(19) **Bundesrepublik Deutschland**
Deutsches Patent- und Markenamt

(10) **DE 699 07 824 T2** 2004.04.15

(12) # Übersetzung der europäischen Patentschrift

(97) **EP 0 981 089 B1**
(21) Deutsches Aktenzeichen: **699 07 824.5**
(96) Europäisches Aktenzeichen: **99 305 518.5**
(96) Europäischer Anmeldetag: **12.07.1999**
(97) Erstveröffentlichung durch das EPA: **23.02.2000**
(97) Veröffentlichungstag der Patenterteilung beim EPA: **14.05.2003**
(47) Veröffentlichungstag im Patentblatt: **15.04.2004**

(51) Int Cl.[7]: **G06F 11/14**

(30) Unionspriorität:
119140 20.07.1998 US

(84) Benannte Vertragsstaaten:
DE, ES, FR, GB, IT, NL

(73) Patentinhaber:
Lucent Technologies Inc., Murray Hill, N.J., US;
Academia Sinica, Taipeh/T'ai-pei, TW

(74) Vertreter:
derzeit kein Vertreter bestellt

(72) Erfinder:
Chung, Pi-Yu, Bellevue, Washington 98006, US;
Huang, Yennun, Bridgewater, New Jersey 08807, US; Liang, Deron, Taiwan 11529, TW; Shih, Chia-Yen, Murray Hill, New Jersey 07974, TW; Yajnik, Shalini, Scotch Plains, New Jersey 07076, TW

(54) Bezeichnung: **Verfahren und Vorrichtung zur Fehlererkennung und Wiederherstellung mit vorbestimmtem Replikationsgrad für verteilte Anwendungen in einem Netzwerk**

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel II § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

Beschreibung

TECHNISCHES GEBIET

[0001] Diese Erfindung betrifft die Erkennung eines Ausfalls eines Anwendungsmoduls, das auf einem Hostcomputer in einem Netzwerk läuft, sowie die Wiederherstellung nach diesem Ausfall.

HINTERGRUND DER ERFINDUNG

[0002] Damit ein Anwendungsmodul, das auf einem Hostcomputer in einem Netzwerk läuft, den darauf zugreifenden Clients akzeptable Leistung zur Verfügung stellen kann, muss das Anwendungsmodul sowohl zuverlässig als auch verfügbar sein. Um akzeptable Leistung zur Verfügung zu stellen, sind Schemata zur Erkennung eines Ausfalls eines Anwendungsmoduls oder des gesamten Hostcomputers, auf dem es läuft, und zur anschließenden raschen Wiederherstellung nach einem derartigen erkannten Ausfall erforderlich. Die Replikation des Anwendungsmoduls auf anderen Hostcomputern in dem Netzwerk ist eine wohl bekannte Technik, die zur Verbesserung von Zuverlässigkeit und Verfügbarkeit des Anwendungsmoduls verwendet werden kann.

[0003] Im Stand der Technik sind drei Strategien zum Betreiben und Konfigurieren des Ersatzschaltprozesses in Anwendung auf die Replikas oder Backup-Kopien eines Anwendungsmoduls bekannt, die einen Vorbereitungsgrad für diese Backups definieren. In der ersten Strategie, die als "kaltes Backup"-Format bekannt ist, läuft nur die primäre Kopie eines Anwendungsmoduls auf einem Hostcomputer, und andere Backup-Kopien bleiben auf anderen Hostcomputern in dem Netzwerk im Leerlauf. Wenn ein Ausfall der primären Kopie des Anwendungsmoduls erkannt wird, wird die primäre Kopie des Anwendungsmoduls entweder auf demselben Hostcomputer erneut gestartet, oder eine der Backup-Kopien des Anwendungsmoduls wird auf einem der anderen Hostcomputer gestartet, wobei die Backup-Kopie dann die neue primäre Kopie wird. Durch Verwendung einer Fixpunktfestlegungstechnik, die periodisch "Aufnahmen" des Laufzustands des primären Anwendungsmoduls macht und diesen Zustand auf einem stabilen Speichermedium speichert, werden die Fixpunktdaten des letzten derartigen gespeicherten Zustands des ausgefallenen primären Anwendungsmoduls dem Backup-Anwendungsmodul zur Verfügung gestellt, damit es die Arbeit als primäres Anwendungsmodul übernehmen und die Verarbeitung ausgehend von diesem zuletzt gespeicherten Zustand des ausgefallenen primären Anwendungsmoduls fortsetzen kann, wenn ein Ausfall des primären Anwendungsmoduls erkannt worden ist.

[0004] Die zweite Strategie ist als "warmes Backup"-Format bekannt. Im Unterschied zu dem kalten Backup-Format, bei dem kein Backup eines Anwendungsmoduls zur gleichen Zeit wie das primäre Anwendungsmodul läuft, laufen beim warmen Backup-Format eine oder mehrere Backup-Anwendungsmodule gleichzeitig mit dem primären Anwendungsmodul. Die Backup-Anwendungsmodule erhalten jedoch keine Anfragen irgendeines Clients und reagieren nicht auf diese, sondern erhalten periodisch Zustandsaktualisierungen von dem primären Anwendungsmodul. Nachdem ein Ausfall des primären Anwendungsmoduls erkannt worden ist, wird rasch eines der Backup-Anwendungsmodule aktiviert, um die Verantwortung des primären Anwendungsmoduls zu übernehmen, ohne dass Initialisierung oder Neustart erforderlich ist, welche die Zeit verlängern, die das Backup benötigt, um die Verarbeitungsfunktionen der ausgefallenen primären Kopie zu übernehmen.

[0005] Die dritte Strategie ist als ein "heißes Backup"-Format bekannt. Gemäß diesem Format sind zwei oder mehr Kopien eines Anwendungsmoduls zu Laufzeit aktiv. Jede laufende Kopie kann Anfragen von Clients verarbeiten, und die Zustände werden unter den mehreren Kopien synchronisiert. Nachdem ein Ausfall in einem der laufenden Anwendungsmodule erkannt worden ist, ist jede der anderen laufenden Kopien in der Lage, die Last der ausgefallenen Kopie sofort zu übernehmen und den Betrieb fortzusetzen.

[0006] Im Unterschied zu der kalten Backup-Strategie, bei der zu jeder gegebenen Zeit nur eine primäre Kopie läuft, können sowohl die warme Backup- als auch die heiße Backup-Strategie vorteilhaft den gleichzeitigen Ausfall von mehr als einer Kopie eines speziellen Anwendungsmoduls tolerieren, das im Netzwerk läuft, da auf dem Netzwerk mehrere Kopien dieses Anwendungsmodultyps simultan laufen.

[0007] Jede der drei Replikationsstrategien bringt unterschiedlichen Verwaltungsaufwand zur Laufzeit mit sich, und sie haben unterschiedliche Wiederherstellungszeiten. Ein auf einem Netzwerk laufendes Anwendungsmodul benötigt möglicherweise auf Grundlage seiner Verfügbarkeitsanforderungen und seiner Laufzeitumgebung eine andere Replikationsstrategie als ein anderes Anwendungsmodul, das auf dem gleichen Hostcomputer oder einem anderen Hostcomputer in dem Netzwerk läuft. Da verteilte Anwendungen oft auf heterogenen Hardware- und Betriebssystemplattformen laufen, müssen die Techniken zur Verbesserung der Zuverlässigkeit und Verfügbarkeit eines Anwendungsmoduls in der Lage sein, sich allen möglichen Replikationsschemata anzupassen.

