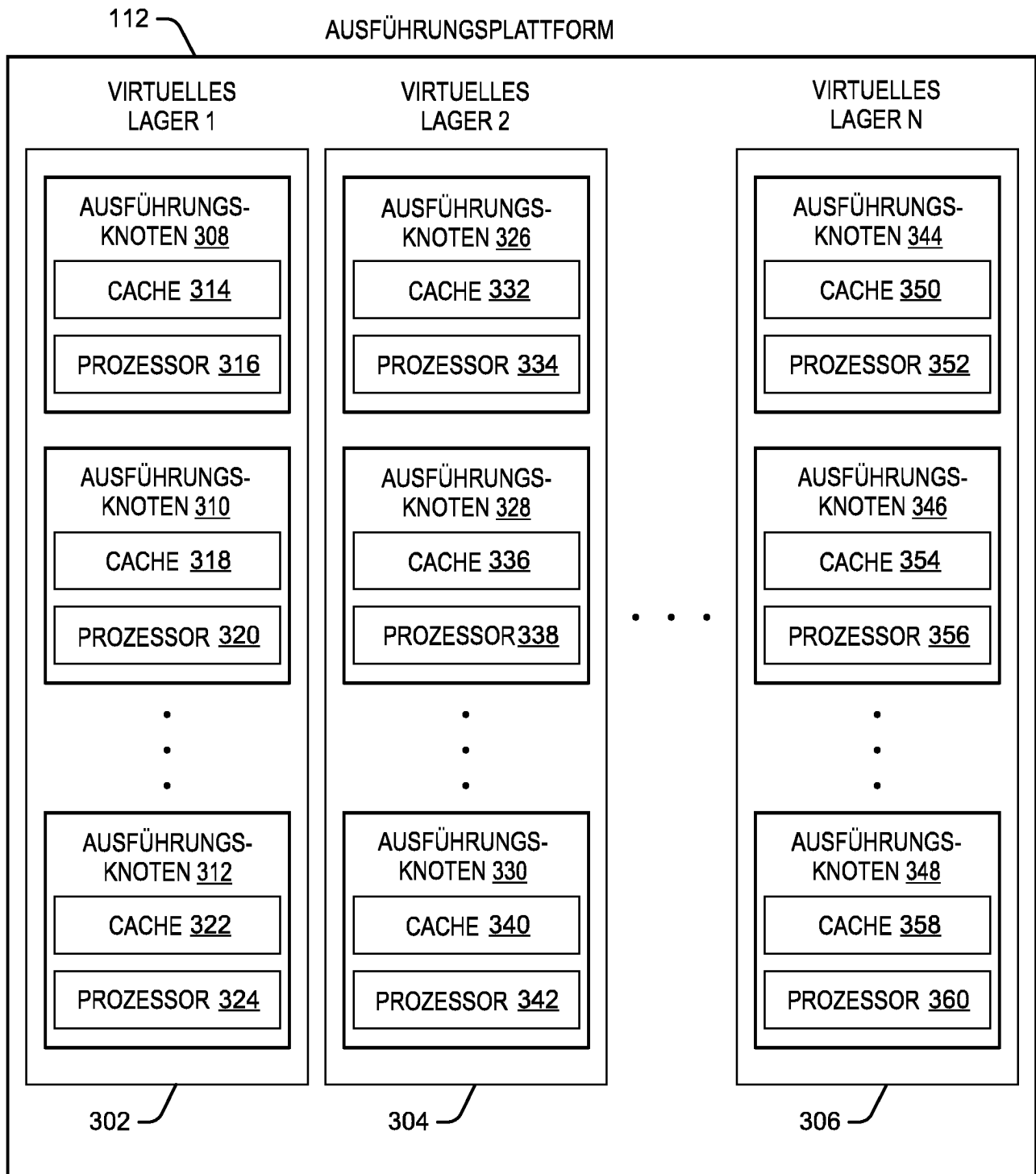


FIG. 3

