



(10) **DE 10 2014 114 108 B4** 2024.12.24

(12) **Patentschrift**

(21) Aktenzeichen: **10 2014 114 108.3**
(22) Anmeldetag: **29.09.2014**
(43) Offenlegungstag: **02.04.2015**
(45) Veröffentlichungstag
der Patenterteilung: **24.12.2024**

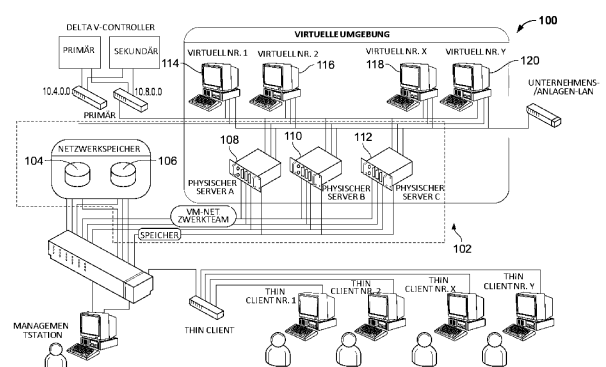
(51) Int Cl.: **G06F 9/44 (2018.01)**
G06F 16/27 (2019.01)
G06F 11/14 (2006.01)
G06F 9/455 (2006.01)
G05B 19/04 (2006.01)

Innerhalb von neun Monaten nach Veröffentlichung der Patenterteilung kann nach § 59 Patentgesetz gegen das Patent Einspruch erhoben werden. Der Einspruch ist schriftlich zu erklären und zu begründen. Innerhalb der Einspruchsfrist ist eine Einspruchsgebühr in Höhe von 200 Euro zu entrichten (§ 6 Patentkostengesetz in Verbindung mit der Anlage zu § 2 Abs. 1 Patentkostengesetz).

<p>(30) Unionspriorität: 61/883,748 27.09.2013 US</p> <p>(73) Patentinhaber: Fisher-Rosemount Systems, Inc., Round Rock, Tex., US</p> <p>(74) Vertreter: Meissner Bolte Patentanwälte Rechtsanwälte Partnerschaft mbB, 80538 München, DE</p>	<p>(72) Erfinder: Thiele, Dirk, Austin, Tex., US; Qui, Shaobo, Round Rock, Tex., US; Nixon, Mark J., Round Rock, Tex., US</p> <p>(56) Ermittelter Stand der Technik: US 2012 / 0 124 012 A1</p>
---	---

(54) Bezeichnung: **Verfahren und computerlesbares Speichermedium für Prozessleitsysteme**

(57) Zusammenfassung: Offenbart werden ein Prozessleitsystem und -verfahren. Ein beispielhaftes Verfahren schließt das Betreiben eines ersten Clusters mit ersten virtuellen Maschinen und ersten Servern sowie das Betreiben eines zweiten Clusters mit zweiten virtuellen Maschinen und zweiten Servern ein. Das beispielhafte Verfahren umfasst ferner das Speichern erster Daten der ersten virtuellen Maschine in einem ersten Datenspeicher des ersten Clusters und das Speichern einer Replikation der ersten Daten in einem zweiten Datenspeicher des zweiten Clusters. Das beispielhafte Verfahren umfasst ferner das Speichern von zweiten Daten der zweiten virtuellen Maschinen im zweiten Datenspeicher und das Speichern einer Replikation der zweiten Daten im ersten Datenspeicher sowie das Identifizieren eines Ausfalls des ersten Clusters. Das Verfahren umfasst auch, als Reaktion auf den Ausfall, ein Neustarten der ersten virtuellen Maschinen unter Verwendung der zweiten Server und der Replikation der ersten Daten im zweiten Datenspeicher.



Beschreibung**VERWANDTE ANMELDUNG**

[0001] Die vorliegende Erfindung betrifft ein Verfahren und ein computerlesbares Speichermedium für Prozessleitsysteme.

HINTERGRUND

[0002] Ein typisches Leitsystem umfasst ein Netzwerk von vielen Arbeitsstationen, Servern, Steuerungen und PO-Untersystemen. Die Einrichtung und Wartung mehrerer Systeme, die für Entwicklung, Tests, Schulungen und Online-Produktion benötigt werden, kann teuer und zeitaufwändig sein.

[0003] Relevanten Stand der Technik stellt die US 2012/0124012 A1 dar. Diese offenbart im Kontext eines Prozessleitsystems ein Verfahren mit folgenden Schritten:

Betreiben eines ersten Clusters, der erste virtuelle Maschinen und erste Server umfasst;

Betreiben eines zweiten Clusters, der zweite virtuelle Maschinen und zweite Server umfasst;

Speichern erster Daten der ersten virtuellen Maschinen in einem ersten Datenspeicher des ersten Clusters und Speichern einer Replikation der ersten Daten;

Speichern zweiter Daten der zweiten virtuellen Maschinen im zweiten Datenspeicher und Speichern einer Replikation der zweiten Daten im ersten Datenspeicher;

Identifizieren eines Ausfalls des ersten Clusters; und als Reaktion auf den Ausfall, Neustarten der ersten virtuellen Maschinen unter Verwendung der zweiten Server und der Replikation der ersten Daten.

[0004] Die genannte Druckschrift offenbart des Weiteren ein Verfahren mit folgenden Schritten:

Betreiben eines ersten Modus zum Betreiben eines ersten Clusters, der erste virtuelle Maschinen und erste Server aufweist;

Betreiben des ersten Clusters im ersten Modus, umfassend Schreiben in einen Cache eines ersten Datenspeichers, um die ersten Daten der ersten virtuellen Maschinen zu speichern;

alternatives Betreiben eines zweiten Modus zum Betreiben des ersten Clusters; und Betreiben des ersten Clusters im zweiten Modus, umfassend Umgehen des Caches und direktes Schreiben auf eine Platte des ersten Datenspeichers, um die ersten Daten der ersten virtuellen Maschinen zu speichern.

[0005] Der Erfindung liegt die Aufgabe zugrunde, verbesserte Verfahren dieser Art anzugeben, die insbesondere eine Wiederherstellung nicht-redundanter Komponenten eines Prozessleitsystems nach Störungen in kürzerer Zeit ermöglichen und somit die schnellere Verfügbarkeit von Anlagendaten ermöglichen und den Verlust solcher Daten besser vermeiden können.

KURZDARSTELLUNG

[0006] Die Aufgabe wird mit den Gegenständen der unabhängigen Patentansprüche gelöst.

[0007] Ein beispielhaftes Verfahren umfasst das Betreiben eines ersten Clusters mit ersten virtuellen Maschinen und ersten Servern und im ersten und zweiten Datenspeicher eines zweiten Clusters mit zweiten virtuellen Maschinen und zweiten Servern. Das Verfahren schließt auch das Speichern erster Daten von den ersten virtuellen Maschinen in einem ersten Datenspeicher des ersten Clusters und das Speichern einer Replikation der ersten Daten in einem zweiten Datenspeicher des zweiten Clusters ein. Das Verfahren schließt auch das Speichern zweiter Daten aus den zweiten virtuellen Maschinen in dem zweiten Datenspeicher und das Speichern einer Replikation der zweiten Daten im ersten Datenspeicher ein. Das Verfahren umfasst das Identifizieren eines Ausfalls des ersten Clusters und als Reaktion auf den Ausfall, das Neustarten der ersten virtuellen Maschinen mithilfe der zweiten Server und der Replikation der ersten Daten in dem zweiten Datenspeicher.

[0008] Ein weiteres Beispielverfahren schließt das Auswählen eines ersten Modus ein, um einen ersten Cluster mit ersten virtuellen Maschinen und ersten Servern zu betreiben, und das Betreiben des ersten Clusters im ersten Modus, einschließlich Schreiben in einen Cache eines ersten Datenspeichers, um die ersten Daten von den ersten virtuellen Maschinen zu speichern. Das Verfahren umfasst das Auswählen eines zweiten Modus, um den ersten Cluster zu betreiben, und das Betreiben des ersten Clusters im zweiten Modus unter Umgehung des Cache, wobei direkt auf eine Platte des ersten Datenspeichers geschrieben wird, um erste Daten von den ersten virtuellen Maschinen zu speichern.

KURZE BESCHREIBUNG DER ZEICHNUNGEN

Fig. 1 zeigt ein beispielhaftes Prozessleitsystem, das verwendet werden kann, um die hierin offenbarten Beispiele zu implementieren.

Fig. 2 zeigt ein weiteres beispielhaftes Prozessleitsystem, das verwendet werden kann, um die hier offenbarten Beispiele zu implementieren.

Fig. 3 zeigt ein weiteres beispielhaftes Prozessleitsystem, das verwendet werden kann, um die hier offenbarten Beispiele zu implementieren.

Fig. 4 zeigt eine beispielhafte Benutzerschnittstelle, die während einer Live-Migration in Übereinstimmung mit den Lehren der vorliegenden Offenbarung verwendet werden kann.

Fig. 5 bis 7 zeigen beispielhafte Benutzerschnittstellen, die verwendet werden können, um das Prozessleitsystem aus **Fig. 1** oder die Prozessleitsysteme aus **Fig. 2 und 3** zu implementieren.

Fig. 8 bis 11 zeigen beispielhafte Benutzerschnittstellen, die zur Implementierung des Prozessleitsystems aus **Fig. 3** oder der Prozessleitsysteme aus **Fig. 1 und 2** benutzt werden können.

Fig. 12 zeigt ein Beispiel einer Benutzerschnittstelle, die Performance-Tests und/oder die Diagnose der hier offenbarten beispielhaften Leitsysteme zeigt.

Fig. 13 zeigt eine beispielhafte Anwenderschnittstelle mit Diagnoseergebnissen der hier offenbarten beispielhaften Leitsysteme.

Fig. 14 zeigt eine beispielhafte Anwenderschnittstelle, die den Replikationsverkehr der virtuellen Maschine in Übereinstimmung mit den Lehren der vorliegenden Offenbarung darstellt.

Fig. 15 zeigt eine beispielhafte Benutzerschnittstelle, die verwendet werden kann, um die hierin offenbarten Beispiele zu implementieren.

Fig. 16 veranschaulicht das Schreiben durch den Cache und direkt auf eine Platte in Übereinstimmung mit den Lehren der vorliegenden Offenbarung.

Fig. 17 veranschaulicht das Schreiben in den Speicher in Übereinstimmung mit den Lehren der vorliegenden Offenbarung.

Fig. 18 stellt Prozessströme in einem Normalmodus und den Betrieb in einem abhängigen Modus dar.

Fig. 19 ist ein Ablaufdiagramm, das ein Verfahren zeigt, das ausgeführt werden kann, um die Prozessleitsysteme aus **Fig. 1 bis 3** zu implementieren.

Fig. 20 zeigt eine Prozessorplattform, die verwendet werden kann, um die Prozessleitsysteme aus **Fig. 1 bis 3** zu implementieren.

AUSFÜHRLICHE BESCHREIBUNG

[0009] Virtuelle Umgebungen können in Unternehmen implementiert werden, um Platz im Rechenzent-

rum zu reduzieren, die Sicherheit zu verbessern, Kosten zu reduzieren und/oder den Energieverbrauch zu reduzieren. In einigen Beispielen virtueller Umgebungen arbeiten virtuelle Maschinen (VMs) auf einem einzigen Server, wobei der Verlust der einzelnen Server den Verlust und/oder Betriebsunfähigkeit der zugehörigen VMs verursacht.

[0010] In einigen beispielhaften Prozessleitsystemen ermöglicht die hier offenbarte beispielhafte Virtualisierungsarchitektur die Replikation der VMs auf andere Hosts, um es den VMs zu ermöglichen, nach einem plötzlichen Ausfall des entsprechenden Servers oder anderer Komponenten relativ schnell wieder gestartet zu werden, um so den Verlust der VMs im Wesentlichen zu vermeiden.

[0011] In anderen beispielhaften Prozessleitsystemen werden beispielhafte Hilfshosts in einer Cluster-Umgebung zusammengefasst, wo die VMs laufen und/oder die Daten von jedem der VMs zentral gespeichert und/oder allgemein zugänglich sind. In einigen solchen Beispielen wird die Cluster-Umgebung konfiguriert und/oder eingerichtet, um VMs im Cluster automatisch mit gespeicherten Daten aus dem Datenspeicher neu zu starten, wenn einer der Hosts ausfällt, auf dem die VMs laufen.

[0012] In anderen beispielhaften Prozessleitsystemen werden beispielhafte Hilfshosts in einer Cluster-Umgebung zusammengefasst, und ein zusätzlicher Server wird für die Replikation verwendet (z. B. um Daten, die mit den VMs und/oder den Hosts verbunden sind, zu speichern). Wenn in solchen Beispielen der Cluster ausfällt, können die VMs relativ schnell auf dem zusätzlichen und/oder Replikationsserver unter Verwendung der gespeicherten Daten neu gestartet werden.

[0013] In anderen beispielhaften Prozessleitsystemen können zwei getrennte Cluster von Hilfshosts verwendet werden, um die Replikation zwischen den Clustern zu ermöglichen. In einigen dieser Beispiele werden Daten sowohl aus dem ersten Cluster (z. B. den VMs des ersten Clusters) und dem zweiten Cluster (z. B. den VMs des zweiten Clusters) repliziert und in separaten Datenbanken mit den jeweiligen Clustern gespeichert. Durch Ausführen einiger der ersten VMs im ersten Cluster und der zweiten VMs im zweiten Cluster können, wenn der erste oder zweite Cluster ausfällt, die VMs, die mit dem ausgefallenen Cluster verbunden sind, relativ einfach neu gestartet werden, und/oder auf dem betriebsfähigen Cluster laufen. Somit wird die Regelung der Anlage unter Verwendung der restlichen Cluster betriebsfähig gehalten. In Beispielen, in denen die ersten VMs redundant zu den zweiten VMs sind, können zusätzlich oder alternativ durch Ausführen der ersten VMs im ersten Cluster und der zweiten VMs im zweiten Cluster, wenn der erste

Cluster oder der zweite Cluster ausfällt, VMs, die dem betriebsfähigen Cluster zugeordnet sind, weiter verwendet werden, um den Prozess zu regeln (z. B. können die Sekundär-VMs verwendet werden, um die Anlage zu betreiben). Mehrere Cluster mit Replikation ermöglichen eine hohe Verfügbarkeit.

[0014] Fig. 1 zeigt ein beispielhaftes Prozessleitsystem 100, das einen Cluster 102 aufweist, der Speicherung, Netzwerkrollen und/oder Knoten in einer einzigen Konsole zuordnet und/oder zusammenführt. Bei einigen Beispielen umfasst der Cluster 102 Netzwerkvorrichtungen (beispielsweise Schalter, Firewalls und/oder Speichervorrichtungen), Speicher und/oder Datenspeicher 104, 106, einen Failover-Cluster und/oder Prozessor und eine oder mehrere Gruppen von Servern 108, 110, 112, die im virtuellen Raum zugeordnet sind. In einigen Beispielen kann eine Gruppe von Servern 108, 110, 112 als Knoten bezeichnet werden, und der Cluster 102 kann eine beliebige Anzahl von Knoten (beispielsweise 22, 50, 64 usw.) umfassen. In einigen Beispielen arbeitet der Failover-Cluster als Manager, der die Komponenten des Clusters einer einzigen Konsole zuordnet. In einigen Beispielen können die Benutzer und Dienste, die einen Cluster verwenden, mit einem virtuellen Namen und/oder Raum verbunden werden, der durch den Failover-Cluster erstellt wird.

[0015] Im Betrieb arbeiten virtuelle Maschinen (VMs) 114, 116, 118 und 120 auf den Servern 108, 110, 112, und Daten, die den Aktionen der VMs 114, 116, 118 und/oder 120 zugeordnet sind, werden auf dem Datenspeicher 104, 106 gespeichert. Wenn also einer der Server 108, 110, 112 ausfällt, werden die VMs 114, 116, 118 und/oder 120, die auf dem ausgefallenen Server 108, 110 und/oder 112 ausgeführt wurden, unter Verwendung von Daten, die der VM 114, 116, 118 und/oder 120 zugeordnet sind und die im Datenspeicher 104, 106 gespeichert sind, auf den Servern 108, 110 und/oder 112, die noch in Betrieb sind, neu gestartet. In Beispielen, in denen der Cluster 102 einen ersten Knoten von Servern, die VMs betreiben, und einen zweiten Knoten von Servern umfasst, die VMs betreiben, werden die VMs, die dem ersten Knoten zugeordnet sind, unter Verwendung des zweiten Knotens neu gestartet, wenn der erste Knoten ausfällt.