[0008] In US-A-5,748,882 sind eine Vorrichtung und ein Verfahren für fehlertolerante Datenverarbeitung offenbart. In diesem Patent ist beschrieben, dass eine Anwendung oder ein Prozess in einem "Wächter"-Dämonprozess registriert wird, der dann die Anwendung oder den Prozess auf Ausfall oder Aufhängen "überwacht". Falls Ausfall oder Aufhängen der überwachten Anwendung erkannt wird, startet der Wächter die Anwendung oder den Prozess neu. In ei-

nem verteilten System mit mehreren Hosts in einem Netzwerk überwacht ein Wächterdämon auf einem Hostcomputer registrierte Anwendungen oder Prozesse auf seinem eigenen Hostcomputer sowie Anwendungen oder Prozesse auf einem anderen Hostcomputer. Falls ein überwachter Hostcomputer ausfällt, startet der Wächterdämon, der den ausgefallenen Hostcomputer überwacht, die registrierten Prozesse oder Anwendungen, die auf dem ausgefallenen überwachten Knoten gelaufen sind, auf seinem eigenen Knoten neu. In Ausführungsformen mit einem einzigen Knoten sowie solchen mit mehreren Knoten ist die Replikationsstrategie zum Neustart des ausgefallenen Prozesses oder der ausgefallenen Anwendung das kalte Backup-Format, d. h. ein neuer Replikaprozess oder eine neue Replikaanwendung wird nur nach Ausfall des primären Prozesses oder der primären Anwendung gestartet.

[0009] Fehlertolerante Methoden des Standes der Technik konnten leider nicht mehrere unterschiedliche Replikationsstrategien handhaben, wie die oben beschriebenen kalten, warmen und heißen Backup-Formate, die am besten jeder individuellen Anwendung von einer Vielzahl unterschiedlicher Anwendungen, die auf einem oder mehreren Rechnern in einem Netzwerk laufen können, zugeordnet werden können, und lassen sich dafür nicht anpassen. Außerdem gibt es im Stand der Technik keine Methode, um für die warmen und heißen Backup-Replikationsformate eine konstante Anzahl laufender Anwendungen in dem Netzwerk aufrechtzuerhalten.

[0010] In US-A-5,408,649 wird eine einzige Datenbankcomputereinrichtung auf Bereitschaftsbasis (Standby) in einem System mit verteiltem Datenzugriff zur Verfügung gestellt, das eine Vielzahl derartiger Datenbankcomputereinrichtungen einschließt. Die Standby-Computereinrichtung übernimmt den Betrieb von jeder der Vielzahl von Computereinrichtungen, bei der ein Ausfall erkannt wird. Eine willkürliche Computereinrichtung, die an alle Datenbankcomputereinrichtungen gekoppelt ist, ist in der Lage, einen Ausfall von jeder der Datenbankcomputereinrichtungen zu erkennen und die Reservedatenbankcomputereinrichtung anstelle der ausgefallenen als Ersatz zu nehmen.

[0011] In einem Artikel von Silvano Maffeis mit dem Titel "Piranha: A CORBA Tool For High Availability", IEEE Computer, April 1997, Seiten 59 bis 66, wird ein Tool auf CORBA-Basis (CORBA steht für Common Object Request Broker Architecture) für ein verteiltes System offenbart, das als Netzwerk-Monitor wirkt, der Ausfälle durch eine graphische Benutzeroberfläche übermittelt und als Manager wirkt, um solche Funktionen wie automatischen Neustart ausgefallener CORBA-Objekte und Durchsetzen festgelegter Replikationsgrade auf Gruppen von Objekten durchzuführen.

[0012] In einem Artikel von Rudi Cuyvers et al. mit dem Titel "A Kernel for Multi-Level Fault-Tolerant Multiprocessing", IEEE Proceedings of the Southeastcon, 1991, Seiten 248 bis 252, wird ein Laufzeit-Kernel offenbart, der die Auswahl der Anzahl von kalten, warmen, heißen und aktiven Backups für jeden Task von unterschiedlichen parallelen Programmen ermöglicht. Mehrere Prozesse mit unterschiedlichen Hardware-Fehlertoleranzniveaus können gleichzeitig auf demselben Multiprozessor laufen, was zu kostengünstiger fehlertoleranter Mehrprogrammverarbeitung führt.

KURZFASSUNG DER ERFINDUNG

[0013] Die erfindungsgemäße Vorrichtung und ein erfindungsgemäßes Verfahren sind in den unabhängigen Ansprüchen beschrieben. Bevorzugte Formen sind in den abhängigen Ansprüchen beschrieben.

[0014] Erfindungsgemäß wird ein auf einem Hostcomputer laufendes Anwendungsmodul zuverlässig gemacht, indem es zuerst seine eigenen Ausfall- und Wiederherstellungsprozesse registrieren lässt. Ein Replikamanagerdämonprozess, der auf demselben Hostcomputer läuft, auf dem das Anwendungsmodul läuft, oder auf einem anderen Hostcomputer läuft, der mit dem Netzwerk verbunden ist, mit dem der Rechner des Anwendungsmoduls verbunden ist, empfängt eine Registrierungsnachricht von dem Anwendungsmodul. Diese Registrierungsnachricht schließt zusätzlich zu der Identifizierung des sich registrierenden Anwendungsmoduls und des Hostrechners, auf dem es läuft, die spezielle Replikationsstrategie (kaltes, warmes oder heißes Backupformat) und den Replikationsgrad ein, der dem registrierten Anwendungsmodul zugeordnet werden soll, wobei die registrierte Replikationsstrategie von dem Replikamanager verwendet wird, um den Betriebszustand von jeder Backupkopie des Anwendungsmoduls zu setzen sowie die Anzahl der Backup-Kopien gemäß dem Replikationsgrad zu halten. Ein Wächterdämonprozess, der auf demselben Hostcomputer wie das registrierte Anwendungsmodul läuft, überwacht dann periodisch das registrierte Anwendungsmodul, um Ausfälle zu erkennen. Wenn der Wächterdämon einen Absturz oder ein Aufhängen des überwachten Anwendungsmoduls erkennt, teilt er den Ausfall dem Replikamanager mit, der wiederum einen Ersatzschaltprozess bewirkt. Wenn somit das Replikationsformat warm oder heiß ist und das ausgefallene Anwendungsmodul nicht auf seinem eigenen Hostcomputer neu gestartet werden kann, wird eine der laufenden Backup-Kopien des primären Anwendungsmoduls als das neue primäre Anwendungsmodul ernannt, und einem Hostcomputer, auf dem sich eine Leerlaufkopie des Anwendungsmoduls befindet, wird über das Netzwerk signalisiert, diese Leerlaufanwendung auszuführen. Der Replikationsgrad bleibt somit erhalten, wodurch Schutz gegen mehrfache Ausfälle dieses Anwendungsmoduls gewährleistet ist. Falls das Replikationsformat kalt ist und die ausgefallene Anwendung nicht auf ihrem eigenen Hostcomputer neu gestartet werden kann, dann wird einem Host-

computer, auf dem sich eine Leerlaufkopie des Anwendungsmoduls befindet, über das Netzwerk signalisiert, diese Leerlaufkopie auszuführen.

