



(10) **DE 11 2012 004 776 T5** 2014.08.07

(12)

Veröffentlichung

der internationalen Anmeldung mit der
(87) Veröffentlichungs-Nr.: **WO 2013/072800**
in deutscher Übersetzung (Art. III § 8 Abs. 2 IntPatÜG)
(21) Deutsches Aktenzeichen: **11 2012 004 776.1**
(86) PCT-Aktenzeichen: **PCT/IB2012/056080**
(86) PCT-Anmeldetag: **01.11.2012**
(87) PCT-Veröffentlichungstag: **23.05.2013**
(43) Veröffentlichungstag der PCT Anmeldung
in deutscher Übersetzung: **07.08.2014**

(51) Int Cl.: **G06F 11/30 (2006.01)**

(30) Unionspriorität:
13/297,314 **16.11.2011** **US**

(71) Anmelder:
**International Business Machines Corp., Armonk,
N.Y., US**

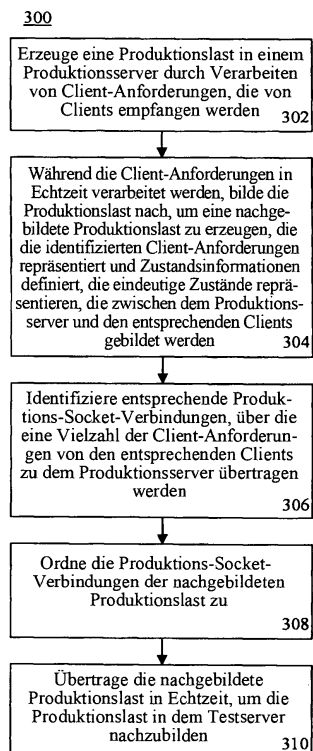
(74) Vertreter:
**Richardt Patentanwälte PartG mbB, 65185,
Wiesbaden, DE**

(72) Erfinder:
**Smith, Richard John, c/o IBM Corp., Austin, Tex.,
US; Williams, Gary Dale, c/o IBM Corp., Austin,
Tex., US**

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

(54) Bezeichnung: **Erzeugen einer Produktionsserver-Lastaktivität für einen Testserver**

(57) Zusammenfassung: Ein Verfahren zum Nachbilden einer Produktionslast eines Produktionsservers in einem Testserver enthält Erzeugen der Produktionslast in dem Produktionsserver durch Verarbeiten von Client-Anforderungen, die von Clients empfangen werden. Während die Client-Anforderungen über einen Prozessor in Echtzeit verarbeitet werden, wird die Produktionslast nachgebildet, um eine nachgebildete Produktionslast zu erzeugen, die die Client-Anforderungen repräsentiert und Zustandsinformationen definiert, die eindeutige Zustände repräsentieren, die zwischen dem Produktionsserver und den entsprechenden Clients gebildet werden. Das Verfahren enthält außerdem Übertragen der nachgebildeten Produktionslast in Echtzeit, um die Produktionslast in dem Testserver nachzubilden.



Beschreibung**HINTERGRUND**

[0001] Die vorliegende Erfindung bezieht sich allgemein auf das Testen von Produktionsservern.

[0002] Bei einem Produktionsserver handelt es sich um einen Server, der eine oder mehrere aktive Anwendungen beherbergt, auf die von Clients über ein Datenübertragungsnetzwerk wie etwa ein Intranet oder das Internet zugegriffen werden kann. Ein Produktionsserver beherbergt üblicherweise die neueste Version seiner entsprechenden Anwendung(en). Vor einem Verlagern einer Anwendung oder einer aktualisierten Version der Anwendung zu einem Produktionsserver, um aktiv zu werden, kann die Anwendung während der Entwicklung von einem Bereitstellungsserver beherbergt werden. Während einer Ausführung auf dem Bereitstellungsserver steht die Anwendung üblicherweise lediglich Entwicklern oder Benutzern zur Verfügung, die gebeten werden, die Funktionalität der Anwendung zu testen. Demzufolge stellt der Bereitstellungsserver eine Plattform bereit, auf der die Anwendung getestet werden kann, bevor sie aktiv wird.

KURZDARSTELLUNG

[0003] Gemäß einem ersten Aspekt der Erfindung wird ein Verfahren bereitgestellt, um eine Produktionslast eines Produktionsservers auf einem Testserver nachzubilden. Das Verfahren beinhaltet Erzeugen der Produktionslast an dem Produktionsserver durch Verarbeiten von Client-Anforderungen, die von Clients empfangen werden. Das Verfahren beinhaltet ferner Nachbilden der Produktionslast in Echtzeit über einen Prozessor, während die Client-Anforderungen verarbeitet werden, um eine nachgebildete Produktionslast zu erzeugen, die die Client-Anforderungen repräsentiert und Zustandsinformationen definiert, die eindeutige Zustände repräsentieren, die zwischen dem Produktionsserver und den entsprechenden Clients gebildet werden. Das Verfahren beinhaltet außerdem Kommunizieren der nachgebildeten Produktionslast in Echtzeit, um die Produktionslast auf dem Testserver nachzubilden.

[0004] Ein weiterer Aspekt beinhaltet ein Verfahren zum Nachbilden einer Produktionslast eines Produktionsservers auf einem Testserver. Das Verfahren beinhaltet Empfangen einer nachgebildeten Produktionslast, die einer Produktionslast an dem Produktionsserver entspricht, die durch Verarbeiten von Client-Anforderungen erzeugt wird. Die nachgebildete Produktionslast wird in Echtzeit erzeugt, während die Client-Anforderungen durch den Produktionsserver verarbeitet werden. Die Nachbildungs-Produktionslast definiert außerdem Zustandsinformationen, die eindeutige Zustände repräsentieren, die zwi-

schen dem Produktionsserver und den Clients gebildet werden, von denen die Client-Anforderungen empfangen werden. In Reaktion auf ein Empfangen der nachgebildeten Produktionslast wird die nachgebildete Produktionslast in Echtzeit über einen Prozessor verarbeitet, um eine Vielzahl von nachgebildeten Client-Anforderungen zu erzeugen. Jede von den nachgebildeten Client-Anforderungen bildet eine entsprechende Client-Anforderung nach, die durch den Produktionsserver empfangen wurde, und wenigstens ein Abschnitt der nachgebildeten Client-Anforderungen definiert die Zustandsinformationen, die dem entsprechenden Client zugehörig sind, von dem die Client-Anforderung empfangen wird. Das Verfahren beinhaltet ferner Übertragen der nachgebildeten Client-Anforderungen in Echtzeit an den Testserver.

[0005] Eine weitere Ausführungsform kann ein Computerprogrammprodukt enthalten, um eine Produktionslast eines Produktionsservers auf einem Testserver nachzubilden. Das Computerprogrammprodukt kann einen computerlesbaren Programmcode enthalten, der darin verkörpert ist, wobei der computerlesbare Programmcode so eingerichtet ist, dass er die verschiedenen Operationen und/oder Funktionen, die in dieser Spezifikation offenbart werden, ausführt.

KURZBESCHREIBUNG DER ZEICHNUNGEN

[0006] Ausführungsformen der vorliegenden Erfindung werden nun lediglich beispielhaft und unter Bezugnahme auf die beigefügten Zeichnungen beschrieben, in denen:

[0007] Fig. 1 ein Funktionsschaubild ist, das ein System zum Nachbilden einer Produktionslast eines Produktionsservers auf einem Testserver gemäß einer Ausführungsform der vorliegenden Erfindung veranschaulicht,

[0008] Fig. 2 ein Blockschaltplan ist, der ein Verarbeitungssystem gemäß einer Ausführungsform der vorliegenden Erfindung veranschaulicht,

[0009] Fig. 3 ein Ablaufplan ist, der ein Verfahren zum Nachbilden einer Produktionslast eines Produktionsservers auf einem Testserver gemäß einer weiteren Ausführungsform der vorliegenden Erfindung veranschaulicht,

[0010] Fig. 4 ein Ablaufplan ist, der ein Verfahren zum Erzeugen einer Nachricht einer nachgebildeten Produktionslast gemäß einer weiteren Ausführungsform der vorliegenden Erfindung veranschaulicht,

[0011] Fig. 5 ein Ablaufplan ist, der ein weiteres Verfahren zum Nachbilden einer Produktionslast eines Produktionsservers auf einem Testserver gemäß ei-

ner weiteren Ausführungsform der vorliegenden Erfindung veranschaulicht, und

[0012] Fig. 6 ein Ablaufplan ist, der ein Verfahren zum Verarbeiten einer nachgebildeten Client-Anforderung auf dem Testserver gemäß einer weiteren Ausführungsform der vorliegenden Erfindung veranschaulicht.