[0016] Die hier offenbarten Beispiele können bis zu etwa 64 Knoten pro Cluster und bis zu etwa 4.000 VMs unterstützen. Die offenbarten Beispiele ermöglichen die automatische Aktualisierung der Cluster (z. B. Cluster-Aware Updating) und die gleichzeitige Live-Migration von einem oder mehreren der Cluster und/oder den diesen zugeordneten Knoten. Die hier beschriebenen Beispiele können Hyper-V-Replica Broker unterstützen, um Hyper-V-Replica in einer Cluster-Umgebung, Maschinenanwendungsüberwachung und/oder Clustervalidierungstests zu ermöglichen

und/oder zu unterstützen. Die hier offenbarten Beispiele ermöglichen Cluster-Upgrade und/oder Migration, Cluster Shared Volumes 2.0 und/oder VM Failover-Priorisierung. In einigen dieser Beispiele können VM-Gesundheit und Position in die DeltaV-Diagnostik integriert werden, um es den Nutzern zu ermöglichen, sich nicht in andere Umgebungen begeben zu müssen, um ihr System zu diagnostizieren. In einigen Beispielen ermöglicht die Failover-Priorisierung kritische VMs zuerst (z. B. Batch-Executive).

[0017] Fig. 2 zeigt ein Beispielprozessleitsystem 200 mit hoher Verfügbarkeit und einem integrierten Replikationsmodul (z. B. Hyper-V Replica, Vermittlerrolle). In einigen Beispielen ermöglicht diese Replikations-Engine es virtuellen Maschinen (VMs) 202, 204, 206, 208 aus einem Cluster 210, auf einem Replikationsserver (z. B. einem entfernt angeordneten Server) 212 mittels einer einzigen Netzwerkverbindung repliziert zu werden. Bezogen auf Replikation ist Hyper-V-Replica Broker eine Rolle und/oder eine Anwendung, die auf einer oder mehreren der VMs 202, 204, 206, 208 ausgeführt wird, wobei der Cluster 210 bestimmt und/oder festlegt, welcher Knoten die Vermittlerrolle übernimmt. Die Vermittlerrolle ist hoch verfügbar, so dass, wenn ein Knoten verloren geht und/oder ausfällt, die Vermittlerrolle auf einen anderen Knoten übertragen wird. In einigen Beispielen weist der Server 212 zusätzlich oder alternativ Hyper-V Replica Service aus, der darauf ausgeführt wird, um Replikationsdaten zu empfangen und/oder zu synchronisieren.

[0018] In einigen Beispielen enthält die Replikations-Engine ein Modul, das in einer virtuellen Festplattendatei nachverfolgt und/oder schreibt, die von allen VMs 202, 204, 206, 208 verwendet wird, und/oder eine Protokolldatei generiert. Durch Ermöglichen der Replikation auf Ebene einer virtuellen Festplatte (VHD) wird die Leichtigkeit der Replikation von einer beliebigen der VMs 202, 204, 206, 208 erhöht. In einigen Beispielen findet die Replikation der VMs 202, 204, 206, 208 periodisch und/oder asynchron über eine Verbindung (zum Beispiel eine HTTP-Verbindung, eine HTTPS-Verbindung usw.) statt.

[0019] Im Betrieb laufen die VMs 202, 204, 206, 208 auf Servern 214, 216, 218, und Daten, die den VMs 202, 204, 206, 208 zugeordnet sind und/oder Aktionen, die von diesen ausgeführt werden, werden in Datenspeichern 220, 222 gespeichert, und eine Replikation derselben wird auf dem Replikationsserver 212 und/oder an einem anderen Ort gespeichert. Wenn also der Cluster 210 und/oder alle Server 214, 216, 218 ausfallen, werden die VMs 202, 204, 206, 208, die auf dem ausgefallenen Cluster 210 und/oder den ausgefallenen Servern 214, 216, 218 ausgeführt wurden, anhand von Daten, die den VMs 202, 204, 206, 208 zugeordnet sind und die

auf dem Replikationsserver 212 und/oder einem anderen Ort gespeichert sind, auf dem und/oder über den Replikationsserver 212 neu gestartet. Das Beispielprozessleitsystem 200 ermöglicht also, dass die VMs 202, 204, 206, 208 nach einem Ausfall weiter ausgeführt werden und/oder betrieben werden, wodurch eine Erfüllung der Bedingungen im Wesentlichen aller Katastrophenwiederherstellungsprogramme gegeben ist.

[0020] In einigen Beispielen werden die Daten von den VMs 202, 204, 206, 208 unter Verwendung eines Netzwerk-Moduls (beispielsweise Datenkompression und/oder -optimierung), das die Arbeitsbelastung durch Arbeiten in langsamen Netzwerkverbindungen (z. B. WANs) optimiert, auf den Replikationsserver 212 repliziert, transportiert und/oder übertragen. In einigen Beispielen werden zur Implementierung des beispielhaften Prozessleitsystems 200 zwei oder mehr Server 212, 214, 216, 218 verwendet, die durch eine Netzwerkverbindung verbunden sind, wobei mindestens einer der Server der Replikationsserver 212 ist. In einigen solchen Beispielen führen die zwei oder mehr Server 212, 214, 216, 218 eine Anwendung wie Hyper-V aus, und es werden keine anderen Drittanbieter-Hardware, Softwareanwendungen und/oder gemeinsam genutzter Speicher benötigt. In einigen Beispielen umfasst das Prozessleitsystem 200 Wiederherstellungspunkte, um es einer oder mehreren der VMs 202, 204, 206 und 208 zu ermöglichen, im Wesentlichen jederzeit neu gestartet werden zu können. Durch Verwendung derartiger Wiederherstellungspunkte wird eine Beschädigung der Datenbank und/oder Virusreplikation reduziert, im Wesentlichen eliminiert und/oder eliminiert.

[0021] In einigen Beispielen ermöglicht und/oder bietet das Prozessleitsystem 200 und/oder Hyper-V Replica kleinen und/oder mittelständischen Unternehmen eine vollständige Lösung zur Infrastrukturwiederherstellung für ihre virtualisierte Umgebung mit nur wenigen Komponenten und zu einem vernünftigen Preis. In einigen Beispielen kann der Replikationsserver 212 durch DeltaV-Primary- und/oder Secondary-Netzwerke ergänzt werden, um es den VMs 202, 204, 206, 208 zu ermöglichen, auf und/oder mit dem Replikationsserver 212 schneller gestartet zu werden.

[0022] Fig. 3 zeigt ein beispielhaftes Prozessleitsystem 300 mit einem ersten Cluster und/oder Virtualisierungscluster 302 und einem zweiten Cluster und/oder Virtualisierungscluster 304, wobei entweder primäre Server 306, 308, 310 des ersten Clusters 302 oder Replikationsserver 312, 314, 316 des zweiten Clusters 304 für eine Vermittlerrolle (z. B. Hyper-V Replica-Vermittlerrolle) konfiguriert sind. Die Vermittlerrolle ermöglicht es dem Cluster 302 und/oder 304, Hyper-V Replica und/oder einem Teil davon

zugeordnet zu werden, um die nahtlose Replikation zwischen den Clustern 302, 304 zu unterstützen und/oder zu ermöglichen.

[0023] Im Betrieb arbeiten virtuelle Maschinen (VMs) 318, 320, 322, 324 auf den Servern 306, 308, 310, und VMs 326, 328, 330, 332 arbeiten auf den Servern 312, 314, 316. In einigen Beispielen sind die VMs 318, 320, 322, 324 Replikationen der VMs 326, 328, 330, 332. In einigen Beispielen unterscheiden sich die VMs 318, 320, 322, 324 teilweise oder vollständig von den VMs 326 328, 330, 332.

[0024] In den Beispielen, in denen die VMs 318, 320, 322, 324 Replikationen der VMs 326, 328, 330, 332 sind, werden die den VMs 318, 320, 322 zugeordneten Daten und/oder die von ihnen ausgeführten Aktionen auf Datenspeichern 334, 336 des ersten Clusters 302 gespeichert, und eine Replikation der Daten wird in Datenspeichern 338, 340 des zweiten Clusters 304 gespeichert. Wenn somit der erste Cluster 302 und/oder alle Server 306, 308, 310 ausfallen, werden die VMs 318, 320, 322, 324, die auf dem ausgefallenen Cluster 302 und/oder den ausgefallenen Server 306, 308, 310 ausgeführt wurden, mithilfe von Daten, die den VMs 318, 320, 322, 324 zugeordnet sind und in den Datenspeichern 338, 340 gespeichert sind auf den und/oder mit den Replikationsservern 312, 314, 316 neu gestartet. Das beispielhafte Prozessleitsystem 300 ermöglicht es den VMs 318, 320, 322, 324, nach einem Ausfall weiter ausgeführt zu werden und/oder in Betrieb zu bleiben.

[0025] In Beispielen, in denen sich die VMs 318, 320, 322, 324 von den VMs 326 328, 330, 332 unterscheiden, werden Daten, die den VMs 318, 320, 322, 324 zugeordnet sind, und/oder von diesen ausgeführte Aktionen auf den Datenspeichern 334, 336 des ersten Clusters 302 gespeichert, und eine Replikation der Daten wird in den Datenspeichern 338, 340 des zweiten Clusters Cluster 304 gespeichert, und Daten, die den VMs 326, 328, 330, 332 zugeordnet sind, und/oder von diesen ausgeführte Aktionen werden in den Datenspeichern 338, 340 des zweiten Clusters 304 gespeichert und eine Replikation der Daten wird in den Datenspeichern 334, 336 des ersten Clusters 302 gespeichert. Wenn also der erste Cluster 302 oder der zweite Cluster 304 ausfällt, werden die VMs 318, 320, 322, 324, 326, 328, 330, 332, die auf dem ausgefallenen ersten Cluster 302 oder dem ausgefallenen zweiten Cluster 304 ausgeführt wurden, anhand von Daten, die den jeweiligen VMs 318, 320, 322, 324, 326, 328, 330, 332 zugeordnet sind und in den Datenspeichern 334, 336, 338 und/oder 340 gespeichert sind, auf und/oder mit den Servern 306, 308, 310 oder den Replikationsservern 312, 314, 316 neu gestartet. Sobald der ausgefallene erste Cluster 302 oder der ausgefallene Cluster 304 wiederhergestellt wurde, können die VMs 318, 320,

322, 324, 326, 328, 330, 332 auf ihren ursprünglichen Cluster 302 oder 304 repliziert und/oder migriert werden, um den normalen Betrieb wieder aufzunehmen.

[0026] Fig. 4 stellt eine beispielhafte Benutzerschnittstelle 400 dar, die während einer Live-Migration verwendet werden kann. In einigen Beispielen kann eine Shared-Nothing-Live-Migration etwa 8 bis 10 Minuten in Anspruch nehmen, eine Storage-Live-Migration kann etwa 5 bis 7 Minuten in Anspruch nehmen, eine Live-Migration über CSV (Comma Separated Value) kann etwa 10 bis 30 Sekunden und eine schnelle Migration über CSV kann etwa 5 Sekunden dauern.

[0027] Fig. 5 bis 7 zeigen beispielhafte Benutzerschnittstellen 500, 600, 700, die verwendet werden können, um die hier offenbarten Beispiele zu implementieren (z. B. Konfiguration und Manipulation von virtuellen Maschinen). In einigen Beispielen können die Benutzerschnittstellen 500, 600, 700 verwendet werden, um das Prozessleitsystem 100 zu implementieren. 1. Die Benutzerschnittstelle 600 aus **Fig. 6** zeigt den Status der verschiedenen Betriebsvorgänge des ausgewählten C1-1-Knotens und/oder ermöglicht die Auswahl der verschiedenen Knoten (z. B. C1-1; C 1-2; C 1-3 usw.) und verschiedenen Cluster (zum Beispiel 1, 10, 11, 2, 3 usw.) in ihre jeweiligen Status. Die Benutzerschnittstelle 700 aus **Abb. 7** zeigt den Status der verschiedenen Betriebsvorgänge des ausgewählten C1-1-Knotens und stellt eine Zusammenfassung des Clusters 1 bereit.

[0028] Fig. 8 bis 11 zeigen beispielhafte Benutzerschnittstellen 800, 900, 1000, 1100, die verwendet werden können, um die hier offenbarten Beispiele zu implementieren. In einigen Beispielen können die Benutzerschnittstellen 800, 900, 1000, 1100 verwendet werden, um das Prozessleitsystem 300 zu implementieren. 3, wenn ein Betriebsvorgang (z. B. OP20) und/oder die virtuelle Maschine 318 sich aufgrund eines Ausfalls vom ersten Cluster 302 zum zweiten Cluster 304 bewegt. **Fig. 9** zeigt die Benutzerschnittstelle 900, die angibt, dass ein Ausfall von OP20 und/oder der virtuellen Maschine 318 aufgetreten ist. **Fig. 10** zeigt die Benutzerschnittstelle 1000, die angibt, dass OP20 und/oder die virtuellen Maschine 318 auf dem zweiten Cluster 304 ausgeführt wird. **Fig. 11** zeigt die Migration der virtuellen Maschine 318 von dem zweiten Cluster 304 zurück zu dem ersten Cluster 302, nachdem der erste Cluster 302 repariert wurde.

[0029] Fig. 12 stellt eine beispielhafte Benutzerschnittstelle 1200 dar, die Leistungstests und/oder Diagnose der hier offenbarten Leitsysteme zeigt. In einigen Beispielen umfassen die beispielhaften Prozessleitsysteme 2 Cluster, die 64 virtuelle Maschinen, 1 ProPlus, 13 Anwendungsstationen, 50

Betriebsstationen und/oder 4 Hosts pro Cluster ausführen. In einigen Beispielen umfassen die ausgeführten oder erzielten Metriken Systemkonfiguration, Betriebslaufzeit, historische Sammlung und/oder Replikation und/oder sind diesen zugeordnet. In einigen Beispielen umfasst die ausgeführte und/oder erzielte Diagnostik Windows-und/oder SolarWinds.

[0030] Fig. 13 stellt eine beispielhafte Benutzerschnittstelle 1300 dar, die Diagnoseergebnisse des hier offenbarten beispielhaften Leitsystems zeigt. Die Benutzerschnittstelle 1300 zeigt eine Hyper-V-Übersicht unter Darstellung der Virtualisierungsanlagen (z. B. C1-1; C1 -2; usw.), die Virtualisierungsanlagenübersicht, die führenden 10 Hosts nach belegtem Speicher und die führenden 10 Hosts nach Netzwerkauslastung .

[0031] Fig. 14 stellt eine beispielhafte Benutzerschnittstelle 1400 dar, die den Replikationsverkehr der virtuellen Maschine zeigt, wobei ein erster Teil 1402 der Zeit entspricht, wenn die Replikation deaktiviert ist (z. B. bei der Konfiguration), und ein zweiter Abschnitt 1404 der Zeit entspricht, wenn das System und/oder ein Cluster (z. B. C2-1) neu gestartet wird, und ein dritter Teil 1406 der Zeit entspricht, wenn die virtuellen Maschinen neu gestartet werden.

[0032] Fig. 15 zeigt eine beispielhafte Benutzerschnittstelle 1800, die verwendet werden kann, um die hier offenbarten Beispiele zu implementieren (z. B. automatisches Erstellen, Konfigurieren und Manipulieren von vorinstallierten virtuellen DeltaV-Maschinen für die Prozessleitung). Insbesondere zeigt die Benutzerschnittstelle 1800 diagnostische Warnungen im Zusammenhang mit einer virtuellen Maschine, die in einem normalen Modus arbeitet, obwohl die virtuelle Maschine eigentlich in einem robusten Modus arbeiten sollte. Somit beschreibt die Benutzerschnittstelle 1800 die Informationen, die bis zur IT-Schicht weitergeleitet werden, um bei Bedarf eine Korrektur des Betriebsmodus zu veranlassen. In einigen Beispielen wird ein Werkzeugtipp 1802 angezeigt, wenn ein Benutzer z. B. einen Maus-Pfeil über die aufgelaufenen diagnostischen Zahlen in der Statusleiste bewegt.