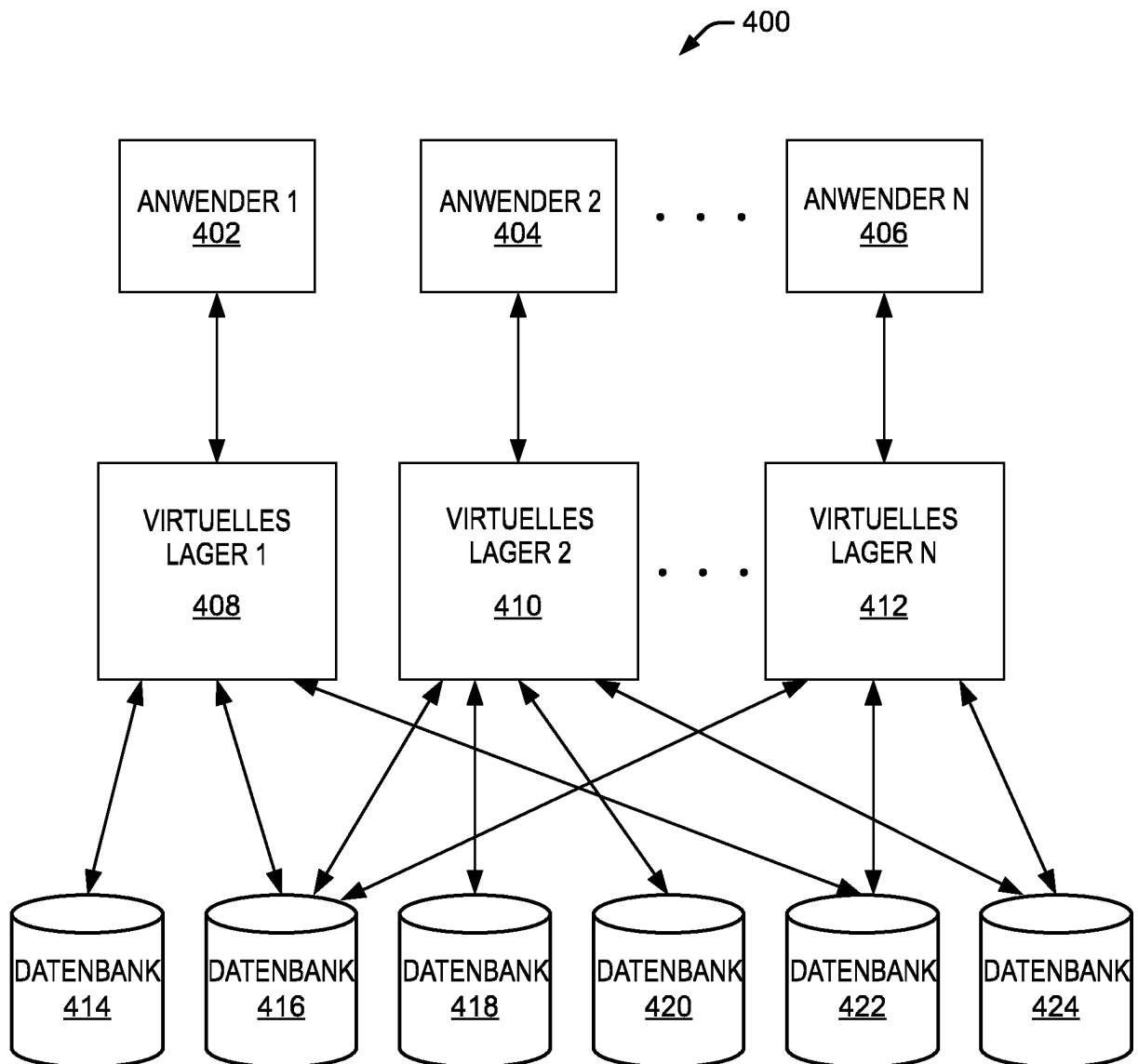


FIG. 4

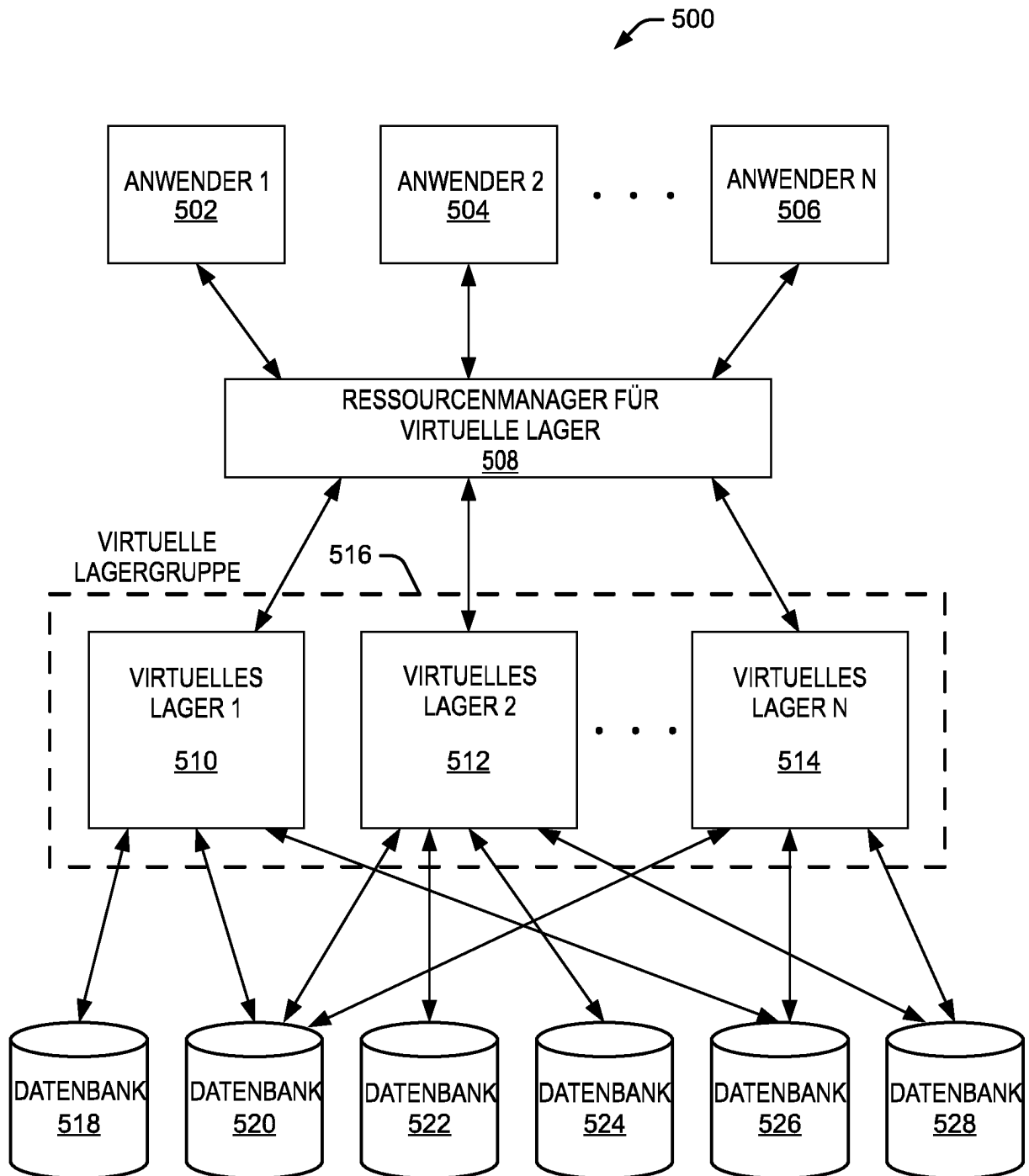


