



(12) **Gebrauchsmusterschrift**

(21) Aktenzeichen: **20 2010 000 226.5**  
(22) Anmeldetag: **19.02.2010**  
(47) Eintragungstag: **19.08.2010**  
(43) Bekanntmachung im Patentblatt: **23.09.2010**

(51) Int Cl.<sup>8</sup>: **G06F 9/46** (2006.01)  
**G06F 9/52** (2006.01)  
**G06F 9/445** (2006.01)

(30) Unionspriorität:  
**61/153,779**      **19.02.2009**      **US**  
**12/706,392**      **16.02.2010**      **US**

(73) Name und Wohnsitz des Inhabers:  
**American Megatrends, Inc., Norcross, Ga., US**

(74) Name und Wohnsitz des Vertreters:  
**Grünecker, Kinkeldey, Stockmair &  
Schwanhäusser, 80802 München**

(56) Recherchenergebnisse nach § 7 Abs. 2 GebrMG:  
**US 2003/00 61 497 A1**

**Computex: BIOS mit graphischer Oberfläche und  
Internet-Zugang.heise online,  
<http://heise.de/newsticker/meldung/print/27943>  
,recherchiert am **23.07.2004** Abs. 1-3**

**BUGNION,E.,et.al.: Disco: Running Commodity  
Operating Systems on Scalable  
Multiprocessors. In: ACM Transactions on  
Computer Systems, Vol.15,Nov.1997, S.412-447  
S.415,416,Abschn.3,Abs.1,2,4,5**

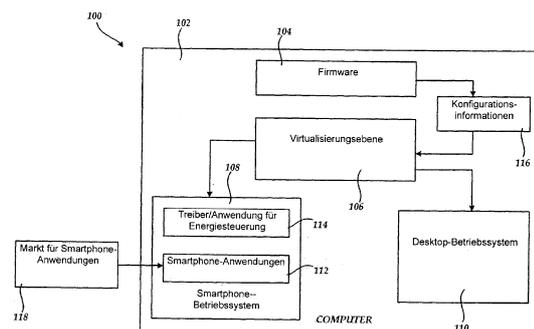
**Google groups, Android Developers: Running  
Android Ulnatively on desktop, not in  
emulator?, luke, Jan.31,2008,3.51 pm.  
[http://groups.google.com/group/android-devel-ops/browse\\_thread/tread/8b74ca45b6a14e9e/f698da3eac2e43e7](http://groups.google.com/group/android-devel-ops/browse_thread/tread/8b74ca45b6a14e9e/f698da3eac2e43e7),rech. am **14.04.10****

**Google groups, android-porting: Android x86?  
markgross,Nov.14,2008,2:04 am.  
[http://groups.google.com/group/android-portin-g/browse\\_thread/thread/ab553116dbc960da/5764eb1ad8ba7571](http://groups.google.com/group/android-portin-g/browse_thread/thread/ab553116dbc960da/5764eb1ad8ba7571),recherchiert am **14.04.10****

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

(54) Bezeichnung: **Gleichzeitiges Ausführen eines Smartphone-Betriebssystems und eines Desktop-Betriebssystems**

(57) Hauptanspruch: Recheneinrichtung, die kein Smartphone ist, und die umfasst:  
eine Zentraleinheit (CPU);  
einen RAM-Speicher (RAM); und  
einen nicht-flüchtigen Speicher, in dem ein Smartphone-Betriebssystem und eine Firmware gespeichert sind, die von der CPU ausgeführt werden können, wobei die Firmware Anweisungen umfasst, die beim Ausführen von der CPU die CPU veranlassen, zu ermitteln, ob die Einrichtung – die kein Smartphone ist – nur unter Verwendung eines Desktop-Betriebssystems, nur unter Verwendung eines Smartphone-Betriebssystems, oder unter gleichzeitiger Verwendung des Smartphone-Betriebssystems und des Desktop-Betriebssystems gebootet werden soll,  
und wobei als Reaktion auf die Feststellung, dass die Einrichtung – die kein Smartphone ist – nur unter Verwendung des Smartphone-Betriebssystems, oder unter gleichzeitiger Verwendung des Smartphone-Betriebssystems und des Desktop-Betriebssystems gebootet werden soll, veranlasst wird, dass eine Virtualisierungsebene auf der Einrichtung – die kein Smartphone ist – ausgeführt wird, und wobei als Reaktion auf die Feststellung, dass die Einrichtung – die kein Smartphone ist...



## Beschreibung

### Querverweis auf verwandte Anmeldungen

**[0001]** Diese Anmeldung beansprucht die Priorität der vorläufigen US-Patentanmeldung Nr. 61/153 779 vom 19. Februar 2009, auf deren vollständigen Wortlaut hiermit ausdrücklich Bezug genommen wird.

### Hintergrund der Erfindung

**[0002]** Ein Smartphone ist ein drahtloses Mobiltelefon, das einige Funktionalitäten bietet, die denen eines Computers (PC) ähnlich sind. Beispielsweise umfassen einige Smartphones Funktionalitäten zum Senden und Empfangen von elektronischen Nachrichten („E-Mail-Nachrichten“), zum Surfen im World Wide Web („Internet“), und zum Führen eines elektronischen Kalenders. Manche Smartphones bieten auch Funktionalitäten zum Ausführen von anderen Anwendungen aus erster Quelle (z. B. diejenigen, die der Smartphone-Hersteller bereitstellt) und von Fremdherstellern.

**[0003]** Obwohl die derzeitige Generation von Smartphones wichtige Funktionalitäten bietet, weisen diese Geräte auch mehrere Nachteile auf. Beispielsweise ist es bei den derzeitigen Smartphones für den Benutzer häufig frustrierend, deren Eingabemechanismen zu benutzen. Beispielsweise sind die Tastaturen mancher Smartphones so klein, das es schwierig ist, darauf zu tippen. Andere Smartphones haben überhaupt keine Tastatur, sondern weisen stattdessen eine Benutzerschnittstelle per Touchscreen auf, deren Benutzung ebenfalls frustrierend sein kann. Ein weiteres Beispiel ist die geringe Bildschirmgröße der meisten Smartphones, die insofern eine Einschränkung darstellt, als nur eine begrenzte Informationsmenge angezeigt werden kann und die angezeigten Informationen für manche Benutzer sogar schwer zu entziffern sind.

**[0004]** Daher bezieht sich die Offenlegung dieser Erfindung auf die oben genannte und weitere Überlegungen.

### Zusammenfassung

**[0005]** Die folgende Offenlegung bezieht sich auf Konzepte und Technologien zum gleichzeitigen Ausführen eines Smartphone-Betriebssystems und eines Desktop-Betriebssystems auf einer Einrichtung bzw. einem Gerät, die bzw. das kein Smartphone ist. Zum Beispiel kann unter Verwendung der hier vorgestellten Technologien ein Smartphone-Betriebssystem mit einer anderen Hardware, wie etwa der eines standardmäßigen Notebooks, Netbooks, oder Desktop-Computers benutzt werden.

**[0006]** Beim Ausführen eines Smartphone-Betriebs-

systems auf einer Hardware, die kein Smartphone ist, auf die hier beschriebene Weise kann eine stabile Betriebsumgebung des Smartphone-Betriebssystems schnell für die Verwendung durch einen Computerbenutzer zur Verfügung gestellt werden. Das Smartphone-Betriebssystem kann viel schneller geladen werden; die Lebensdauer des Akkus ist länger als bei einem herkömmlichen Desktop-Betriebssystem; dabei bietet es dennoch signifikante Funktionalitäten, wie z. B. das Abspielen von Video- oder Audiodateien, das Lesen von E-Mail-Nachrichten, oder das Surfen im Internet. Nach einigen Ausführungsformen dieser Erfindung kann ein Benutzer Smartphone-Anwendungen des Herstellers und von Dritten Quellen unter dem Smartphone-Betriebssystem ausführen und gleichzeitig die Vorteile genießen, die eine Hardware, die kein Smartphone ist, bietet, wie z. B. eine größere Tastatur, ein größeres Display, normal große Eingabevorrichtungen für den Benutzer, und ähnliches.

**[0007]** Nach hier vorgestellten Aspekten dieser Erfindung werden auch Technologien bereitgestellt, die es dem Benutzer gestatten, die Hardware, die kein Smartphone ist, wie z. B. einen standardmäßiger Laptop oder einen PC, entweder in ein Smartphone-Betriebssystem oder in ein Desktop-Betriebssystem zu booten. In einer Ausführungsform kann das Smartphone-Betriebssystem gleichzeitig mit dem Desktop-Betriebssystem gebootet werden. Nach einer anderen Ausführungsform wird das Betriebssystem des Smartphones zuerst gebootet, um dem Benutzer zu gestatten, die Funktionalitäten zu nutzen, die das Smartphone-Betriebssystem bietet, während das Desktop-Betriebssystem bootet. Die Funktionalitäten, die das Smartphone-Betriebssystem bietet, können bereits genutzt werden, während das Desktop-Betriebssystem noch gebootet wird.

**[0008]** Nach den Ausführungsformen wird eine Virtualisierungsebene bereitgestellt, um das Smartphone-Betriebssystem und das Desktop-Betriebssystem in virtuellen Maschinen auszuführen. Wenn der Benutzer geeignete Schritte durchführt, wie z. B. die Auswahl einer vordefinierten Taste, Tastenkombination, oder die Betätigung eines Schalters, dann kann das Computersystem dynamisch zwischen dem Smartphone-Betriebssystem und dem Desktop-Betriebssystem hin- und herwechseln. Des Weiteren können Dateien und Ordner synchronisiert und/oder von dem Smartphone-Betriebssystem dem Desktop-Betriebssystem gemeinsam verwendet werden. Dadurch wird ein Zugriff auf diese Dateien unter dem jeweiligen Betriebssystem ermöglicht, auch wenn das jeweils andere Betriebssystem gerade nicht ausgeführt wird.