[0015] Um den Ausfall eines Hostcomputers oder des Wächterdämons zu erkennen, der auf einem Hostcomputer läuft, erhält ein Superwächterdämonprozess, der auf demselben Hostcomputer wie der Replikamanager läuft, Eingaben von jedem Hostcomputer. Nachdem ein Ausfall eines Hostcomputers durch den Superwächterdämon infolge des Fehlens einer Eingabe von diesem Hostcomputer erkannt worden ist, wird auf den Replikamanager zugegriffen, um die Anwendungsmodule zu ermitteln, die auf diesem Hostcomputer gelaufen sind. Diese Anwendungsmodule werden dann individuell in der in dem Replikamanager eingerichteten und gespeicherten Weise vor Ausfall geschützt.

KURZE BESCHREIBUNG DER ZEICHNUNG

[0016] **Fig.** 1 ist ein Blockdiagramm eines Computernetzwerks, das in veranschaulichender Weise eine Vielzahl von Hostcomputern zeigt, auf denen Anwendungsmodule laufen, die erfindungsgemäß vor Ausfall geschützt sind; und

[0017] **Fig.** 2 zeigt eine Tabelle, die in dem Replikamanagerdämon gespeichert ist und auf einem Hostcomputer in dem Netzwerk von **Fig.** 1 läuft, die für jeden Anwendungsmodultyp Informationen zuordnet, die zum erfindungsgemäßen Bewirken von Ausfallschutz verwendet werden.

DETAILLIERTE BESCHREIBUNG

[0018] In Bezug auf **Fig.** 1 ist ein Netzwerk **100** gezeigt, mit dem eine Vielzahl von Hostcomputern verbunden sind. Das Netzwerk **100** kann ein Ethernet, ein ATM-Netzwerk oder jeder andere Typ von Datennetzwerk sein. Lediglich zu veranschaulichenden Zwecken sind sechs Hostcomputer H1, H2, H3, H4, H5 und H6, mit Ziffern als 101, 102, 103, 104, 105 beziehungsweise 106 bezeichnet, mit dem Netzwerk **100** verbunden. Jeder Hostcomputer weist eine Vielzahl unterschiedlicher Anwendungsmodule auf, die sich in seinem Arbeitsspeicher befinden. Diese Anwendungsmodule, die in **Fig.** 1 als vom Typ A, B und C bezeichnet werden, weisen jeweils eine primäre Kopie auf, die auf mindestens einem dieser sechs Hostcomputer ausgeführt wird und läuft. Speziell läuft in diesem veranschaulichenden Beispiel eine primäre Kopie des Anwendungsmoduls vom Typ A, Anwendungsmodul A_1, auf Hostcomputer H1, eine primäre Kopie des Anwendungsmoduls vom Typ B, Anwendungsmodul B_1, läuft auf Hostcomputer H4, und eine primäre Kopie des Anwendungsmoduls vom Typ C, Anwendungsmodul C_1, läuft auf Hostcomputer H3. Andere Kopien von jedem Anwendungsmodultyp sind entweder, wie nachfolgend beschrieben wird, gespeichert und aus dem Arbeitsspeicher auf mindestens einem der anderen Hostcomputer in einem Leerlaufzustand verfügbar, der auf die spätere Ausführung wartet, oder laufen als Backup-Kopien oder zweite primäre Kopien von Anwendungsmodulen.

[0019] Ein Anwendungsmodul, das wie zuvor beschrieben auf einem Hostcomputer läuft, wird durch eine oder mehrere Backupkopien des Anwendungsmoduls vor Ausfall geschützt, die im Zustand der Vorbereitung betrieben werden, der durch eine von drei bekannten Replikationsformaten definiert ist. Jedes Replikationsformat hat sein eigenes Verfahren, einem Anwendungsmodul, das mittels Absturz oder Aufhängen ausfällt, oder allen jenen Anwendungsmodulen, die sich auf einem Hostcomputer befinden, der seinerseits ausfällt, Backup zur Verfügung zu stellen. Erfindungsgemäß wird jeder Anwendungsmodultyp durch das spezielle Replikationsformat (kaltes Backup, warmes Backup, heißes Backup) vor Ausfall geschützt, das für seine eigenen Verarbeitungsanwendungen am besten geeignet ist. Zudem wird erfindungsgemäß jeder Anwendungsmodultyp mit einem Replikationsgrad vor Ausfall geschützt, der für dieses Anwendungsmodul spezifiziert ist, wodurch eine konstante Anzahl von Kopien dieses Anwendungsmoduls zum Schutz gegen mehrere Ausfälle dieses Anwendungsmodultyps in einem laufenden Zustand gehalten wird.

[0020] Damit ein im Leerlauf befindliches oder Backup-Anwendungsmodul die Funktion eines ausgefallenen primären Anwendungsmoduls nach Erkennung eines Ausfalls mit einer minimalen Unterbrechung die Verarbeitung übernehmen kann, muss dem Backup- oder im Leerlauf befindlichen Anwendungsmodul nach seiner Ausführung vom Leerlauf zustand oder nach seiner Ernennung als neues primäres Anwendungsmodul der letzte Betriebszustand des ausgefallenen Anwendungsmoduls zur Verfügung gestellt werden. Ein Fixpunktserver **110**, der mit Netzwerk **110** verbunden ist, erhält periodisch von jedem vor Ausfall geschützten Anwendungsmodul, das auf dem Netzwerk läuft, den aktuellsten Zustand dieser Anwendung, wobei der Zustand dann in seinem Arbeitsspeicher gespeichert wird. Nach Erkennung eines Ausfalls eines Anwendungsmoduls wird der zuletzt gespeicherte Zustand dieses ausgefallenen Anwendungsmoduls von dem Speicher eines Fixpunktservers **110** abgerufen und dem neuen primären Anwendungsmodul für eine Fortsetzung der Verarbeitung zur Verfügung gestellt.

[0021] Erfindungsgemäß wird ein Anwendungsmodul zuverlässig gemacht, indem es selbst seine eigene Ausfallerkennung und Wiederherstellung registrieren lässt. Speziell erhält ein zentralisierter Replikamanagerdämonprozess **112**, der auf einem der Hostcomputer (Hostcomputer H2 in **Fig.** 1) in dem Netzwerk läuft, eine Registrierungsanforderung von jedem gegen Ausfall geschützten Anwendungsmodul. Die Registrierungsanforderung schließt für dieses spezielle Anwendungsmodul das Replikationsformat (d. h. heiß, warm und kalt), den Replikationsgrad,

eine Liste der Hostcomputer, auf denen sich das Anwendungsmodul befindet, und den Ort, an dem sich auf jedem derartigen Hostcomputer das ausführbare Programm befindet, und ein Umschaltformat ein. Der Replikationsgrad spezifiziert die Gesamtanzahl der Kopien eines Anwendungsmoduls. Bei einem heißen oder warmen Replikationsformat definiert der Replikationsgrad die Gesamtanzahl der laufenden Kopien eines Anwendungsmoduls, die in dem Netzwerk gehalten werden sollen. Bei einem kalten Replikationsformat spezifiziert der Replikationsgrad die Anzahl der Hostcomputer in dem Netzwerk, von denen aus das Anwendungsmodul ausgeführt werden kann. Das Umschaltformat spezifiziert eine Ersatzschaltstrategie, die festlegt, wann ein Anwendungsmodul von einem Hostcomputer auf einen anderen Hostcomputer migriert werden soll. In Bezug auf das letztere kann, wenn ein Ausfall eines Anwendungsmoduls erkannt worden ist, es entweder auf demselben Hostcomputer, auf dem der Ausfall stattgefunden hat, neu gestartet werden, oder es kann auf einen anderen Hostcomputer migriert werden, auf dem sich eine im Leerlauf befindliche oder laufende Backup-Kopie befindet. Zwei Ersatzschaltstrategien können bei Registrierung des Anwendungsmoduls mit dem Replikamanager spezifiziert werden. Bei der Ersten, die als OnOverThreshold (über dem Schwellenwert liegend) bekannt ist, wird ein Anwendungsmodul auf einen anderen Hostcomputer migriert, nachdem die Anzahl der Ausfälle des Anwendungsmoduls auf einem gegebenen Hostcomputer einen gegebenen Schwellenwert überschritten hat. Bei dieser Strategie wird das ausgefallene Anwendungsmodul daher auf seinem eigenen Hostcomputer neu gestartet, bis die Anzahl der Ausfälle dieses Anwendungsmoduls die Schwellenwertzahl erreicht hat. Danach wird das ausgefallene Anwendungsmodul auf einen anderen Hostcomputer migriert. Bei der zweiten Ersatzschaltstrategie, die als OnEachFailure (bei jedem Ausfall) bekannt ist, wird ein ausgefallenes Anwendungsmodul bei jedem Auftreten eines Ausfalls auf einen anderen Hostcomputer migriert.