GENAUE BESCHREIBUNG

[0013] Einem Fachmann ist klar, dass Aspekte der vorliegenden Erfindung als System, Verfahren oder Computerprogrammprodukt verkörpert werden können. Dementsprechend können Aspekte der vorliegenden Erfindung die Form einer reinen Hardware-Ausführungsform, einer reinen Software-Ausführungsform (darunter Firmware, residente Software, Mikrocode usw.) oder einer Ausführungsform, die Software- und Hardware-Aspekte kombiniert, annehmen, die hier alle als "Schaltung", "Modul" oder "System" bezeichnet werden können. Des Weiteren können Aspekte der vorliegenden Erfindung die Form eines Computerprogrammprodukts annehmen, das in einem oder mehreren computerlesbaren Medien verkörpert wird und beispielsweise darin gespeichert ist, die computerlesbaren Programmcode aufweisen, der darin ausgeführt wird.

[0014] Jede Kombination aus einem oder mehreren computerlesbaren Medien kann genutzt werden. Bei dem computerlesbaren Medium kann es sich um ein computerlesbares Signalmedium oder ein computerlesbares Speichermedium handeln. Ein computerlesbares Speichermedium kann z. B. ein elektronisches, magnetisches, optisches, elektromagnetisches, Infrarot- oder Halbleitersystem, -vorrichtung oder -einheit oder jede geeignete Kombination des Vorhergehenden sein, ist jedoch nicht darauf beschränkt. Zu spezifischeren Beispielen (eine nicht erschöpfende Liste) des computerlesbaren Speichermediums würde Folgendes gehören: eine elektrische Verbindung mit einer oder mehreren Leitungen, eine tragbare Computerdiskette, ein Festplattenlaufwerk (HDD), ein Halbleiterlaufwerk (SSD), ein Direktzugriffsspeicher (RAM), ein Festwertspeicher (ROM), ein löschbarer programmierbarer Festwertspeicher (EPROM oder Flash-Speicher), ein Lichtwellenleiter, ein tragbarer Compactdisk-Festwertspeicher (CD-ROM), eine Digital Versatile Disk (DVD), eine optische Speichereinheit, eine magnetische Speichereinheit oder jede geeignete Kombination des Vorhergehenden. Im Kontext dieses Dokuments kann ein computerlesbares Speichermedium jedes materielle Medium sein, das ein Programm zur Verwendung durch oder in Verbindung mit einem System, einer Vorrichtung oder einer Einheit zur Befehlsausführung enthalten oder speichern kann.

[0015] Ein computerlesbares Signalmedium kann ein sich ausbreitendes Datensignal mit computerlesbarem Programmcode enthalten, der darin z. B. im Basisband oder als Teil einer Trägerwelle verkörpert wird. Ein derartiges sich ausbreitendes Signal kann jede von einer Vielfalt von Formen annehmen, zu denen elektromagnetische, optische Formen oder jede geeignete Kombination hiervon gehören, jedoch nicht darauf beschränkt sind. Ein computerlesbares Signalmedium kann jedes computerlesbare Medium sein, das kein computerlesbares Speichermedium ist und ein Programm zur Verwendung durch oder in Verbindung mit einem System, einer Vorrichtung oder einer Einheit zur Befehlsausführung kommunizieren, verbreiten oder transportieren kann.

[0016] Programmcode, der auf einem computerlesbaren Medium verkörpert ist, kann unter Verwendung jedes geeigneten Mediums übertragen werden, darunter drahtlose, leitungsgestützte, Lichtwellenleiterkabel-, Hochfrequenz-(HF-)Medien usw. oder jede geeignete Kombination aus dem Vorhergehenden, ohne darauf beschränkt zu sein. Computerprogrammcode zum Ausführen von Operationen für Aspekte der vorliegenden Erfindung kann in jeder Kombination aus einer oder mehreren Programmiersprachen geschrieben sein, darunter eine objektorientierte Programmiersprache wie Java™, Smalltalk, C++ oder dergleichen und herkömmliche prozedurale Programmiersprachen wie etwa die Programmiersprache "C" oder ähnliche Programmiersprachen. Der Programmcode kann nur auf dem Computer eines Benutzers, teilweise auf dem Computer eines Benutzers, als ein eigenständiges Software-Paket, teilweise auf dem Computer eines Benutzers und teilweise auf einem fernen Computer oder nur auf dem fernen Computer oder Server ausgeführt werden. In dem zuletzt genannten Szenario kann der ferne Computer mit dem Computer des Benutzers durch jeden Netzwerktyp verbunden sein, darunter ein lokales Netzwerk (LAN) oder ein Weitverkehrsnetz (WAN), oder die Verbindung kann zu einem externen Computer (z. B. über das Internet unter Verwendung eines Internet-Diensteanbieters) hergestellt werden.

[0017] Aspekte der vorliegenden Erfindung werden hier unter Bezugnahme auf Ablaufplan-Darstellungen und/oder Blockschaltbilder von Verfahren, Vorrichtungen (Systemen) und Computerprogrammprodukten gemäß Ausführungsformen der Erfindung beschrieben. Es ist klar, dass jeder Block der Ablaufplan-Darstellungen und/oder Blockschaltbilder und Kombinationen von Blöcken in den Ablaufplan-Darstellungen und/oder Blockschaltbildern durch Computerprogrammbefehle umgesetzt werden können. Diese Computerprogrammbefehle können einem Prozessor eines Universalcomputers, eines Spezialcomputers oder einer anderen programmierbaren Datenverarbeitungsvorrichtung bereitgestellt werden, um eine Maschine zu bilden, so dass

Befehle, die über den Prozessor des Computers, der anderen programmierbaren Datenverarbeitungsvorrichtung oder anderer Einheiten ausgeführt werden, Mittel zum Umsetzen der Funktionen/Wirkungen, die in dem Block oder den Blöcken des Ablaufplans und/oder Blockschalbilds spezifiziert sind, erzeugen.

[0018] Diese Computerprogrammbefehle können außerdem in einem computerlesbaren Medium gespeichert sein, das einen Computer, eine andere programmierbare Datenverarbeitungsvorrichtung oder andere Einheiten anweisen kann, in einer bestimmten Weise zu funktionieren, so dass die in dem computerlesbaren Medium gespeicherten Befehle einen Herstellungsgegenstand produzieren, der Befehle enthält, die die Funktion/Wirkung umsetzen, die in dem Block/den Blöcken des Ablaufplans und/oder Blockschalbilds spezifiziert sind.

[0019] Computerprogrammbefehle können außerdem in einen Computer, andere programmierbare Datenverarbeitungsvorrichtungen oder andere Einheiten geladen werden, um eine Reihe von Operationsschritten zu bewirken, die auf dem Computer, der anderen programmierbaren Datenverarbeitungsvorrichtung oder anderen Einheiten ausgeführt werden sollen, um einen durch einen Computer umgesetzten Prozess zu erzeugen, so dass die Befehle, die auf dem Computer oder der anderen programmierbaren Vorrichtung ausgeführt werden, Prozesse zum Umsetzen der Funktionen/Wirkungen, die in dem Block oder Blöcken des Ablaufplans und/oder Blockschalbilds spezifiziert sind, bereitstellen.

[0020] Mehrere Definitionen, die in diesem Dokument durchgängig gelten, werden im Folgenden präsentiert. Der hier verwendete Ausdruck „Echtzeit“ bezeichnet ein Maß des Antwortverhaltens der Verarbeitung, das ein Benutzer oder ein System für einen bestimmten Prozess oder eine Feststellung, die getroffen werden soll, als in ausreichendem Maße als unmittelbar empfindet, oder das ermöglicht, dass der Prozessor mit einem bestimmten externen Prozess Schritt hält.

[0021] Der hier verwendete Ausdruck „Produktionsserver“ bezeichnet ein Verarbeitungssystem, das wenigstens einen Prozessor enthält und eine oder mehrere aktive Anwendungen beherbergt, auf die über ein Datenübertragungsnetzwerk zugegriffen werden kann. Eine Anwendung, die aktiv bleibt, bedeutet in diesem Zusammenhang eine Anwendung, die funktionsfähig ist, um Clients über ein Datenübertragungsnetzwerk wie etwa ein Intranet oder das Internet in der vorgesehenen Weise zu dienen. Der hier verwendete Ausdruck „Testserver“ bezeichnet ein Verarbeitungssystem, das wenigstens einen Prozessor aufweist und eine oder mehrere Anwendungen für den Zweck des Testens dieser Anwendung(en) beherbergt. Der hier verwendete Ausdruck „Zwischen-

server“ bezeichnet ein Verarbeitungssystem, das wenigstens einen Prozessor enthält, der Daten von einem Produktionsserver empfängt und entsprechende Daten zu einem Testserver überträgt.

[0022] Der hier verwendete Ausdruck „Client“ bezeichnet ein Verarbeitungssystem, das wenigstens einen Prozessor aufweist und Client-Anforderungen erzeugt, die zu einem Produktionsserver übertragen werden. Der hier verwendete Ausdruck „Client-Anforderung“ bezeichnet eine Nachricht, die fordert, dass eine bestimmte Aktion durch einen Produktionsserver ausgeführt wird. Eine Client-Anforderung kann beispielsweise fordern, dass Informationen abgerufen oder verarbeitet werden. Der hier verwendete Ausdruck „Produktionslast“ bezieht sich auf diskrete, unabhängig bedeutsame Einheiten, die als „Produktionslasteinheiten“ bezeichnet werden. Bei einer Ausführungsform kann jede Produktionslasteinheit einer separaten Client-Anforderung entsprechen.