**[0009]** Es sollte einzusehen sein, dass der oben beschriebene Gegenstand auch als Computersystem, als computergesteuertes Gerät, als Computerpro-

zess, oder als Erzeugnis, wie z. B. als maschinenlesbares Medium umgesetzt werden kann. Diese und verschiedene andere Eigenschaften werden beim Durchlesen der nachfolgenden genauen Beschreibung und unter Bezugnahme auf die dazugehörigen Zeichnungen ersichtlich.

**[0010]** Diese Zusammenfassung dient zur Einführung in eine Auswahl der Konzepte in vereinfachter Form, die nachfolgend in der genauen Beschreibung näher beschrieben werden. Diese Zusammenfassung ist weder dafür vorgesehen, die Hauptmerkmale oder wichtige Merkmale des beanspruchten Gegenstands zu identifizieren, noch ist beabsichtigt, dass diese Zusammenfassung dafür verwendet wird, den Schutzzumfang des beanspruchten Gegenstands einzuschränken. Des Weiteren ist der beanspruchte Gegenstand nicht auf Umsetzungen beschränkt, die einen, mehrere, oder alle Probleme lösen, die im Laufe dieser Offenlegung erwähnt werden.

#### Kurze Beschreibung der Zeichnungen

**[0011]** [Fig. 1](#) ist ein Diagramm einer Architektur für eine Computersoftware und -hardware und zeigt eine anschauliche Betriebskonfiguration zusammen mit mehreren Softwarekomponenten, die nach hier offen gelegten Ausführungsformen bereitgestellt werden;

**[0012]** [Fig. 2](#) ist ein Blockdiagramm und zeigt Aspekte einer hier offen gelegten Anwendung zum Synchronisieren von Daten zwischen einem Smartphone-Betriebssystem und einem Desktop-Betriebssystem;

**[0013]** [Fig. 3](#) ist ein Flussdiagramm und zeigt einen hier vorgestellten anschaulichen Prozess zum Ausführen eines Smartphone-Betriebssystems und eines Desktop-Betriebssystems;

**[0014]** [Fig. 4](#) ist ein Diagramm einer Computer-Architektur und zeigt eine anschauliche Computer-Architektur, die zum Verwirklichen eines Computersystems verwendet werden könnte, das die verschiedenen hier vorgestellten Konzepte und Technologien umfasst.

#### Genaue Beschreibung

**[0015]** Die folgende genaue Beschreibung bezieht sich auf Technologien zum Ausführen eines Smartphone-Betriebssystems und eines Desktop-Betriebssystems auf einer Hardware, die kein Smartphone ist. Der hier beschriebene Gegenstand wird zwar im allgemeinen Kontext von Programmbausteinen vorgestellt, die in Verbindung mit der Ausführung eines Betriebssystems und von Anwenderprogrammen in einem Computersystem ausgeführt werden; der Kenner der Technik wird jedoch erkennen, dass auch andere Umsetzungen in Kombination mit anderen Ty-

pen von Programmbausteinen möglich sind.

**[0016]** Im Allgemeinen umfassen Programmbausteine Routinen, Programme, Komponenten, Datenstrukturen und andere Strukturtypen, die bestimmte Aufgaben ausführen oder bestimmte abstrakte Datentypen implementieren. Des Weiteren wird es der Kenner der Technik zu schätzen wissen, dass der hier beschriebene Gegenstand mit anderen Computersystemkonfigurationen in die Praxis umgesetzt werden kann, einschließlich Handgeräte, Mehrprozessorsysteme, mit Mikroprozessoren ausgerüstete oder programmierbare Geräte aus der Unterhaltungselektronik, Minicomputer, Großrechner, und ähnliches.

**[0017]** In der folgenden Beschreibung wird auf die beigefügten Zeichnungen Bezug genommen, die Teil dieser Beschreibung sind, und in denen zur Veranschaulichung spezifische Ausführungsformen oder Beispiele dargestellt sind. Unter Bezugnahme auf die Zeichnungen werden nun Aspekte eines Computersystems und einer Methodik für das gleichzeitige Ausführen eines Smartphone-Betriebssystems eines Desktop-Betriebssystems auf einer Einrichtung bzw. einem Gerät, die bzw. das kein Smartphone ist, beschrieben. Hierbei werden in allen Zeichnungen für die jeweils beschriebenen Bestandteile die gleichen Bezugszeichen verwendet.

**[0018]** Zunächst werden mit Bezug auf [Fig. 1](#) eine anschauliche Betriebskonfiguration und mehrere Softwarekomponenten genauer beschrieben, die mit den hier vorgestellten Ausführungsformen bereitgestellt werden. [Fig. 1](#) zeigt insbesondere Aspekte eines Systems **100** zum Ausführen eines Smartphone-Betriebssystems **108** und eines Desktop-Betriebssystems **110** auf einem Gerät, das kein Smartphone ist, wie etwa auf dem Computer **102**. In einer Ausführungsform umfasst der Computer **102** ein standardmäßiges Laptop- oder Desktop-Computersystem, das ein Desktop-Betriebssystem **110** ausführen kann. Wie nachfolgend noch genauer beschrieben wird, ist der Computer **102** in hier offen gelegten Ausführungsformen auch so konfiguriert, dass er ein Smartphone-Betriebssystem **108** ausführen kann, obwohl das Smartphone-Betriebssystem **108** im Allgemeinen nicht für die Ausführung auf dem Computer **102** vorgesehen ist. Auf diese Weise wird der Computer **102** in die Lage versetzt, die Abläufe eines Smartphone nachzuahmen bzw. zu simulieren und behält dabei gleichzeitig die Fähigkeit, das Desktop-Betriebssystem **110** auszuführen.

**[0019]** Durch das Simulieren eines Smartphone bietet der Computer **102** viele der Funktionen eines Smartphones, das mit einer Smartphone-Software betrieben wird, einschließlich des Smartphone-Betriebssystems **108** und der Smartphone-Anwendungen **112**, die Anwendungen des ersten oder eines

ritten Herstellers sein können. Dies gestattet es dem Computer **102**, ein standardmäßiges Smartphone einschließlich eines Großteils der Funktionalitäten, die ein Smartphone bietet, einschließlich der Funktionalitäten der Anwendungen **112** zu simulieren. Diese Funktionalitäten werden zusätzlich zu den standardmäßigen Funktionalitäten des Computers **102** bei der Ausführung des Desktop-Betriebssystems **110** bereitgestellt

**[0020]** Wie nachfolgend noch genauer beschrieben wird, ist der Computer **102** in hier offen gelegten Ausführungsformen so konfiguriert, dass er gleichzeitig das Smartphone-Betriebssystem **108** und das Desktop-Betriebssystem **110** ausführen kann. Den Aspekten entsprechend kann ein Benutzer eines Computers **102** zwischen dem Betrieb unter dem Smartphone-Betriebssystem **108** und dem Betrieb unter dem Desktop-Betriebssystem **110** hin- und herwechseln. Zusätzlich können Ordner und Dateien sowohl vom Smartphone-Betriebssystem **108** und vom Desktop-Betriebssystem **110** gemeinsam genutzt werden, auch wenn eines der Betriebssysteme gerade nicht ausgeführt wird. Weitere Einzelheiten in Bezug auf diese Merkmale des Computers **102** nach dieser Erfindung werden nachfolgend beschrieben.

**[0021]** Wie oben kurz beschrieben, umfasst der Computer **102** ein standardmäßiges Notebook-, Laptop- oder Desktop-Computersystem, das ein Desktop-Betriebssystem **110** ausführen kann. Es sollte jedoch einzusehen sein, dass der Computer **100** andere Arten von Recheneinrichtungen umfassen kann, einschließlich eines Multimedia-Internetgeräts („MID“ – Multimedia Internet Device), eines Netbooks, oder eines anderen Typs einer Recheneinrichtung im Kleinformat der bzw. die kein Smartphone ist. Es sollte ebenfalls einzusehen sein, dass das Desktop-Betriebssystem **110** jedes Betriebssystem sein kann, das für die Bereitstellung der vollständigen Funktionalität eines Desktop- oder Laptop-Computersystems konfiguriert ist. Beispielsweise kann das Desktop-Betriebssystem **110** je nach Ausführungsform eines der Betriebssysteme der WINDOWS-Familie der MICROSOFT CORPORATION, das Betriebssystem OSX von APPLE COMPUTER, INC., das Betriebssystem LINUX, oder ein anderer Typ eines Desktop-Betriebssystems nach dem Stand der Technik sein. In dieser Hinsicht bietet der Computer **102** die Funktionalitäten zum Booten und Ausführen des Desktop-Betriebssystems **110** auf herkömmliche Weise.

**[0022]** Wie oben ebenfalls kurz beschrieben, bietet der nach dieser Erfindung bereitgestellte Computer **102** auch die Funktionalität zum Ausführen eines Smartphone-Betriebssystems **108**. Das Smartphone-Betriebssystem **108** umfasst ein Betriebssystem, das zum Ausführen auf einem Smartphone konfiguriert ist. Wie oben kurz beschrieben, ist ein Smartpho-

ne ein drahtloses Mobiltelefon, das hochentwickelte Fähigkeiten aufweist, die über diejenigen eines typischen Mobiltelefons hinausgehen, häufig mit computerähnlichen Funktionalitäten. Die Funktionalitäten mancher Smartphones umfassen das Senden von E-Mail-Nachrichten, das Surfen im Internet, das Abspielen von Audio- und Videodateien, das Spielen von Computerspielen und die Nutzung anderer Funktionen. Ein Gerät, das kein Smartphone ist, ist eine Recheneinrichtung – aber eben kein Smartphone – die in der Lage ist, die hier offen gelegte Funktionalität auszuführen.

**[0023]** Smartphones besitzen typischerweise relative kleine Bildschirme und können eine Reihe von Eingabegeräten aufweisen, wie etwa eine kleine Tastatur und/oder einen Touchscreen. Das Smartphone-Betriebssystem **108** kann das Betriebssystem SYMBIAN OS von SYMBIAN LIMITED umfassen, das Betriebssystem IPHONE OS von APPLE INC., das Betriebssystem RIM BLACKBERRY von RESEARCH IN MOTION, INC., das Betriebssystem WINDOWS MOBILE von MICROSOFT, die von PALM, INC. entwickelten Betriebssysteme PALM OS oder PALM PRE WEBOS, oder das Smartphone-Betriebssystem ANDROID von GOOGLE, INC. Andere Betriebssysteme für Smartphones können ebenfalls auf die hier vorgestellte Weise zum Einsatz kommen.