FIG. 5

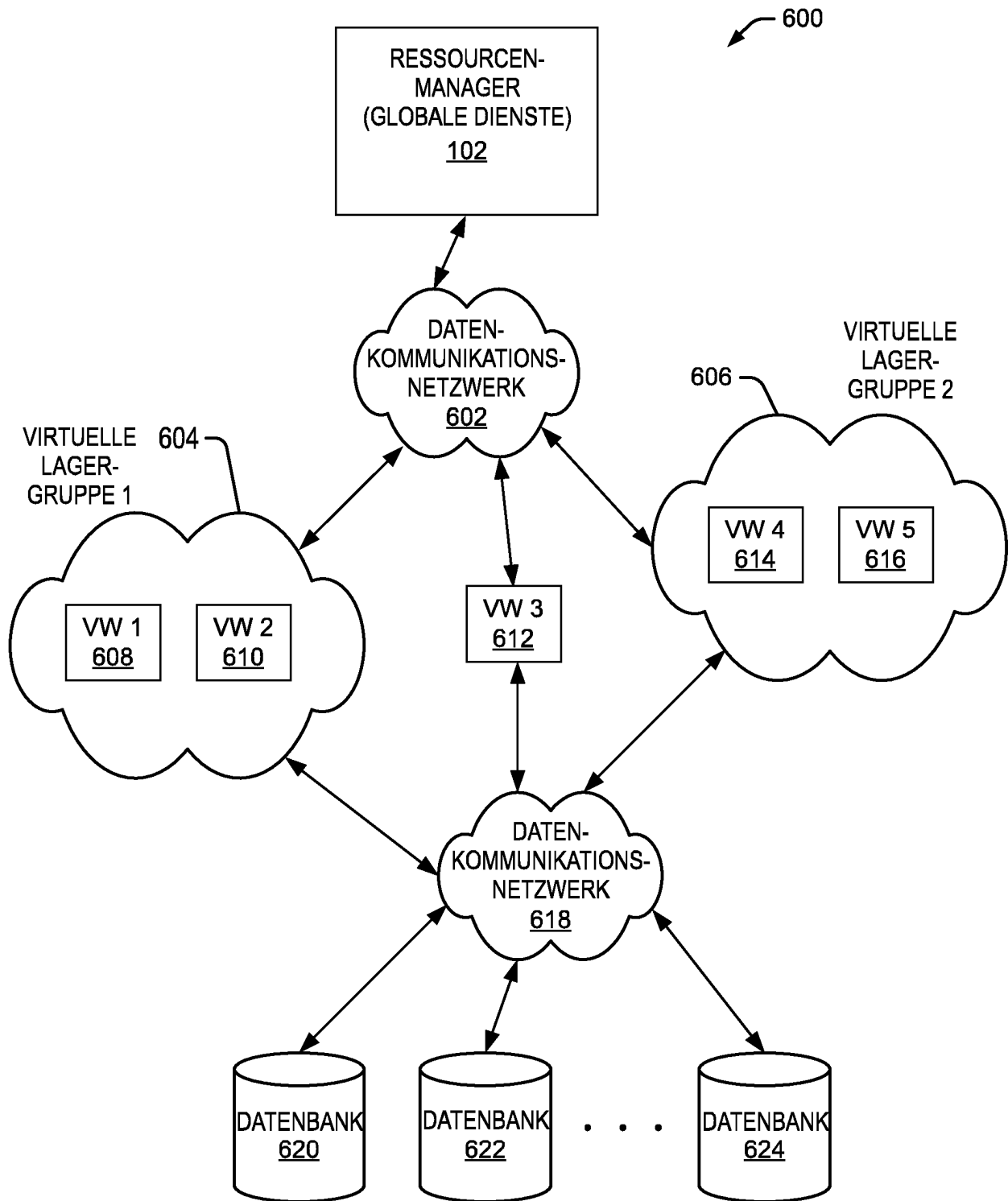