[0023] Ausführungsformen, die hier beschrieben werden, beziehen sich auf ein Nachbilden einer Produktionslast eines Produktionsservers in einem Testserver. Während der Produktionsserver eine oder mehrere aktive Anwendungen beherbergt, kann der Produktionsserver Client-Anforderungen von einem oder mehreren Clients empfangen, um auf die aktive(n) Anwendung(en) zuzugreifen. Der Produktionsserver kann derartige Client-Anwendungen verarbeiten und Antworten auf die Client-Anforderungen erzeugen, wobei somit eine Produktionslast auf dem Produktionsserver erzeugt wird. Wenn die Client-Anforderungen durch den Produktionsserver verarbeitet werden, kann die durch die Client-Anforderungen repräsentierte Produktionslast nachgebildet werden, um eine nachgebildete Produktionslast zu erzeugen. Die nachgebildete Produktionslast repräsentiert die Client-Anforderungen. Die nachgebildete Produktionslast kann des Weiteren Zustandsinformationen definieren, die eindeutige Zustände repräsentieren, die zwischen dem Produktionsserver und den Clients gebildet werden, von denen die identifizierten Client-Anforderungen empfangen werden.

[0024] Die nachgebildete Produktionslast kann zu einem Verteiler-Proxy übertragen werden, der die nachgebildete Produktionslast verarbeiten kann, um nachgebildete Client-Anforderungen zu erzeugen. Jede nachgebildete Client-Anforderung kann eine entsprechende Client-Anforderung, die durch den Produktionsserver empfangen wird, sowie die Zustandsinformationen nachbilden, die der entsprechenden Client-Anforderung zugehörig sind. Der Verteiler-Proxy kann die nachgebildeten Client-Anforderungen zu dem Testserver senden.

[0025] Der Testserver kann eine oder mehrere Kopien von Anwendungen beherbergen, die durch den Produktionsserver beherbergt werden, die die nach-

gebildeten Client-Anforderungen verarbeiten können. Eine derartige Verarbeitung kann durch eine Analyse-Anwendung überwacht werden, um Fehler zu identifizieren, die in der Anwendung (den Anwendungen) gelegentlich als „Bugs“ bezeichnet werden, oder um die Leistungsfähigkeit einer Testversion einer Anwendung mit einer Produktionsversion der Anwendung zu vergleichen. In diesem Zusammenhang können die Kopien der Anwendungen, die von dem Testserver beherbergt werden, mit Code instrumentiert werden, um eine derartige Analyse zu erleichtern. Die oben erwähnten Prozesse können insbesondere in Echtzeit ausgeführt werden und können so lange wie gewünscht fortgesetzt werden. Dementsprechend können die von dem Produktionsserver beherbergten Anwendungen analysiert werden, während die Produktionsanwendungen aktiv bleiben. Somit muss der Produktionsserver nicht offline geschaltet werden, um die Analyse auszuführen, und kann weiterhin Clients dienen. Darüber hinaus kann die Analyse unter Verwendung derselben aktiven Bedingungen ausgeführt werden, denen der Produktionsserver ausgesetzt ist. Diese Bedingungen können sich zeitlich ändern, wobei derartige Änderungen in dem Testserver nachgebildet werden können.

[0026] Fig. 1 ist ein Funktionsschaubild, das ein System **100** zum Nachbilden einer Produktionslast eines Produktionsservers auf einem Testserver gemäß einer Ausführungsform der vorliegenden Erfindung veranschaulicht. In diesem Zusammenhang kann das System einen oder mehrere Clients **110**, einen Produktionsserver **120**, einen Testserver **140** und wahlweise einen Zwischenserver **140** enthalten.

[0027] Der Produktionsserver **120** kann eine oder mehreren Anwendungen **122** beherbergen (z. B. ausführen). Zu Beispielen einer Anwendung **122** gehören eine Website, ein Lightweight-Directory-Access-Protocol-(LDAP-)Server, eine Datenbank, eine webgestützte Anwendung, eine netzwerkgestützte Anwendung und dergleichen, ohne auf diese beschränkt zu sein. Der Produktionsserver **120** kann außerdem eine Prüf-Einrichtung **124** beherbergen, die hier genauer beschrieben wird. Bei einer Ausführungsform kann der Zwischenserver **130** einen Verteiler-Proxy **132** beherbergen, der ebenfalls hier genauer beschrieben wird. Bei einer weiteren Ausführungsform kann der Verteiler-Proxy **132** in dem Testserver **140** beherbergt werden.

[0028] Der Testserver **140** kann eine oder mehrere Host-Anwendungen **142** und eine Analyse-Anwendung **144** beherbergen. Bei der bzw. den Anwendungen **142** kann es sich um eine bzw. mehrere Kopien von Anwendungen **122** handeln, die in dem Produktionsserver **120** beherbergt werden, die durch die Analyse-Anwendung **144** zu analysieren sind. Bei einer Anwendung **142** kann es sich um dieselbe Version handeln wie bei einer entsprechenden Anwendung

122, wenn beispielsweise die Anwendung in Bezug auf mögliche Defekte analysiert wird, oder es kann sich bei der Anwendung **142** um eine andere Version als bei der Anwendung **122** handeln, wenn beispielsweise Leistungsverbesserungen an der Anwendung getestet werden. In diesem Zusammenhang können die Anwendung(en) **142** als ein oder mehrere Computerprogramme betrachtet werden, die getestet werden. Im Folgenden kann auf eine Anwendung **122** und/oder eine Anwendung **142** Bezug genommen werden, es ist jedoch klar, dass eine derartige Bezugnahme eine Vielzahl der entsprechenden Anwendungen **122** und/oder Anwendungen **142** betreffen kann.

[0029] Die Clients **110** können mit dem Produktionsserver **120** zum Zweck des Datenaustauschs verbunden sein, der Produktionsserver kann mit dem Zwischenserver **130** zum Zweck des Datenaustauschs verbunden sein und der Zwischenserver **130** kann mit dem Testserver **130** zum Zweck des Datenaustauschs verbunden sein. Bei einer Anordnung, bei der der Verteiler-Proxy **132** von dem Testserver **140** beherbergt ist, kann der Produktionsserver **120** mit dem Testserver **140** anstelle des Zwischenservers **130** zum Zweck des Datenaustauschs verbunden sein. Ein oder mehrere Netzwerke können genutzt werden, um derartige Datenübertragungsverbindungen bereitzustellen. Zu den Netzwerken können öffentliche Netzwerke, z. B. das Internet, und/oder private Netzwerke wie etwa LANs, WANs oder dergleichen gehören.

[0030] Im Betrieb kann der Produktionsserver **120** funktionsfähig sein und die Anwendung **122** ausführen, um die Clients **110** zu bedienen. Der Produktionsserver **120** kann eine Vielzahl von Client-Anforderungen **112** von den Clients **110** empfangen und identifizieren, die an die Anwendung **122** gerichtet sind. Die Anwendung **122** kann die Client-Anforderungen **112** verarbeiten und in einer Weise, die für die Anwendung **122** üblich ist, Antworten **114** an die Clients **110** zurückgeben. Wenn die Client-Anforderungen **112** durch den Produktionsserver **120** in Echtzeit empfangen werden, kann die Prüf-Einrichtung **124** Prüf-Datensätze erzeugen, die den Client-Anforderungen **112** entsprechen, und die Prüf-Datensätze in einem Prüf-Protokoll **126** speichern. Die Prüf-Einrichtung **124** kann des Weiteren eine Nachricht erzeugen, die sich auf eine bestimmte Client-Anforderung **112** bezieht.

[0031] Die Prüf-Einrichtung **124** kann die Nachricht zu dem Verteiler-Proxy **132** übertragen. Wenn der Verteiler-Proxy **132** von dem Zwischenserver **130** beherbergt wird, kann die Nachricht zu dem Zwischenserver **130** gesendet werden. Wenn der Verteiler-Proxy **132** von dem Testserver **140** beherbergt wird, kann die Nachricht zu dem Testserver **140** gesendet werden. Während des normalen Betriebs kann der Produktionsserver **120** eine Vielzahl von Client-

Anforderungen **112** empfangen, und die Prüf-Einrichtung kann eine Vielzahl von entsprechenden Nachrichten erzeugen und übertragen, von denen jede eine Produktionslasteinheit repräsentiert. Die Vielzahl derartiger Nachrichten kann gemeinsam die Produktionslast des Produktionsservers **120** nachbilden, und sie werden hier daher gemeinsam als eine nachgebildete Produktionslast **128** bezeichnet.