**[0024]** Zum Ausführen des Smartphone-Betriebssystems **108** auf dem Computer **102** wird in einer Ausführungsform eine Virtualisierungsebene **106** verwendet. Wie nach dem Stand der Technik bekannt, kann die Virtualisierung dafür verwendet werden, einen bestimmten Typ einer virtuellen Maschinenumgebung zu implementieren, der die Simulation der zugrunde liegenden Hardware gestattet. Auf diese Weise kann das Betriebssystem des Smartphones durch die Virtualisierungsebene **106** ausgeführt werden, obwohl das Smartphone-Betriebssystem **108** nicht speziell zum Ausführen auf der Hardware des Computers **102** geschrieben wurde.

**[0025]** Nach den Ausführungsformen kann eine selektive Virtualisierung durch die Virtualisierungsebene **106** genutzt werden, um nur die Teile der Hardware zu virtualisieren, die von dem Smartphone-Betriebssystem **108** benötigt werden. Beispielsweise wird das Smartphone-Betriebssystem **108** bei einer Anwendung in einem nicht-flüchtigen Speicher des Computers **102** gespeichert. In diesem Beispiel wäre es nicht erforderlich, ein mit dem Computer **102** verbundenes Festplattenlaufwerk zu virtualisieren, da das Smartphone-Betriebssystem **108** die vom Festplattenlaufwerk bereitgestellte Funktionalität nicht benötigt oder benutzt. Es sollte ebenfalls einzusehen sein, dass auch andere Hardwarekomponenten des Computers **102** auf ähnliche Weise selektiv virtualisiert werden könnten.

**[0026]** Es sollte einzusehen sein, dass die die Virtualisierungsebene **106** auf eifern Massenspeicher des Computers **102** gespeichert werden kann und von dort aus ausgeführt werden kann. Alternativ dazu kann die Virtualisierungsebene **106** innerhalb der Firmware **104** des Computers **102** implementiert und in einem nicht-flüchtigen Speicher des Computers **102** gespeichert werden. Beispielsweise wird die Virtualisierungsebene **106** in einer Anwendung als Teil einer BIOS- oder Extensible-Firmware-Interface- („EFI“)-Firmware des Computers **102** für das ausgeführt. Kenner der Technik werden erkennen, welche anderen Anwendungen ferner möglich sind.

**[0027]** Nach einer Ausführungsform wird ein Benutzer des Computers **102** aufgefordert, Konfigurationsinformationen **116** einzugeben, die angeben, wie der Computer **102** gebootet werden soll. Nach den Aspekten der Erfindung kann der Computer **102** auf herkömmliche Weise direkt in das Desktop-Betriebssystem **110** gebootet werden. Alternativ dazu kann der Computer **102** durch die Verwendung der oben beschriebenen Virtualisierungsebene **106** direkt in das Smartphone-Betriebssystem **108** gebootet werden. Auf diese Weise kann ein Benutzer des Computers **102** spezifizieren, welches der Betriebssysteme **108** oder **110** zum Booten des Computers **102** verwendet werden soll.

**[0028]** Nach einer anderen Ausführungsform kann ein Benutzer des Computers **102** spezifizieren, dass das Smartphone-Betriebssystem **108** und das Desktop-Betriebssystem **110** zur gleichen Zeit gebootet werden sollen. Zum Durchführen dieses Prozesses wird die Virtualisierungsebene **106** genutzt, um sowohl das Smartphone-Betriebssystem **108** als auch das Desktop-Betriebssystem **110** in virtuellen Maschinen auszuführen. Auf diese Weise können beide Betriebssysteme, **108** und **110**, gleichzeitig auf dem Computer **102** ausgeführt werden. In einer Ausführungsform kann eine vordefinierte Tastenkombination oder eine andere Form der Eingabe durch den Benutzer vom Computer **102** empfangen werden, um im Display des Computers **102** vom Videoausgang des Smartphone-Betriebssystems **108** zum Videoausgang des Desktop-Betriebssystems **110** zu wechseln und umgekehrt. Auf diese Weise kann ein Benutzer des Computers **102** wechselweise beide Betriebssysteme, **108** und **110**, gleichzeitig verwenden.

**[0029]** Zum Empfangen der Konfigurationsinformationen **116** von einem Benutzer kann eine Firmware **104** des Computers **102** ein Menü auf der Benutzeroberfläche oder einen anderen Typ einer Steuerung über die Benutzeroberfläche anzeigen, über die ein Benutzer des Computers **102** spezifizieren kann, auf welche Weise der Computer **102** gebootet werden soll. Wenn der Benutzer sich entscheidet, nur das Desktop-Betriebssystem **110** zu starten, dann bootet der Computer **102** das Desktop-Betriebssystem **110**

auf herkömmliche Weise. Wenn ein Benutzer des Computers **102** sich entscheidet, dass entweder das Smartphone-Betriebssystem **108** booten soll, oder sowohl das Smartphone-Betriebssystem **108** als auch das Desktop-Betriebssystem **110** booten soll, dann wird die Virtualisierungsebene **106** geladen. Die Virtualisierungsebene **106** wiederum lädt das Smartphone-Betriebssystem **108** und das Desktop-Betriebssystem **110**.

**[0030]** Nach den Ausführungsformen kann das Smartphone-Betriebssystem **108** geladen und einem Benutzer des Computers **102** zur Verfügung gestellt werden, bevor das Desktop-Betriebssystem **110** geladen wird. Da das Smartphone-Betriebssystem **108** viel kleiner ist als das Desktop-Betriebssystem **110**, kann der Computer **102** schneller booten; und dadurch ist es möglich, dass ein Benutzer des Computers **102** vom Smartphone-Betriebssystem **108** bereitgestellte Funktionen sogar schon nutzt, bevor das Desktop-Betriebssystem **110** vollständig geladen ist. In dieser Hinsicht kann die Virtualisierungsebene **106** so konfiguriert werden, dass der Status des Smartphone-Betriebssystems **108** als eine Virtual-Machine-(VM)-Image-Datei der virtuellen Maschine gespeichert wird und dass diese VM-Image-Datei beim nächsten Booten des Computers **102** geladen wird. Dieser Prozess reduziert auch die Zeit, die erforderlich ist, um das Smartphone-Betriebssystem **108** zu laden und das Smartphone-Betriebssystem **108** einem Benutzer des Computers **102** zur Verfügung zu stellen.

**[0031]** Es sollte einzusehen sein, dass das Smartphone-Betriebssystem **108** zusätzliche Funktionalitäten bietet, die in manchen der gebräuchlichen Betriebssysteme nicht verfügbar ist. Beispielsweise gestattet das Smartphone-Betriebssystem **108** einem Benutzer des Computers **102**, Smartphone-Anwendungen **112** zu installieren, die Anwendungen des ersten oder eines dritten Herstellers sein können. Smartphone-Anwendungen **112** sind Anwendungen, die zum Ausführen unter einem Smartphone-Betriebssystem **108** programmiert wurden.

**[0032]** Nach den Ausführungsformen kann der Zugang zu einem Markt für Smartphone-Anwendungen **118** über ein Fernnetzwerk („WAN“ – Wide Area Network) erfolgen, um Smartphone-Anwendungen **112** herunterzuladen und zu installieren. Die Smartphone-Anwendungen **112** können Anwendungen zum Senden und Empfangen von E-Mail-Nachrichten, zum Abspielen von Videodateien, zum Abspielen von Audiodateien, zum Surfen im Internet, zum Spielen von Computerspielen und für die Nutzung vieler anderer Funktionen enthalten. Nach den Ausführungsformen ist eine der Smartphone-Anwendungen **112** ein Programm für Voice Over Internet Protocol („VOIP“), das es einem Benutzer des Computers **102** ermöglicht, Telefongespräche als anrufender und als

angerufener Teilnehmer zu führen. Zusätzlich kann ein Benutzer des Smartphone-Betriebssystems **108**, wenn der Computer **102** eine dafür geeignete Hardware umfasst, die Möglichkeit haben, Telefongespräche als anrufender sowie als angerufener Teilnehmer auf herkömmliche Weise über ein drahtloses Telefonnetz oder über ein Funknetz zu führen.

**[0033]** Nach den Ausführungsformen werden ein spezieller Treiber für eine Energiesteuerung und ein Anwenderprogramm **114** mit dem Smartphone-Betriebssystem **108** ausgeführt. Da das Smartphone-Betriebssystem **108** für eine andere Hardwarekonfiguration programmiert ist, als diejenige, des Computers **102**, kann das Smartphone-Betriebssystem **108** energiesparend betrieben werden, wenn es auf dem Computer **102** ausgeführt wird. Dementsprechend gestatten der Treiber für die Energiesteuerung und das Anwenderprogramm **114** dem Smartphone-Betriebssystem **108**, mit Energiesparfunktionen zusammenzuwirken, die von der Hardware des Computers **102** bereitgestellt werden, und ermöglichen es demzufolge, energiesparender zu arbeiten

**[0034]** Der Treiber für die Energiesteuerung bietet die Funktionalität zum Zusammenwirken mit der eigentlichen Hardware des Computers **102**, um den Stromverbrauch zu reduzieren, während das Smartphone-Betriebssystem **108** ausgeführt wird. Die Anwendung der Energiesteuerung gestattet es einem Benutzer des Computers **102**, Einstellungsoptionen zu wählen, die sich auf die Stromsparschaltung des Smartphone-Betriebssystems **108** beziehen. Auf diese Weise kann der Computer **102** so konfiguriert werden, dass er beim Ausführen des Smartphone-Betriebssystems **108** weniger Strom verbraucht, als beim Ausführen des Desktop-Betriebssystems **110**. Daraus ergibt sich, dass ein Benutzer den Computer **102** dazu nutzen kann, das Smartphone-Betriebssystem **108** auszuführen und gemeinsame Funktionen über einen längeren Zeitraum zu nutzen, als beim Ausführen des Desktop-Betriebssystems **110**.

