



(12)

## Veröffentlichung

der internationalen Anmeldung mit der  
(87) Veröffentlichungs-Nr.: **WO 2011/126499**  
in deutscher Übersetzung (Art. III § 8 Abs. 2 IntPatÜG)  
(21) Deutsches Aktenzeichen: **11 2010 005 473.8**  
(86) PCT-Aktenzeichen: **PCT/US2010/049816**  
(86) PCT-Anmeldetag: **22.09.2010**  
(43) Veröffentlichungstag der PCT Anmeldung  
in deutscher Übersetzung: **24.01.2013**

(51) Int Cl.: **G06F 9/48 (2013.01)**  
**G06F 9/46 (2013.01)**  
**G06F 9/54 (2013.01)**

(30) Unionspriorität:  
**61/321,616**                      **07.04.2010**    **US**

(71) Anmelder:  
**Apple Inc., Cupertino, Calif., US**

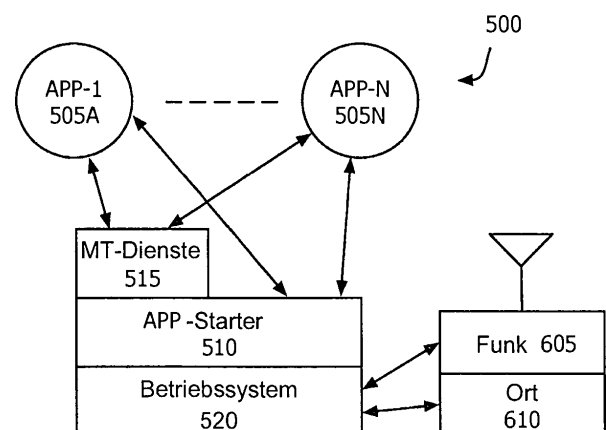
(74) Vertreter:  
**BARDEHLE PAGENBERG Partnerschaft**  
**Patentanwälte, Rechtsanwälte, 81675, München,**  
**DE**

(72) Erfinder:  
**ALBERT, Eric, Cupertino, US; Chapman,**  
**Gregory R., Cupertino, Calif., US; Forstall,**  
**Scott, Cupertino, Calif., US; Crane, Neil Gareth,**  
**Cupertino, Calif., US; Litzinger, Darren, Cupertino,**

**Calif., US; Myszewski, David, Cupertino, Calif.,**  
**US; Schreyer, Richard, Cupertino, Calif., US;**  
**Srisuwananukorn, Charles, Cupertino, Calif.,**  
**US; Duffy, Thomas Brogan, Cupertino, Calif.,**  
**US; Freedman, Gordon J., Cupertino, Calif., US;**  
**Goodwin, David William, Cupertino, Calif., US;**  
**Iarocci, John, Cupertino, Calif., US; Marcellino,**  
**Christopher, Cupertino, Calif., US; Mason,**  
**Henry Graham, Cupertino, Calif., US; Masputra,**  
**Cahya, Cupertino, Calif., US; Novick, Gregory,**  
**Cupertino, Calif., US; Rothert, Curtis, Cupertino,**  
**Calif., US; Ballard, Lucia, Cupertino, US; Sokol,**  
**Joseph, Cupertino, Calif., US; Sorresso, Damien,**  
**Cupertino, Calif., US; Milligan, Michael Van,**  
**Cupertino, Calif., US; Waston, Matthew G.,**  
**Cupertino, Calif., US**

(54) Bezeichnung: **OPPORTUNISTISCHES MULTITASKING**

(57) Zusammenfassung: Dienste für eine persönliche elektronische Vorrichtung werden bereitgestellt, durch welche eine Form von Hintergrundverarbeitung oder Multitasking unterstützt wird. Die offenbarten Dienste erlauben es Nutzeranwendungen Vorteile aus Hintergrundverarbeitung zu ziehen, ohne signifikante negative Konsequenzen auf eine Nutzererfahrung des Vordergrundprozesses oder auf die Leistungsressourcen der persönlichen elektronischen Vorrichtung. Um das offenbarte Multitasking zu bewirken, werden ein oder mehrere einer Anzahl von operationellen Beschränkungen durchgesetzt. Eine Konsequenz solcher Beschränkungen kann es sein, dass ein Prozess in dem Hintergrundzustand nicht das tun kann, was es zu tun in der Lage wäre, wenn es in dem Vordergrundzustand wäre. Die Implementierung der offenbarten Dienste kann im Wesentlichen erkennbar für die ausführenden Nutzeranwendungen sein und kann, in einigen Fällen, ohne die explizite Kooperation der Nutzeranwendungen ausgeführt werden.