[0033] Fig. 16 veranschaulicht das Schreiben durch den Cache und direkt auf eine Platte (z. B. Umgehen des Caches; robuster Modus) und **Fig. 17** veranschaulicht das Schreiben in den Speicher (beispielsweise Normalbetrieb). **Fig. 18** stellt Prozessströme in einem Normalmodus und den Betrieb in einem abhängigen Modus dar. In einigen Beispielen ermöglicht es die Änderung des Caches dem Prozessleitsystem, einem Hardware-Fehler standzuhalten, ohne dass die zur Regelung verwendeten Konfigurationsdaten beschädigt werden.

[0034] Ein Ablaufdiagramm, das ein beispielhaftes Verfahren für die Durchführung der Prozessleitsysteme der Abbildungen darstellt, ist in **Fig. 19** gezeigt. In diesem Beispiel kann das Verfahren unter Verwendung von maschinenlesbaren Befehlen implementiert werden, die ein Programm zur Ausführung durch einen Prozessor umfassen, wie etwa den Prozessor 2312, der in der beispielhaften Prozessorplattform 2300 dargestellt ist, die nachstehend in Verbindung mit **Fig. 20** erörtert wird. Das Programm kann in Software verkörpert sein, die auf einem greifbaren computerlesbaren Speichermedium, wie einer CD-ROM, einer Diskette, einer Festplatte, einer Digital Versatile Disk (DVD), einer Blu-ray-Disk oder einem Speicher gespeichert ist, der dem Prozessor 2312 zugeordnet ist, doch können das gesamte Programm und/oder Teile davon alternativ auch von einer anderen Vorrichtung als dem Prozessor 2312 ausgeführt werden und/oder in Firmware oder dedizierter Hardware verkörpert sein. Ferner können, obwohl das beispielhafte Programm unter Bezugnahme auf das Flussdiagramm beschrieben aus **Fig. 19** beschrieben wurde, viele andere Verfahren der Implementierung der Prozessleitsysteme der aus **Fig. 1** bis 3 alternativ verwendet werden. Zum Beispiel kann die Reihenfolge der Ausführung der Blöcke verändert werden, und/oder einige der beschriebenen Blöcke können verändert, eliminiert oder kombiniert werden.

[0035] Wie oben erwähnt, kann das beispielhafte Verfahren aus **Fig. 19** unter Verwendung codierter Anweisungen implementiert werden (z. B. computer- und/oder maschinenlesbarer Anweisungen) und auf einem greifbaren computerlesbaren Speichermedium gespeichert werden, wie einer Festplatte, einem Flash-Speicher, einem Lesespeicher (ROM), einer Compact Disk (CD), einer Digital Versatile Disk (DVD), einem Cache, einem Schreib-/Lesespeicher (RAM) und/oder einer beliebigen anderen Speichervorrichtung oder Speicherplatte, auf der Informationen für jede beliebige Zeitdauer gespeichert werden können (beispielsweise für längere Zeiträume, dauerhaft, für kurze Momente, zur vorübergehenden Pufferung und/oder zur Zwischenspeicherung der Informationen). Im hier verwendeten Sinne ist der Begriff des greifbaren computerlesbaren Speichermediums ausdrücklich derart definiert, dass er jede Art von computerlesbarem Speichermedium und/oder computerlesbarer Speicherplatte einschließt und die Verbreitung von Signalen sowie Übertragungsmedien ausschließt. Im hier verwendeten Sinne sind die Begriffe „greifbares computerlesbares Speichermedium“ und „greifbares maschinenlesbares Speichermedium“ austauschbar. Zusätzlich oder alternativ können die beispielhaften Prozesse aus **Fig. 1** bis 3 unter Verwendung codierter Anweisungen (z. B. computer- und/oder maschinenlesbarer Anweisungen), die auf einem nicht-transitorischen Computer und/oder maschinenlesbaren

Medium gespeichert sind, wie einer Festplatte, einem Flash-Speicher, einem Lesespeicher, einer CD, einer DVD, einem Cache, einem Schreib-/Lesespeicher und/oder einer beliebigen anderen Speichervorrichtung oder Speicherplatte, auf der Informationen für jede beliebige Dauer gespeichert werden (z. B. über längere Zeiträume, dauerhaft, für kurze Momente, zur vorübergehenden Pufferung und/oder zur Zwischenspeicherung der Informationen), implementiert werden. Im hier verwendeten Sinne ist der Begriff „nicht-transitorisches computerlesbares Medium“ ausdrücklich derart definiert, dass er jede Art von computerlesbarem Speichermedium und/oder computerlesbarer Speicherplatte einschließt und die Verbreitung von Signalen sowie Übertragungsmedien ausschließt. Wenn im hier verwendeten Sinne der Begriff „wenigstens“ als Übergangsbegriff im Oberbegriff eines Anspruchs verwendet wird, so ist er ebenso offen wie der Begriff „umfassend“.

[0036] Das beispielhafte Verfahren aus **Fig. 19** beginnt mit einem ersten Modus, der gewählt wird, um den ersten Cluster 302 zu betreiben (Block 2202). Der erste Cluster 302 umfasst die virtuellen Maschinen 318, 320, 322, 324 und die Server 306, 308, 310. Der erste Modus wird ausgewählt, um den zweiten Cluster 304 zu betreiben (Block 2204). Der zweite Cluster 304 umfasst die zweiten virtuellen Maschinen 326, 328, 330, 332 und die Server 312, 314, 316. Bei einigen Beispielen umfasst der erste Modus das Schreiben in einen Cache-Datenspeicher 334, 336, 338 und/oder 340 und kann einem Konfigurationsbetriebsmodus zugeordnet sein. Die ersten virtuellen Maschinen 318, 320, 322, 324 werden im ersten Modus betrieben (Block 2206). Die zweiten virtuellen Maschinen 326, 328, 330, 332 werden ebenfalls im ersten Modus betrieben (Block 2208).

[0037] Erste Daten von den ersten virtuellen Maschinen 318, 320, 322, 324 werden auf den Datenspeichern 334 und/oder 336 des ersten Cluster 302 gespeichert, und eine Replikation der ersten Daten von den ersten virtuellen Maschinen 318, 320, 322, 324 wird auf den Datenspeichern 338 und/oder 340 des zweiten Clusters 304 gespeichert (Block 2210). In einigen Beispielen werden die ersten Daten und die Replikation der ersten Daten parallel und/oder in einer einheitlichen Weise unter Verwendung von Einheitlichkeit auf Dateiebene gespeichert. Zweite Daten von den zweiten virtuellen Maschinen 326, 328, 330, 332 werden auf dem Datenspeicher 338 und/oder 340 des zweiten Clusters 304 gespeichert und eine Replikation der zweiten Daten der zweiten virtuellen Maschinen 326, 328, 330, 332 wird auf den Datenspeichern 334 und/oder 336 des ersten Clusters 302 gespeichert (Block 2212). In einigen Beispielen werden die zweiten Daten und die Replikation der zweiten Daten parallel und/oder in

einer einheitlichen Weise unter Verwendung von Einheitlichkeit auf Dateiebene gespeichert.

[0038] Nachdem das Prozessleitsystem 300 konfiguriert wurde, kann ein zweiter Modus ausgewählt werden, um den ersten Cluster 302 zu betreiben (Block 2214). Der zweite Modus wird gewählt, um den zweiten Cluster 304 zu betreiben (Block 2216). Bei einigen Beispielen umfasst der zweite Modus die Umgehung des Caches der Datenspeicher 334, 336, 338 und/oder 340 und das direkte Schreiben auf eine Platte (n) von den Datenspeichern 334, 336, 338 und/oder 340, um die Zeit zu reduzieren, in der die Daten in einem flüchtigen Speicher gehalten werden. Die ersten virtuellen Maschinen 318, 320, 322, 324 werden im zweiten Modus betrieben (Block 2218). Die zweiten virtuellen Maschinen 326, 328, 330, 332 werden im zweiten Modus betrieben (Block 2220).

[0039] Bei Block 2222 bestimmt ein Prozessor, ob ein Ausfall des ersten Clusters 302 aufgetreten ist. Wenn der erste Cluster 302 ausgefallen ist, werden die ersten virtuellen Maschinen 318, 320, 322, 324 unter Verwendung der Server 312, 314 und/oder 316 des zweiten Clusters 304 und der Replikation der ersten Daten von den Datenspeichern 338 und/oder 340 neu gestartet (Block 2224). In einigen Beispielen kann ein Alarm erzeugt und/oder übermittelt werden, wenn die ersten virtuellen Maschinen 318, 320, 322, 324 unter Verwendung der Server 312, 314 und/oder 316 neu gestartet werden und/oder falls der erste Cluster 302 ausfällt (Block 2226). In einigen Beispielen gibt der Alarm an, dass sowohl die ersten als auch die zweiten virtuellen Maschinen 318, 320, 322, 324, 326, 328, 330, 332 unter Verwendung der Server 312, 314 und/oder 316 des zweiten Clusters 304 ausgeführt werden. Nachdem der Ausfall des ersten Clusters 302 behoben wurde, kann eine Live-Migration vom Betriebs der ersten virtuellen Maschinen 318, 320, 322, 324 mit den Servern 338, 340 des zweiten Clusters 304 zum Betrieb der ersten virtuellen Maschinen 318, 320, 322, 324 mit den Servern 334, 336 des ersten Clusters 302 durchgeführt werden (Block 2228).

[0040] Bei Block 2230 bestimmt ein Prozessor, ob ein Ausfall des zweiten Clusters 304 aufgetreten ist. Wenn der zweite Cluster 304 ausgefallen ist, werden die zweiten virtuellen Maschinen 326, 328, 330, 332 unter Verwendung der Server 306, 308 und/oder 310 des ersten Clusters 302 und der Replikation der ersten Daten von den Datenspeichern 334 und/oder 336 (Block 2232) neu gestartet. In einigen Beispielen kann ein Alarm erzeugt und/oder übermittelt werden, wenn die zweiten virtuellen Maschinen 326, 328, 330, 332 mit den Servern 306, 308, 310 neu gestartet wurden und/oder der zweite Cluster 304 ausgefallen ist (Block 2234). In einigen Beispielen gibt der Alarm an, dass sowohl die ersten als auch die zwei-

ten virtuellen Maschinen 318, 320, 322, 324, 326, 328, 330, 332 unter Verwendung der Server 306, 308, 310 des ersten Clusters 302 ausgeführt werden. Sobald der Ausfall des zweiten Clusters 304 behoben wurde, kann eine Live-Migration vom Betrieb der zweiten virtuellen Maschinen 326, 328, 330, 332 mit den Servern 334, 336 des ersten Clusters 302 zum Betrieb der zweiten virtuellen Maschinen 326, 328, 330, 332 mit den Servern 338, 340 des zweiten Clusters 304 durchgeführt werden (Block 2236). Bei Block 2238 bestimmt der Prozess, ob er beendet werden soll oder nicht. Wenn der Prozess fortgesetzt werden soll, wird an Block 2240 eine Bestimmung vorgenommen, ob der Betriebsmodus zwischen dem zweiten Betriebsmodus und dem ersten Betriebsmodus gewechselt wird.

[0041] Die Prozessorplattform 2300 des in **Fig. 20** dargestellten Beispiels umfasst einen Prozessor 2312. Der Prozessor 2312 aus dem dargestellten Beispiel ist Hardware. Zum Beispiel kann der Prozessor 2312 mit einem oder mehreren integrierten Schaltkreisen, Logikschaltungen, Mikroprozessoren oder Steuerungen einer beliebigen Familie oder eines beliebigen Herstellers implementiert werden.

[0042] Der Prozessor 2312 des dargestellten Beispiels umfasst einen lokalen Speicher 2313 (beispielsweise einen Cache). Der Prozessor 2312 des dargestellten Beispiels steht über einen Bus 2318 in Kommunikationsverbindung mit einem Hauptspeicher, der einen flüchtigen Speicher 2314 und einem nicht-flüchtigen Speicher 2316. Der flüchtige Speicher 2314 kann durch Synchronous Dynamic Random Access Memory (SDRAM), Dynamic Random Access Memory (DRAM), RAMBUS Dynamic Random Access Memory (RDRAM) und/oder jede andere Art von Direktzugriffsspeichervorrichtung implementiert werden. Der nicht-flüchtige Speicher 2316 kann durch Flash-Speicher und/oder jede andere gewünschte Art von Speichereinrichtung implementiert werden. Der Zugriff auf den Hauptspeicher 2314, 2316 wird durch eine Speichersteuerung gesteuert.

[0043] Die Prozessorplattform 2300 im dargestellten Beispiel weist auch eine Schnittstellenschaltung 2320 auf. Die Schnittstellenschaltung 2320 kann durch jede Art von Schnittstellenstandard, wie beispielsweise eine Ethernet-Schnittstelle, einen universellen seriellen Bus (USB) und/oder PCI-Express-Schnittstelle implementiert werden.

[0044] In dem dargestellten Beispiel sind eine oder mehrere Eingabevorrichtungen 2322 mit der Schnittstellenschaltung 2320 verbunden. Die Eingabevorrichtung(en) 2322 erlauben es einem Benutzer, Daten und Befehle in den Prozessor 2312 einzugeben. Die Eingabevorrichtung(en) können beispielsweise durch einen Audio-Sensor, ein Mikrofon, eine

Kamera (Standbild oder Video), eine Tastatur, einen Knopf, eine Maus, einen Touchscreen, ein Track-Pad, einen Trackball, einen Isopoint und/oder ein Spracherkennungssystem realisiert werden.

[0045] Eine oder mehrere Ausgabevorrichtungen 2324 sind ebenfalls mit der Schnittstellenschaltung 2120 des gezeigten Beispiels verbunden. Die Ausgabevorrichtungen 23124 können zum Beispiel durch Anzeigevorrichtungen (beispielsweise eine Leuchtdiode (LED), eine organische Leuchtdiode (OLED), eine Flüssigkristallanzeige, eine Kathodenstrahlröhrenanzeige (CRT), einen Touchscreen, ein taktiles Ausgabegerät, eine Leuchtdiode (LED), einen Drucker und/oder Lautsprecher) implementiert werden. Die Schnittstellenschaltung 2320 des gezeigten Beispiels umfasst somit typischerweise eine Grafiktreiberkarte, einen Grafiktreiberchip oder einen Grafiktreiber-Prozessor.

[0046] Die Schnittstellenschaltung 2320 des gezeigten Beispiels weist auch eine Kommunikationsvorrichtung auf, wie etwa einen Sender, einen Empfänger, einen Sende-Empfänger, ein Modem und/oder eine Netzwerkschnittstellenkarte, um den Austausch von Daten mit externen Geräten (z. B. Computer jeglicher Art) über ein Netzwerk 2326 zu ermöglichen (z. B. eine Ethernet-Verbindung, eine digitale Teilnehmerleitung (DSL), eine Telefonleitung, ein Koaxialkabel, ein Mobiltelefonsystem usw.).

[0047] Die Prozessorplattform 2300 im dargestellten Beispiel weist auch eine oder mehrere Massenspeichervorrichtungen 2328 zum Speichern von Software bzw. Daten auf. Beispiele solcher Massenspeichervorrichtungen 2328 umfassen Diskettenlaufwerke, Festplatten, CD-Laufwerke, Blu-ray-Disc-Laufwerke, RAID-Systeme und Digital Versatile Disk(DVD)-Laufwerke.

[0048] Aus dem Vorstehenden wird deutlich, dass die hier offenbarten Beispiele Computerarchitektur (en), Vernetzung bzw. Virtualisierung betreffen, die die Entwicklung von wirksamen, leicht zu verwaltenen, virtualisierten Rechenumgebungen ermöglichen, wodurch Kosten reduziert werden und die Systembetriebszeit bzw. -anlaufzeit verbessert wird.