FIG. 6

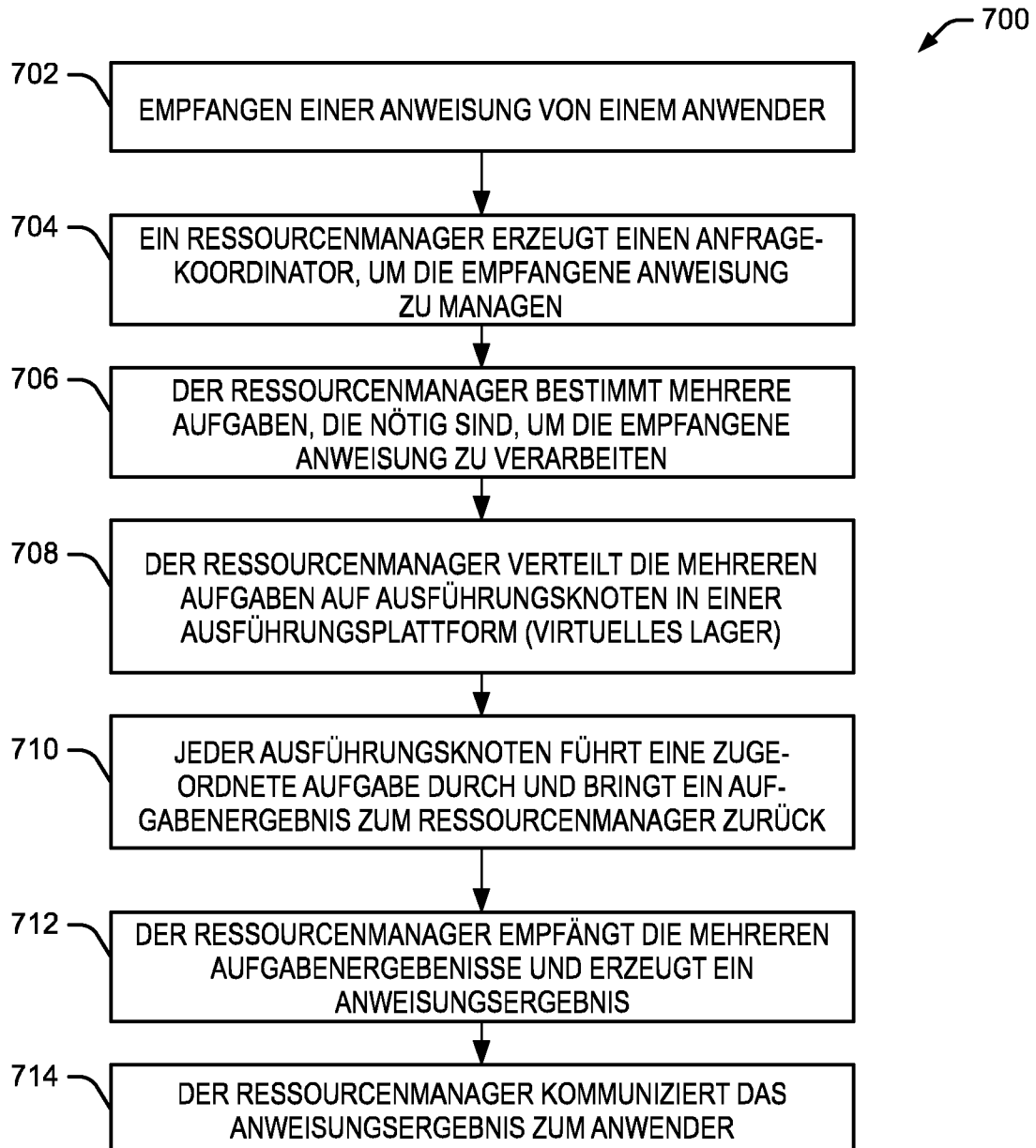


FIG. 7