### Beschreibung

**[0001]** Die Erfindung, welche in dieser Anmeldung zu offenbaren und zu beanspruchen ist, wurde vorzeitig und ohne die Genehmigung von Apple der Öffentlichkeit offenbart, als ein Prototyp des Iphone 4 von Apple offensichtlich einem Apple-Ingenieur am 25. März 2010 gestohlen wurde. Die US-Prioritätsanmeldung, auf welcher diese Anmeldung basiert, wurde nicht vor diesem offensichtlichen Diebstahl eingereicht.

#### Querverweis auf verwandte Anmeldungen

**[0002]** Dies ist eine internationale Anmeldung basierend auf der vorläufigen US-Anmeldung Nummer 61/321, 616 mit dem Titel „Opportunistisches Multitasking (Opportunistic Multitasking)“ eingereicht am 7. April 2010 und welches hiermit in seiner Gesamtheit durch Verweis aufgenommen wird.

#### Hintergrund

**[0003]** Diese Offenbarung bezieht sich im Allgemeinen auf das Gebiet der Computerwissenschaft, insbesondere bezieht sich diese Offenbarung auf eine Technik zum Verbessern der Nutzererfahrung und der Leistungsverwaltung in einer persönlichen elektronischen Vorrichtung.

**[0004]** Leistungsbeschränkte, handgehaltene Vorrichtungen (z. B. Mobiltelefone, persönliche Unterhaltungsvorrichtungen und elektronische Pad-Computer) sind ressourcenbeschränkt, im Vergleich zu größeren, festen Systemen, wie beispielsweise Desktop-, Arbeitsplatz-, und Notebook-Computer. In solchen Systemen kann die Rechenleistung, welche verfügbar ist, nicht einfach die Ausführung einer großen Zahl von gleichzeitigen Prozessen/Threads unterstützen ohne signifikant die Nutzererfahrung zu vermindern und die limitierten Leistungsressourcen der Vorrichtung zu verbrauchen. Im Lichte dieser Erkenntnis haben Systemdesigner für diese Typen von Vorrichtungen traditionell nicht Multitasking auf der Nutzeranwendungsebene unterstützt. Während diese Herangehensweise den Vorteil des Minimierens der Inanspruchnahme der begrenzten Leistungsressourcen der Vorrichtung hat, beschränkt es auch die Fähigkeit dem Nutzer (über eine Nutzeranwendung) eine interaktive Umgebung bereitzustellen.

#### Zusammenfassung

**[0005]** Dienste für eine persönliche elektronische Vorrichtung werden bereitgestellt, durch welche eine Form von Hintergrundverarbeitung oder Multitasking unterstützt wird. Die offenbarten Dienste erlauben Nutzeranwendungen Vorteile aus Hintergrundverarbeitungen zu ziehen ohne signifikante negative Konsequenzen auf eine Nutzererfahrung der Vordergrundprozesse oder die Leistungsressourcen der persönlichen elektronischen Vorrichtung. Die Implementierung der offenbarten Dienste kann im Wesentlichen erkennbar für die ausführenden Nutzeranwendungen sein und, in einigen Fällen, ohne die explizite Kooperation der Nutzeranwendung ausgeführt werden. Um das offenbarte Multitasking zu bewirken, kann eine Zahl von operationellen Beschränkungen durchgesetzt werden. Eine Konsequenz solcher Beschränkungen kann es sein, dass ein Prozess nicht in der Lage sein wird im Hintergrund das zu tun, was es in der Lage wäre zu tun, wenn es im Vordergrund wäre.