[0022] Der Replikamanagerdämonprozess **112** hat in seinem Speicher die Replikationsinformationen für alle registrierten Anwendungsmodule in dem Netzwerk vereinigt. Für jeden im Netzwerk laufenden Anwendungsmodultyp speichert der Replikamanager die notwendigen Informationen, um die Wiederherstellung eines laufenden Anwendungsmoduls oder eines ganzen Hostcomputers zu bewirken, auf dem mehrere unterschiedliche Anwendungsmodule laufen. **Fig.** 2 veranschaulicht im Format einer Tabelle 200 den Typ der gespeicherten Information für die drei Typen von Anwendungsmodulen, die auf den sechs Hostcomputern in **Fig.** 1 laufen. Als Beispiel wird Anwendungsmodul vom Typ A in Eintrag **201** mit einem warmen Backupformat mit einem Replikationsgrad von drei registriert. Ein primäres Anwendungsmodul läuft somit immer zusammen mit zwei Backup-Kopien, wobei jede beliebige der Backup-Kopien in der Lage ist, die Funktion als primäre Kopie nach Ausfall der primären Kopie zu übernehmen. Wie aus den **Fig.** 1 und 2 hervorgeht, ist die primäre Kopie (die in Block **202** mit "P" bezeichnet ist), A_1, veranschaulichend als auf H1 laufend gezeigt, und die Backup-Kopien (die in Blöcken **203** und **204** mit "B" bezeichnet sind), A_2 und A_3, sind als auf H2 beziehungsweise H3 laufend gezeigt. Eine weitere Kopie von Anwendungsmodul vom Typ A, A_4, ist als im Speicher von H4 befindlich im Leerlaufzustand (in Block **205** mit "I" bezeichnet) gezeigt. Die Pfadbezeichnung von jeder Kopie des Anwendungsmoduls auf dem Hostcomputer ist veranschaulichend gezeigt. Anwendungsmodul vom Typ B ist in Eintrag **206** mit einem heißen Backupformat mit einem Grad von zwei registriert und gespeichert. Daher werden zwei primäre Kopien dieses Anwendungsmoduls aktiv und laufend gehalten, wobei jede Client-Anfragen verarbeitet und ihre Zustände synchronisiert werden. Die erste primäre Kopie, B_1, ist veranschaulichend als auf H4 befindlich gezeigt, und die zweite primäre Kopie, B_2, ist als auf H1 befindlich gezeigt. Eine Leerlaufkopie, B_3, befindet sich auf H5. Das dritte Anwendungsmodul, Typ C, ist in Eintrag **207** mit einem kalten Backupformat mit einem Grad von zwei registriert. Eine primäre Kopie, C_1, ist veranschaulichend als auf H3 laufend gezeigt, und eine einzige Leerlaufkopie ist veranschaulichend als auf H6 befindlich gezeigt.

[0023] Wie erörtert wird, wird nach Erkennung eines Ausfalls eines primären Anwendungsmoduls mit einem OnEachFailure-Umschaltformat oder einem OnOverThreshold-Umschaltformat, bei dem der Schwellenwert erreicht worden ist, in Tabelle 200 ein Backup-Anwendungsmodul als neues primäres Anwendungsmodul ernannt. Falls das ausgefallene Anwendungsmodul ein warmes oder heißes Backupformat hat, wird eine Leerlaufkopie dieses Anwendungsmodultyps auf seinem als Host dienenden Computer ausgeführt, um dasselbe Replikationsniveau in dem Netzwerk aufrechtzuerhalten. Falls in ähnlicher Weise ein Ausfall einer laufenden Backup-Kopie eines Anwendungsmoduls erkannt worden ist, wird eine Leerlaufkopie dieses Anwendungsmoduls auf einem anderen Hostcomputer gestartet, um dieselbe Anzahl laufender Kopien in dem Netzwerk aufrechtzuerhalten, wie durch den registrierten Replikationsgrad spezifiziert ist. Nach Erkennen eines Ausfalls auf einem Hostcomputer wird, wie weiterhin erörtert wird, auf Tabelle 200 zugegriffen, um die Identitäten der Anwendungsmodule zu ermitteln, die auf diesem Computer als entweder primäre Kopien oder Backup-Kopien laufen. Jede derartige primäre oder Backup-Kopie auf dem ausgefallenen Hostcomputer wird dann in der gleichen Weise vor Ausfall geschützt, als wenn jede individuell ausgefallen wäre.

[0024] Wiederun in Bezug auf **Fig.** 1 wird die Erkennung eines Ausfalls über einen Wächterdämonprozess bewirkt, der auf jedem Hostcomputer läuft. Jeder derartige Wächterdämonprozess führt die Funkti-

on der Überwachung dieses laufenden Anwendungsmoduls und aller anderen registrierten und laufenden Anwendungsmodule auf seinem Hostcomputer durch, nachdem ein Anwendungsmodul in dem Replikamanager **112** registriert worden ist. Demzufolge überwacht Wächterdämon **113-1** die registrierten Anwendungsmodule A_1 und B_2, die auf Hostcomputer H1 laufen; Wächterdämon **113-2** überwacht das registrierte Anwendungsmodul A_2, das auf Hostcomputer H2 läuft; Wächterdämon **113-3** überwacht die registrierten Anwendungsmodule A_3 und C_1, die auf Hostcomputer H3 laufen; und Wächterdämon **113-4** überwacht das Anwendungsmodul B_1, das auf Hostcomputer H4 läuft. Da Anwendungsmodul A_4 im Speicher von Hostcomputer H4 im Leerlauf ist, überwacht Wächterdämon **113-4** es erst dann, wenn es später aktiv gemacht wird. Leerlaufanwendungsmodul B3 auf Hostcomputer H5 und Leerlaufanwendungsmodul C_2 auf Hostcomputer H6 werden in ähnlicher Weise von Wächterdämonen **113-5** beziehungsweise 113-6 erst dann überwacht, wenn sie ausgeführt werden.