**[0035]** Es wird einzusehen sein, dass das Smartphone-Betriebssystem **108** im Allgemeinen zum Ausführen auf einem Smartphone konfiguriert ist, das typischerweise ein sehr kleines Display besitzt. Unter Verwendung der hier vorgestellten Ausführungsformen kann ein Smartphone-Betriebssystem **108** für einen größeren Bildschirm genutzt werden, wie bei einem Laptop- oder einem Desktop-Computer. Des Weiteren kann das Smartphone-Betriebssystem **108** bei der Verwendung auf einem Computer mit einem kleineren Bildschirm, wie etwa bei einem Netbook, eine bessere Anzeige für die Verwendung auf einem kleinen Bildschirm liefern, als das Desktop-Betriebssystem **110**. Dies kann nützliche Vorteile gegenüber der Ausführung des Desktop-Betriebssystems **110** bieten, da dieses im Allgemeinen für die Verwendung eines größeren Bildschirm vorgesehen ist.

**[0036]** Es wird ebenso einzusehen sein, dass der Computer **102** beim Ausführen des Smartphone-Betriebssystems **108** auf die gleiche Weise wie ein typisches Smartphone betrieben werden kann. Auf diese Weise kann ein Benutzer des Smartphone-Betriebssystems **108** die Smartphone-Anwendungen **112** nutzen, um im Internet zu surfen und auf Daten wie etwa Kontakte, Kalender, sowie andere Datentypen zuzugreifen. Dieser Prozess gestattet den Smartphone-Anwendungen **112** den Zugriff auf größere Massenspeicher, die der Computer **102** für Archivierungszwecke bereitstellt.

**[0037]** Es wird einzusehen sein, dass der Betrieb der Virtualisierungsebene **106** für einen Benutzer des Computers **102** unsichtbar ist, obwohl die Virtualisierungsebene **106** die Funktionalität zum gleichzeitigen Ausführen des Smartphone-Betriebssystems **108** und des Desktop-Betriebssystems **110** bereitstellt. Insbesondere wird ein Benutzer des Computers **102** nicht bemerken, dass die Virtualisierungsebene **106** vorhanden ist und auf dem Computer **102** ausgeführt wird. Auf diese Weise kann ein Benutzer durch eine Eingabe über die Tastatur oder einen anderen geeigneten Mechanismus über eine Benutzerschnittstelle nahtlos zwischen dem Smartphone-Betriebssystem **108** und dem Desktop-Betriebssystem **110** hin- und herwechseln. Wie oben kurz besprochen, kann die Virtualisierungsebene **106** in einem nicht-flüchtigen Speicher des Computers **102** gespeichert werden. Auf ähnliche Weise kann das Smartphone-Betriebssystem **108** ebenfalls in einem nicht-flüchtigen Speicher des Computers **102** gespeichert und vor dort aus ausgeführt werden. Alternativ dazu kann das Smartphone-Betriebssystem **108** auf einem Massenspeicher des Computers **102**, wie etwa auf einem Festplattenlaufwerk, gespeichert und vor dort aus ausgeführt werden.

**[0038]** Unter Bezugnahme auf [Fig. 2](#) wird nun ein Aspekt der hier offen gelegten Erfindung zum gemeinsamen Zugriff auf Ordner und Dateien unter dem Smartphone-Betriebssystem **108** und dem Desktop-Betriebssystem **110** beschrieben. Nach einer Ausführungsform wird der Computer **102** beim ersten Booten in das Desktop-Betriebssystem **110** gebootet. Beim Booten des Computers **102** in das Desktop-Betriebssystem **110** wird ein Synchronisationsagent **202B** in das Desktop-Betriebssystem **110** installiert.

**[0039]** Der Synchronisationsagent **202B** gestattet es einem Benutzer des Computers **102** zu spezifizieren, welche der unter dem Desktop-Betriebssystem **110** zugänglichen Ordner oder Dateien in einen gemeinsam genutzten Speicherort **204** synchronisiert werden sollen. In dem in [Fig. 2](#) dargestellten Beispiel wurden die Dateien **206** als Dateien identifiziert, die von dem Desktop-Betriebssystem **110** und dem Smartphone-Betriebssystem **108** gemeinsam ge-

nutzt werden sollen. Diese Dateien werden dann von dem Synchronisationsagenten **202B** in den gemeinsam genutzten Speicherort **204** kopiert. Nach den Ausführungsformen kann der Synchronisationsagent **202B** Ordner oder andere Speicherorte auf Änderungen überwachen und die Änderungen in den gemeinsam genutzten Speicherort **204** synchronisieren. Nach den Ausführungsformen kann der gemeinsam genutzte Speicherort **204** einen Speicherort in einem Massenspeicher umfassen, wie etwa ein Festplattenlaufwerk, oder in einer nicht-flüchtigen Speichervorrichtung. Die Vorrichtung, in der der gemeinsam genutzte Speicherort **204** gespeichert wird, wird zur Nutzung durch das Smartphone-Betriebssystem **108** virtualisiert.

[0040] Wie in [Fig. 2](#) ebenfalls dargestellt, weist die Konfiguration des Smartphone-Betriebssystems **108** in einer Ausführungsform auch einen Synchronisationsagenten **202A** auf. Der Synchronisationsagent **202A** kann ebenfalls auf ähnliche Weise so konfiguriert werden, dass er Dateien aus dem Smartphone-Betriebssystem **108** in den gemeinsam genutzten Speicherort **204** synchronisiert. In dem in [Fig. 2](#) dargestellten Beispiel wurden die durch den Synchronisationsagenten **202A** in den gemeinsam genutzten Speicherort **204** kopierten Dateien **206** vom Synchronisationsagenten **202A** in das Smartphone-Betriebssystem **108** kopiert.

[0041] Wie in [Fig. 2](#) dargestellt, wird eine 2-Wege-Synchronisation zwischen dem Smartphone-Betriebssystem **108** und dem Desktop-Betriebssystem **110** aktiviert. Auf diese Weise kann ein Benutzer des Computers **102** auf die Dateien **206** unter dem Smartphone-Betriebssystem **108** zugreifen, auch wenn das Desktop-Betriebssystem **110** gerade nicht ausgeführt wird. Auf ähnliche Weise kann ein Benutzer des Computers **102** unter dem Desktop-Betriebssystem **110** auf die Dateien **206** zugreifen, auch wenn das Smartphone-Betriebssystem **108** gerade nicht ausgeführt wird.

[0042] Unter Bezugnahme auf [Fig. 3](#) folgen nun weitere Einzelheiten der hier vorgestellten Ausführungsformen eines Computersystems, das in der Lage ist, ein Smartphone-Betriebssystem und ein Desktop-Betriebssystem auszuführen. Insbesondere ist [Fig. 3](#) ein Flussdiagramm und zeigt Aspekte beim Betrieb eines hier vorgestellten Prozesses **300** zum Booten eines Computers **102**, der in der Lage ist, gleichzeitig ein Smartphone-Betriebssystem **108** und ein Desktop-Betriebssystem **110** auf die hier offengelegte Weise auszuführen.

[0043] Es wird einzusehen sein, dass die hier beschriebenen logischen Operationen (1) als eine Sequenz der von einem Computer implementierten Schritte oder der auf einem Computer laufenden Programmmodule implementiert werden, und/oder (2)

als miteinander verbundene Geräte-Schaltkreise oder Schaltkreismodule innerhalb des Computersystems. Die Anwendung kann frei gewählt werden und hängt von der Leistung und anderen Anforderungen des Computersystems ab. Dementsprechend wird auf die hier beschriebenen logischen Operationen unterschiedlich Bezug genommen, die Zustände, Operationen, Bausteine, Vorgänge oder Module sein können. Diese Operationen, Bausteine, Vorgänge und Module können als Software, Firmware, digitale Spezial-Logik, und jegliche Kombination daraus implementiert sein. Es sollte ebenfalls einzusehen sein, dass auch weniger oder mehr Operationen durchgeführt werden können, als in den Figuren gezeigt und hier beschrieben ist. Diese Operationen können auch in einer anderen Reihenfolge als hier beschrieben ausgeführt werden.

[0044] Die Routine **300** beginnt bei der Operation **302**, bei der die Firmware **104** des Computers **102** ermittelt, ob der Computer **102** zum ersten Mal gebootet wird. Wenn das der Fall ist, veranlasst die Firmware den Computer **102**, das Desktop-Betriebssystem **110** ohne Verwendung der Virtualisierungsebene **106** bei der Operation **304** zu booten. Sobald der Computers **102** unter Verwendung des Desktop-Betriebssystems **110** gebootet wurde, wird ein Synchronisationsagent **202B** in das Desktop-Betriebssystem **110** installiert. Dies erfolgt bei Operation **306**. Sobald der Synchronisationsagent **202B** installiert ist, läuft die Routine **300** weiter von Operation **306** bis Operation **302**. In einer alternativen Ausführungsform wird der Synchronisationsagent **202B** in das Desktop-Betriebssystem **110** installiert, wenn es zuvor noch nicht darin installiert war, unabhängig davon, wie oft das Desktop-Betriebssystem **110** zuvor gebootet worden war.

[0045] Wenn die Firmware **104** des Computers **102** bei Operation **302** feststellt, dass der Computer **102** nicht zum ersten Mal gebootet wird, läuft die Routine **300** von Operation **302** weiter zu Operation **308**. Bei Operation **308** erhält der Computer **102** die Konfigurationsinformationen **116** vom Benutzer des Computers **102**. Wie oben besprochen, kann die Firmware **104** des Computers **102** so konfiguriert werden, dass der Benutzer des Computers **102** aufgefordert wird, die Konfigurationsinformationen **116** einzugeben. Wie ebenfalls oben besprochen, sind die Konfigurationsinformationen **116** Daten, die anzeigen, ob der Computer **102** nur in das Desktop-Betriebssystem **110** gebootet werden soll, oder nur in das Smartphone-Betriebssystem **108**, oder gleichzeitig in das Desktop-Betriebssystem **110** und in das Smartphone-Betriebssystem **108**. Sobald die Konfigurationsinformationen **116** installiert sind, läuft die Routine **300** weiter von Operation **308** bis Operation **310**.