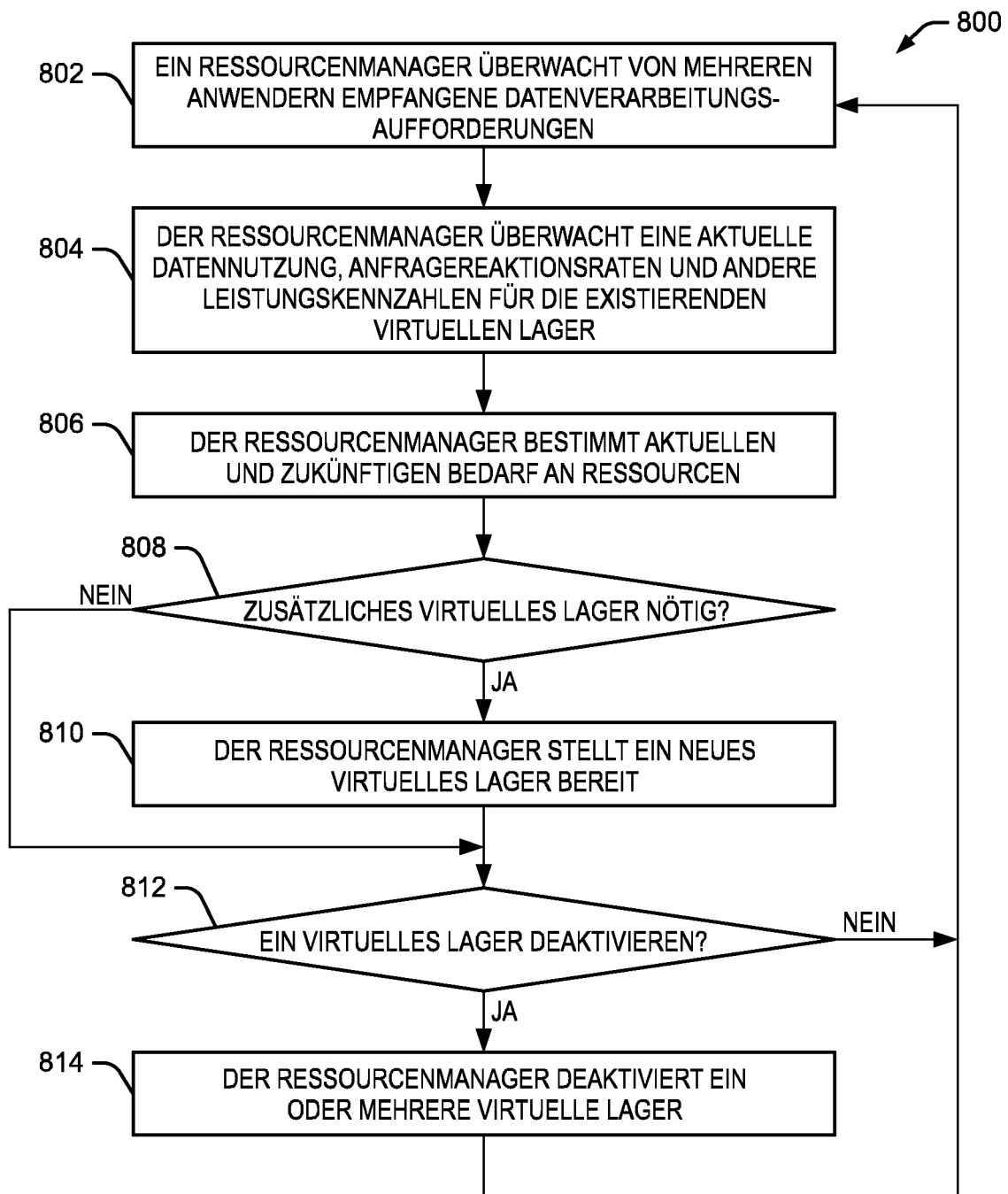


FIG. 8