[0049] In einigen Prozessleitsystemen sind wenigstens einige kritische Hardwarekomponenten (z. B. Controller) redundant ausgelegt, während andere Komponenten (z. B. VMs, Server usw.) nicht redundant sind oder anderweitig in der Lage sind, sich schnell von Störungen wie Stromausfällen, Computerhardwarefehlern usw. zu erholen oder damit umzugehen. Wenn Ausfälle in solchen Systemen auftreten, können die redundanten kritischen Komponenten (z. B. die Steuerung) der Anlage ermöglichen, weiter zu arbeiten. Jedoch kann die Wiederherstellung der nicht redundanten Komponenten (z.

B. VMs, Server etc.) bzw. der Anlage in den normalen Betriebszustand eine erhebliche Menge an Zeit verbrauchen, in der Anlagendaten (z. B. wichtige Daten) eventuell nicht sichtbar oder nicht abrufbar sind bzw. verloren gehen.

[0050] Im Gegensatz zu einigen bekannten Systemen bieten bzw. ermöglichen die hier offenbarten Beispiele für Prozessleitsysteme eine hohe Verfügbarkeit bzw. Replikation, wodurch bei Störungen die Ausfallzeiten in virtuellen Umgebungen reduziert werden. Ferner beugen die offenbarten Beispiele deutlich dem Datenverlust bzw. der Beschädigung der Daten vor, indem dem Prozessleitsystem ermöglicht wird, Protokolldateien für alle Systemkomponenten zu schreiben oder verfügbar zu machen bzw. indem sichergestellt wird, dass das System sich der hochgradig verfügbaren Virtualisierungsarchitektur des Gesamtsystems bewusst ist. Einfach ausgedrückt, bieten die hier offenbarten Beispiele ein sofort funktionsbereites Hochverfügbarkeitsleitsystem.

[0051] Um den Zugriff auf die Protokolldateien der Komponenten zu ermöglichen, greifen mehrere Host-Computer zur gleichen Zeit parallel auf redundante Festplatten zu oder schreiben darauf (beispielsweise eine redundante Anordnung unabhängiger Festplatten (RAID)). Unter Verwendung der offenbarten Beispiele sind virtuelle Festplatten virtueller Maschinen sichtbar bzw. können von mehreren Host-Computern gesehen werden. In solchen Fällen, wenn in einem der Host-Computer ein Fehler auftritt (z. B. ein Hardware-Fehler, ein Software-Fehler), kann ein anderer Host-Computer eine zugehörige virtuelle Maschine übernehmen bzw. ausführen und eine zugehörige virtuelle Maschine ausgehend von den auf der Festplatte gespeicherten Daten neu starten.

[0052] In einigen hier offenbarten Beispielen für Prozessleitsysteme werden die Daten zum besseren Schutz gegen Datenverlust in einer einheitlichen Art und Weise mit einer Einheitlichkeit auf Dateiebene geschrieben, zum Beispiel unter Verwendung eines NTFS-Dateisystems, eines NTFS-Logs (Protokoll), einer Volume Shadow Copy (VSS) bzw. eines Transactional NTFS. Zusätzlich bzw. alternativ ermöglichen die hier offenbarten Prozessleitsysteme es dem System, angewiesen zu werden, über den Cache direkt auf ein Speichermedium, eine Festplatte bzw. eine virtuelle Festplatte zu schreiben und zu speichern. Um in solchen Beispielen die logische Dateieinheitlichkeit in einer objektorientierten Datenbank möglich zu machen, wird das beispielhafte Prozessleitsystem und/oder eine Komponente davon in einem robusten Modus ausgeführt, der die direkte Schreibfreigabe über den Cache ermöglicht und deutlich die Zeit minimiert, in der sich zum Beispiel flüchtige Daten im Speicher befinden. In einigen

Beispielen führt das Schreiben durch den Cache zu verminderter Leistung, weil Festplatten langsamer als der Speicher schreiben. Wenn beispielsweise der Cache verwendet wird, gibt die Anwendung bzw. der Benutzer Daten an den Cache weiter und das System schreibt die Daten später weiter. Da jedoch ein Großteil der Konfiguration erfolgt, bevor das System online geht, um eine Anlage bzw. ein Prozessleitsystem zu regeln, kann der robuste Modus während des anfänglichen Aufbaus und/oder der anfänglichen Konfiguration abgeschaltet werden, da Datenverluste und/oder -beschädigungen zu dieser Zeit keine negativen Auswirkungen auf den Betrieb der Anlage haben (da kein Prozess geregelt wird).

[0053] In einigen Beispielen werden die Leitsystemfunktionen und die Virtualisierungsfunktionen integriert, um zu ermöglichen, dass das gesamte System und/oder wesentliche Teile davon einheitlich sind. Um zum Beispiel diese Integration zu ermöglichen, werden die Einstellungen für den robusten bzw. normalen Betriebsmodus an die Virtualisierungs-Management-Software übermittelt, um die Sichtbarkeit auf dem gleichen Niveau wie die Einstellungen für Virtualisierung und Hochverfügbarkeit zu ermöglichen.

[0054] In anderen Beispielen werden Leitsystemfunktionen und Virtualisierungsfunktionen mit einem Prozessleitsystem (z. B. DVS) integriert, das die VM-Knotenredundanz und Hochverfügbarkeit übereinstimmend verwaltet. In einigen dieser Beispiele wird ein Alarm (z. B. DVS-Alarm) ausgelöst, wenn ein redundantes Paar von Anwendungsstationen (z. B. DeltaV-Anwendungsstationen) dem gleichen Host-Computer zugeordnet werden, da dem gleichen Host zugeordnete Anwendungsstationen eine ungewöhnlich lange Unterbrechung verursachen und/oder auslösen können, wenn der Host ausfällt. In einigen Beispielen werden Redundanzen und Standbys über mehrere Hosts verteilt, um einen Knotenwechsel zu ermöglichen, wenn ein Fehler in einem der Hosts auftritt.

[0055] Virtualisierte Rechnerplattformen, wie hier offenbart, ermöglichen es, nichttraditionelle Messungen mit Wireless-Infrastruktur zur fortgeschrittenen Analyse zu integrieren, um prädiktive Algorithmen und Automatisierung zu kombinieren, damit Online-Vorhersagefähigkeiten und/oder neue Auslegungen und/oder Plattformen verwendet werden können, um Menschen in Bewegung und im Kontrollraum zu verbinden. Solche Beispiele können zur Virtualisierung und Big Data in der Prozessregelung nützlich sein.

[0056] Einige der offenbarten Beispiele ermöglichen eine hohe Verfügbarkeit, Failover-Cluster-Management, Replikation virtueller Maschinen, Katastro-

phenwiederherstellung, Leistungsdaten- und/oder Plattform-Konfigurationen. Einige der offenbarten Beispiele ermöglichen die Hochskalierung von Virtualisierungen, Live-Migration (z. B. schnelle, uneingeschränkte, simultane Live-Migrationen virtueller Maschinen bzw. Speichermigrationen), Shared-Nothing-Migrationen, hochverfügbares Failover-Clustering (z. B. iSCSI, Fibre-Channel, SMB), Priorisierung der Ausfallsicherung, NIC-Teaming bzw. Verwaltung und/oder Replikationsunterstützung.

[0057] Einige der offenbarten Beispiele ermöglichen einem Host, bis zu etwa 160 logische Prozessoren bzw. bis zu etwa 2 Terabyte (TB) an Speicher und/oder bis zu etwa 1024 virtuelle Zentralprozessoreinheiten (CPUs) pro Host zu beinhalten. Einige der offenbarten Beispiele ermöglichen einer virtuellen Maschine, bis zu etwa 32 virtuelle CPUs pro Host, bis zu etwa 1 TB Speicher pro VM und/oder bis zu etwa 1024 aktive VMs pro Host zu beinhalten und/oder diesen zugeordnet zu sein. Einige der offenbarten Beispiele ermöglichen es einem Cluster, etwa 64 Knoten und/oder etwa 4000 VMs und/oder diesen zugeordnet zu sein.

[0058] Die hier offenbarten Beispiele können verwendet werden, um Datenströme (z. B. Big Data) abzurufen bzw. zu analysieren, um neue Erkenntnisse zu gewinnen, bessere Entscheidungen zu treffen und/oder eine verbesserte Systemerfahrung zu erhalten. In einigen Beispielen können die hier offenbarten Beispiele den Empfang periodischer unkomprimierter Daten unterstützen, die an allen Prozessmessungen und/oder -ausgängen gesammelt werden, damit verschiedene Arten von Daten analysiert oder unterstützt werden können. Einige hier offenbarte Beispiele können Informationsquellen wie traditionelle Arten von Alarmen, Chargen und/oder kontinuierliche neue Quellen wie einen On-Stream-Analysatoren unterstützen. Einige hier offenbarte Beispiele können über Anlagengrenzen hinweg (z. B. ganze Ölfelder) Datenagilität unterstützen und/oder Anwendern über Browser und/oder mobile Geräte verwertbare Informationen liefern. Zum Beispiel kann die Datenuntersuchung beispielsweise mit drahtlosen Sensoren (z. B. Vibrationssensoren) zeigen und/oder identifizieren, dass Informationen, die bisher nicht aufgezeichnet wurden, nun aufgezeichnet werden sollten.

[0059] Die beispielhaften Prozessleitsysteme ermöglichen es, alle oder im Wesentlichen alle Daten in hoher Auflösung aufzubereiten, um eine Analyse zu ermöglichen. Die erhaltenen Daten können bereinigt und/oder gefiltert und/oder unter Verwendung von beispielsweise Hochfrequenz-Echtzeit-Daten untersucht werden, die einen besseren Einblick bieten und/oder ermöglichen. Die beispielhaften Prozessleitsysteme ermöglichen und/oder bringen übergreifende Analysen aus Systemen,

Plattformen und/oder Anlagen zusammen. Die beispielhaften Prozessleitsysteme ermöglichen und/oder unterstützen zusätzliche Messungen, die über drahtlose und/oder externe Messungen erhalten werden, zum Beispiel über REST API, die mit jeder Sprache aufgerufen werden können. Die Prozessleitsysteme bieten umsetzbare Ergebnisse für den Menschen und/oder geschlossene Regelkreise.

[0060] In einigen Beispielen werden Vorlagen für Hochverfügbarkeit und Replikation entwickelt, um Benutzern zu ermöglichen, diese Funktionen als Teil der Vorlage zu aktivieren und dann die Möglichkeiten zu nutzen, während wir Instanzen aus der Vorlage erstellen. In einigen Beispielen wird die Netzwerkinfrastruktur in virtualisierten Umgebungen vordefiniert, um leicht ein Regelungsnetzwerk (z. B. ein primäres Regelungsnetzwerk, ein sekundäres Regelungsnetzwerk), das Host-/Management-Netzwerk und/oder andere Netzwerke (z. B. das Anlagenetzwerk) zu unterscheiden. In einigen Beispielen wird ein Domänencontroller in die Infrastruktur integriert, damit Benutzer ihre Rechenressourcen nicht einrichten und verwalten müssen. In einigen Beispielen wird die Diagnostik in das Regelungssystem integriert, um dem Benutzer zu ermöglichen, keine Werkzeuge (z. B. SolarWinds) verwenden zu müssen, um Informationen bezüglich seiner Netze, Festplatten-Subsysteme, CPU-Ressourcen, Speicherressourcen, IO-Durchsatz, IO-Verzögerungen, Cluster und/oder Replikationssysteme zu sammeln. In einigen Beispielen wird die Diagnostik im Kontext der Regelung gemeldet, im Gegensatz zum eher abstrakten Wesen von IT-Produkten. In einigen Beispielen wird ein Beispiel-Verwaltungszentrum mit DeltaV-Konfiguration und Diagnose integriert. In einigen Beispielen kann die Diagnose des Verkehrs und/oder der Nutzung von Regelungssystemen durchgeführt werden (z. B. für Warnungen, wenn sich Nutzungsmuster ändern usw.). In einigen Beispielen wird das System durch Segmentieren des Netzwerkverkehrs und Sicherung des Infrastrukturverkehrs gesichert und/oder verwaltet. In einigen Beispielen kann eine Hardware-Plattform unter Virtualisierung verwendet werden, um die offenbarten Beispiele zu implementieren. In einigen Beispielen ermöglicht Virtual Studio, DeltaV-Infrastruktur, Hardware (z. B. VRTX-Plattform von Dell) eine virtuelle Anwendung der Prozessregelung. In einigen Beispielen können die beispielhaften Prozessregelungssysteme Vorlagen, Diagnose- und/oder Konfigurationswerkzeuge und/oder -aspekte beinhalten und/oder verwenden.

[0061] Wie hier dargelegt, schließt ein beispielhaftes Verfahren den Betrieb eines ersten Clusters mit ersten virtuellen Maschinen und ersten Servern sowie den Betrieb eines zweiten Cluster mit zweiten virtuellen Maschinen und zweiten Servern ein. Das beispielhafte Verfahren schließt Speichern von ers-

ten Daten der ersten virtuellen Maschine in einem ersten Datenspeicher des ersten Clusters und Speichern einer Replikation der ersten Daten in einem zweiten Datenspeicher des zweiten Clusters und Speichern von zweiten Daten von der zweiten virtuellen Maschine im zweiten Datenspeicher sowie Speichern einer Replikation der zweiten Daten in dem ersten Datenspeicher ein. Das Verfahren umfasst auch die Identifizierung von Fehlern des ersten Clusters und als Reaktion auf den Ausfall, das Neustarten der ersten virtuellen Maschinen unter Verwendung der zweiten Server und der Replikation der ersten Daten im zweiten Datenspeicher.

[0062] In einigen Beispielen schließt das Speichern der ersten Daten die Umgehung eines Caches des ersten Datenspeichers und das direkte Schreiben auf ein Speichermedium des ersten Datenspeichers im normalen Betriebsmodus ein, um die Zeit, in der Daten im flüchtigen Speicher gehalten werden, zu reduzieren. In einigen Beispielen umfasst das Speichern der ersten Daten das Schreiben in einen Cache des ersten Datenspeichers während eines Konfigurationsbetriebsmodus. In einigen Beispielen reduziert der Neustart der virtuellen Maschinen unter Verwendung der zweiten Server Ausfallzeiten. In einigen Beispielen schließt das Speichern der ersten Daten von den ersten virtuellen Maschinen im ersten Datenspeicher und das Speichern der Replikation der ersten Daten im zweiten Datenspeicher das parallele Speichern der ersten Daten im ersten und zweiten Datenspeicher ein. In einigen Beispielen schließt das Speichern der ersten Daten von den ersten virtuellen Maschinen im ersten Datenspeicher und das Speichern der Replikation der ersten Daten im zweiten Datenspeicher das einheitliche Speichern der ersten Daten in den ersten und zweiten Datenspeichern unter Nutzung von Einheitlichkeit auf Dateiebene ein. Bei einigen Beispielen umfasst das Verfahren das Erzeugen eines Alarms, wenn die ersten virtuellen Maschinen unter Verwendung der zweiten Server neu gestartet werden.

[0063] In einigen Beispielen schließt das Verfahren auch das Durchführen einer im Wesentlichen live stattfindenden Migration vom Betrieb der ersten virtuellen Maschinen unter Verwendung der zweiten Server, zum Betrieb der ersten virtuellen Maschinen unter Verwendung der ersten Server ein, nachdem der Ausfall des ersten Clusters behoben wurde. Bei einigen Beispielen umfasst das Verfahren den Neustart der ersten virtuellen Maschinen unter Verwendung der zweiten Server und der Replikation der ersten Daten im zweiten Datenspeicher, umfasst einen automatischen Neustart der ersten virtuellen Maschinen unter Verwendung der zweiten Server und der Replikation der ersten Daten im zweiten Datenspeicher. In einigen Beispielen ist das erste Cluster zum zweiten Cluster redundant. Bei einigen Beispielen

umfasst das Verfahren das automatische Aktualisieren des zweiten Clusters.

[0064] Ein weiteres beispielhaftes Verfahren beinhaltet das Auswählen eines ersten Betriebsmodus eines ersten Clusters, der erste virtuelle Maschinen und erste Server aufweist, und das Betreiben des ersten Clusters im ersten Modus umfasst das Schreiben in einen Cache des ersten Datenspeichers, um die ersten Daten von den ersten virtuellen Maschinen zu speichern, und das Auswählen eines zweiten Modus, um den ersten Cluster zu betreiben. Das Verfahren umfasst auch das Betreiben des ersten Clusters im zweiten Modus, wobei der Cache umgangen und direkt auf eine Platte des ersten Datenspeichers geschrieben wird, um erste Daten der ersten virtuellen Maschinen zu speichern.