[0032] Jede Nachricht kann spezielle Informationen enthalten, die eine bestimmte Client-Anforderung **112** betreffen, z. B. eine Aktion oder Aktionen, die von der Anwendung **122** oder anderen Prozessen in dem Produktionsserver **120** gefordert werden. Des Weiteren kann jede Nachricht Zustandsinformationen enthalten, die einen eindeutigen Zustand repräsentieren, der zwischen dem Produktionsserver **120** und dem Client **110** gebildet wird, von dem die Client-Anforderung **112** empfangen wird. Zu derartigen Zustandsinformationen können z. B. gehören: eine Verbindungskennung für die Socket-Verbindung, über die die Client-Anforderung **112** empfangen wurde, ein oder mehrere Cookies, die der Client-Anforderung **112** zugehörig sind, auf der Grundlage einer vorherigen Nutzung des Produktionsservers **120** durch den Client **110**, Page-Dienste, die dem Client **110** bereitgestellt werden, Verbindungsroutrinen, die dem Client **110** zugehörig sind, Bindungsinformationen, die dem Client **110** entsprechen (z. B. Benutzername und/oder Passwort) und/oder alle weiteren Informationen, die einen eindeutigen Zustand repräsentieren, der zwischen dem Produktionsserver **120** und dem Client **110** gebildet wird.

[0033] Wenn der Verteiler-Proxy **132** die Nachrichten in Echtzeit empfängt, kann der Verteiler-Proxy **132** Daten aus jeder Nachricht einer Syntaxprüfung unterziehen und eine nachgebildete Client-Anforderung **134** erzeugen, die der Client-Anforderung **112** entspricht, auf der die Nachricht beruht. Jede nachgebildete Client-Anforderung **134** kann die speziellen Informationen, die ihre entsprechende Client-Anforderung **112** betreffen, sowie die Zustandsinformationen enthalten, die den eindeutigen Zustand repräsentieren, der zwischen dem Produktionsserver **120** und dem Client **110** gebildet wird, von dem die Client-Anforderung **112** empfangen wurde.

[0034] Bei einer Ausführungsform kann der Verteiler-Proxy **112** die Nachrichten der nachgebildeten Produktionslast **128** filtern, um Client-Anforderungen **112** zu filtern, die unter Verwendung der nachgebildeten Client-Anforderungen **134** für den Testserver **140** nachgebildet werden. Bei einer weiteren Ausführungsform können die Prüf-Datensätze und somit die Client-Anforderungen **112** selektiv gefiltert werden, um einen Teil der Client-Anforderungen **112** zu identifizieren, für die entsprechende Nachrichten erzeugt werden. Nachrichten, die z. B. Client-Anforderungen **112** entsprechen, die bestimmten Benutzern

oder Passwörtern zugehörig sind, können ignoriert werden, oder Nachrichten, die bestimmten Typen von Client-Anforderungen **112** entsprechen, können ignoriert werden, wobei entsprechende nachgebildete Client-Anforderungen **134** nicht erzeugt werden müssen. Des Weiteren kann das selektive Filtern Nachrichten identifizieren, die bestimmten Typen von Client-Anforderungen **112** entsprechen, die zu verarbeitet sind, um die nachgebildeten Client-Anforderungen **134** zu erzeugen. Beispielsweise können lediglich Client-Anforderungen **112** nachgebildet werden, die LDAP-Verzeichnis- oder Datenbank-Suchvorgänge betreffen.

[0035] Wenn eine bestimmte Nachricht einer Client-Anforderung **112** entspricht, die fordert, dass eine Socket-Verbindung (Produktions-Socket-Verbindung) für einen entsprechenden Client **110** durch den Produktionsserver **120** aufgebaut wird, kann die entsprechende nachgebildete Client-Anforderung **134** fordern, dass eine Socket-Verbindung (Test-Socket-Verbindung) durch den Testserver **140** aufgebaut wird. Die Kennung der Test-Socket-Verbindung, die erzeugt wird, ist möglicherweise nicht dieselbe Kennung, die für die Produktions-Socket-Verbindung erzeugt wird. Der Verteiler-Proxy **132** kann trotzdem eine Zuordnung zwischen den entsprechenden Kennungen erzeugen und diese Zuordnung in einer Zuordnungstabelle speichern. Auf der Grundlage der Zuordnung kann der Verteiler-Proxy **132** die Produktions-Socket-Verbindung der Test-Socket-Verbindung zuordnen. Wenn demzufolge eine Vielzahl von Client-Anforderungen **112** dem Produktionsserver **120** über eine bestimmte Produktions-Socket-Verbindung bereitgestellt wird, kann der Verteiler-Proxy **132** sicherstellen, dass ihre entsprechenden nachgebildeten Client-Anforderungen **134** über dieselbe Test-Socket-Verbindung gesendet werden.

[0036] Wenn jede nachgebildete Client-Anforderung **134** erzeugt wird, kann der Verteiler-Proxy die nachgebildete Client-Anforderung **134** in Echtzeit zu dem Testserver **140** übertragen, der die nachgebildete Client-Anforderung zu der Anwendung **142** leiten kann. Die nachgebildete Client-Anforderung **134** kann gemäß einem Protokoll formatiert sein, das für die Anwendung **142** verständlich ist. Wenn beispielsweise eine bestimmte Client-Anforderung **112** gemäß LDAP formatiert ist, kann die entsprechende nachgebildete Client-Anforderung **134** gemäß LDAP formatiert sein. Dabei handelt es sich lediglich um ein Beispiel eines Formats der Client-Anforderung, und die Erfindung ist in dieser Hinsicht nicht beschränkt. Die Anwendung **122** kann die nachgebildeten Client-Anforderungen **134** verarbeiten und nachgebildete Antworten **136** in einer Weise, die für die Anwendung **122** üblich ist, an den Verteiler-Proxy **132** zurückgeben.

[0037] Wenn die Anwendung **142** die nachgebildeten Client-Anforderungen **134** verarbeitet, kann die

Analyse-Anwendung die Ausführung der Anwendung **142** analysieren, um Defekte zu identifizieren, die möglicherweise in dem Code der Anwendung **142** vorhanden sind, oder um die Leistungsfähigkeit der Anwendung **142** zu analysieren. Die Anwendung **142** kann beispielsweise mit Programmcode instrumentiert werden, um Ereignisse zu erzeugen, wenn die Anwendung **142** ausgeführt wird, und derartige Ereignisse können durch die Analyse-Anwendung **144** erkannt werden. Eine Anwendungs-Instrumentierung und Analyse-Anwendungen sind Fachleuten bekannt.

[0038] Fig. 2 ist ein Blockschaltplan, der ein Beispiel eines Verarbeitungssystems **200** gemäß einer Ausführungsform der vorliegenden Erfindung veranschaulicht. Bei dem Verarbeitungssystem **200** kann es sich um einen Client, einen Produktionsserver, einen Zwischenserver oder einen Testserver handeln. Das Verarbeitungssystem **200** kann wenigstens einen Prozessor **205** enthalten, der über einen Systembus **215** mit Speicherelementen **210** verbunden ist. Daher kann das Verarbeitungssystem **200** Programmcode in den Speicherelementen **210** speichern. Der Prozessor **205** kann den Programmcode ausführen, auf den über den Systembus **215** in den Speicherelementen **210** zugegriffen werden kann. Bei einem Aspekt kann z. B. das Verarbeitungssystem **200** als ein Computer umgesetzt werden, der in der Lage ist, Programmcode zu speichern und/oder auszuführen. Es sollte jedoch klar sein, dass das Verarbeitungssystem **200** in Form eines beliebigen Systems umgesetzt werden kann, das einen Prozessor und einen Speicher aufweist und in der Lage ist, die in dieser Spezifikation beschriebenen Funktionen auszuführen.

[0039] Zu den Speicherelementen **210** können ein oder mehrere physische Speichereinheiten gehören wie z. B. ein lokaler Speicher **220** und ein oder mehrere Massenspeichereinheiten **225**. Als lokaler Speicher **220** werden ein Direktzugriffsspeicher oder andere nichtdauerhafte Speichereinheit(en) bezeichnet, die im Allgemeinen während der eigentlichen Ausführung des Programmcode verwendet werden. Die Massenspeichereinheit(en) **225** können als HDD, SSD oder andere dauerhafte Datenspeichereinheiten umgesetzt werden. Das Verarbeitungssystem **200** kann außerdem einen oder mehrere Cache-Speicher (nicht gezeigt) enthalten, die eine vorübergehende Speicherung wenigstens eines Teils der Programmcode gewährleisten, um die Häufigkeit zu vermindern, wie oft Programmcode während der Ausführung von der Massenspeichereinheit **225** abgerufen werden muss.

[0040] Ein oder mehrere Netzwerkadapter **230** können mit dem Verarbeitungssystem **200** entweder direkt mit dem Systembus **215** oder über dazwischliegende E/A-Steuereinheiten verbunden sein, damit

das Verarbeitungssystem **200** mit anderen Systemen, Computersystemen, fernen Druckern und/oder fernen Speichereinheiten über private oder öffentliche Netzwerke verbunden werden kann. Modems, Kabel-Modems, Ethernet-Karten-Drahtlos-Netzwerkadapter sind Beispiele von unterschiedlichen Typen von Netzwerkadaptern, die bei dem Verarbeitungssystem **200** verwendet werden können, wobei die Erfindung in dieser Hinsicht nicht beschränkt ist. Weitere E/A-Einheiten (nicht gezeigt) können außerdem mit dem Verarbeitungssystem **200** verbunden werden.