[0046] Bei Operation **310** wird ermittelt, ob der Benutzer die Eingabe vorgenommen hat, dass nur das

Desktop-Betriebssystem **110** gebootet werden soll. Wenn das der Fall ist, läuft die Routine **300** weiter von Operation **310** bis Operation **312**. Hier wird das Desktop-Betriebssystem **110** ohne Verwendung der Virtualisierungsebene **106** gebootet. Das Desktop-Betriebssystem **110** kann dann auf herkömmliche Weise genutzt werden.

**[0047]** Wenn der Computer **102** bei Operation **310** feststellt, dass der Benutzer keine Eingabe vorgenommen hat, dass nur das Desktop-Betriebssystem **110** gebootet werden soll, läuft die Routine **300** von Operation **310** weiter zu Operation **314**. Bei Operation **314** ermittelt der Computer **102**, ob der Benutzer die Eingabe vorgenommen hat, dass nur das Smartphone-Betriebssystem **108** gebootet werden soll. Wenn das der Fall ist, läuft die Routine **300** weiter von Operation **314** bis Operation **316**. Hier wird die Virtualisierungsebene **106** ausgeführt. Vor Operation **316** läuft die Routine dann weiter zu Operation **318**. Hier lädt die Virtualisierungsebene das Smartphone-Betriebssystem **108**. Wie oben besprochen, kann die Virtualisierungsebene **106** ein Image des gerade ausgeführten Smartphone-Betriebssystem **108** speichern und dieses Image beim Booten laden, um das Booten des Computers **102** zu beschleunigen.

**[0048]** Wenn der Computer **102** bei Operation **314** feststellt, dass der Benutzer keine Eingabe vorgenommen hat, dass nur das Smartphone-Betriebssystem **108** gebootet werden soll, läuft die Routine **300** weiter zu Operation **322**. Bei Operation **322** ermittelt der Computer **102**, ob der Benutzer die Eingabe vorgenommen hat, dass das Smartphone-Betriebssystem **108** und das Desktop-Betriebssystem **110** gemeinsam gebootet werden sollen. Wenn das der Fall ist, läuft die Routine **300** weiter zu Operation **324**. Hier wird die Virtualisierungsebene **106** entweder von einem nicht-flüchtigen Speicher oder von einem flüchtigen Massenspeicher geladen.

**[0049]** Sobald die Virtualisierungsebene **106** geladen ist, läuft die Routine **300** weiter bis Operation **326**. Hier wird das Smartphone-Betriebssystem **108** auf die oben beschriebene Weise geladen. Wie oben kurz besprochen, lädt die Virtualisierungsebene **106** in einer Ausführungsform das Smartphone-Betriebssystem **108** vollständig und stellt einem Benutzer des Computers **102** die vom Smartphone-Betriebssystem **108** bereitgestellten Funktionalitäten zur Verfügung, bevor das Desktop-Betriebssystem **110** geladen wird. In anderen Ausführungsformen kann das Smartphone-Betriebssystem **108** gleichzeitig mit dem Desktop-Betriebssystem **110** geladen werden.

**[0050]** Von Operation **326** läuft die Routine **300** weiter bis Operation **328**. Hier wird das Desktop-Betriebssystem **110** geladen und in einer virtuellen Maschine ausgeführt, die von der Virtualisierungsebene **106** bereitgestellt wird. Wie oben besprochen kann,

wenn das Smartphone-Betriebssystem **108** und das Desktop-Betriebssystem **110** gleichzeitig ausgeführt werden, ein Mechanismus bereitgestellt werden, der es einem Benutzer des Computers **102** gestattet, zwischen den beiden Betriebssystemen hin- und herzuwechseln. Von Operation **312**, **318**, und **328** läuft die Routine **300** weiter zu Operation **320** und ist damit abgeschlossen.

**[0051]** [Fig. 4](#) zeigt eine anschauliche Computer-Architektur für einen Computer **400**, der in den hier beschriebenen Anwendungen verwendet werden kann. Die in [Fig. 4](#) dargestellte Architektur kann verwendet werden, um ein Gerät, das kein Smartphone ist, wie etwa den in [Fig. 1](#) gezeigten und oben beschriebene Computer **102** darzustellen.

**[0052]** Der Computer **400** umfasst eine Grundplatte oder „Motherboard“, das eine Leiterplatte ist, mit der eine Vielzahl von Komponenten oder Einrichtungen über einen Systembus oder einen anderen elektrischen Übertragungsweg verbunden werden können. In einer anschaulichen Ausführungsform arbeitet eine CPU **422** in Verbindung mit einem Chipsatz **452**. Die CPU **422** ist ein standardmäßiger Zentralprozessor, der die arithmetischen und logischen Operationen durchführt, die für den Betrieb des Computers erforderlich sind. Der Computer **400** kann eine Vielzahl von CPUs **422** umfassen.

**[0053]** Der Chipsatz **452** umfasst eine North-Bridge **424** und eine South-Bridge **426**. Die North-Bridge **424** liefert eine Schnittstelle zwischen der CPU **422** und dem Rest des Computers **400**. Die North-Bridge **424** liefert auch eine Schnittstelle zu einem RAM-Speicher („RAM“ – Random Access Memory), der als Hauptspeicher **454** im Computer **400** verwendet wird, und möglicherweise auch zu einer Grafikkarte **430**. Die North-Bridge **424** kann auch die Funktionalitäten zum Bereitstellen einer Netzwerkfunktionalität über einen Gigabit-Ethernet-Adapter **428** umfassen. Der Gigabit-Ethernet-Adapter **428** ist dazu geeignet, den Computer **400** über ein Netzwerk mit einem anderen Computer zu verbinden. Die über den Netzwerkadapter **428** möglichen Verbindungen können Verbindungen über ein lokales Netzwerk („LAN“) oder über ein Fernnetzwerk („WAN“) umfassen. Netzwerkkombinationen mit LAN und WAN werden in Büros, unternehmensweiten Computernetzwerken, Intranets und im Internet alltäglich verwendet. Die North-Bridge **424** ist mit der South-Bridge **426** verbunden.

**[0054]** Die South-Bridge **426** ist dafür verantwortlich, viele der Ein- und Ausgabefunktionen des Computers **400** zu steuern. Insbesondere die South-Bridge **426** kann einen oder mehrere Universal-Serial-Bus-Anschlüsse („USB-Ports“) **432**, einen Sound-Adapter **446**, einen Ethernet-Controller **460**, sowie einen oder mehrere Allzweck-Eingangs- und

Ausgangspins („GPIO-Pins“) **434** aufweisen. Die South-Bridge **426** kann also einen Bus als Schnittstelle für periphere Karten, wie etwa einen Grafik-Adapter **462** aufweisen. In einer Ausführungsform umfasst der Bus einen Peripheral-Component-Interconnect-Bus („PCI-Bus“), es können jedoch auch andere Bus-Typen verwendet werden.

**[0055]** Die South-Bridge **426** ist auch dazu geeignet, eine oder mehrere Schnittstellen zum Anschließen von Massenspeichern an den Computer **400** bereitzustellen. Beispielsweise umfasst die South-Bridge **426** nach einer Ausführungsform einen Serial-Advanced-Technology-Attachment-Adapter („SATA-Adapter“) zum Bereitstellen eines oder mehrerer serieller ATA-Ports **436** und einen ATA-100-Adapter zum Bereitstellen eines oder mehrerer ATA-100-Ports **444**. Die seriellen ATA-Ports **436** und die ATA-100-Ports **444** können wiederum mit einem oder mehreren Massenspeichern, wie etwa mit dem SATA-Plattenlaufwerk **438**, verbunden sein, auf dem/denen ein Betriebssystem **440** und Anwenderprogramme gespeichert werden. Wie oben besprochen, umfasst ein Betriebssystem **440** einen Satz von Programmen, die die Operationen eines Computers und die Zuweisung von Ressourcen steuern. Ein Anwenderprogramm ist eine Software, die auf dem Betriebssystem oder in einer anderen Laufzeitumgebung läuft, und Ressourcen des Computers nutzt, um vom Benutzer gewünschte anwendungsspezifische Aufgaben ausführt. Wie ebenso oben besprochen, kann der Computer **400** auf die oben beschriebene Weise so konfiguriert werden, dass er ein Smartphone-Betriebssystem **108**, ein Desktop-Betriebssystem **110**, oder beide Betriebssysteme gleichzeitig ausführt

**[0056]** Die mit der South-Bridge **426** verbundenen Massenspeicher und die ihnen zugewiesenen maschinenlesbaren Medien stellen den nicht-flüchtigen Speicher für den Computer **400** bereit. Die hier enthaltene Beschreibung der maschinenlesbaren Medien bezieht sich zwar auf Massenspeicher, wie etwa ein Festplatten- oder ein CD-ROM-Laufwerk, der Kenner der Technik wird es jedoch zu schätzen wissen, dass die maschinenlesbaren Medien jedes erdenkliche verfügbare Medium sein kann, auf das der Computer **400** zugreifen kann.

**[0057]** Maschinenlesbare Medien umfassen flüchtige und nicht-flüchtige, herausnehmbare und nicht herausnehmbare Medien, die nach beliebigen Verfahren oder Technologien zum Speichern von Informationen implementiert sind, wie etwa maschinenlesbare Anweisungen, Datenstrukturen, Programmbausteine oder andere Daten. Maschinenlesbare Medien umfassen beispielsweise RAM, ROM, EPROM, EEPROM, Flash-Memory oder eine andere Halbleiterspeicher-Technologie, CD-ROM, DVD, HD-DVD, BLU-RAY, oder andere optische Daten-

speicher, Magnetkassetten, Magnetbänder, Magnetplattenspeicher oder andere magnetische Datenspeicher, oder jedes andere Medium, das zum Speichern der gewünschten Information verwendet werden kann und auf das der Computer zugreifen kann.