[0065] In einigen Beispielen schließt der erste Modus einen Konfigurationsmodus und der zweite Modus einen robusten Modus ein.

[0066] In einigen Beispielen schließt das Verfahren das Auswählen eines dritten Betriebsmodus eines zweiten Clusters ein, der zweite virtuelle Maschinen und zweiter Server aufweist, wobei das Betreiben des dritten Clusters im dritten Betriebsmodus das Schreiben in einen Cache des zweiten Datenspeichers umfasst, um die zweiten Daten von den zweiten virtuellen Maschinen zu speichern. Das Verfahren umfasst das Auswählen eines vierten Modus, um den zweiten Cluster zu betreiben, und den zweiten Cluster im vierten Modus zu betreiben, wobei der Cache umgangen und direkt auf eine Platte des zweiten Datenspeichers geschrieben wird, um zweite Daten der zweiten virtuellen Maschinen zu speichern.

[0067] Bei einigen Beispielen umfasst der erste Cluster im zweiten Modus auch das Speichern einer Replikation der ersten Daten im zweiten Datenspeicher des zweiten Clusters, wobei das Betreiben des zweiten Clusters das Speichern einer Replikation der zweiten Daten im ersten Datenspeicher umfasst. Bei einigen Beispielen umfasst das Verfahren auch die Identifizierung von Fehlern des ersten Clusters und das Neustarten der ersten virtuellen Maschinen unter Verwendung der zweiten Server und der Replikation der ersten Daten im zweiten Datenspeicher. Bei einigen Beispielen umfasst das Verfahren das Erzeugen eines Alarms, wenn die ersten virtuellen Maschinen unter Verwendung der zweiten Server neu gestartet werden. In einigen Beispielen schließt das Verfahren auch das Durchführen einer im Wesentlichen live stattfindenden Migration vom Betrieb der ersten virtuellen Maschinen unter Verwendung der zweiten Server zum Betrieb der ersten virtuellen Maschinen unter Verwendung der ersten Server ein, nachdem der Ausfall des ersten Clusters behoben wurde.

[0068] In einigen Beispielen umfasst das Speichern der ersten Daten von den ersten virtuellen Maschinen im ersten Datenspeicher und das Speichern der Replikation der ersten Daten im zweiten Datenspeicher das parallele Speichern der ersten Daten im ersten und zweiten Datenspeicher. In einigen Beispielen schließt das Speichern der ersten Daten von den ersten virtuellen Maschinen im ersten Datenspeicher und das Speichern der Replikation der ersten Daten im zweiten Datenspeicher das einheitliche Speichern der ersten Daten im ersten und zweiten Datenspeicher unter Nutzung von Einheitlichkeit auf Dateiebene ein.

Patentansprüche

1. Verfahren, Folgendes umfassend:

Betreiben eines ersten Clusters, der erste virtuelle Maschinen und erste Server umfasst;
Betreiben eines zweiten Clusters, der zweite virtuelle Maschinen und zweite Server umfasst;
Speichern erster Daten der ersten virtuellen Maschinen in einem ersten Datenspeicher des ersten Clusters und Speichern einer Replikation der ersten Daten in einem zweiten Datenspeicher des zweiten Clusters;
Speichern zweiter Daten der zweiten virtuellen Maschinen im zweiten Datenspeicher und Speichern einer Replikation der zweiten Daten im ersten Datenspeicher; Identifizieren eines Ausfalls des ersten Clusters; und
als Reaktion auf den Ausfall, Neustarten der ersten virtuellen Maschinen unter Verwendung der zweiten Server und der Replikation der ersten Daten im zweiten Datenspeicher, wobei das Speichern der ersten Daten ein Umgehen eines Caches des ersten Datenspeichers und direktes Schreiben auf ein Speichermedium des ersten Datenspeichers in einem normalen Betriebsmodus umfasst, um die Zeit, in der Daten in flüchtigem Speicher gehalten werden, zu reduzieren.

2. Verfahren, Folgendes umfassend:

Betreiben eines ersten Clusters, der erste virtuelle Maschinen und erste Server umfasst;
Betreiben eines zweiten Clusters, der zweite virtuelle Maschinen und zweite Server umfasst;
Speichern erster Daten der ersten virtuellen Maschinen in einem ersten Datenspeicher des ersten Clusters und Speichern einer Replikation der ersten Daten in einem zweiten Datenspeicher des zweiten Clusters;
Speichern zweiter Daten der zweiten virtuellen Maschinen im zweiten Datenspeicher und Speichern einer Replikation der zweiten Daten im ersten Datenspeicher;
Identifizieren eines Ausfalls des ersten Clusters; und
als Reaktion auf den Ausfall, Neustarten der ersten virtuellen Maschinen unter Verwendung der zweiten Server und der Replikation der ersten Daten im

zweiten Datenspeicher, wobei das Speichern der ersten Daten ein Schreiben in einen Cache des ersten Datenspeichers während eines Konfigurationsbetriebsmodus umfasst.

3. Verfahren nach einem der Ansprüche 1 bis 2, wobei das Speichern der ersten Daten von den ersten virtuellen Maschinen im ersten Datenspeicher und das Speichern der Replikation der ersten Daten im zweiten Datenspeicher ein paralleles Speichern der ersten Daten im ersten und zweiten Datenspeicher umfasst, und/oder unter Nutzung von Einheitlichkeit auf Dateiebene.

4. Verfahren nach einem der Ansprüche 1 bis 3, ferner umfassend Erzeugen eines Alarms, wenn die ersten virtuellen Maschinen unter Verwendung der zweiten Server neu gestartet werden.

5. Verfahren nach einem der Ansprüche 1 bis 4, ferner umfassend Durchführen einer im Wesentlichen live stattfindenden Migration vom Betrieb der ersten virtuellen Maschinen unter Verwendung der zweiten Server zum Betrieb ersten virtuellen Maschinen unter Verwendung der ersten Server, nachdem der Ausfall des ersten Clusters behoben wurde.

6. Verfahren nach einem der Ansprüche 1 bis 5, wobei der erste Cluster zu dem zweiten Cluster redundant ist.

7. Verfahren nach einem der Ansprüche 1 bis 6, ferner umfassend automatisches Aktualisieren des zweiten Clusters.

8. Verfahren, Folgendes umfassend:
Auswählen eines ersten Modus zum Betreiben eines ersten Clusters, der erste virtuelle Maschinen und erste Server aufweist;
Betreiben des ersten Clusters im ersten Modus, umfassend Schreiben in einen Cache eines ersten Datenspeichers, um die ersten Daten der ersten virtuellen Maschinen zu speichern;
Auswählen eines zweiten Modus zum Betreiben des ersten Clusters; und
Betreiben des ersten Clusters im zweiten Modus, umfassend Umgehen des Caches und direktes Schreiben auf eine Platte des ersten Datenspeichers, um die ersten Daten der ersten virtuellen Maschinen zu speichern;
Auswählen eines dritten Modus zum Betreiben eines zweiten Clusters, der zweite virtuelle Maschinen und zweite Server umfasst;
Betreiben des zweiten Clusters im dritten Modus, umfassend Schreiben in einen Cache eines zweiten Datenspeichers, um die zweiten Daten der zweiten virtuellen Maschinen zu speichern;
Auswählen eines vierten Modus zum Betreiben der zweiten Cluster; und

Betreiben des zweiten Clusters im vierten Modus, umfassend ein Umgehen des Caches und direktes Schreiben auf eine Platte des zweiten Datenspeichers, um die zweiten Daten der zweiten virtuellen Maschinen zu speichern.

9. Verfahren nach Anspruch 8, wobei der erste Modus ferner einen Konfigurationsmodus und der zweite Modus einen robusten Modus umfasst.

10. Verfahren nach Anspruch 8, wobei das Betreiben des ersten Clusters im zweiten Modus ferner ein Speichern einer Replikation der ersten Daten im zweiten Datenspeicher des zweiten Clusters umfasst und das Betreiben des zweiten Clusters ein Speichern einer Replikation der zweiten Daten im ersten Datenspeicher umfasst.

11. Verfahren nach Anspruch 10, ferner umfassend Identifizieren eines Ausfalls des ersten Clusters und Neustarten der ersten virtuellen Maschinen unter Verwendung der zweiten Server und der Replikation der ersten Daten im zweiten Datenspeicher.

12. Verfahren nach Anspruch 11, ferner umfassend Erzeugen eines Alarms, wenn die ersten virtuellen Maschinen unter Verwendung der zweiten Server neu gestartet werden, und/oder Durchführen einer im Wesentlichen live stattfindenden Migration vom Betrieb der ersten virtuellen Maschinen unter Verwendung der zweiten Server zum Betrieb der ersten virtuellen Maschinen unter Verwendung der ersten Server, nachdem der Ausfall des ersten Clusters behoben wurde.

13. Verfahren nach einem der Ansprüche 10 bis 12, wobei das Speichern der ersten Daten von den ersten virtuellen Maschinen im ersten Datenspeicher und das Speichern der Replikation der ersten Daten im zweiten Datenspeicher ein paralleles und/oder unter Nutzung von Einheitlichkeit auf Dateiebene umfassendes Speichern der ersten Daten im ersten und zweiten Datenspeicher umfasst.

14. Ein computer-lesbares Speichermedium umfassend computer-lesbare Anweisungen, um die Schritte in den vorhergehenden Verfahrensansprüchen bei Ausführung durch einen Prozessor und/oder ein Gerät durchzuführen.