[0041] Beispielhaft und nicht als Einschränkung können ein oder mehrere Programme/Dienstprogramme **240**, von denen jedes eine Gruppe (wenigstens ein) von Programmmodulen **242** aufweist, sowie ein Betriebssystem, ein oder mehrere Anwendungsprogramme, weitere Programmmodule und Programmdateien in den Speicherelementen **210** gespeichert werden. Programmmodule **142** führen im Allgemeinen die Funktionen und/oder Methodiken von Ausführungsformen der hier beschriebenen Erfindung aus.

[0042] Wenn es sich bei dem Verarbeitungssystem **200** um den Produktionsserver **120** von Fig. 1 handelt, können die Programme/Dienstprogramme **240** in der Darstellung die Anwendung(en) **122** und die Prüf-Einrichtung **124** enthalten. Das Prüf-Protokoll **126** kann außerdem in den Speicherelementen **210** gespeichert werden. Wenn es sich bei dem Verarbeitungssystem **200** um den Zwischenserver **130** von Fig. 1 handelt, können die Programme/Dienstprogramme **240** den Verteiler-Proxy **132** enthalten. Wenn es sich bei dem Verarbeitungssystem **200** um den Testserver **140** von Fig. 1 handelt, können die Programme/Dienstprogramme **240** die Anwendung(en) **142**, die Analyseanwendung **144** und wahlweise den Verteiler-Proxy **132** enthalten (wenn der Zwischenserver **130** nicht verwendet wird).

[0043] Fig. 3 ist ein Ablaufplan, der ein Verfahren **300** zum Nachbilden einer Produktionslast eines Produktionsservers in einem Testserver gemäß einer weiteren Ausführungsform der vorliegenden Erfindung veranschaulicht. Das Verfahren **300** kann durch die Prüf-Einrichtung **124** von Fig. 1 umgesetzt werden. Im Schritt **302** kann eine Produktionslast in einem Produktionsserver durch Verarbeiten von Client-Anforderungen erzeugt werden, die von Clients empfangen werden. Während die Client-Anforderungen in Echtzeit verarbeitet werden, kann im Schritt **304** die Produktionslast nachgebildet werden, um eine nachgebildete Produktionslast zu erzeugen, die die identifizierten Client-Anforderungen repräsentiert und Zustandsinformationen definiert, die eindeutige Zustände repräsentieren, die zwischen dem Produktionsserver und den entsprechenden Clients gebildet werden. Im Schritt **306** können entsprechende Produktions-

Socket-Verbindungen identifiziert werden, über die eine Vielzahl der Client-Anforderungen von den entsprechenden Clients zu dem Produktionsserver übertragen werden. Im Schritt **308** können die Produktions-Socket-Verbindungen der nachgebildeten Produktionslast zugeordnet werden. Im Schritt **310** kann die nachgebildete Produktionslast in Echtzeit übertragen werden, um die Produktionslast in dem Testserver nachzubilden. Die nachgebildete Produktionslast kann z. B. zu dem Verteiler-Proxy **132** von **Fig. 1** übertragen werden.

[0044] **Fig. 4** ist ein Ablaufplan, der ein Verfahren **400** zum Erzeugen einer Nachricht über eine nachgebildete Produktionslast gemäß einer weiteren Ausführungsform der vorliegenden Erfindung veranschaulicht. Das Verfahren **400** kann wiederum durch die Prüf-Einrichtung **124** von **Fig. 1** umgesetzt werden.

[0045] Im Schritt **402** kann Speicherplatz für erste Datenstrukturen für Client-Anforderungsdaten zugewiesen werden. Im Schritt **404** kann Speicherplatz für zweite Datenstrukturen für einen Verteiler-Proxy-Datensatz zugewiesen werden. Im Schritt **406** kann eine Client-Anforderung durch den Produktionsserver empfangen werden, und die Prüf-Einrichtung kann die Client-Anforderungsdaten in einen ersten Prüf-Protokoll-Datensatz eintragen. Im Schritt **408** kann die Prüf-Einrichtung Zustandsinformationen abrufen, die der Client-Anforderung zugehörig sind, und die Zustandsinformationen wie auch die Client-Anforderungsdaten in einen zweiten Prüf-Protokoll-Datensatz eintragen. Im Schritt **410** kann die Prüf-Einrichtung den zweiten Prüf-Protokoll-Datensatz in ein Socket schreiben, der dem Verteiler-Proxy zugehörig ist, wodurch eine Nachricht erzeugt wird, die eine Produktionslasteinheit repräsentiert, und die Nachricht zu dem Verteiler-Proxy übertragen wird. Im Schritt **414** kann die Prüf-Einrichtung den ersten Prüf-Protokoll-Datensatz in eine Prüf-Protokoll-Datei in dem Produktionsserver schreiben.

[0046] **Fig. 5** ist ein Ablaufplan, der ein weiteres Verfahren **500** zum Nachbilden einer Produktionslast eines Produktionsservers in einem Testserver gemäß einer weiteren Ausführungsform der vorliegenden Erfindung veranschaulicht. Das Verfahren **500** kann durch den Verteiler-Proxy **132** von **Fig. 1** umgesetzt werden.

[0047] Im Schritt **502** kann eine nachgebildete Produktionslast empfangen werden. Die nachgebildete Produktionslast kann einer Produktionslast in dem Produktionsserver entsprechen, die durch Verarbeiten von Client-Anforderungen erzeugt wird. Die nachgebildete Produktionslast kann in Echtzeit erzeugt werden, während die Client-Anforderungen durch den Produktionsserver verarbeitet werden, und kann Zustandsinformationen definieren, die eindeutige Zu-

stände repräsentieren, die zwischen dem Produktionsserver und den Clients gebildet werden, von denen die Client-Anforderungen empfangen werden.

[0048] Im Schritt **504** kann in Reaktion auf ein Empfangen der nachgebildeten Produktionslast in Echtzeit die nachgebildete Produktionslast über einen Prozessor verarbeitet werden, um eine Vielzahl von nachgebildeten Client-Anforderungen zu erzeugen. Jede der nachgebildeten Client-Anforderungen kann eine entsprechende Client-Anforderung nachbilden, die durch den Produktionsserver empfangen wurde. Wenigstens ein Teil der nachgebildeten Client-Anforderungen kann des Weiteren die Zustandsinformationen nachbilden, die dem entsprechenden Client zugehörig sind, von dem die Client-Anforderung empfangen wird. Mit anderen Worten, einige der nachgebildeten Client-Anforderungen können entsprechende Zustandsinformationen nachbilden oder sämtliche nachgebildeten Client-Anforderungen können entsprechende Zustandsinformationen nachbilden.

[0049] Im Schritt **506** können für eine Vielzahl von Client-Anforderungen entsprechende Produktions-Socket-Verbindungen entsprechenden Test-Socket-Verbindungen zugeordnet werden, die den entsprechenden nachgebildeten Client-Anforderungen zugehörig sind. Im Schritt **508** können die nachgebildeten Client-Anforderungen an den Testserver in Echtzeit übertragen werden.

[0050] **Fig. 6** ist ein Ablaufplan, der ein Verfahren zum Verarbeiten einer nachgebildeten Client-Anforderung in einem Testserver gemäß einer weiteren Ausführungsform der vorliegenden Erfindung veranschaulicht. Das Verfahren **600** kann wiederum durch den Verteiler-Proxy **132** von **Fig. 1** umgesetzt werden.

[0051] Im Schritt **602** kann der Verteiler-Proxy ein Socket öffnen, um von einem Produktionsserver Nachrichten zu empfangen, die Komponenten einer nachgebildeten Produktionslast darstellen. Die Nachrichten können beispielsweise von einer prüfenden Einrichtung oder einem anderen geeigneten Programm/Dienstprogramm empfangen werden, die zum Erzeugen derartiger Nachrichten eingerichtet sind. Im Schritt **604** kann eine Nachricht von dem Socket gelesen (d. h. empfangen) werden. Im Schritt **606** kann die Nachricht verarbeitet werden, um an Anforderungsdaten und Zustandsinformationen eine Syntaxprüfung auszuführen.

[0052] Wenn an dem Entscheidungskasten **608** die Client-Anforderungsdaten eine Bindung fordern, kann im Schritt **610** eine Bindung einer Test-Socket-Verbindung mit dem Testserver erzeugt werden, und die Produktions-Socket-Verbindung kann der Test-Socket-Verbindung zugeordnet werden. Wenn die Client-Anforderungsdaten keine Bindung fordern,

kann der Prozess zum Entscheidungskasten **612** vorgehen. Wenn an dem Entscheidungskasten **612** die Client-Anforderungsdaten das Aufheben einer Bindung fordern, kann bei **614** das Aufheben der Bindung der Test-Socket-Verbindung mit dem Testserver umgesetzt werden, und das Zuordnen der Produktions-Socket-Verbindung zu der Test-Socket-Verbindung kann aufgehoben werden. Wenn an dem Entscheidungskasten **612** die Client-Anforderungsdaten kein Aufheben einer Bindung fordern, kann an dem Entscheidungskasten **616** ermittelt werden, ob die Nachricht ausgefiltert wird. Wenn das nicht der Fall ist, kann die Verbindungszuordnung verwendet werden, um die nachgebildete Client-Anforderung zum Testserver zu senden, und eine Antwort kann gelesen werden. Wenn die Nachricht gefiltert wird, kann die Nachricht im Schritt **620** als nicht unterstützt identifiziert werden, und somit muss die nachgebildete Client-Anforderung nicht zu dem Testserver gesendet werden.