[0025] Die auf jedem Hostcomputer laufenden Wächterdämonen **113** unterstützen zwei Ausfallerkennungsmechanismen: Abrufbetrieb (Polling) und Herzschlag. Beim Abrufbetrieb schickt der Wächterdämon periodisch eine Ping-Nachricht an das Anwendungsmodul, das er überwacht. Wenn das Ping fehlschlägt, wird angenommen, dass das Anwendungsmodul abgestürzt ist. Der Abrufbetrieb kann auch verwendet werden, um die Unversehrtheit eines Anwendungsmoduls zu prüfen, indem eine Unversehrtheitsprüfungsmethode im Inneren des Anwendungsmoduls aufgerufen wird. Im Herzschlagmechanismus sendet ein Anwendungsmodul aktiv Herzschlagsignale auf entweder periodischer Basis oder auf-Anfrage-Basis an den Wächterdämon. Falls der Wächterdämon nicht innerhalb eines bestimmten Zeitintervalls ein Herzschlagsignal erhält, wird das Anwendungsmodul als aufgehängt angesehen. Der Herzschlagmechanismus ist in der Lage, Ausfälle durch Absturz sowie durch Aufhängen eines Anwendungsmoduls oder eines Hostcomputers zu erkennen, während der Abrufbetrieb nur Ausfall durch Absturz erkennen kann. Ein Anwendungsmodul kann auf Basis seiner Zuverlässigkeitsanforderungen einen dieser beiden Ansätze auswählen.

[0026] Wenn ein Wächterdämon einen Absturz oder ein Aufhängen eines Anwendungsmoduls erkennt, das er "überwacht", teilt er den Ausfall dem Replikamanager **112** mit, um eine Ersatzschaltaktion zu bewirken. Falls das ausgefallene Anwendungsmodul sich, wie bereits gesagt, mit einer OnEachFailure-Ersatzschaltstrategie registriert hat, wird das ausgefallene Anwendungsmodul auf einen anderen Host migriert. Falls das ausgefallene Anwendungsmodul eine primäre Kopie ist, wird eine der Backup-Anwendungsmodule als neue primäre Kopie ernannt und ein im Leerlauf befindliches Anwendungsmodul wird ausgeführt, um denselben Replikationsgrad aufrechtzuerhalten, der für den Anwendungsmodultyp eingetragen worden ist. Nach Beförderung eines Anwendungsmoduls vom Backup-Zustand zum primären Zustand wird seine Bezeichnung in Tabelle 200 geändert, ebenso wie bei der im Leerlauf befindlichen Anwendung, die ausgeführt wird. Falls das ausgefallene Anwendungsmodul eine Backup-Kopie ist, dann wird eine im Leerlauf befindliche Kopie ausgeführt, und seine Bezeichnung in Tabelle 200 wird geändert, um diese Änderung wiederzugeben.

[0027] Wie in **Fig.** 1 dargestellt ist, ist Replikamanager **112** zentralisiert, d. h. nur eine Kopie des Replikamanagers läuft in dem Netzwerk. Die Replikationsinformationen für das Anwendungsmodul, das in dem Netzwerk läuft, laufen in Tabelle 200 zusammen, die im Speicher von Replikamanager **112** gehalten wird. Um den Verlust dieser Informationen im Fall von Ausfällen zu vermeiden, setzt diese Replikamanagertabelle Fixpunkte in Fixpunktserver **110**.

[0028] Zusätzlich zu der Funktionalität der Wächterdämonen, die auf jedem Hostcomputer laufen, wird ein zentralisierter Superwächterdämonprozess **115-1** verwendet, um Absturz des Hosts zu erkennen und die Wiederherstellung durchzuführen. Alle Wächterdämonen lassen sich bei dem Superwächterdämon auf Erkennung von Hostabstürzen registrieren. Ausfallschutz wird durch eine Herzschlagerkennungsstrategie bewirkt. Jeder der Wächterdämonen **113** sendet periodisch ein Herzschlagsignal an den Superwächterdämon **115-1**. Falls der Superwächterdämon **115-1** kein Herzschlagsignal von irgendeinem der Wächter **113** erhält, geht er davon aus, dass der Wächter und der Hostcomputer, auf dem er läuft, ausgefallen sind. Dann leitet er die Wiederherstellung nach dem Ausfall ein, indem der Replikamanager **112** über den Ausfall dieses Hostcomputers informiert wird. Da ein zentralisierter Superwächterdämon selbst zu einem Einzelausfallpunkt werden kann, repliziert er sich selbst, und die Replikas werden in einem warmen Replikationsformat gehalten. In **Fig.** 1 sind Superwächter-Backup-Kopien **115-2** und **115-3** von Superwächter **115-1** als auf Hostcomputern H5 beziehungsweise H6 befindlich gezeigt. Die drei Superwächterdämonen bilden eine logische Ringstruktur. Jeder Superwächterdämon sendet periodisch Herzschlagssignale an einen benachbarten Superwächter. In **Fig.** 1 sendet der primäre Superwächter **115-1** somit periodisch ein Herzschlagsignal an Superwächter **115-2**, der wiederum periodisch ein Herzschlagsignal an Superwächter **115-3** sendet, der wiederum periodisch ein Herzschlagsignal zurück an Superwächter **115-1** sendet. Falls ein Superwächter kein Herzschlagsignal von seinem Nachbarn in dem Ring erhält, geht er davon aus, dass ein Ausfall stattgefunden hat. Eine Ersatzschaltprozedur für einen ausgefallenen Superwächter wird nachfolgend beschrieben.

[0029] Als Beispiel für die Wiederherstellung eines abgestürzten oder aufgehängten Anwendungsmoduls beziehen wir uns auf Anwendungsmodul A, das

von Replikamanager **112** mit einem warmen Replikationsformat mit einem Grad von drei und einem Umschaltformat von OnEachFailure registriert worden ist. Am Anfang läuft Anwendungsmodul A1 auf Hostcomputer H1, wobei Backups A_2 und A3 auf Hostcomputern H2 beziehungsweise H3 laufen. Anwendungsmodul A_1 wird bei seinem lokalen Wächter **113-1** mit dem Erkennungsformat Abrufbetrieb registriert, so dass Wächter **113-1** Anwendungsmodul A_1 periodisch abfragt. Zu irgendeiner Zeit stürzt Anwendungsmodul A_1 auf Hostcomputer H1 ab, wobei der Ausfall von Wächter **113-1** erkannt wird. Wächter **113-1** teilt diesen Ausfall Replikamanager **112** mit, der in seiner internen Tabelle 200 nachschlägt und feststellt, dass ein primäres Anwendungsmodul vom Typ A ausgefallen ist und Backup-Anwendungen auf Hostcomputern H2 und H3 laufen. Es stuft eines dieser Backups (beispielsweise A_2) hinauf auf den primären Zustand und ändert den Zustand von A_2 in Tabelle 200 von Backup auf primär. Er stellt dann fest, dass eine Leerlaufkopie, A_4, sich auf Hostcomputer H4 an der Pfadbezeichnung /home/chung/A.exe befindet, und startet dieses neue Backup, indem der Wächter **113-4** auf Hostcomputer H4 angewiesen wird, diese Kopie auszuführen. Somit laufen in dem Netzwerk nach Erkennen des Ausfalls von Anwendungsmodul A_1 auf Hostcomputer H1 und dessen Wiederherstellung insgesamt drei Kopien von Anwendungsmodul A weiter, wodurch die Anzahl der laufenden Anwendungsmodule in dem Netzwerk auf drei entsprechend dem registrierten Replikationsgrad gehalten wird. Die Ausfallerkennung und Wiederherstellung eines aufgehängten Anwendungsmoduls ist genau gleich, außer dass in diesem Fall Herzschlagsignale anstelle von Abrufbetrieb als Mittel zur Ausfallerkennung verwendet werden.