Es folgen 20 Seiten Zeichnungen

Anhängende Zeichnungen

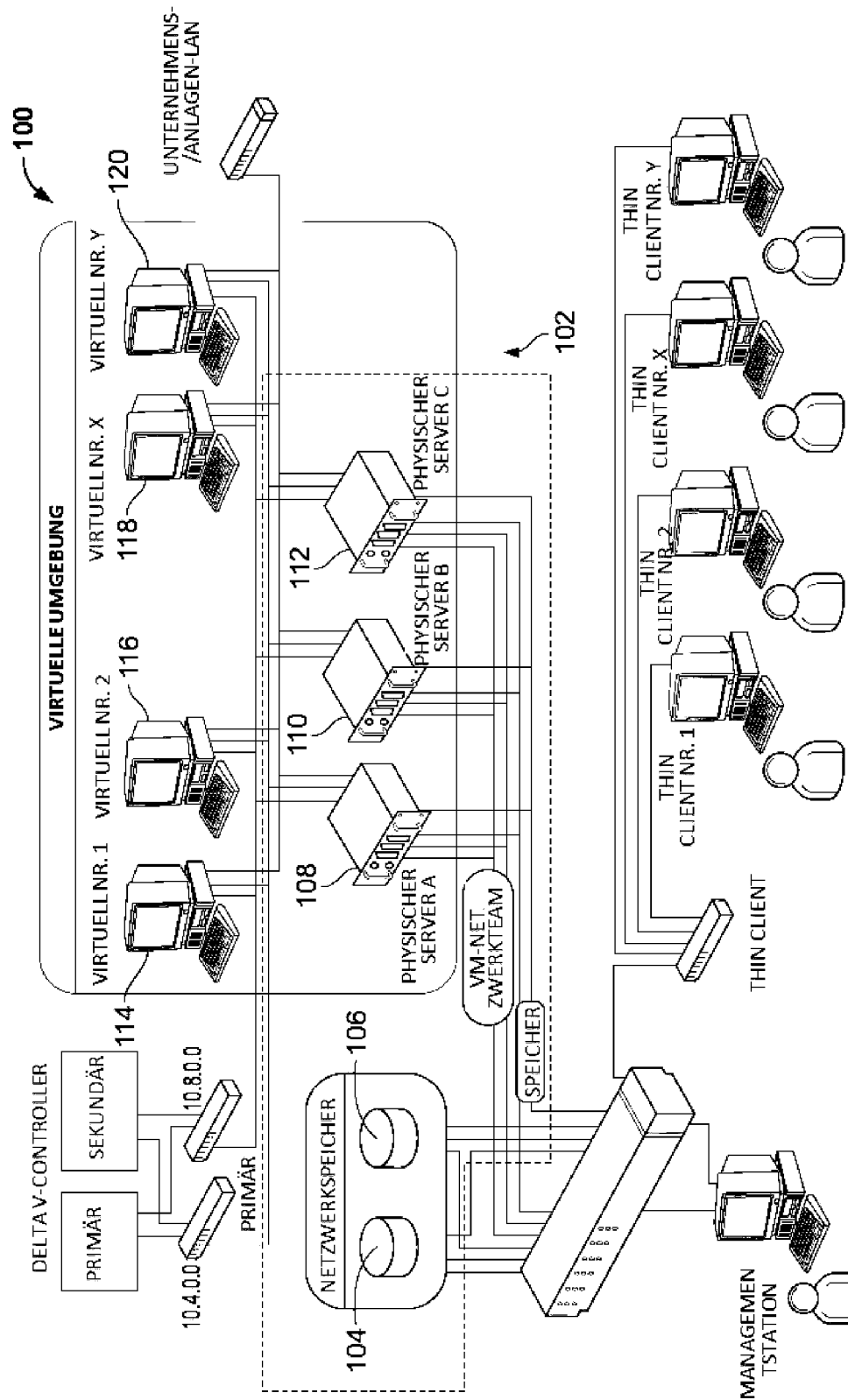


FIG. 1

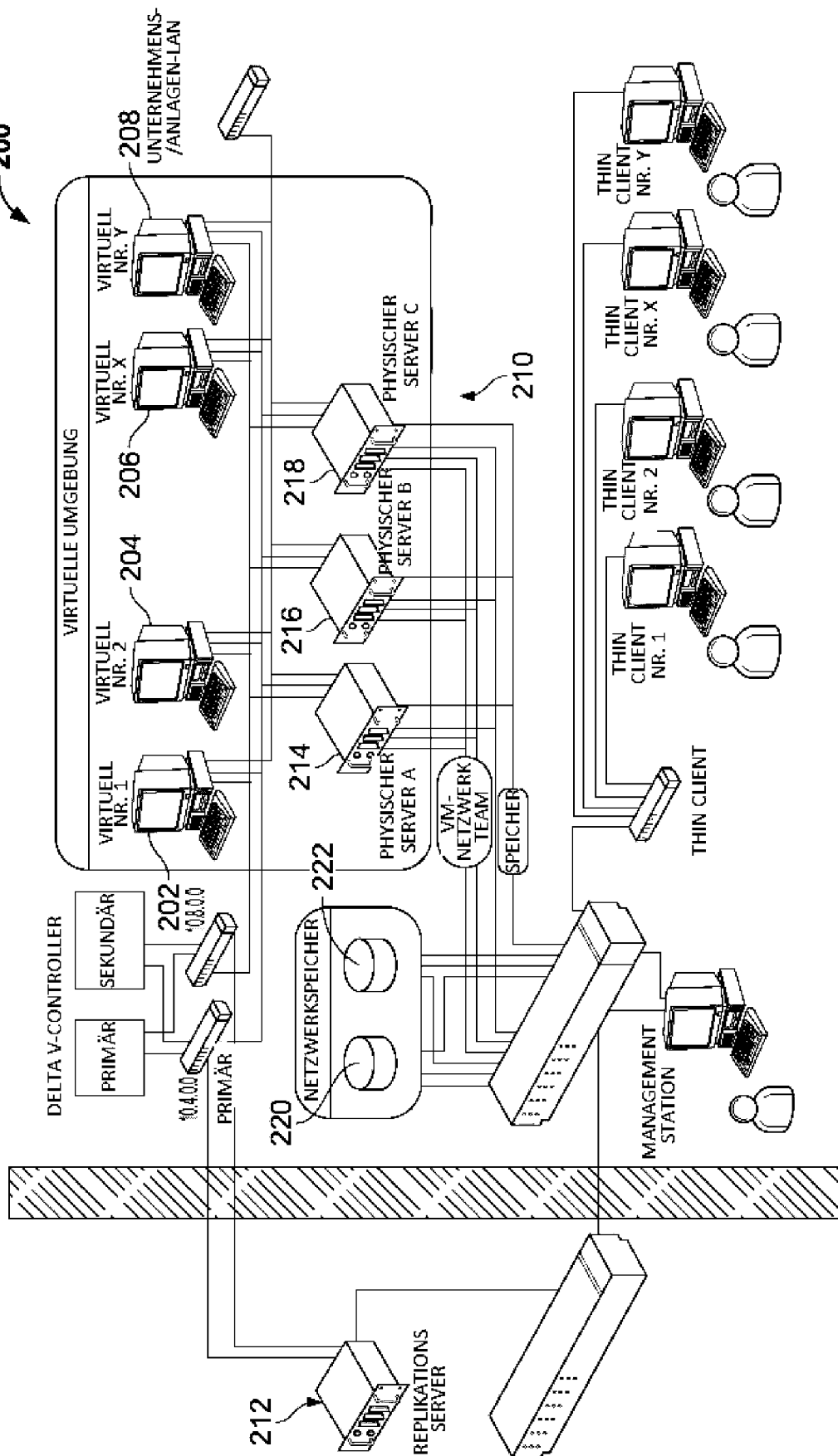


FIG. 2

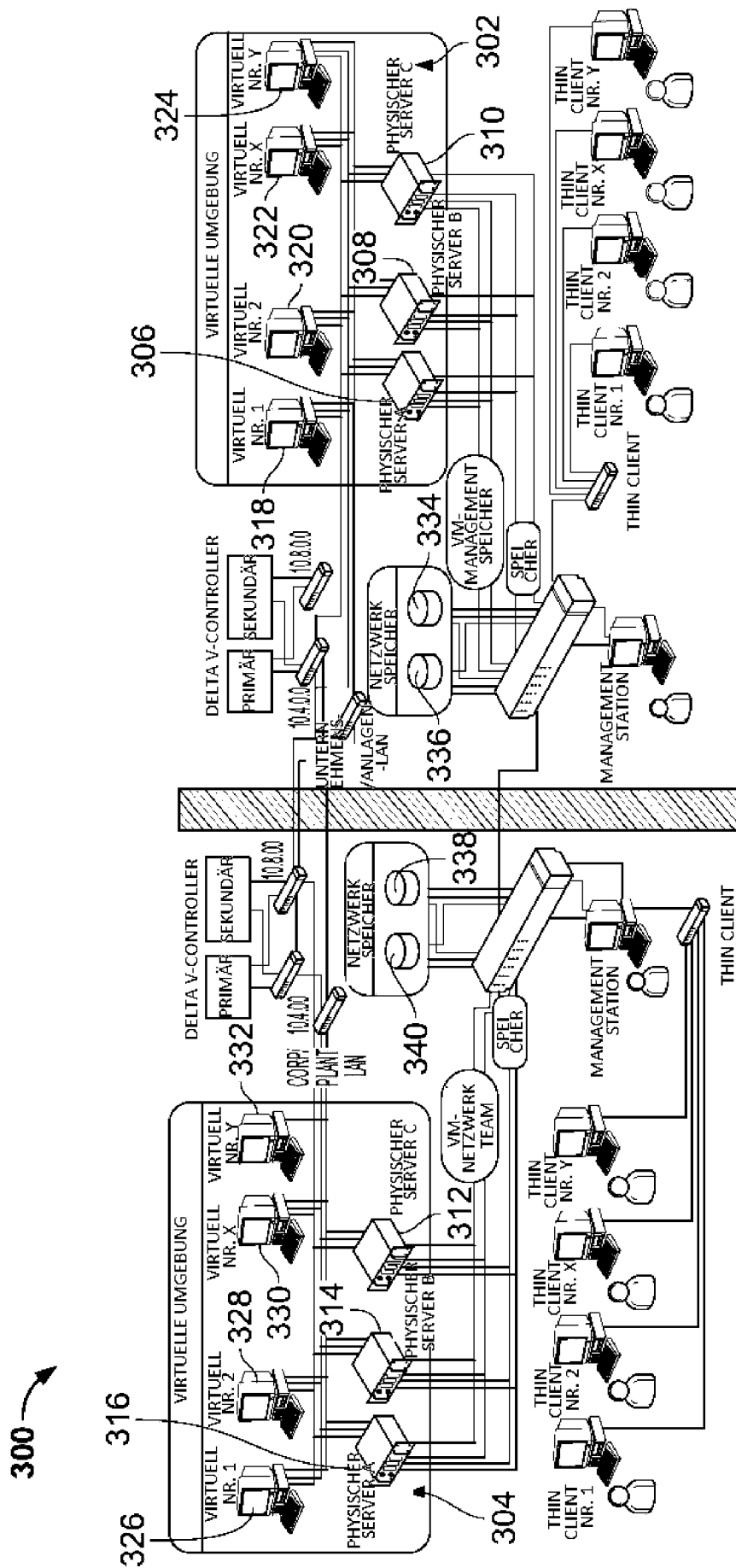

















FIG. 3

	MANAGEN ...	
	REPLIKATION	▶
	VERSCHIEBEN	▶
	LIVE-MIGRATION ABBRECHEN	▶
	STARTUP-PRIORITÄT ÄNDERN	▶
	DETAILANGABEN...	
	KRITISCHE EREIGNISSE ANZEIGEN	
	SPEICHER HINZUFÜGEN	▶
	RESSOURCE HINZUFÜGEN	▶
	WEITERE AKTIONEN	▶
	ENTFERNEN	
	EIGENSCHAFTEN	

	LIVE-MIGRATION	▶
	SCHNELLMIGRATION	▶
	SPEICHER VIRTUELLER MASCHINE	



	BESTMÖGLICHER KNOTEN
	KNOTEN WÄHLEN...

FIG. 4

500

FAILOVER-CLUSTER-MANAGER	
DATEI AKTION ANSICHT HILFE	
<div> FAILOVER CLUSTER-MANAGER </div>	
<div> CLUSTER2.RESEARCH.LOCAL </div> <div> FUNKTIONEN ▶ KNOTEN ▶ SPEICHER ▶ NETZWERKE CLUSTER-EREIGNISSE </div>	<div> CLUSTER1.RESEARCH.LOCAL </div> <div> FUNKTIONEN ▶ KNOTEN ▶ SPEICHER ▶ NETZWERKE CLUSTER-EREIGNISSE </div>
<div> CLUSTER1.RESEARCH.LOCAL </div> <div> ÜBERSICHT VON CLUSTER CLUSTER1 CLUSTER HAT 64 CLUSTER-FUNKTIONEN UND 4 KNOTEN. NAME: CLUSTER1.RESEARCH.LOCAL NETZWERKE: CLUSTER-NETZWERK 1, CLUSTER-NETZWERK 10,... SUBNETZE: 8 IPV4 UND 0 IPV6 AKTUELLER HOSTSERVER: C1-1 QUORUM-KONFIGURATION: MEHRHEIT DER KNOTEN UND FESTPLATTE LETZTE CLUSTER-EREIGNISSE: KEINER DER LETZTEN STUNDE ▲ KONFIGURIEREN </div> <div> HOHE VERFÜGBARKEIT FÜR EINE SPEZIFISCHE CLUSTER-FUNKTION KONFIGURIEREN, EINEN ODER MEHRERE SERVER (KNOTEN) HINZUFÜGEN, ODER SERVICES UND ANWENDUNGEN VON EINEM CLUSTER, DER WINDOWS SERVER 2012 AUSFÜHRT, AUF WINDOWS SERVER 2008 R2, ODER WINDOWS SERVER 2008 MIGRIEREN. <input type="checkbox"/> FUNKTION KONFIGURIEREN <input type="checkbox"/> FUNKTIONEN WERDEN FÜR HOHE VERFÜGBARKEIT KONFIGURIERT <input type="checkbox"/> CLUSTER VALIDIEREN <input type="checkbox"/> CLUSTER-VALIDIERUNGSTEST VERSTEHEN <input type="checkbox"/> KNOTEN HINZUFÜGEN <input type="checkbox"/> EINEN SERVER ZU IHREM CLUSTER HINZUFÜGEN <input type="checkbox"/> FUNKTIONEN MIGRIEREN <input type="checkbox"/> MIGRIEREN EINES CLUSTERS VON WINDOWS SERVER 2012, WINDOWS SERVER 2008 R2, ODER WINDOWS SERVER 2008. CLUSTER-BEWUSSTER UPDATE <input type="checkbox"/> SOFTWAREAKTUALISIERUNGEN AN KNOTEN IM CLUSTER ANWENDEN </div> <div> ▲ NAVIGIEREN <input type="checkbox"/> FUNKTIONEN <input type="checkbox"/> KNOTEN <input type="checkbox"/> SPEICHER <input type="checkbox"/> NETZWERKE <input type="checkbox"/> CLUSTER-EREIGNISSE ▲ CLUSTER-KERNRESSOURCEN NAME STATUS CLUSTER-NAME <input type="checkbox"/> NAME: CLUSTER1 <input type="checkbox"/> ONLINE (LÖSUNGSBENENNUNG NICHT...) SPEICHER <input type="checkbox"/> CLUSTER-FESTPLATTE2 <input type="checkbox"/> ONLINE </div>	
<div> AKTIONEN </div> <div> <input type="checkbox"/> CLUSTER1.RESEARCH.LOCAL ▲ <input type="checkbox"/> FUNKTION KONFIGURIEREN <input type="checkbox"/> CLUSTER VALIDIEREN <input type="checkbox"/> VALIDIERUNGSBEREICH ANZEIGEN <input type="checkbox"/> KNOTEN HINZUFÜGEN <input type="checkbox"/> VERBINDUNG BEENDEN <input type="checkbox"/> KURZLICHE EREIGNISSE ZURÜCKSETZEN <input type="checkbox"/> WEITERE AKTIONEN ANSICHT <input type="checkbox"/> AUFRISCHEN <input type="checkbox"/> EIGENSCHAFTEN <input type="checkbox"/> HILFE NAME: CLUSTER1 <input type="checkbox"/> ONLINE BRINGEN <input type="checkbox"/> OFFLINE NEHMEN <input type="checkbox"/> KRITISCHE EREIGNISSE ANZEIGEN <input type="checkbox"/> WEITERE AKTIONEN <input type="checkbox"/> EIGENSCHAFTEN <input type="checkbox"/> HILFE </div>	