[0053] Von den Schritten **610**, **614**, **618** und **620** kann der Prozess zum Schritt **604** zurückkehren, und eine weitere Nachricht kann von dem Socket gelesen werden.

[0054] Der Prozess kann solange wiederholt werden, bis eine Entscheidung getroffen wird, den Prozess anzuhalten.

[0055] Der Ablaufplan und die Blockschaltbilder in den Figuren veranschaulichen die Architektur, Funktionalität und Operation von möglichen Umsetzungen von Systemen, Verfahren und Computerprogrammen gemäß verschiedener Ausführungsformen der vorliegenden Erfindung. In diesem Zusammenhang kann jeder Block in dem Ablaufplan oder Blockschaltbildern ein Modul, Segment oder Abschnitt von Code repräsentieren, der einen oder mehrere ausführbare Befehle zum Umsetzen der spezifizierten logischen Funktion(en) aufweist. Es sollte außerdem angemerkt werden, dass bei einigen alternativen Umsetzungen die in dem Block angegebenen Funktionen nicht in der in den Figuren angegebenen Reihenfolge auftreten. Zum Beispiel können zwei Blöcke, die nacheinander gezeigt sind, tatsächlich im Wesentlichen gleichzeitig ausgeführt werden oder die Blöcke können gelegentlich in Abhängigkeit von der beteiligten Funktionalität in der umgekehrten Reihenfolge ausgeführt werden. Es wird außerdem angemerkt, dass jeder Block in den Blockschaltbildern und/oder Ablaufplan-Darstellungen und Kombinationen von Blöcken in den Blockschaltbildern und/oder Ablaufplan-Darstellung durch Systeme, die auf spezieller Hardware beruhen, die die spezifizierten Funktionen oder Wirkungen ausführen, oder Kombinationen aus spezieller Hardware und Computerbefehlen umgesetzt werden können.

[0056] Die hier verwendete Terminologie dient lediglich dem Zweck der Beschreibung bestimmter Ausführungsformen und ist nicht vorgesehen, die Erfindung einzuschränken. Es ist vorgesehen, dass die hier verwendeten Singularformen "ein" und "der/die/das" ebenso die Pluralformen einschließen, falls im Kontext nicht anders deutlich angegeben. Es ist ferner klar, dass die Ausdrücke "weist auf" und/oder "aufweisen", wenn sie in dieser Beschreibung verwendet werden, das Vorhandensein von angegebenen Merkmalen, Ganzzahlen, Schritten, Operationen, Elementen und/oder Komponenten spezifizieren, jedoch nicht das Vorhandensein oder die Hinzufügung von einem oder mehreren anderen Merkmalen, Ganzzahlen, Schritten, Operationen, Elementen, Komponenten und/oder Gruppen hiervon ausschließen.

[0057] Es ist vorgesehen, dass die entsprechenden Strukturen, Materialien, Wirkungen und Ersetzungen aller Mittel oder Schritte plus Funktionselementen in den nachfolgenden Ansprüchen jede Struktur, jedes Material oder jede Wirkung zum Ausführen der Funktion in Kombination mit anderen beanspruchten Elementen in der speziell beanspruchten Weise enthalten. Die Beschreibung der vorliegenden Erfindung wurde für Zwecke der Erläuterung und Beschreibung dargestellt, es ist jedoch nicht vorgesehen, dass sie in der beschriebenen Form für die Erfindung erschöpfend oder einschränkend ist. Viele Modifikationen und Variationen werden für einen Fachmann ersichtlich sein, ohne vom Umfang und Erfindungsgedanken der Erfindung abzuweichen. Die Ausführungsform wurde ausgewählt und beschrieben, um die Grundgedanken der Erfindung und die praktische Anwendung am besten zu erläutern und um andere Fachleute zu befähigen, die Erfindung zu verstehen, da verschiedene Ausführungsformen mit zahlreichen Modifikationen für die vorgesehene bestimmte Verwendung geeignet sind.

Patentansprüche

1. Verfahren zum Nachbilden einer Produktionslast eines Produktionsservers in einem Testserver, wobei das Verfahren aufweist:
Erzeugen der Produktionslast in dem Produktionsserver durch Verarbeiten von Client-Anforderungen, die von Clients empfangen werden;
während die Client-Anforderungen über einen Prozessor in Echtzeit verarbeitet werden, Nachbilden der Produktionslast, um eine nachgebildete Produktionslast zu erzeugen, die die Client-Anforderungen repräsentiert und Zustandsinformationen definiert, die eindeutige Zustände repräsentieren, die zwischen dem Produktionsserver und den entsprechenden Clients gebildet werden; und
Übertragen der nachgebildeten Produktionslast in Echtzeit, um die Produktionslast in dem Testserver nachzubilden.

2. Verfahren nach Anspruch 1, das ferner aufweist: Einrichten einer Prüf-Einrichtung, um die nachgebildete Produktionslast automatisch in Echtzeit zu erzeugen, während die Client-Anforderungen durch den Produktionsserver verarbeitet werden.

3. Verfahren nach Anspruch 1, das ferner aufweist: Identifizieren entsprechender Produktions-Socket-Verbindungen, über die eine Vielzahl der Client-Anforderungen von den entsprechenden Clients zu dem Produktionsserver übertragen werden; und Zuordnen der Produktions-Socket-Verbindungen zu der nachgebildeten Produktionslast.

4. Verfahren nach Anspruch 1, wobei die Zustandsinformationen Cookies angeben, die den Client-Anforderungen zugehörig sind, die auf der Grundlage einer vorherigen Nutzung des Produktionsservers variieren.

5. Verfahren nach Anspruch 1, wobei die Zustandsinformationen Page-Dienste angeben, die den entsprechenden Clients bereitgestellt werden.

6. Verfahren nach Anspruch 1, wobei die Zustandsinformationen Verbindungs-Steuerungselemente angeben, die den entsprechenden Client-Anforderungen zugehörig sind.

7. Verfahren nach Anspruch 1, wobei die Zustandsinformationen Bindungsinformationen angeben, die den entsprechenden Client-Anforderungen zugehörig sind.

8. Verfahren nach Anspruch 1, wobei die Produktionslast nachgebildet wird, ohne ein Verhalten des Produktionsservers in Bezug auf Dienste zu ändern, die durch den Produktionsserver für die Clients bereitgestellt werden.

9. Verfahren zum Nachbilden einer Produktionslast eines Produktionsservers in einem Testserver, wobei das Verfahren aufweist:

Empfangen einer nachgebildeten Produktionslast, die einer Produktionslast in dem Produktionsserver entspricht, die durch Verarbeiten von Client-Anforderungen erzeugt wird, wobei die nachgebildete Produktionslast in Echtzeit erzeugt wird, während die Client-Anforderungen durch den Produktionsserver verarbeitet werden, und Zustandsinformationen definiert, die eindeutige Zustände repräsentieren, die zwischen dem Produktionsserver und den Clients gebildet werden, von denen die Client-Anforderungen empfangen werden; in Reaktion auf ein Empfangen der nachgebildeten Produktionslast in Echtzeit über einen Prozessor Verarbeiten der nachgebildeten Produktionslast, um eine Vielzahl von nachgebildeten Client-Anforderungen zu erzeugen, wobei jede der nachgebildeten Client-Anforderungen eine entsprechende Client-Anforderung

nachbildet, die durch den Produktionsserver empfangen wird, und wenigstens ein Teil der nachgebildeten Client-Anforderungen die Zustandsinformationen definiert, die dem entsprechenden Client zugehörig sind, von dem die Client-Anforderung empfangen wird; und

Übertragen der nachgebildeten Client-Anforderungen in Echtzeit zu dem Testserver.

10. Verfahren nach Anspruch 9, wobei: die Client-Anforderungen von den Clients zu dem Produktionsserver über Produktions-Socket-Verbindungen übertragen werden, die zwischen den entsprechenden Clients und dem Produktionsserver aufgebaut werden; und

Verarbeiten der nachgebildeten Produktionslast, um die Vielzahl von nachgebildeten Client-Anforderungen zu erzeugen, für eine Vielzahl der Client-Anforderungen Zuordnen entsprechender Produktions-Socket-Verbindungen zu entsprechenden Test-Socket-Verbindungen aufweist, die den entsprechenden nachgebildeten Client-Anforderungen zugehörig sind.

11. Verfahren nach Anspruch 9, wobei die Zustandsinformationen Cookies angeben, die entsprechenden Clients zugehörig sind, die auf der Grundlage einer vorherigen Nutzung des Produktionsservers variieren.

12. Verfahren nach Anspruch 9, wobei die Zustandsinformationen Page-Dienste angeben, die den entsprechenden Clients bereitgestellt werden.