**[0006]** In einem Dienst, wird eine Vordergrundnutzeranwendung in einen nicht-ausgeführten Zustand überführt, weil er aus dem Vordergrundzustand herausbewegt wird. In einem anderen Dienst, wird einem Hintergrundprozess eine maximale Menge von Zeit gegeben, um eine Aufgabe fertigzustellen, bevor er in einen nicht-ausführenden Zustand überführt wird. In noch einem anderen Dienst, wird es Audio-Anwendungen (z. B. Nutzeranwendungen, welche Audio-Signale erzeugen oder aufzeichnen) erlaubt im Hintergrund ausgeführt zu werden, außer sie werden durch einen Nutzer pausiert. In noch einem anderen Dienst, können Kommunikations-Socket(s), welche für eine Nutzeranwendung instanziiert werden, aufrechterhalten werden, selbst wenn die Nutzeranwendung, welche sie instanziiert, in einen nicht-ausgeführten Zustand gebracht wird. Ein anschaulicher Typ von Anwendung, welcher Vorteile aus diesem Dienst ziehen kann, sind voice over internet protocol (VOIP) Nutzeranwendungen. In noch weiteren Ausführungsformen wird es Nutzeranwendungen erlaubt Benachrichtigungen zu empfangen (z. B. Ortseignisse), wenn sie in dem nicht-ausgeführten Zustand sind. Jeder der offenbarten Dienste stützt sich auf oder verwendet einen oder mehrere Beschränkungen, welche, im Betrieb, Vollzeit-Multitasking-Operationen beeinträchtigen oder verhindern können. D. h. die offenbarten Dienste erlauben Multitasking nur, wenn es nicht signifikant Vordergrundprozesse beeinträchtigt oder übermäßig die Leistung der persönlichen elektronischen Vorrichtung verwendet.

## Kurze Beschreibung der Zeichnungen

- [0007] [Fig. 1](#) zeigt ein veranschaulichendes Thread-Planungs-Schema gemäß einer Ausführungsform.
- [0008] [Fig. 2](#) zeigt, in Flussdiagrammform, eine Aufgabenfertigstellungsdienstoperation gemäß einer Ausführungsform.
- [0009] [Fig. 3](#) zeigt, in Flussdiagrammform, eine Audiodienstoperation gemäß einer Ausführungsform.
- [0010] [Fig. 4](#) zeigt, in Flussdiagrammform, eine Netzwerkdienstoperation gemäß einer Ausführungsform.
- [0011] [Fig. 5](#) zeigt, in Blockdiagrammform, eine persönliche elektronische Vorrichtung gemäß einer Ausführungsform.
- [0012] [Fig. 6](#) zeigt, in Blockdiagrammform, eine persönliche elektronische Vorrichtung gemäß einer anderen Ausführungsform.
- [0013] [Fig. 7](#) zeigt, in Blockdiagrammform, Funk- und Ortsverarbeitende Elemente gemäß einer Ausführungsform.
- [0014] [Fig. 8](#) zeigt, in Blockdiagrammform, eine persönliche elektronische Vorrichtung, gemäß einer Ausführungsform.

## Detaillierte Beschreibung

[0015] Dienste für eine persönliche elektronische Vorrichtung werden bereitgestellt, durch welche eine begrenzte Form von Hintergrundverarbeitung oder Multitasking unterstützt wird. Die Verwendung einer oder mehrerer der offenbarten Dienste, in Kombination mit offenbarten Betriebsbeschränkungen, erlaubt es Nutzeranwendungen im Hintergrund ausgeführt zu werden, um dem Nutzer eine robuste interaktive Umgebung mit geringer Auswirkung auf die Leistungsressourcen der Vorrichtung zu geben. In einigen Ausführungsformen kann die Hintergrundverarbeitung beschränkt sein auf die Fertigstellung einer spezifischen Aufgabe, eine spezifische Zeitmenge (z. B. 5 bis 30 Minuten) oder einen spezifischen Typ von Aufgabe (z. B. Audiooperationen). In anderen Ausführungsformen kann eine Nutzeranwendung die Operation einstellen, nachdem dem Betriebssystem gemeldet wird, dass sie neu belebt werden kann nach dem Auftritt einer oder mehrerer spezifizierter Ereignisse. Bei Neubelebung, kann die Anwendung zusätzliche Operationen im Hintergrund ausführen. Die Implementierung der offenbarten Dienste kann im Wesentlichen erkennbar für die ausführenden Nutzeranwendungen sein, und in einigen Fällen, ausgeführt ohne die explizite Operation der Nutzeranwendung werden.