FIG. 5



700

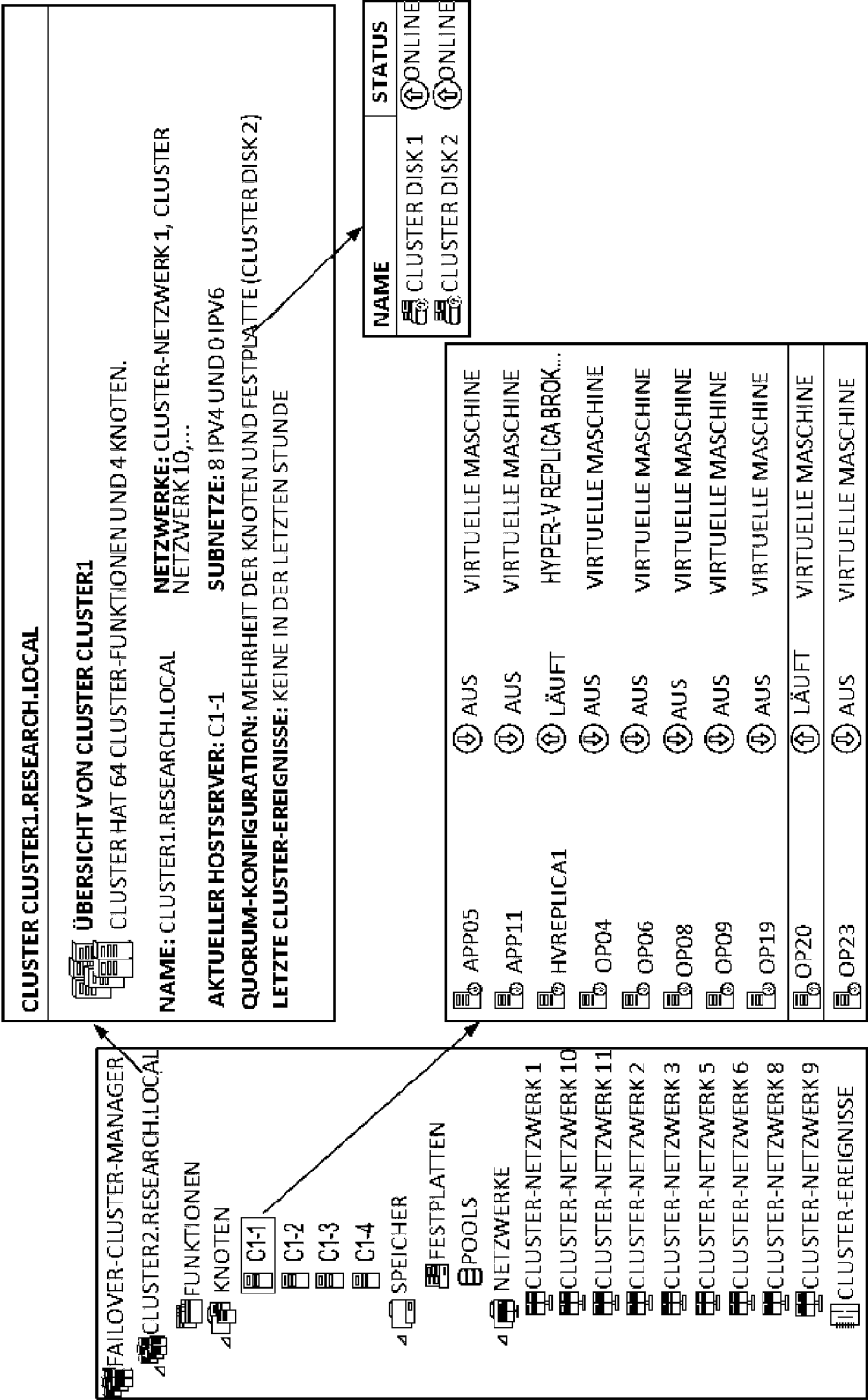


FIG. 7

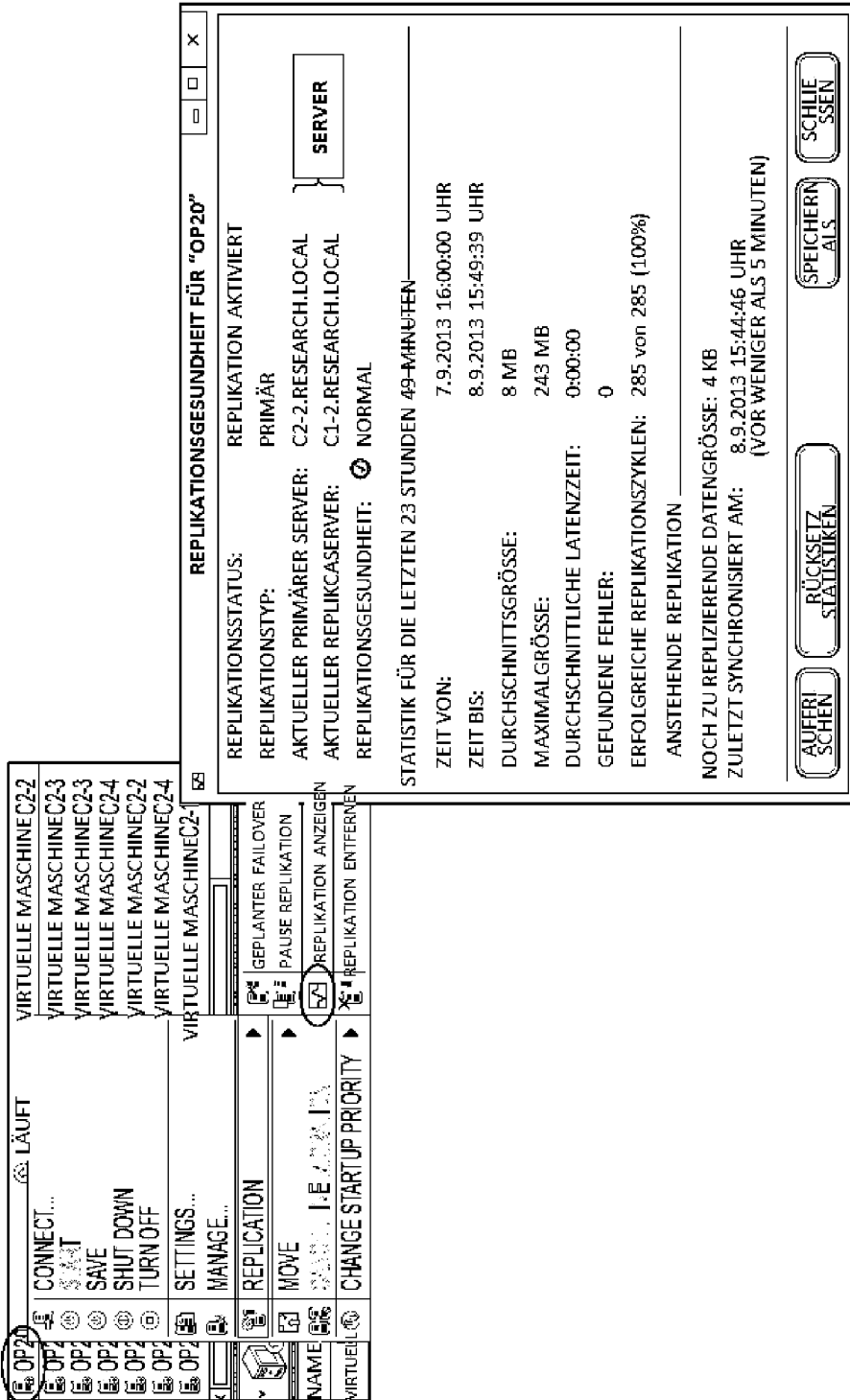



FIG. 8



REPLIKATION	►	GEPLANTER FAILOVER ...	
MO		DATEI-DEKLARATION	x
CH		GEPLANTER FAILOVER	
<p>AUF FAILOVER KLIKEN, UM DEN GEPLANTEN FAILOVER-PROZESS FÜR OP20 ZU STARTEN. ALLE ÄNDERUNGEN AUF DER PRIMÄREN VIRTUELLEN MASCHINE, DIE NICHT SCHON REPLIZIERT WURDEN, WERDEN REPLIZIERT UND DIE VIRTUELLE REPLIKATIONSMASCHINE WIRD STARTBEREIT SEIN.</p> <p><input checked="" type="checkbox"/> FAILOVER STARTEN:</p>			
<p>PRÜFUNG DER VORAUSSETZUNGEN</p> <p>PRÜFEN, DASS DIE VIRTUELLE MASCHINE AUSGESCHALTET IST.</p> <p>KONFIGURATION ÜBER MÖGLICHKEIT DER RÜCKREPLIKATION PRÜFEN.</p> <p>AKTIONEN</p> <p>DATEN, DIE NOCH NICHT REPLIZIERT WURDEN, AN REPLIKATIONSSERVER SENDEN.</p> <p>FAILOVER AN REPLIKATIONSSERVER.</p> <p>REPLIKATIONSRICHTUNG UMGKEHREN.</p> <p>VIRTUELLE REPLIKATIONSMASCHINE STARTEN.</p>			
<p>GEPLANTER FAILOVER</p> <p>AUF FAILOVER KLIKEN, UM DEN GEPLANTEN FAILOVER-PROZESS FÜR OP20 ZU STARTEN. ALLE ÄNDERUNGEN AUF DER PRIMÄREN VIRTUELLEN MASCHINE, DIE NICHT SCHON REPLIZIERT WURDEN, WERDEN REPLIZIERT UND DIE VIRTUELLE REPLIKATIONSMASCHINE WIRD STARTBEREIT SEIN.</p> <p><input checked="" type="checkbox"/> VIRTUELLE REPLIKATIONSMASCHINE NACH FAILOVER STARTEN:</p> <p>PRÜFUNG DER VORAUSSETZUNGEN</p> <p>PRÜFEN, DASS DIE VIRTUELLE MASCHINE AUSGESCHALTET IST. ERFOLGREICH</p> <p>KONFIGURATION ÜBER MÖGLICHKEIT DER RÜCKREPLIKATION PRÜFEN. ERFOLGREICH</p> <p>AKTIONEN</p> <p>DATEN, DIE NOCH NICHT REPLIZIERT WURDEN, AN REPLIKATIONSSERVER SENDEN. ERFOLGREICH</p> <p>FAILOVER AN REPLIKATIONSSERVER. ERFOLGREICH</p> <p>REPLIKATIONSRICHTUNG UMGKEHREN. ERFOLGREICH</p> <p>VIRTUELLE REPLIKATIONSMASCHINE STARTEN. ERFOLGREICH</p> <p>GEPLANTER FAILOVER x</p> <p> FAILOVER ERFOLGREICH DURCHFÜHRT:</p> <p>VIRTUELLE MASCHINE AUF C1-2.RESEARCH.LOCAL WURDE ERFOLGREICH GESTARTET.</p> <p>SCHLIESSEN</p>			

1000

FAILOVER-CLUSTER-MANAGER

CLUSTER2.RESEARCH.LOCAL

FUNKTIONEN

KNOTEN

SPEICHER

NETZWERKE

CLUSTER-EREIGNISSE

CLUSTER1.RESEARCH.LOCAL

FUNKTIONEN

KNOTEN

C1-1

C1-2

C1-3

C1-4

SPEICHER

NETZWERKE

CLUSTER-EREIGNISSE

FUNKTIONEN AUF C1-2 (21)

SUCHE

SUCHANFRAGE

NAME

STATUS

TYP

PRIORITÄT

INFORMATION

APP02	AUS	VIRTUELLE MASCHINE	MITTEL	
APP06	LÄUFT	VIRTUELLE MASCHINE	MITTEL	
APP07	LÄUFT			
APP08	LÄUFT			
APP09	LÄUFT			
APP10	LÄUFT			
OP07	AUS			
OP11	LÄUFT			
OP13	AUS			
OP15	AUS			
OP17	AUS			
OP19	AUS			
OP20	LÄUFT			
OP24	AUS			
OP28	AUS			
OP29	AUS			

REPLIKATIONSGESUNDHEIT FÜR „OP20“

REPLIKATIONSSTATUS: REPLIKATION AKTIVIERT

REPLIKATIONSTYP: PRIMÄR

AKTUELLER PRIMÄRER SERVER: C1-2.RESEARCH.LOCAL

AKTUELLER REPLIKATIONSSERVER: HVREPLICA2.RESEARCH.LOCAL

REPLIKATIONSGESUNDHEIT: NORMAL

STATISTIKEN FÜR DIE LETZTEN 2 MINUTEN

ZEIT AB: 8.9.2013 15:52:30 UHR

ZEIT BIS: 8.9.2013 15:54:59 UHR

DURCHSCHNITTSGRÖSSE: 0 KB

MAXIMALGRÖSSE: 0 KB

DURCHSCHNITTLICHE LATENZZEIT: 0:00:00

GEFUNDENE FEHLER: 0

ERFOLGREICHE REPLIKATIONSZYKLEN: 0

ANSTEHENDE REPLIKATION

NOCH ZU REPLIZIERENDE DATENGROSSE: 24 MB

ZULETZT SYNCHRONISIERT AM: 8.9.2013 15:52:27 UHR

(VOR WENIGER ALS 5 MINUTEN)

AUFRISCHEN

STATISTIKEN ZURÜCKSETZEN

SPEICHERN ALS

SCHLIESSEN

FIG. 10

23/33

1200 ↗






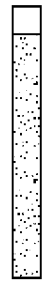










10 TOP-HOSTS NACH KÜRZLICH VERWENDETEM SPEICHER				BEARBEITEN	HILFE
HOST	MACHINENTYP	VERWENDETER SPEICHER			
 C1-2	HYPER-V	16015 MB	98%		
 C1-3	HYPER-V	15700 MB	96%		
 C1-1	HYPER-V	14713 MB	90%		
 C2-1	HYPER-V	14644 MB	90%		
 C2-3	HYPER-V	14457 MB	88%		
 C1-4	HYPER-V	13201 MB	81%		
 C2-4	HYPER-V	13125 MB	80%		
 C2-2	HYPER-V	9665 MB	59%		

FIG. 12

1300

HOME

NETZWERK

VIRTUALISIERUNG

VIRTUALISIERUNG ÜBERBLICK VMWARE HYPER-V

HYPER-V ÜBERBLICK

VIRTUALISIERUNGS-ASSETS

VIRTUALISIERUNGS-EINSTELLUNGEN

BEARBEITEN

HILFE

10 TOP-HOSTS NACH PROZENT DES VERWENDETEN SPEICHERS

HOST	MASCHINENTYP	VERWENDETER SPEICHER
C1-2	HYPER-V	16015 MB <div>98%</div>
C1-3	HYPER-V	15700 MB <div>96%</div>
C1-1	HYPER-V	14713 MB <div>90%</div>
C2-1	HYPER-V	14644 MB <div>90%</div>
C2-3	HYPER-V	14457 MB <div>88%</div>
C1-4	HYPER-V	13201 MB <div>81%</div>
C2-4	HYPER-V	13125 MB <div>80%</div>
C2-2	HYPER-V	9665 MB <div>59%</div>

VIRTUALISIERUNGS-ANLAGEN ÜBERBLICK

BEARBEITEN

HILFE

INSGESAMT

ANZAHL HOSTS 8

ANZAHL VMS 49 LAUFEN, 133 INSGESAMT

GESAMTANYAL PHYSISCHE CPU-KERNE 48

RAM INSGESAMT 127,6 GB

LETZTE ABFRAGE VOR 3 MINUTEN

HYPER-V

ANZAHL HOSTS 8

ANZAHL VMS 49 LAUFEN, 133

GESAMTANYAL PHYSISCHE CPU-KERNE 48

RAM INSGESAMT 127,6 GB

LETZTE ABFRAGE VOR 3 MINUTEN

10 TOP-HOSTS NACH NETZWERK VERWENDUNG

BEARBEITEN

HILFE

HOST MASCHINENTYP EMPFANG ÜBERMITTLUNG INSGESAMTVERWENDUNG

C1-1	HYPER-V	1,88 MBPS	9,71 MBPS	11,59 MBPS	<div>0%</div>
C1-2	HYPER-V	5,40 MBPS	2,15 MBPS	7,55 MBPS	<div>0%</div>
C1-4	HYPER-V	1,57 MBPS	0,14 MBPS	1,71 MBPS	<div>0%</div>
C2-3	HYPER-V	1,21 MBPS	0,16 MBPS	1,37 MBPS	<div>0%</div>
C2-2	HYPER-V	1,07 MBPS	0,12 MBPS	1,19 MBPS	<div>0%</div>
C2-4	HYPER-V	0,95 MBPS	0,13 MBPS	1,07 MBPS	<div>0%</div>
C1-3	HYPER-V	0,93 MBPS	0,13 MBPS	1,06 MBPS	<div>0%</div>
C2-1	HYPER-V	0,90 MBPS	0,13 MBPS	1,03 MBPS	<div>0%</div>