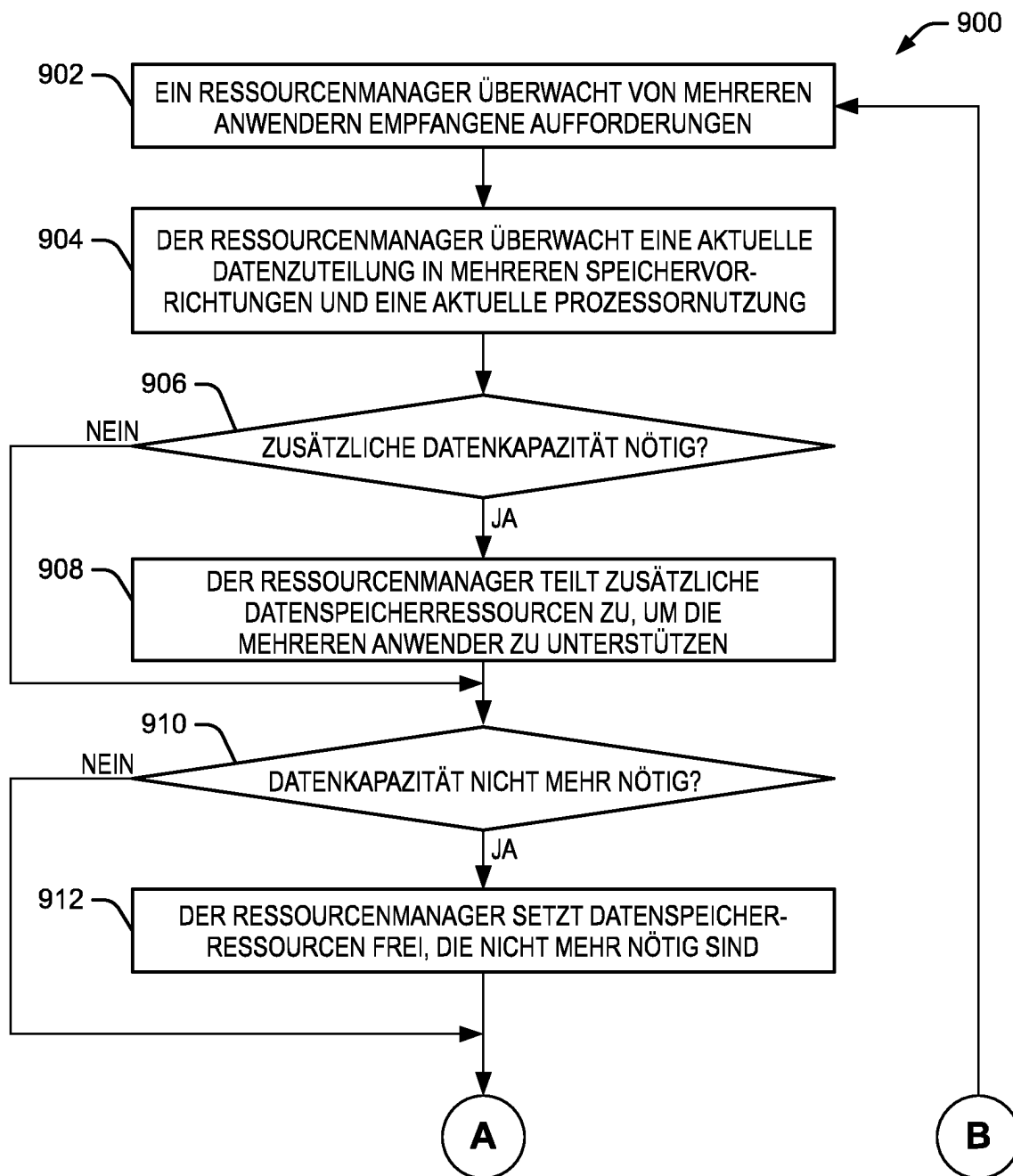


FIG. 9A

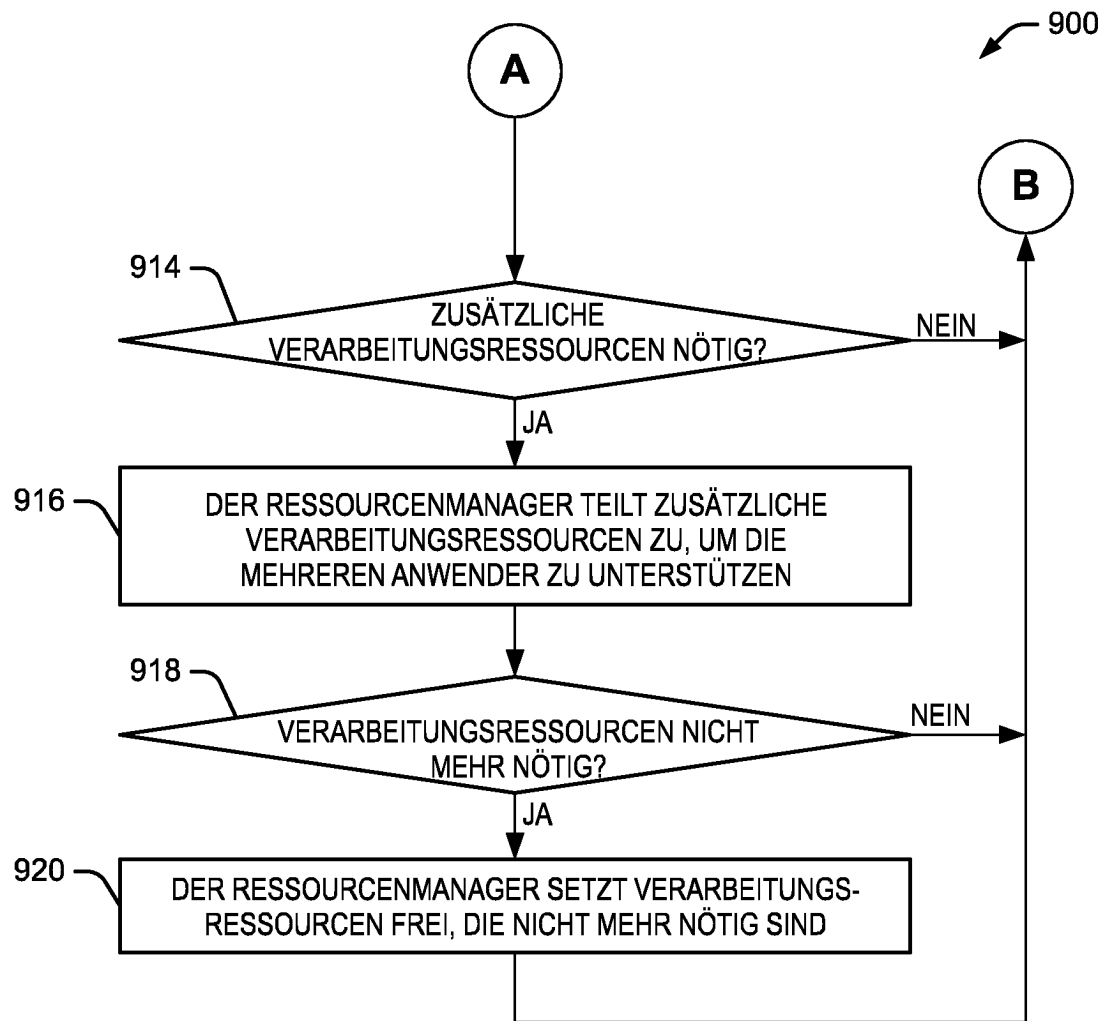