**[0058]** Eine Schnittstelle mit geringer Anschlusszahl („LPC-Schnittstelle“) kann auch durch die South-Bridge **426** bereitgestellt werden, um eine „Super-I/O-Einrichtung“ **470** anzuschließen. Die Super-I/O-Einrichtung **470** ist dafür verantwortlich eine Anzahl von Eingabe-/Ausgabeports bereitzustellen, einschließlich eines Tastatur-Ports, eines Maus-Ports, einer seriellen Schnittstelle **472**, einer parallelen Schnittstelle und anderer Typen von Eingabe-/Ausgabeports. Die LPC-Schnittstelle kann also mit einem Computer-Speichermedium, wie etwa einem ROM, oder einem Flash-Memory, wie etwa einem NVRAM **448**, zum Speichern der Firmware **104** verbunden sein, die den Programmcode mit den Basisroutinen enthält, die zum Hochfahren des Computers **400** benötigt werden, sowie zum Übertragen von Informationen zwischen Elementen innerhalb des Computers **400**. Wie oben besprochen, kann in den Ausführungsformen ein Smartphone-Betriebssystem **108**, eine Virtualisierungsebene **106** und ein anderer Programmcode ebenfalls im NVRAM **448** gespeichert sein.

**[0059]** Es sollte einzusehen sein, dass der Computer **400** andere Typen von Recheneinrichtungen umfassen kann, die kein Smartphone sind, wie Handheld-Computer, Netbooks, MIDs, eingebettete Rechensysteme, PDAs, sowie andere Typen von Recheneinrichtungen, die dem Kenner der Technik bekannt sind. Es wird auch in Betracht gezogen, dass der Computer **400** nicht alle der in [Fig. 4](#) dargestellten Komponenten umfasst, dass er andere Komponenten umfasst, die nicht explizit in [Fig. 4](#) dargestellt sind, oder ggf. eine völlig andere Architektur nutzt, als die in [Fig. 4](#) dargestellte Architektur.

**[0060]** Es sollte einzusehen sein, dass die hier beschriebenen Softwarekomponenten – wenn sie in die CPU **422** geladen und ausgeführt werden – die CPU **422** und den gesamten Computer **400** von einem Allzweck-Rechensystem in ein maßgeschneidertes Spezial-Rechensystem umwandeln, das die hier vorgestellten Funktionalitäten erleichtert. Die Konstruktion der CPU **422** kann aus einer beliebigen Anzahl von Transistoren oder anderen diskreten Schaltkreiselementen bestehen, die einzeln oder zusammen jede Anzahl von Zuständen annehmen können. Genauer gesagt kann die CPU **422** als endlicher Automat arbeiten, und auf ausführbare Anweisungen antworten, die in den hier offen gelegten Programmbausteinen enthalten sind. Diese vom Computer ausführbaren Anweisungen können die CPU **422** umwandeln, indem sie spezifizieren, wie die CPU **422** von einem Zustand in einen anderen übergeht, und

dabei die Transistoren oder andere diskrete Hardware-Elemente umwandelt, aus denen die CPU **422** besteht.

**[0061]** Das Codieren der hier genannten Softwarebausteine kann auch die physikalische Struktur der hier genannten maschinenlesbaren Medien umwandeln. Die spezifische Umwandlung der physikalischen Struktur kann in den unterschiedlichen Ausführungsformen in dieser Beschreibung von verschiedenen Faktoren abhängen. Beispiele solcher Faktoren können sein (sind aber nicht darauf beschränkt): Die Technologie, die zum Implementieren der maschinenlesbaren Medien verwendet wird, und ob die maschinenlesbaren Medien als primärer oder als sekundärer Speicher gekennzeichnet sind, und ähnliches. Wenn das maschinenlesbare Medium beispielsweise als Speicher auf Halbleiterbasis implementiert ist, kann die hier offen gelegte Software auf dem maschinenlesbaren Medium codiert werden, indem der physikalische Zustand des Halbleiterspeichers verändert wird. Beispielsweise kann die Software den Zustand von Transistoren, Kondensatoren, oder anderen diskreten Schaltkreiselementen umwandeln, aus denen der Halbleiterspeicher besteht. Die Software kann auch den physikalischen Zustand solcher Komponenten umwandeln, um Daten darauf zu speichern.

**[0062]** In einem anderen Beispiel können die hier offen gelegten maschinenlesbaren Medien unter Verwendung von magnetischer oder optischer Technologie implementiert sein. In solchen Anwendungen kann die hier vorgestellte Software den physikalischen Zustand von magnetischen oder optischen Medien umwandeln, wenn die Software darin codiert wird. Diese Umwandlungen können das Verändern der magnetischen Eigenschaften von besonderen Speicherorten innerhalb des magnetischen Mediums umfassen. Diese Umwandlungen können auch das Verändern der physikalischen Merkmale oder Eigenschaften von besonderen Speicherorten innerhalb des jeweiligen optischen Mediums umfassen, um die optischen Eigenschaften dieser Speicherorte zu verändern. Andere Umwandlungen von physikalischen Medien sind möglich, ohne vom Schutzzumfang dieser Erfindung und von Erfindergedanken abzuweichen. Dabei sollen die oben genannten Beispiele diese Diskussion lediglich fördern.

**[0063]** Angesichts der oben gegebenen Erläuterungen wird es einzusehen sein, dass viele Typen von physikalischen Umwandlungen im Computer **400** stattfinden, um die hier vorgestellten Softwarekomponenten zu speichern und auszuführen. Es sollte einzusehen sein, dass der Computer **400** andere Typen von Recheneinrichtungen umfassen kann, die keine Smartphones sind, wie Handheld-Computer, eingebettete Rechnersysteme, PDAs, sowie andere Typen von Recheneinrichtungen, die dem Kenner

der Technik bekannt sind. Es wird auch in Betracht gezogen, dass der Computer **400** möglicherweise nicht alle der in **Fig. 4** dargestellten Komponenten umfasst, dass er andere Komponenten umfasst, die nicht explizit in **Fig. 4** dargestellt sind, oder möglicherweise eine völlig andere Architektur nutzt, als die in **Fig. 4** dargestellte Architektur.

**[0064]** Basierend auf den oben gegebenen Erläuterungen wird es einzusehen sein, dass Technologien zum gleichzeitigen Ausführen eines Smartphone-Betriebssystems und eines Desktop-Betriebssystems hier bereit gestellt werden. Obwohl der hier vorgestellte Gegenstand in einer Sprache beschrieben ist, die spezifisch für die Struktur und die Merkmale von Computern, methodischen und transformatorischen Schritten, spezifischen Rechanlagen und maschinenlesbaren Medien ist, versteht es sich von selbst, dass die hier offen gelegten Inhalte nicht auf die hier beschriebenen spezifischen Merkmale, Schritte oder Medien beschränkt sind.

**[0065]** Der oben beschriebene Gegenstand dient lediglich der Veranschaulichung und sollte nicht als einschränkend aufgefasst werden. Verschiedene Abwandlungen und Veränderungen können am hier beschriebenen Gegenstand vorgenommen werden, ohne den beispielhaften Ausführungsformen und den abgebildeten und beschriebenen Anwendungen zu folgen, ohne dass damit vom wahren Erfindergedanken und vom Schutzzumfang der hier offen gelegten Erfindung abgewichen wird.

### Schutzansprüche

1. Recheneinrichtung, die kein Smartphone ist, und die umfasst:  
 eine Zentraleinheit (CPU);  
 einen RAM-Speicher (RAM); und  
 einen nicht-flüchtigen Speicher, in dem ein Smartphone-Betriebssystem und eine Firmware gespeichert sind, die von der CPU ausgeführt werden können, wobei die Firmware Anweisungen umfasst, die beim Ausführen von der CPU die CPU veranlassen, zu ermitteln, ob die Einrichtung – die kein Smartphone ist – nur unter Verwendung eines Desktop-Betriebssystems, nur unter Verwendung eines Smartphone-Betriebssystems, oder unter gleichzeitiger Verwendung des Smartphone-Betriebssystems und des Desktop-Betriebssystems gebootet werden soll, und wobei als Reaktion auf die Feststellung, dass die Einrichtung – die kein Smartphone ist – nur unter Verwendung des Smartphone-Betriebssystems, oder unter gleichzeitiger Verwendung des Smartphone-Betriebssystems und des Desktop-Betriebssystems gebootet werden soll, veranlasst wird, dass eine Virtualisierungsebene auf der Einrichtung – die kein Smartphone ist – ausgeführt wird, und wobei als Reaktion auf die Feststellung, dass die Einrichtung – die kein Smartphone ist – nur unter Ver-

wendung des Smartphone-Betriebssystems gebootet werden soll, dann veranlasst wird, dass das Smartphone-Betriebssystem unter Verwendung der Virtualisierungsebene auf der Einrichtung – die kein Smartphone ist – ausgeführt wird; und wobei als Reaktion auf die Feststellung, dass die Einrichtung – die kein Smartphone ist – unter gleichzeitiger Verwendung des Smartphone-Betriebssystems und des Desktop-Betriebssystems gebootet werden soll, dann veranlasst wird, dass das Smartphone-Betriebssystem und das Desktop-Betriebssystem auf der Einrichtung – die kein Smartphone ist – unter Verwendung der Virtualisierungsebene ausgeführt werden.

2. Einrichtung nach Anspruch 1, wobei das Veranlassen der Ausführung des Smartphone-Betriebssystems und des Desktop-Betriebssystems auf der Einrichtung – die kein Smartphone ist – unter Verwendung der Virtualisierungsebene umfasst, dass das Smartphone-Betriebssystem unter Verwendung der Virtualisierungsebene vollständig vor dem Booten des Desktop-Betriebssystems gebootet wird.

3. Einrichtung nach Anspruch 2, wobei das Veranlassen der Ausführung des Smartphone-Betriebssystems und des Desktop-Betriebssystems auf der Einrichtung – die kein Smartphone ist – unter Verwendung der Virtualisierungsebene des Weiteren umfasst, dass die Funktionalitäten, die das Smartphone-Betriebssystem bietet, zur Verfügung gestellt werden, während das Desktop-Betriebssystem gebootet wird.

4. Einrichtung nach Anspruch 3, wobei das Veranlassen der Ausführung des Smartphone-Betriebssystems und des Desktop-Betriebssystems auf der Einrichtung – die kein Smartphone ist – unter Verwendung der Virtualisierungsebene umfasst, dass das Smartphone-Betriebssystem gleichzeitig unter Verwendung der Virtualisierungsebene gebootet werden.