[0030] Der auf jedem Hostcomputer laufende Wächter sendet Herzschlagsignale an den primären Superwächter in dem Netzwerk. Wächter **113-1** bis **113-6** senden somit Herzschlagsignale an Superwächter **115-1**. Wenn ein Hostabsturz stattfindet, stürzt der darauf laufende Wächter ab, und Superwächter **115-1** erhält keine weiteren Herzschlagsignale von diesem Wächter. Falls beispielsweise Rost H1 abstürzt, erhält Superwächter **115-1** keine weiteren Herzschlagsignale von Wächter **113-1**. Dann erklärt er Hostcomputer H1 für tot und teilt diesen Ausfall Replikamanager **112** mit. Replikamanager **112** greift auf Tabelle 200 zu, um zu ermitteln, dass Anwendungsmodule A1 und B2 auf Hostcomputer H1 gelaufen sind. Die Wiederherstellung für A_1 wird wie zuvor beschrieben eingeleitet. Anwendungsmodul B_2 wird als primäre Kopie eingesetzt. Die Leerlaufkopie B_3, die sich auf Hostcomputer H5 befindet, wird dann ausgeführt, wodurch zwei laufende primäre Kopien von Anwendungsmodultyp B in dem Netzwerk aufrechterhalten werden. Der Zustand von B_3 wird dann in Tabelle 200 von Leerlauf auf primär aktualisiert. Der Ausfall eines Wächterdämons, der auf einem Hostcomputer läuft, wird in der gleichen Weise wie ein Hostabsturz behandelt.

[0031] Wenn der Hostcomputer, auf dem ein Superwächterdämon läuft, abstürzt, erhält der Superwächter auf dem nächsten Hostcomputer des logischen Rings keine Herzschlagsignale mehr. Falls somit Hostcomputer H6 ausfällt oder Superwächter **115-3** auf dem Hostcomputer abstürzt, erhält Superwächter **115-1** auf Hostcomputer H2 keine weiteren Herzschlagsignale von Superwächter **115- 3**. Er erklärt Superwächter **115-3** für tot und prüft, ob der tote Superwächter **115-3** ein primärer Superwächter war. Da Superwächter **115-3** ein Backup ist, muss er im Namen dieses Superwächters keine Aktionen durchführen. Der Superwächter **115-2** erhält dann eine Ausnahme (Exception), wenn er versucht, sein Herzschlagsignal an den Superwächter auf Hostcomputer H6 zu senden. Als Teil der Ausnahmebehandlung ermittelt Superwächter **115- 2** den Handle für Superwächter **115-1** auf Hostcomputer H1, registriert sich selbst darin und fängt an, Herzschlagsignale daran zu senden.

[0032] Falls Hostcomputer H2 ausfällt oder Superwächter **115-1** abstürzt, erkennt dann Superwächter **115-2** auf Hostcomputer H5 den Ausfall und ermittelt, dass der primäre Superwächter ausgefallen ist. Backup-Superwächter **115-2** übernimmt dann die Rolle des primären und startet den Replikamanagerdämon auf Hostcomputer H5. Die Wächter **113-1** bis **113-6** auf Hostcomputern H1 bis H6 erhalten jeweils Ausnahmen, wenn sie versuchen, Herzschlagsignale an den Superwächter **115-1** auf Hostcomputer H2 zu senden (der primär war). Als Teil der Ausnahmebehandlungsroutine ermittelt jeder Wächterdämon den neuen primären Superwächter **115-2**, und Replikamanager **112** registriert sich selbst bei dem neuen primären Superwächter **115-2** und beginnt, ihm periodisch Herzschlagsignale zu senden. Da in dem Netzwerk nur eine Kopie des Replikamanagers läuft, wird der Zustand des Replikamanagers persistent gemacht, indem die Tabelle 200 in dem Fixpunktserver **110** gespeichert wird. Wenn somit der Replikamanager mit dem neuen primären Superwächter **115-2** auf Hostcomputer H5 migriert wird, lädt der auf diesem Host gestartete Replikamanager seinen Zustand von dem Fixpunktserver **110** und initialisiert seine interne Tabelle aus seinem gespeicherten Zustand neu. Wenn Replikamanager **112** ausfällt, wird in ähnlicher Weise sein Ausfall durch Superwächter **115-1** durch das Fehlen von Herzschlagsignalen erkannt. Superwächter **115-1** startet dann Replikamanager **112** auf demselben Hostcomputer neu, lädt seinen Zustand von dem Fixpunktserver **110** und initialisiert erneut seine interne Tabelle 200 aus seinem gespeicherten Zustand.

[0033] Die oben beschriebene Ausführungsform veranschaulicht die Prinzipien der vorliegenden Erfindung. Andere Ausführungsformen können von Fachleuten ersonnen werden, ohne von dem Umfang der vorliegenden Erfindung abzuweichen.

Patentansprüche

1. Fehlermanagementcomputervorrichtung, umfassend:
einen Wächterdämonprozess (**113-1**), der auf einem ersten Hostcomputer (**101**) läuft, der in einem Netzwerk (**100**) mit anderen Hostcomputern (**102**, **103**, **104**, **105**, **106**) verbunden ist, auf denen jeweils ihr eigener Wächterdämonprozess (**113-2**, **113-3**, **113-4**, **113-5**, **113-6**) läuft, wobei der auf dem ersten Hostcomputer laufende Wächterdämonprozess eine Kopie von mindestens einem Anwendungsmodul (A-1) überwacht, die auf dem ersten Hostcomputer läuft, um einen Ausfall dieser Kopie des Anwendungsmoduls zu erkennen, wobei mindestens einer der anderen Hostcomputer eine Backup-Kopie (A2, A3, A4) des Anwendungsmoduls hat, die entweder läuft oder sich im Leerlauf zustand befindet, wobei die auf den anderen Hostcomputern laufenden Wächterdämonprozesse jegliche Backup-Kopie des auf ihnen laufenden Anwendungsmoduls überwachen, ein Replikationsgrad die Gesamtanzahl der Kopien des Anwendungsmoduls auf den Hostcomputern in dem Netzwerk spezifiziert, die in einem laufenden Zustand gehalten werden sollen; und
einen Replikamanagerdämonprozess (**112**), der auf einem der Hostcomputer (**102**) in dem Netzwerk läuft, wobei der Replikamanagerdämonprozess so arbeitet, dass er eine Angabe von dem Wächterdämonprozess des ersten Hostcomputers erhält, dass die auf den ersten Hostcomputer laufende Kopie des Anwendungsmoduls ausgefallen ist, und/oder eine Angabe von dem auf beliebigen der anderen Hostcomputern laufenden Wächterdämonprozess erhält, dass die darauf laufende Backup-Kopie des Anwendungsmoduls ausgefallen ist, wobei der Replikamanagerdämonprozess als Reaktion auf das Erhalten einer Angabe des Ausfalls der auf beliebigen der Hostcomputer laufenden Kopie des Anwendungsmoduls die Wiederherstellung mit der Backup-Kopie auf mindestens einem der anderen Hostcomputer einleitet, auf dem sich eine Backup-Kopie des Anwendungsmoduls befindet,
dadurch gekennzeichnet, dass
der Replikamanagerdämonprozess automatisch mindestens eine der im Leerlauf befindlichen Backup-Kopien des Anwendungsmoduls auf mindestens einem Hostcomputer ausführt, wenn ein Ausfall einer auf beliebigen der Hostcomputer laufenden Kopie des Anwendungsmoduls erhalten worden ist, um die Gesamtanzahl der laufenden Kopien des Anwendungsmoduls auf den Hostcomputern in dem Netzwerk auf einem Replikationsgrad zu halten, der in einer Registrierungsnachricht spezifiziert ist, die der Replikadämonprozess von der auf dem ersten Hostcomputer laufenden Kopie des Anwendungsmoduls erhalten hat.