13. Verfahren nach Anspruch 9, wobei die Zustandsinformationen Verbindungssteuerungselemente angeben, die für die entsprechenden Clients zugeordnet sind.

14. Verfahren nach Anspruch 9, wobei die Zustandsinformationen Bindungsinformationen angeben, die den entsprechenden Client-Anforderungen zugehörig sind.

15. Verfahren nach Anspruch 9, das ferner aufweist:

Filtern der nachgebildeten Produktionslast, um zu verhindern, dass bestimmte Client-Anforderungen in dem Testserver nachgebildet werden.

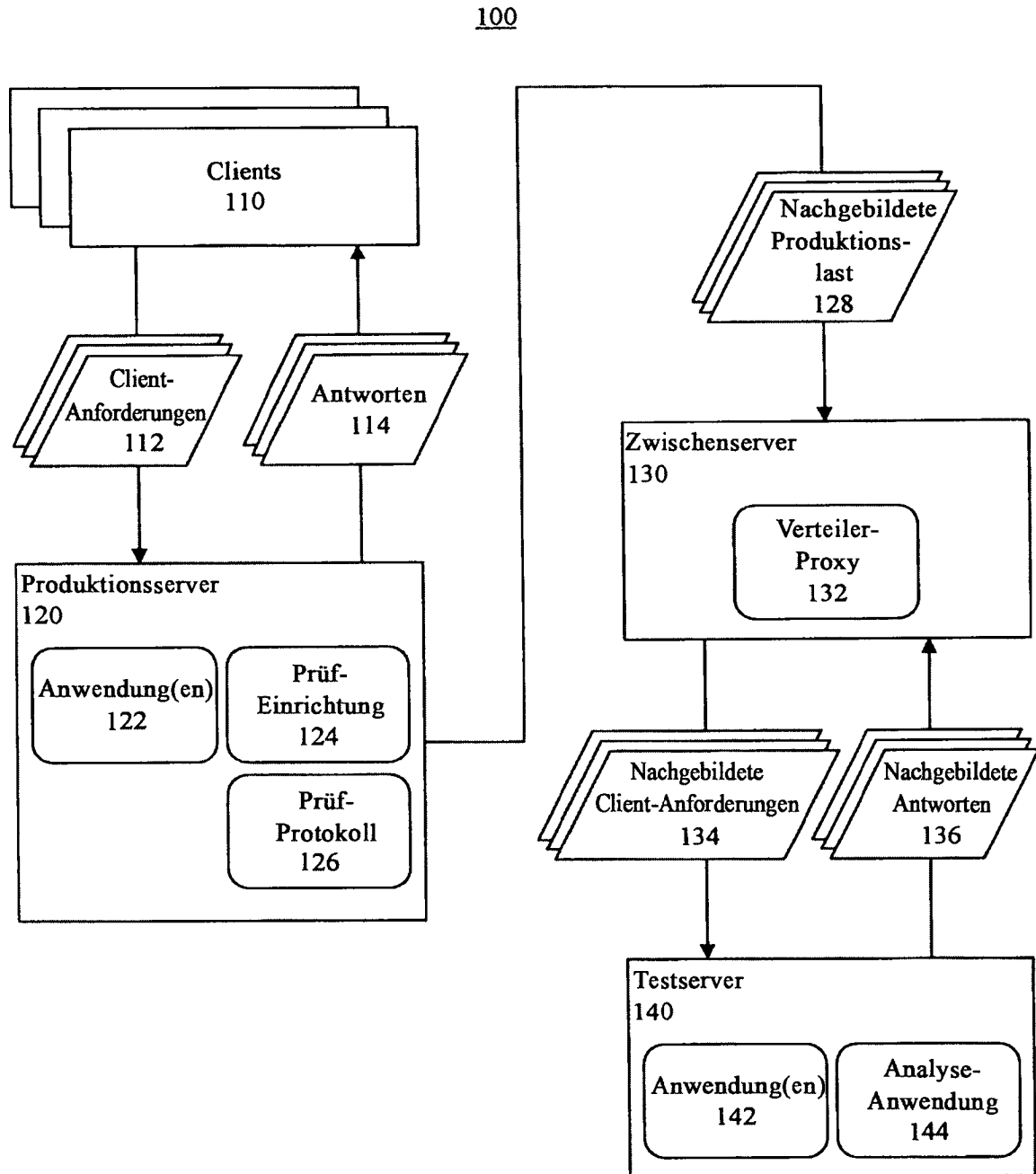
16. Computerprogrammprodukt zum Nachbilden einer Produktionslast eines Produktionsservers in einem Testserver, wobei das Computerprogrammprodukt aufweist:

ein computerlesbares Speichermedium mit computerlesbarem Programmcode, der darin verkörpert ist, der beim Ausführen durch einen Prozessor die Schritte nach einem der vorhergehenden Ansprüche ausführt.

17. System zum Nachbilden einer Produktionslast eines Produktionsservers in einem Testserver, das aufweist:
einen Prozessor und ein Computerprogrammprodukt nach Anspruch 16.

Es folgen 5 Seiten Zeichnungen

Anhängende Zeichnungen

**FIG. 1**

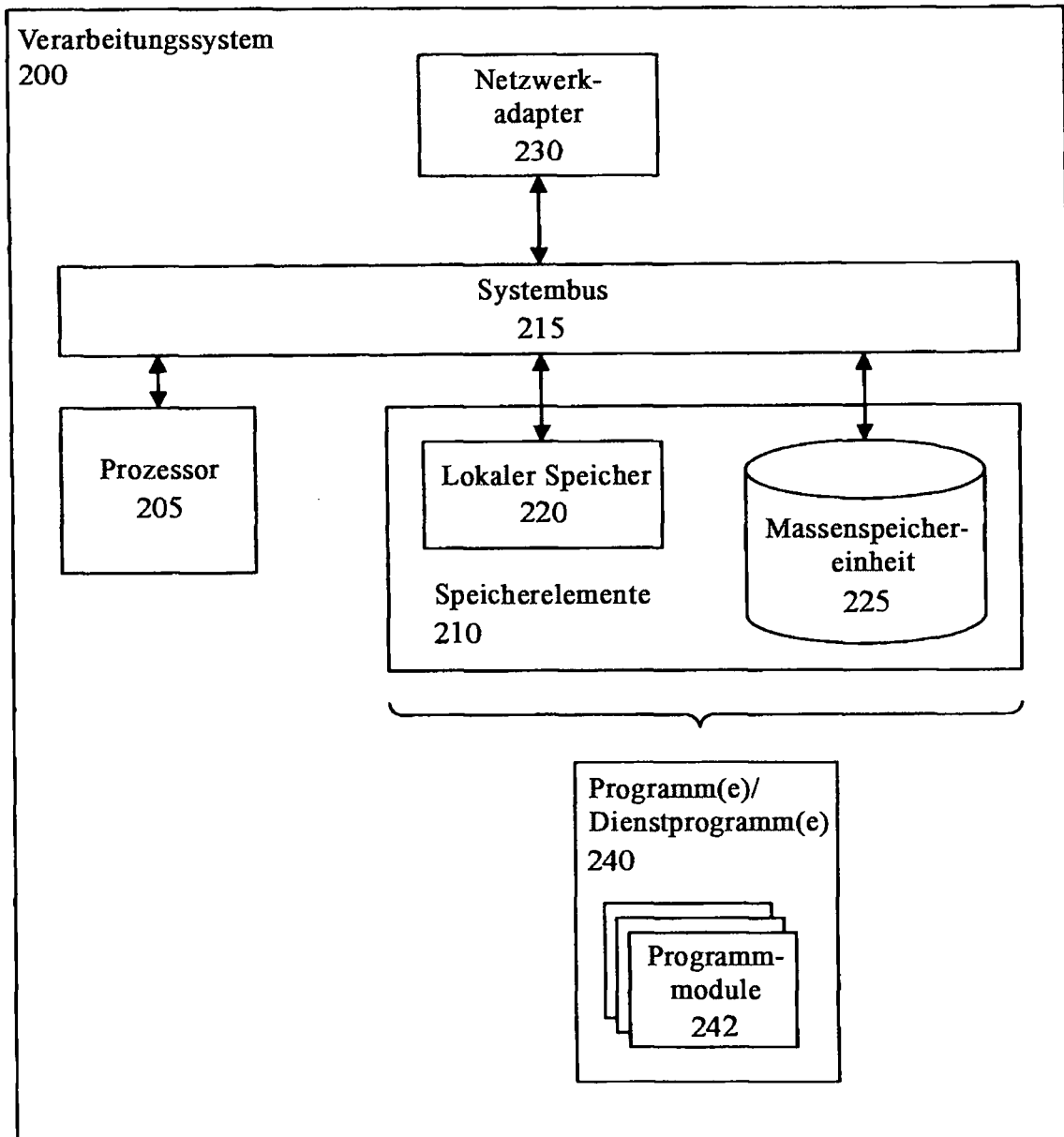
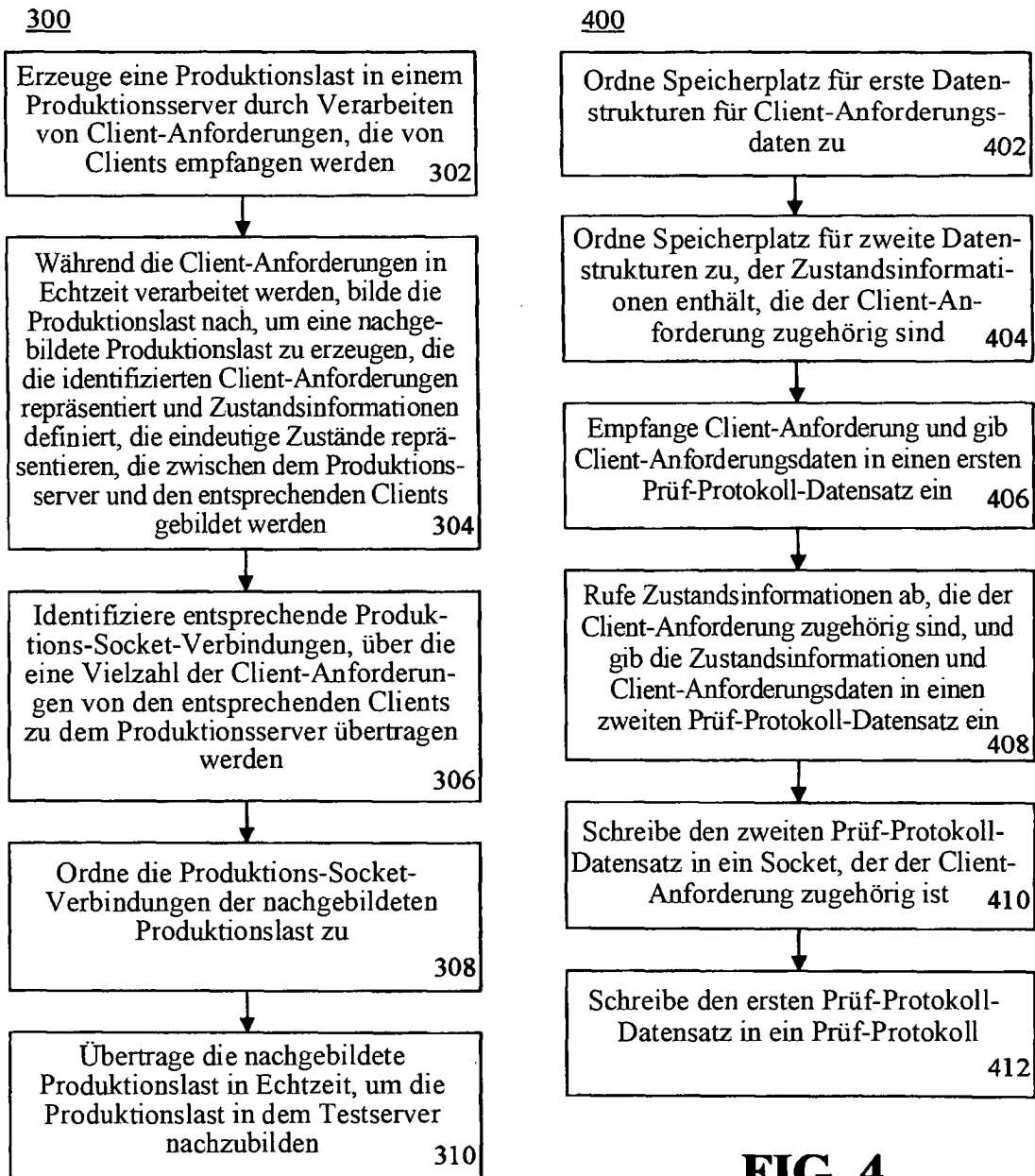