FIG. 9B

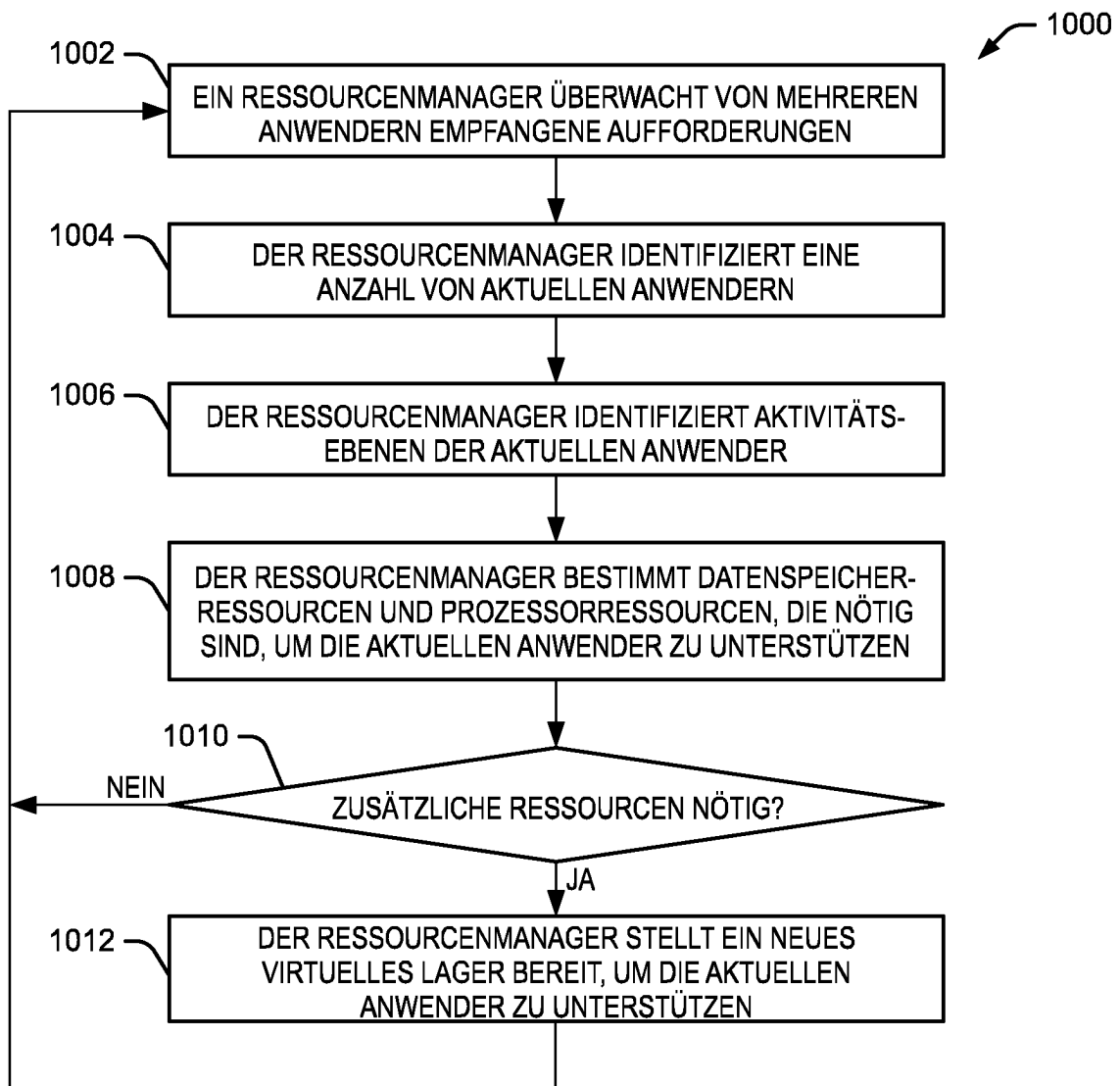