2. Fehlermanagementcomputervorrichtung nach Anspruch 1, die weiterhin einen Fixpunktserver (**110**) umfasst, der mit dem Netzwerk verbunden ist, wobei der Fixpunktserver periodisch die Zustände der auf dem ersten Hostcomputer laufenden Kopie des Anwendungsmoduls und des Replikamanagerdämonprozesses speichert.

3. Fehlermanagementcomputervorrichtung nach Anspruch 2, bei der die Wiederherstellung mit einer Backup-Kopie nach einem erkannten Ausfall der auf dem ersten Hostcomputer laufenden Kopie des Anwendungsmoduls eingeleitet wird, wobei die Backup-Kopie die Verarbeitungsfunktionen der ausgefallenen Kopie des Anwendungsmoduls übernimmt und von dem Fixpunktserver den letzten gespeicherten Zustand der ausgefallenen Kopie des Anwendungsmoduls abruft.

4. Fehlermanagementcomputervorrichtung nach Anspruch 3, die weiterhin einen Superwächterdämonprozess (**115-1**) umfasst, der auf demselben Hostcomputer wie der Replikadämonprozess läuft, wobei der Superwächterdämonprozess den ersten Hostcomputer auf Ausfall überwacht.

5. Fehlermanagementcomputervorrichtung nach Anspruch 4, bei der, nachdem der Superwächterdämonprozess einen Ausfall des ersten Hostcomputers erkannt hat, einer Backup-Kopie der auf dem ausgefallenen ersten Hostcomputer laufenden Kopie des Anwendungsmoduls signalisiert wird, die Verarbeitungsfunktionen dieser Kopie des Anwendungsmoduls zu übernehmen, wobei die Backup-Kopie von dem Fixpunktserver den letzten gespeicherten Zustand dieser Kopie des Anwendungsmoduls abruft.

6. Fehlermanagementcomputervorrichtung nach Anspruch 5, bei der die Registrierungsnachricht, die der Replikamanagerdämonprozess von dem Anwendungsmodul erhält, weiterhin eine Ersatzschaltstrategie spezifiziert, die angibt, ob die Backup-Kopie des Anwendungsmoduls die Verarbeitungsfunktionen der auf dem ersten Hostcomputers laufenden Kopie des Anwendungsmoduls jedes Mal übernehmen soll, wenn der Replikamanager einen Ausfall dieser Kopie feststellt, oder ob die Backup-Kopie diese Verarbeitungsfunktionen nur nach einer festgelegten Anzahl von Ausfällen dieser Kopie übernehmen soll.

7. Verfahren zur fehlertoleranten Datenverarbeitung in einem Netzwerk (**100**), in dem mehrer Hostcomputer (**101**, **102**, **103**, **104**, **105**, **106**) verbunden sind, das die Schritte umfasst, in denen
auf einem ersten Hostcomputer (**101**) mindestens ein auf dem ersten Hostcomputer laufendes Anwendungsmodul (A1) überwacht wird, um einen Ausfall dieser Kopie des Anwendungsmoduls zu erkennen, wobei mindestens einer der anderen Hostcomputer eine laufende oder im Leerlauf befindliche Backup-Kopie (A2, A3, A4) des Anwendungsmoduls hat, wobei die laufende Backup-Kopie des Anwendungs-

moduls auf dem anderen Hostcomputer überwacht wird, um einen Ausfall dieser Backup-Kopie zu erkennen, wobei ein Replikationsgrad die Gesamtanzahl der laufenden Kopien des Anwendungsmoduls auf den Hostcomputern in dem Netzwerk spezifiziert;
auf einem der Hostcomputer eine Angabe erhalten wird, dass die auf dem ersten Hostcomputer laufenden Kopie des Anwendungsmoduls ausgefallen ist, und/oder eine Angabe erhalten wird, dass die auf einem der anderen Hostcomputer laufende Backup-Kopie des Anwendungsmoduls ausgefallen ist; und
von dem Hostcomputer, auf dem die Angabe des Ausfalls einer Kopie des Anwendungsmoduls erhalten worden ist, die Wiederherstellung der ausgefallenen Kopie des Anwendungsmoduls mit der Backup-Kopie auf mindestens einem der anderen Hostcomputer eingeleitet wird, auf dem sich eine Backup-Kopie befindet;
dadurch gekennzeichnet, dass das Verfahren weiterhin die Schritte umfasst, in denen
auf dem Hostcomputer, von dem die Wiederherstellung von der auf dem ersten Hostcomputer laufenden Kopie des Anwendungsmoduls eingeleitet wird, eine Registrierungsnachricht erhalten wird, die einen Replikationsgrad für das Anwendungsmodul spezifiziert; und
als Reaktion auf die Angabe des Ausfalls der auf einem beliebigen der Hostcomputer laufenden Kopie des Anwendungsmoduls automatisch mindestens eine der im Leerlauf befindlichen Backup-Kopien des Anwendungsmoduls auf mindestens einem Hostcomputer ausgeführt wird, auf dem sich eine im Leerlauf befindliche Kopie des Anwendungsmoduls befindet, um die Gesamtanzahl der laufenden Kopien des Anwendungsmoduls in dem Netzwerk auf dem Replikationsgrad zu halten, der in der erhaltenen Registrierungsnachricht spezifiziert ist.

8. Verfahren nach Anspruch 7, bei dem die Registrierungsnachricht weiterhin eine Ersatzschaltstrategie spezifiziert, die angibt, ob die Backup-Kopie des Anwendungsmoduls die Verarbeitungsfunktionen der auf dem ersten Hostcomputer laufenden Kopie dieses Anwendungsmoduls des Anwendungsmoduls jedes Mal übernehmen soll, wenn ein Ausfall dieser Kopie dieses Anwendungsmoduls erkannt wird, oder ob diese Backup-Kopie diese Verarbeitungsfunktionen nur nach einer festgelegten Anzahl von Ausfällen übernehmen soll.