FIG. 13

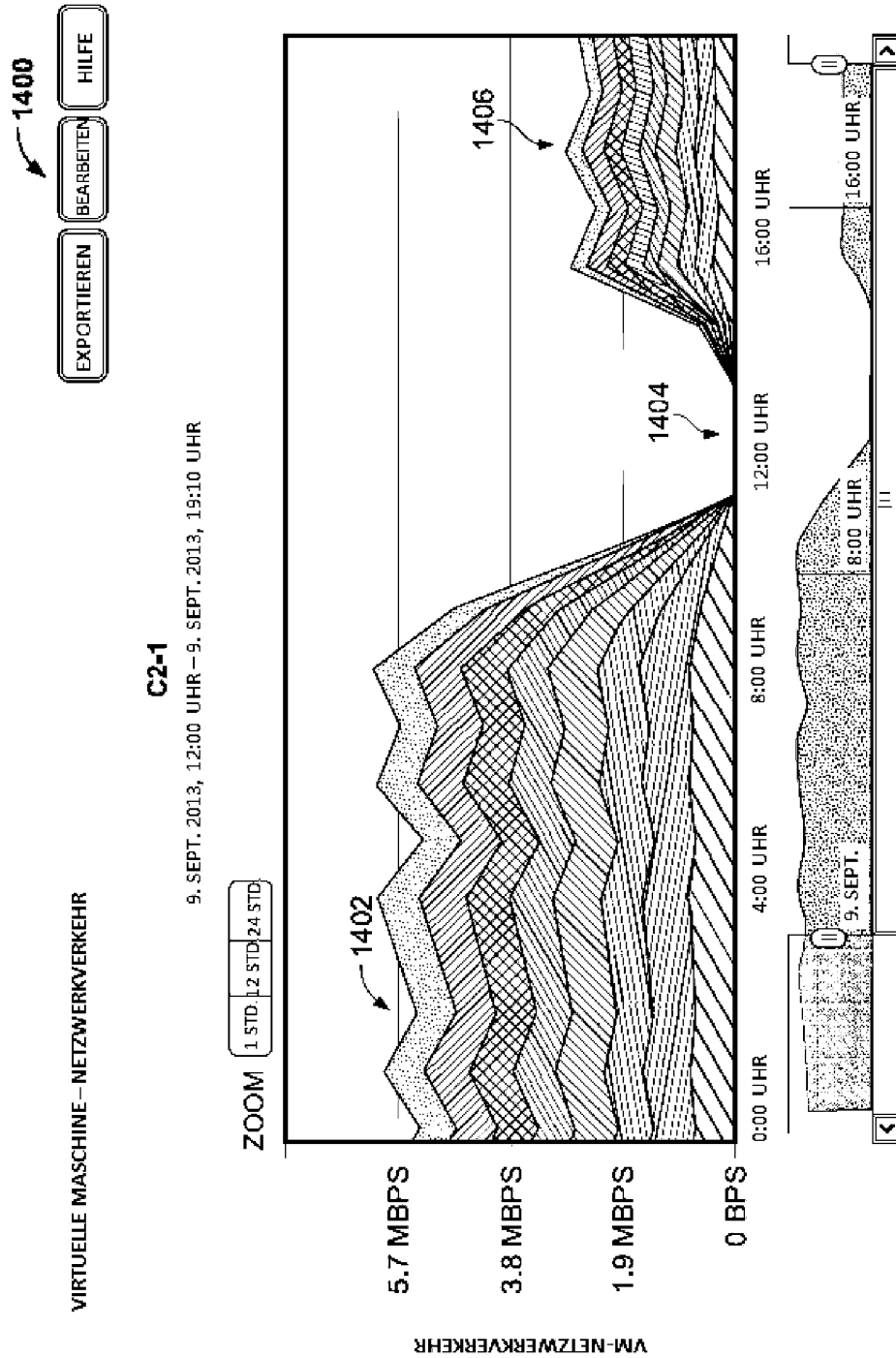


FIG. 14

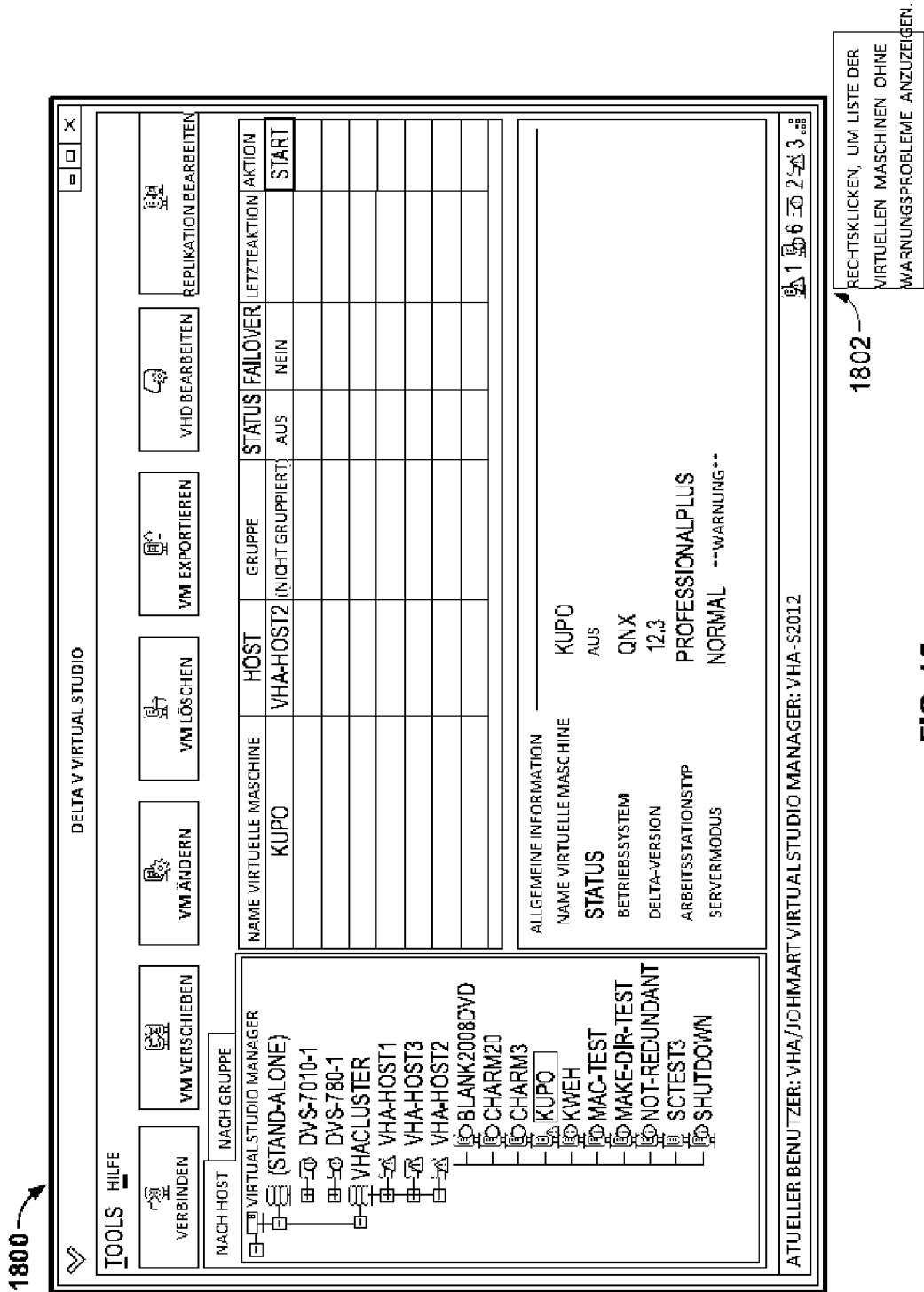


FIG. 15

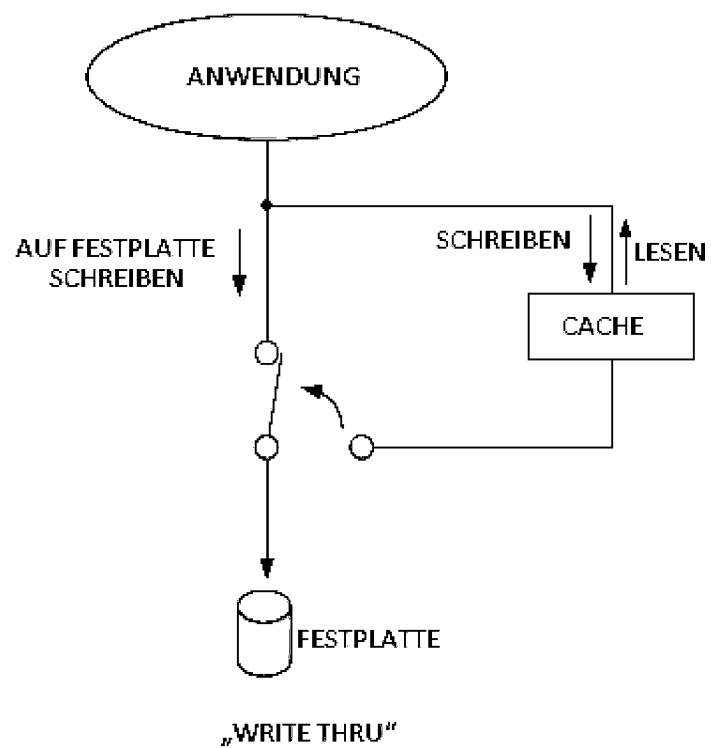


FIG. 16

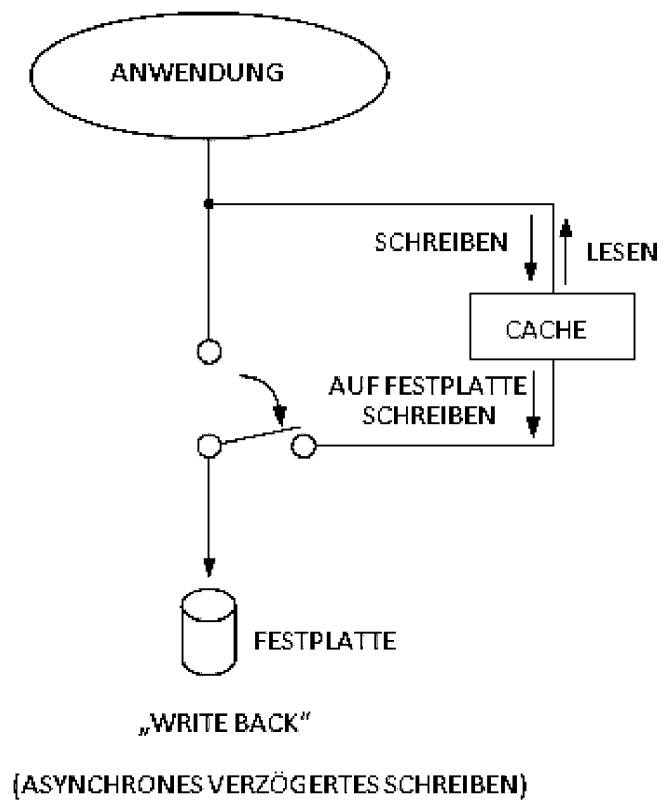


FIG. 17

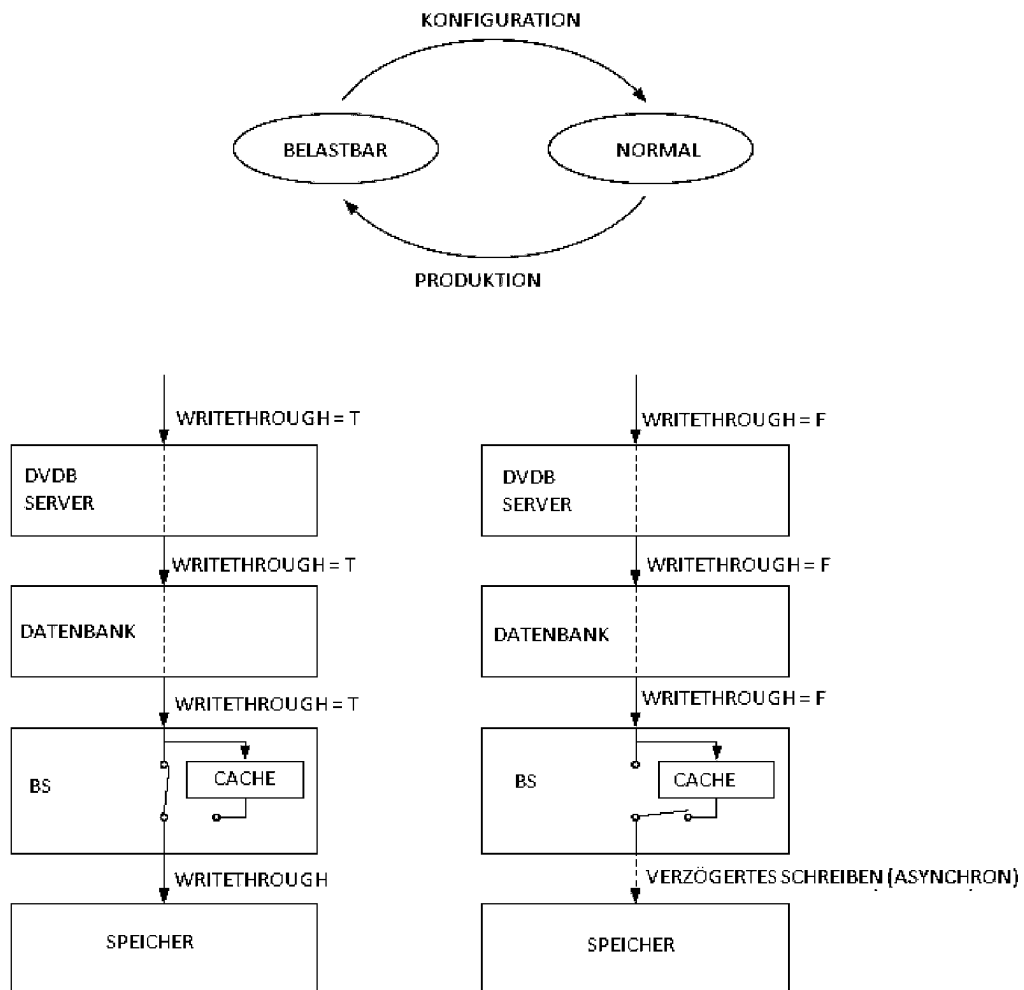


FIG. 18

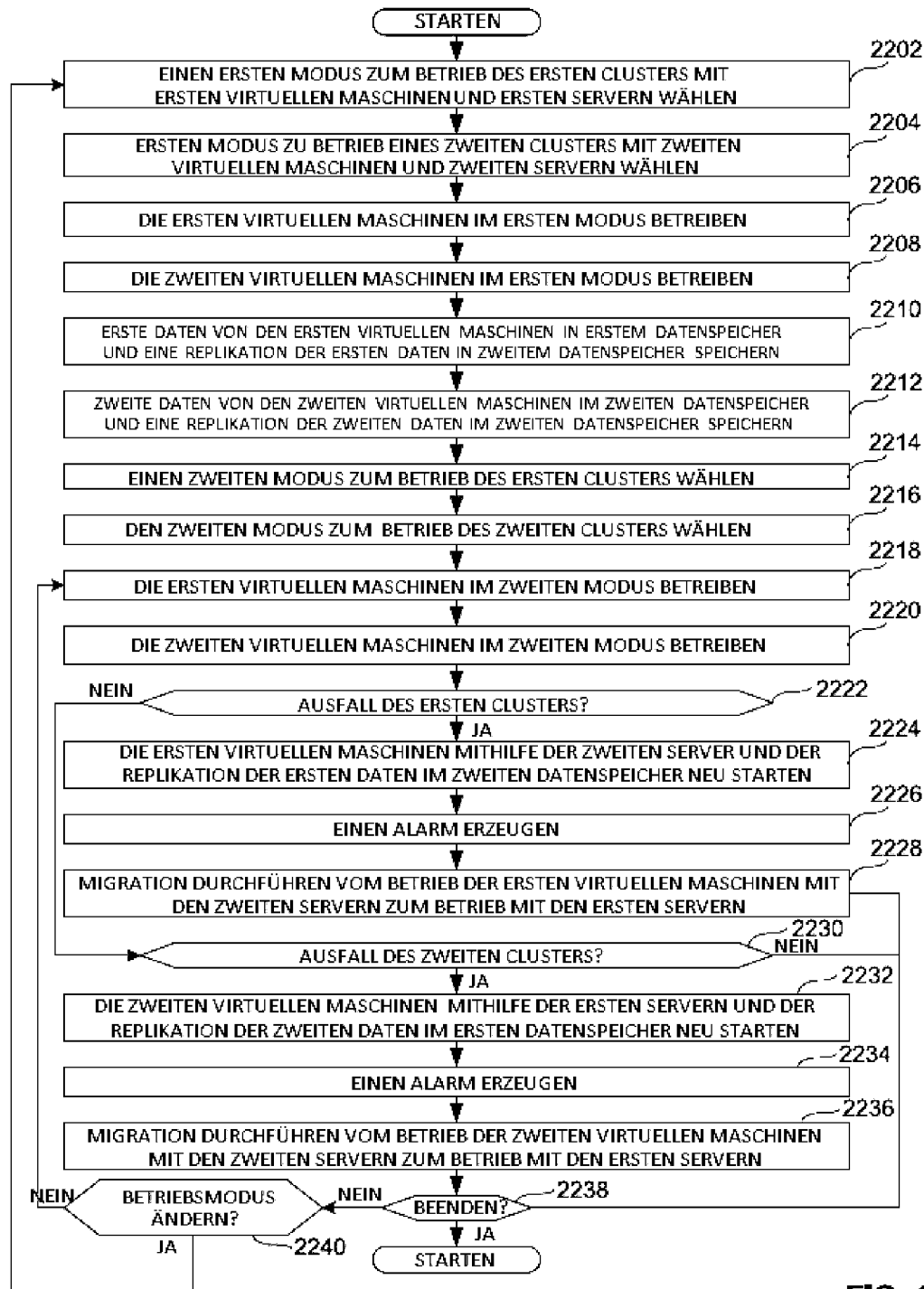


FIG. 19

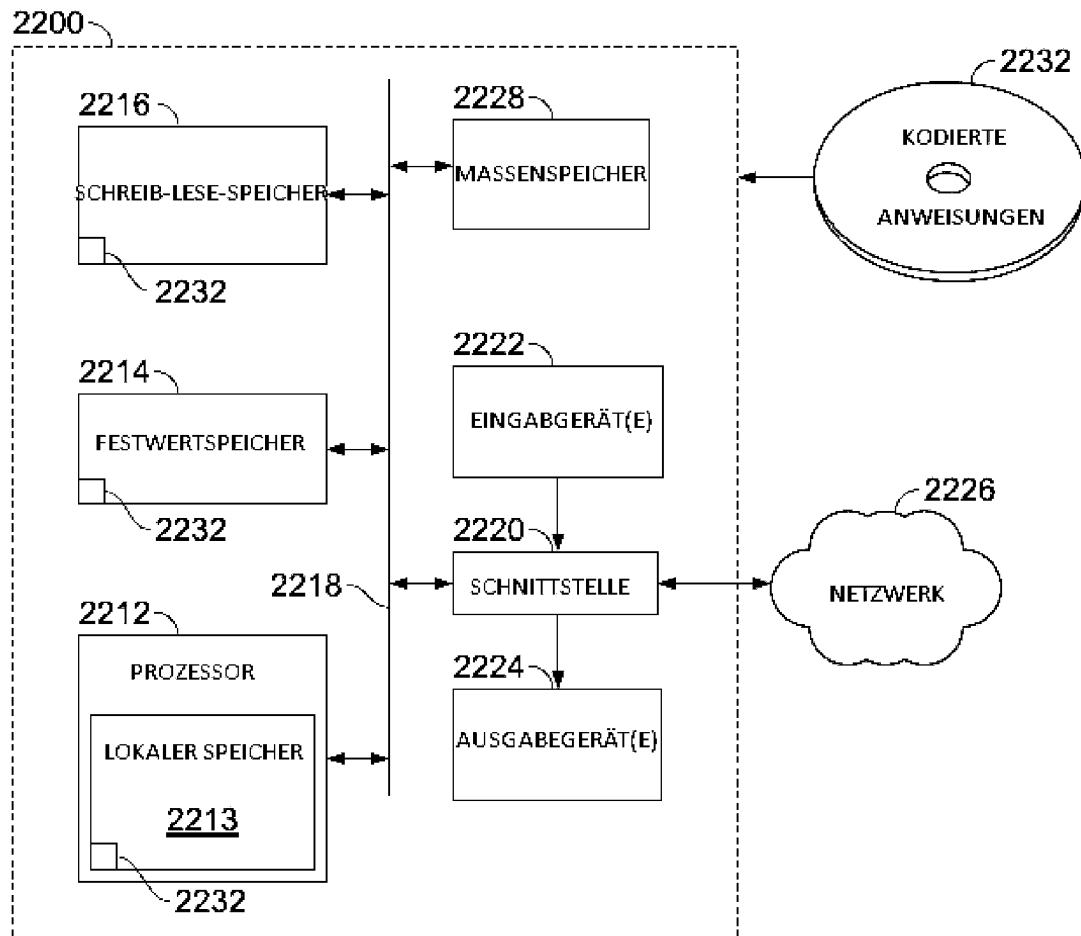


FIG. 20