FIG. 10

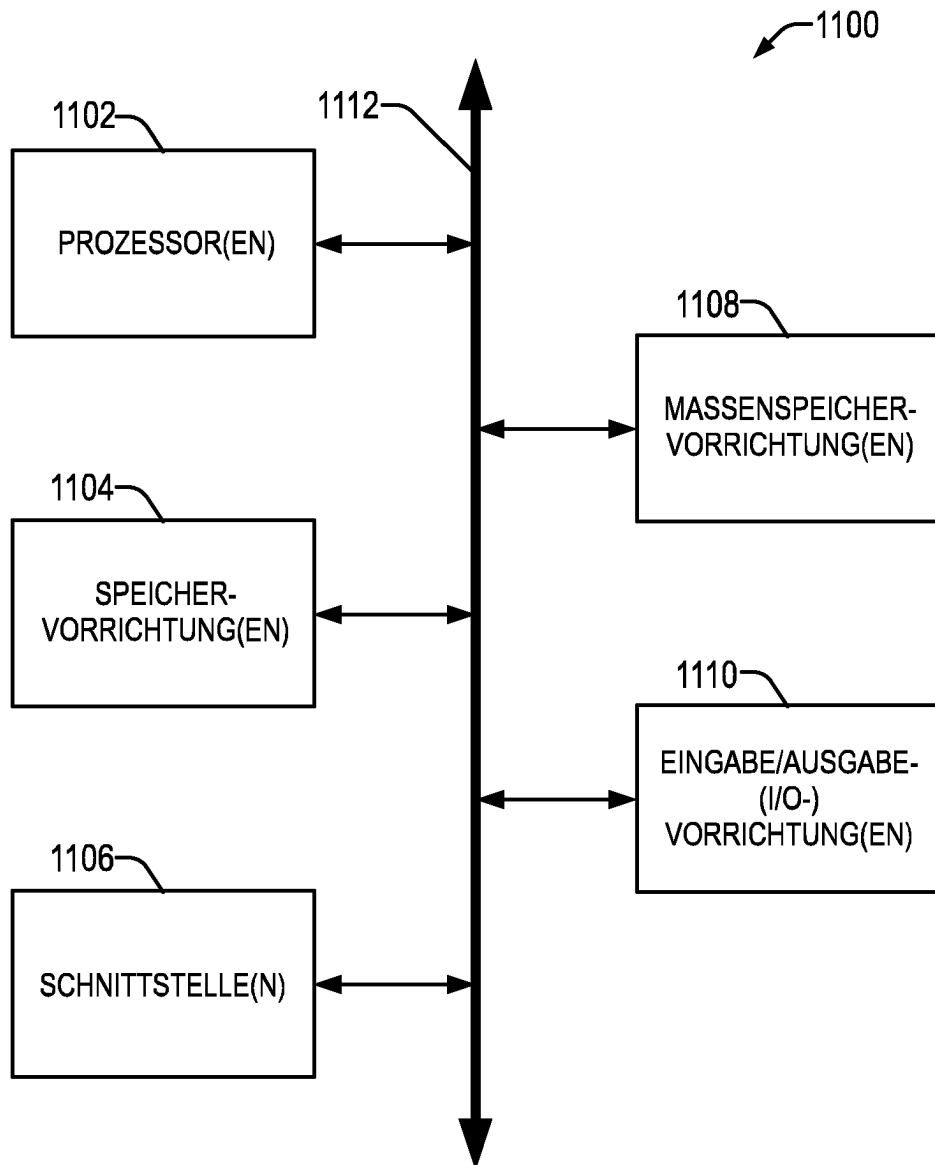


FIG. 11