Es folgen 2 Blatt Zeichnungen

Anhängende Zeichnungen

FIG. 1

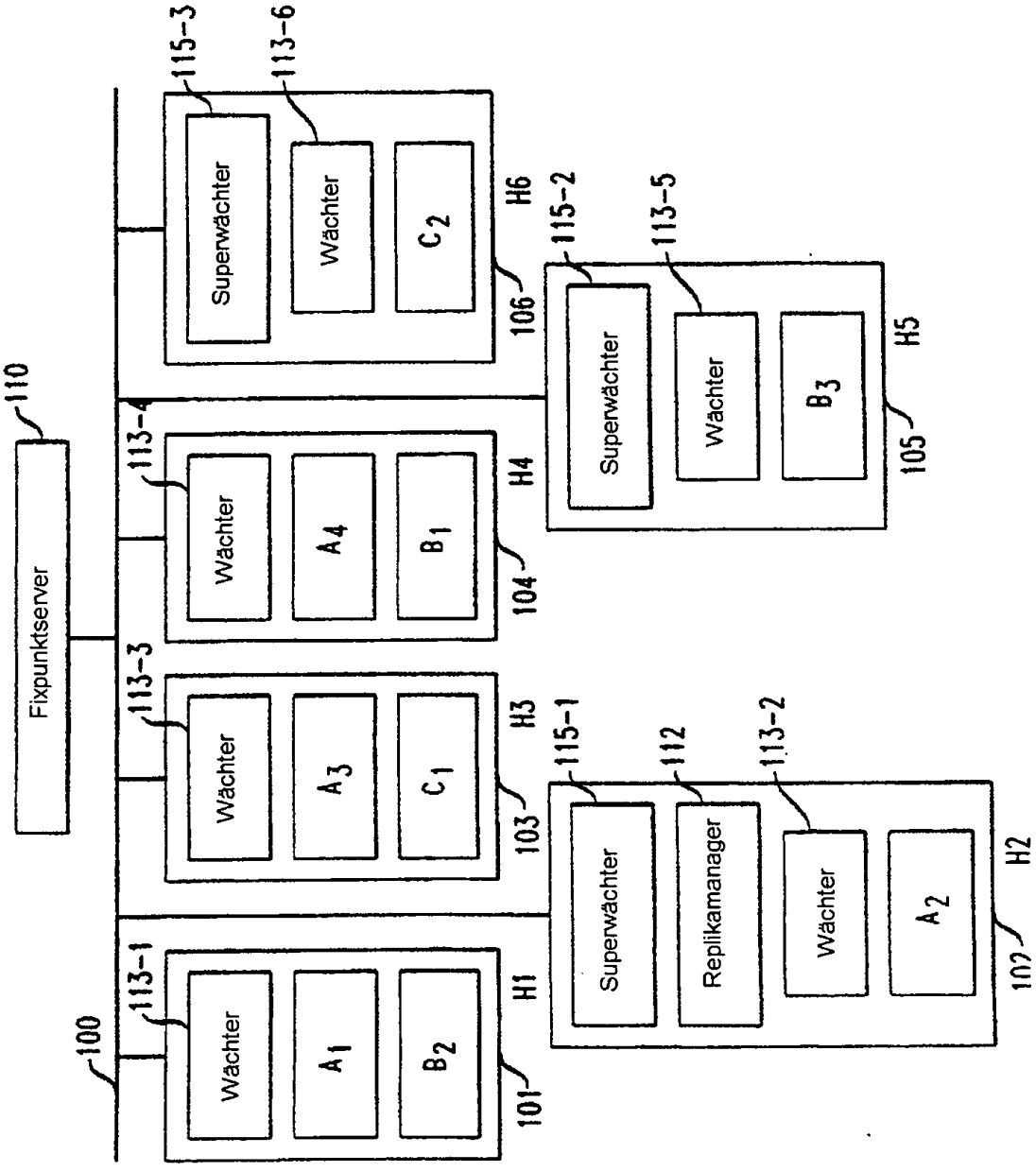
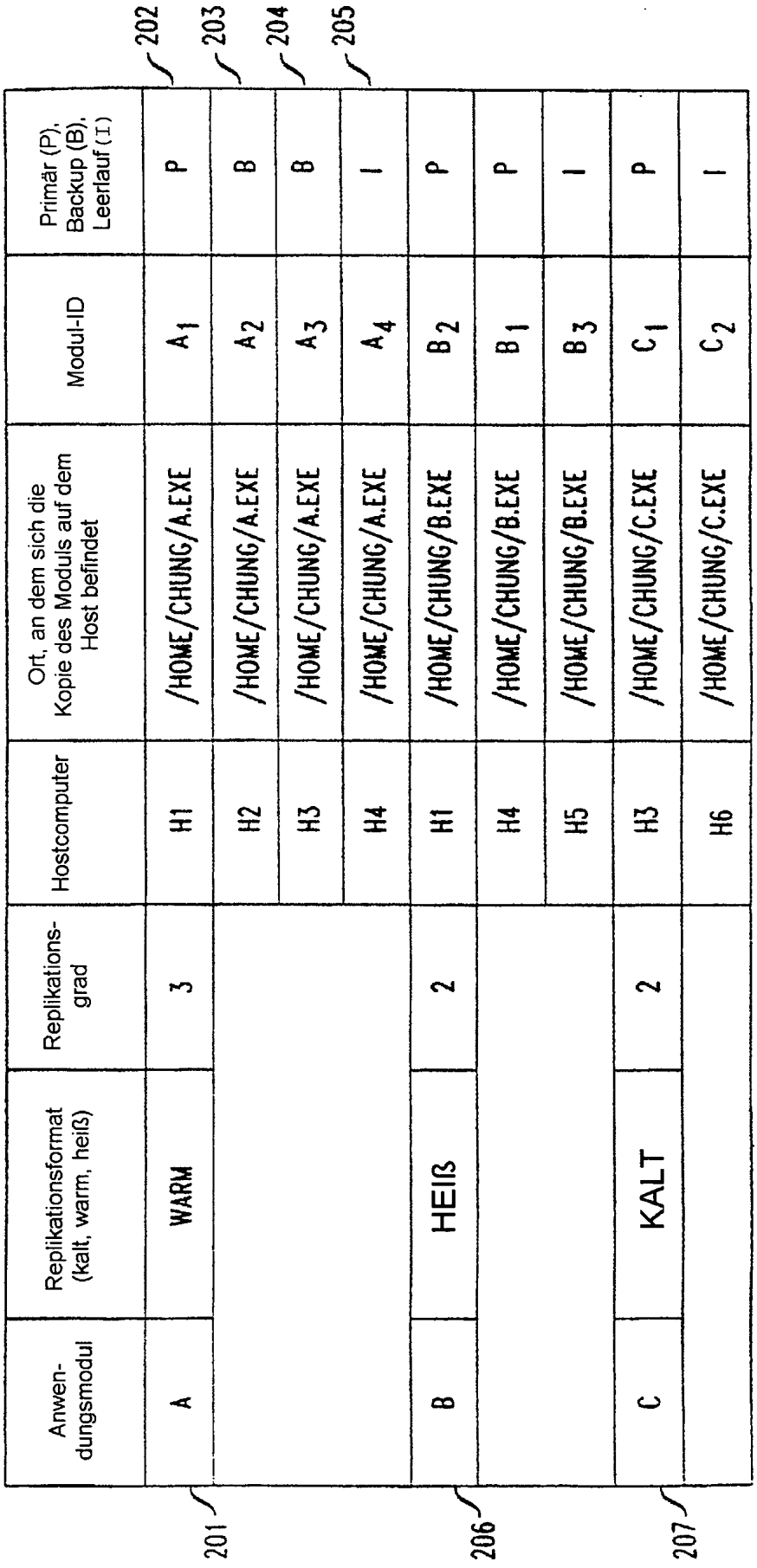


FIG. 2


200
Anwendungsmodul
Replikationsformat (kalt, warm, heiß)
Replikationsgrad
Hostcomputer
Ort, an dem sich die Kopie des Moduls auf dem Host befindet
Modul-ID
Primär (P), Backup (B), Leerlauf (I)
202
203
204
205
A
WARM
3
H1
/HOME/CHUNG/A.EXE
A1
P
H2
/HOME/CHUNG/A.EXE
A2
B
H3
/HOME/CHUNG/A.EXE
A3
B
H4
/HOME/CHUNG/A.EXE
A4
I
201
B
HEIß
2
H1
/HOME/CHUNG/B.EXE
B2
P
H4
/HOME/CHUNG/B.EXE
B1
P
H5
/HOME/CHUNG/B.EXE
B3
I
206
C
KALT
2
H3
/HOME/CHUNG/C.EXE
C1
P
H6
/HOME/CHUNG/C.EXE
C2
I
207