FIG. 2

**FIG. 3****FIG. 4**

500

Empfange eine nachgebildete Produktionslast, die einer Produktionslast in dem Produktionsserver entspricht, die durch Verarbeiten von Client-Anforderungen erzeugt wird, wobei die nachgebildete Produktionslast in Echtzeit erzeugt wird, während die Client-Anforderungen durch den Produktionsserver verarbeitet werden, und Zustandsinformationen definiert, die eindeutige Zustände darstellen, die zwischen dem Produktionsserver und den Clients gebildet werden, von denen die Client-Anforderungen empfangen werden

502

In Reaktion auf ein Empfangen der nachgebildeten Produktionslast in Echtzeit über einen Prozessor verarbeite die nachgebildete Produktionslast, um eine Vielzahl von nachgebildeten Client-Anforderungen zu erzeugen, wobei jede der nachgebildeten Client-Anforderungen eine entsprechende Client-Anforderung nachbildet, die durch den Produktionsserver empfangen wurde, und wenigstens ein Teil der nachgebildeten Client-Anforderungen die Zustandsinformationen definiert, die dem entsprechenden Client zugehörig sind, von dem die Client-Anforderung empfangen wurde

504

Ordne für eine Vielzahl von Client-Anforderungen entsprechende Produktions-Socket-Verbindungen entsprechenden Test-Socket-Verbindungen zu, die den entsprechenden nachgebildeten Client-Anforderungen zugehörig sind

506

Übertrage die nachgebildeten Client-Anforderungen in Echtzeit zu dem Testserver

508

FIG. 5

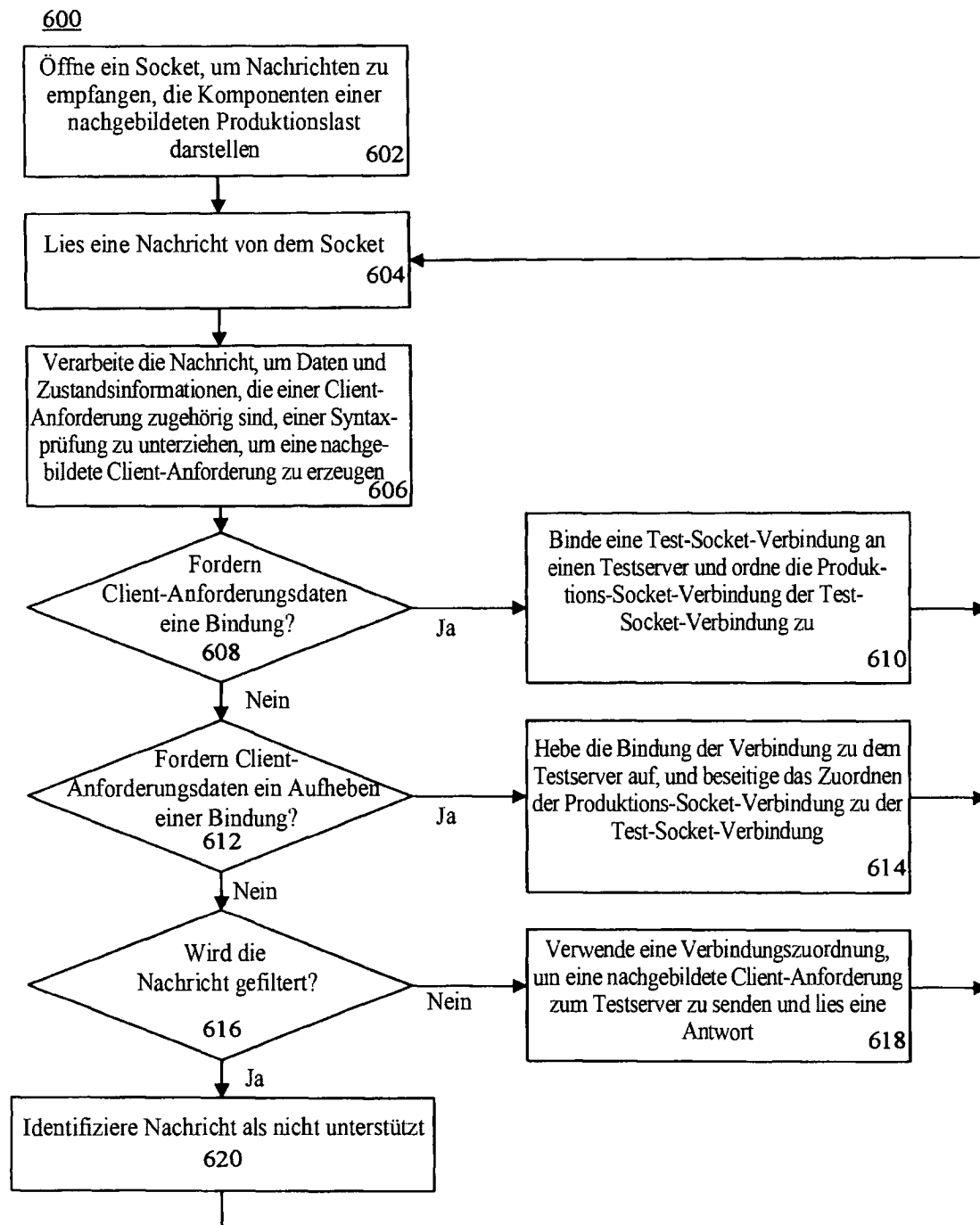


FIG. 6