5. Einrichtung nach Anspruch 4, wobei das Smartphone-Betriebssystem von einer vorher gespeicherten Virtual-Machine-Image-Datei gebootet wird.

6. Einrichtung nach Anspruch 5, wobei die Firmware des Weiteren so konfiguriert ist, dass sie beim Ausführen durch die CPU: ermittelt, ob die Einrichtung – die kein Smartphone ist – zum ersten Mal gebootet wird; und als Reaktion auf die Feststellung, dass die Einrichtung – die kein Smartphone ist – zum ersten Mal gebootet wird, die Einrichtung – die kein Smartphone ist – veranlasst das Desktop-Betriebssystem ohne Verwendung der Virtualisierungsebene zu booten, und veranlasst, dass ein Synchronisationsagent in dem Desktop-Betriebssystem installiert wird.

7. Einrichtung nach Anspruch 6, wobei der Synchronisationsagent so konfiguriert ist, dass er Dateien und Ordner, auf die das Desktop-Betriebssystem zugreifen kann, in einen gemeinsam genutzten Speicherort synchronisiert, auf den das Smartphone-Betriebssystem zugreifen kann.

8. Einrichtung nach Anspruch 7, wobei das Smartphone-Betriebssystem des Weiteren einen zweiten Synchronisationsagenten umfasst, der so konfiguriert ist, dass er Dateien und Ordner, auf die das Smartphone-Betriebssystem zugreifen kann, in den gemeinsam genutzten Speicherort synchronisiert.

9. Einrichtung nach Anspruch 8, wobei das Smartphone-Betriebssystem auf den gemeinsam genutzten Speicherort auch dann zugreifen kann, wenn das Desktop-Betriebssystem nicht läuft und auf den das Desktop-Betriebssystem auch dann zugreifen kann, wenn das Smartphone-Betriebssystem nicht läuft.

10. Einrichtung nach Anspruch 9, wobei die Firmware des Weiteren so konfiguriert ist, dass sie beim Ausführen durch die CPU im Smartphone-Betriebssystem einen Treiber für die Energiesteuerung ausführt, der so konfiguriert ist, dass er die Funktionalität einer Energiesteuerung bereitstellt, die mit der Einrichtung – die kein Smartphone ist – kompatibel ist.

11. Einrichtung nach Anspruch 10, wobei das Smartphone-Betriebssystem so konfiguriert ist, dass es das Ausführen von Smartphone-Anwendungen des ersten und von dritten Herstellern gestattet.

12. Einrichtung nach Anspruch 11, wobei die Virtualisierungsebene so konfiguriert ist, dass sie eine selektive Virtualisierung der Hardware der Einrichtung – die kein Smartphone ist – für das Smartphone-Betriebssystem durchführt.

13. Einrichtung nach Anspruch 12, wobei die Virtualisierungsebene und das Smartphone-Betriebssystem in einem nicht-flüchtigen Speicher der Einrichtung – die kein Smartphone ist – gespeichert ist.

14. Einrichtung nach Anspruch 13, wobei die Virtualisierungsebene in einer Extensible-Firmware-Interface-(EFI)-Firmware der Einrichtung – die kein Smartphone ist – implementiert ist.

15. Recheneinrichtung – die kein Smartphone ist – und die umfasst:  
eine Zentraleinheit (CPU);  
einen RAM-Speicher (RAM); und  
einen nicht-flüchtigen Speicher, in dem ein Smartphone-Betriebssystem und eine Firmware gespeichert sind, die von der CPU ausgeführt werden können, wobei die Firmware Anweisungen umfasst, die

beim Ausführen von der CPU die CPU veranlassen, zu ermitteln, ob die Einrichtung – die kein Smartphone ist – nur unter Verwendung eines Desktop-Betriebssystems, nur unter Verwendung eines Smartphone-Betriebssystems, oder unter gleichzeitiger Verwendung des Smartphone-Betriebssystems und des Desktop-Betriebssystems gebootet werden soll, und wobei als Reaktion auf die Feststellung, dass die Einrichtung – die kein Smartphone ist – nur unter Verwendung eines Smartphone-Betriebssystems, oder unter gleichzeitiger Verwendung des Smartphone-Betriebssystems und des Desktop-Betriebssystems gebootet werden soll, veranlasst wird, dass eine Virtualisierungsebene auf der Einrichtung – die kein Smartphone ist – ausgeführt wird, und wobei als Reaktion auf die Feststellung, dass die Einrichtung – die kein Smartphone ist – nur unter Verwendung des Smartphone-Betriebssystems gebootet werden soll, veranlasst wird, dass das Smartphone-Betriebssystem unter Verwendung der Virtualisierungsebene auf der Einrichtung – die kein Smartphone ist – ausgeführt wird, und wobei als Reaktion auf die Feststellung, dass die Einrichtung – die kein Smartphone ist – unter gleichzeitiger Verwendung des Smartphone-Betriebssystems und des Desktop-Betriebssystems gebootet werden soll, veranlasst wird, dass das Smartphone-Betriebssystem auf der Einrichtung – die kein Smartphone ist – unter Verwendung der Virtualisierungsebene gebootet wird, und wobei veranlasst wird, dass das Desktop-Betriebssystem auf der Einrichtung – die kein Smartphone ist – unter Verwendung der Virtualisierungsebene gebootet wird, nachdem das Smartphone-Betriebssystem vollständig gebootet ist, und die Funktionalität, die das Smartphone-Betriebssystem bietet, einem Benutzer der Einrichtung – die kein Smartphone ist – zur Verfügung stellt, während das Desktop-Betriebssystem bootet.