[0016] Wie hierin verwendet ist der Ausdruck „persönliche elektronische Vorrichtung“ eine persönliche digitale Vorrichtung, wie beispielsweise ein Mobiltelefon, ein persönlicher digitaler Assistent, eine handgehaltene Unterhaltungsvorrichtung, ein Pad- oder ein Tablet-Computersystem oder eine Set-Top-Box (z. B. ein Apple TV® oder eine Kabel-Konverter-Box) (Apple TV ist eine registrierte Marke der Apple Inc.). Wie hierin verwendet bezieht sich der Ausdruck „Dienst“ auf ein Dienstprogramm, Funktion- oder Programmcodemodul, welches einige Aufgaben für einen aufrufenden Prozess ausführt, welcher keine Nutzerschnittstelle hat und auf welchen programmatisch über eine Aufrufschnittstelle zugegriffen wird, wie beispielsweise eine Anwendungsprogrammierschnittstelle (API).

[0017] In der folgenden Beschreibung, zum Zwecke der Erklärung, werden zahlreiche spezifische Details dargelegt, um ein gründliches Verständnis der Erfindung bereitzustellen. Jedoch wird es für einen Fachmann offensichtlich sein, dass die Erfindung ohne diese spezifischen Details ausgeführt werden kann. In anderen Fällen werden Strukturen und Vorrichtungen in Blockdiagrammform gezeigt, um das Unklarmachen der Erfindung zu Vermeiden. Verweise auf Zahlen ohne Indizes sollen auf alle Instanzen von Indizes, welche der referenzierten Zahl entsprechen, verweisen. Außerdem ist die Sprache, welche in dieser Offenbarung verwendet wird, prinzipiell ausgewählt worden für Lesbarkeit und Instruktionszwecke und wurde nicht ausgewählt, um den erfinderischen Gegenstand präzise zu schildern oder einzuschränken, zurückgreifend auf die Ansprüche, welche notwendig sind, um solch einen erfinderischen Gegenstand zu bestimmen. Der Verweis in dieser Beschreibung auf „eine Ausführungsform“ bedeutet, dass eine spezifische Eigenschaft, Struktur oder Charakteristikum, beschrieben in Verbindung mit den Ausführungsformen, in mindestens einer Ausführungsform der Erfindung umfasst ist, und mehrere Verweise auf „eine Ausführungsform“ sollten nicht verstanden werden als notwendigerweise auf die gleiche Ausführungsform verweisend.

**[0018]** Es ist zu beachten, dass in der Entwicklung irgendeiner aktuellen Implementierung zahlreiche Programmier- und Komponentenentscheidungen gemacht werden müssen, um die spezifischen Ziele der Entwickler zu erreichen (z. B. Einhaltung der System- und Wirtschaftsbezogenen Beschränkungen) und dass diese Ziele von einer Implementierung zu einer anderen variieren werden. Es ist auch zu beachten, dass solches Entwicklungsbestreben komplex und zeitaufwendig sein kann, aber nichtsdestotrotz für den Fachmann in dem Feld von persönlicher Computervorrichtungsentwicklung, welche den Nutzen dieser Offenbarung hat, ein Routinevorhaben sein kann.

**[0019]** Um Hintergrundverarbeitung oder Multitasking ohne signifikante negative Konsequenzen auf eine Nutzererfahrung der Vordergrundprozesse oder der Leistungsressourcen der persönlichen elektronischen Vorrichtung zu bewirken, kann eine Zahl von operationellen Beschränkungen durchgesetzt werden. Eine Konsequenz solcher Beschränkungen kann es sein, dass ein Prozess nicht in der Lage sein wird im Hintergrund das zu tun, was er in der Lage wäre zu tun, wenn er in dem Vordergrund wäre.