Es folgen 4 Blatt Zeichnungen

Anhängende Zeichnungen

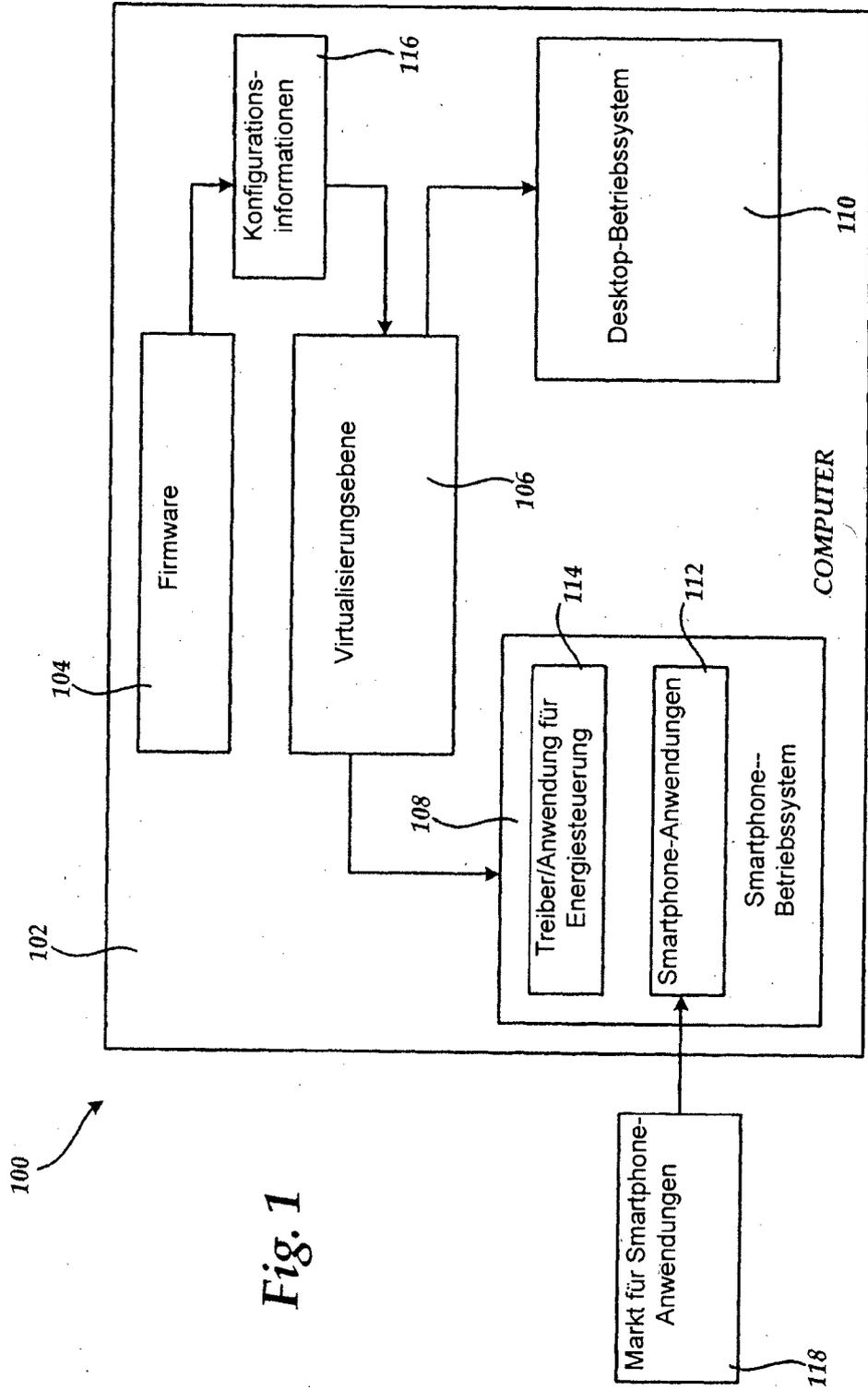


Fig. 1

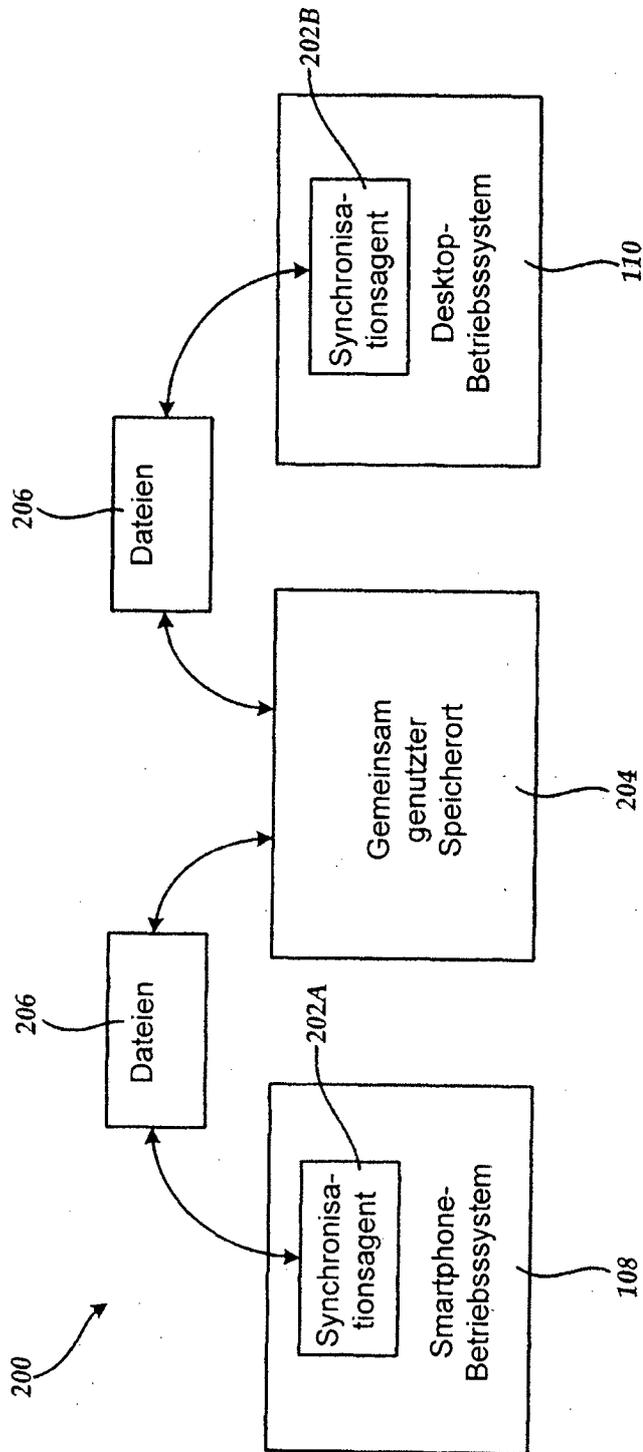
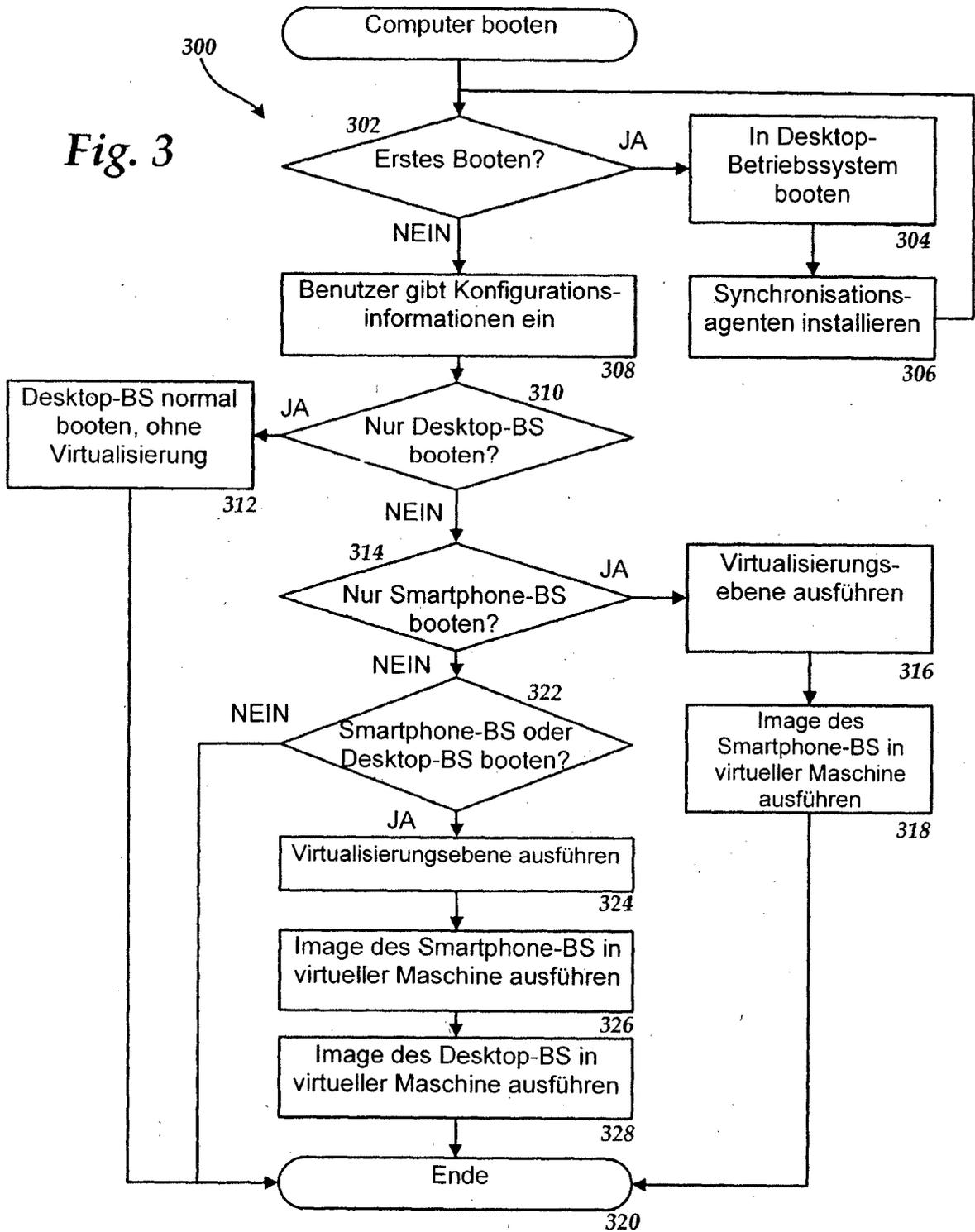


Fig. 2

Fig. 3



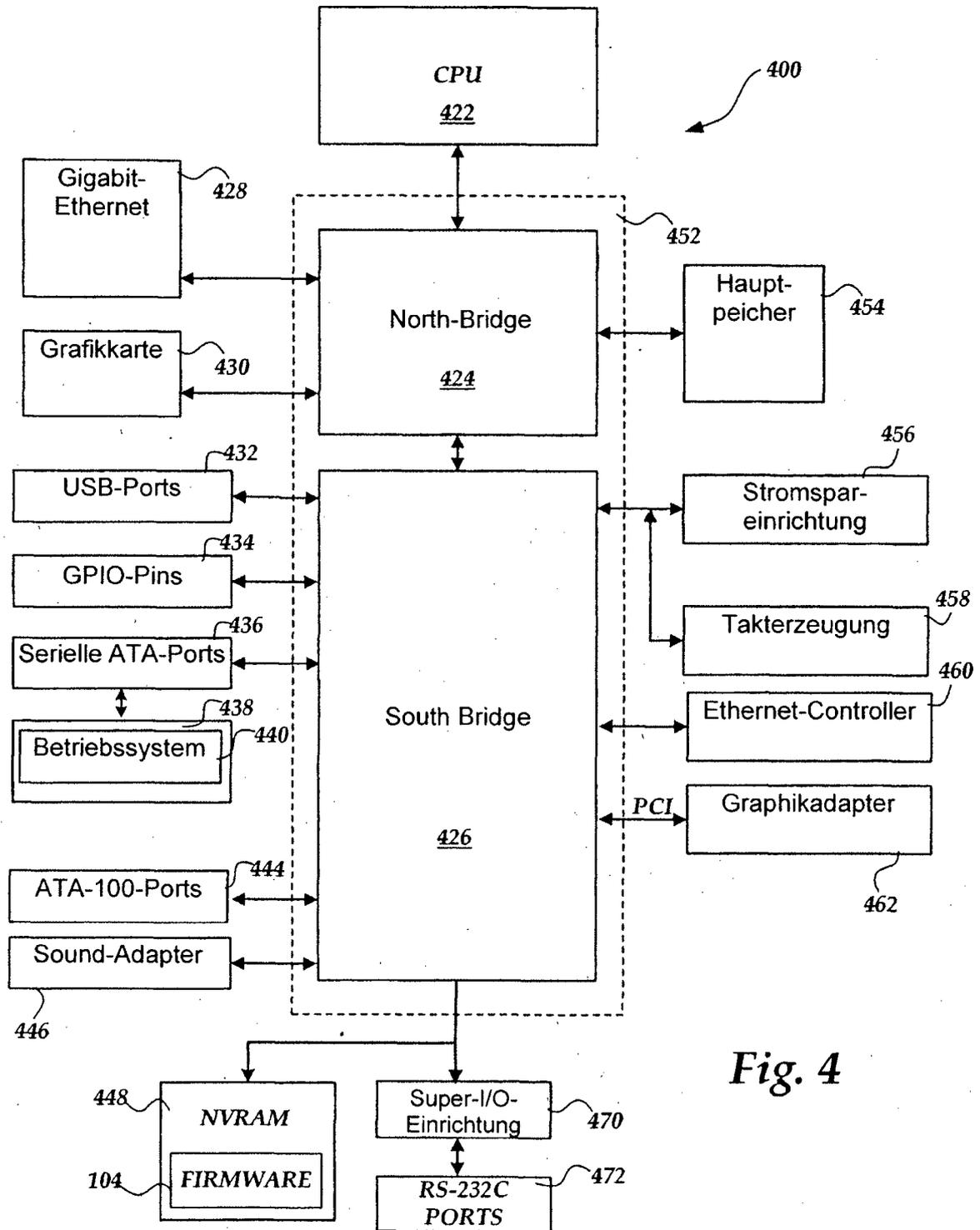


Fig. 4