**[0020]** Wie hierin verwendet bedeutet der Ausdruck „Vordergrundprozess“, dass jener Prozess, welcher derzeitig Zugriff auf Systemressourcen hat (z. B. die zentrale Verarbeitungseinheit und graphische Verarbeitungseinheit der Plattform) und eine Nutzerschnittstelle (UI) oder graphische Nutzerschnittstelle (GUI) einem Nutzer präsentiert (dementsprechend ist der „Vordergrund“ ein Zustand, in welchem ein Vordergrundprozess ausgeführt wird). Im Gegensatz dazu ist ein „Hintergrundprozess“ ein Prozess, welcher geplant werden kann, um auf Systemressourcen zuzugreifen, aber welcher derzeitig nicht eine UI/GUI einem Nutzer präsentiert (dementsprechend ist der „Hintergrund“ ein Zustand, in welchem ein Hintergrundprozess ausgeführt wird). Wie hierin verwendet bedeutet der Ausdruck „Prozess“ eine Nutzeranwendung. Eine Nutzeranwendung wiederum ist ein ausführbares Codemodul/sind ausführbare Codemodule, welche/s fähig ist/sind eine UI/GUI zu präsentieren.

**[0021]** Operationelle Beschränkungen: Die folgende veranschaulichende Liste von operationellen Beschränkungen kann durchgesetzt werden, um die schädliche Wirkung zu begrenzen, welche einen Hintergrundprozess auf einen Vordergrundprozess haben kann.

1. Prozessorplanung: Vordergrund- und Hintergrundprozesse können unterschiedlichen Prozessorplanungs-„Prioritäten“ zugewiesen werden, auf solche Weise, dass Hintergrundprozesse nicht Vordergrundprozesse beeinträchtigen. Ein Fachmann wird erkennen, dass in operationellen Umgebungen, wie beispielsweise UNIX<sup>®</sup> und Mach, Prozesse nicht geplant werden, Threads aber schon (UNIX ist eine registrierte Marke der American Telephone and Telegraph Company). Es wird weiter erkannt werden, dass ein Thread jener Abschnitt eines Programms, einer Anwendung oder eines Prozesses ist, welcher unabhängig von, und gleichzeitig mit anderen Abschnitten des Programms, der Anwendung oder des Prozesses laufen kann. Mit Bezug auf [Fig. 1](#) können in einer Ausführungsform Hintergrund-Threads eine Priorität zugewiesen werden aus einem Bereich von Prioritäten, welche in Teilen mit jenen Prioritätsebenen, welche Vordergrund-Threads zugewiesen werden, überlappt. Die überlappenden Prioritätsbereiche vergrößern die Wahrscheinlichkeit, dass ein Hintergrundprozess/-Thread vorankommt (d. h. ausgeführt wird), selbst wenn es einen Vordergrundprozess/-Thread gibt, welcher läuft. In dieser Ausführungsform, um Thread-Ansprechbarkeit zu verbessern, ist es als vorteilhaft gefunden worden, jene Threads zurückzustufen (d. h. ihre Planungspriorität zu reduzieren), welche ihre gesamte Menge (Quantum) ohne das Blockieren für Eingabe/Ausgabe-(I/O)Operation verwenden. In einer Ausführungsform wird die Priorität eines Threads durch eine einzelne Abzählung jedes Mal, wenn es seine voll zugeteilte Menge ohne Blockieren verwendet (es wird erkannt werden, dass eine Verwendung von Prozessen einer vollen Menge ein Indikator ist, dass der Prozess die zentral verarbeitende Einheit verwendet). In einer anderen Ausführungsform kann eine Priorität eines Threads reduziert werden durch mehr als eine Abzählung, wenn diese auftritt. Die Menge der „Überlappung“ in Prioritätsebenen zwischen dem Hintergrundbereich und dem Vordergrundbereich kann 1 sein oder mehr.
2. Plattenplanung: In einer Ausführungsform wird Vordergrund-Threads Priorität über Hintergrund-Threads gegeben, beim Zugreifen auf Systemspeichereinheiten (magnetische Festplatten und Solid-State-Festplatteneinheiten). Zusätzlich können Hintergrundprozesse (d. h. Threads, welche mit Prozessen in dem Hintergrundzustand assoziiert sind) häufigkeitsbegrenzt in ihrem Zugriff auf Systemspeichereinheiten sein.
3. Eingehende Netzwerkaktivität: In einer Ausführungsform kann eingehender Netzwerkverkehr geleitet werden durch Fallenlassen aller Pakete, welche das Ziel haben, eine Anwendung in dem Hintergrund oder in einem nicht-ausführenden Zustand zu erreichen (z. B. ausgesetzt). Weiterhin können künstlich kleine Puffergrößen an Fernnetzwerkorte gemeldet werden, um eingehenden TCP(transmission control protocol)-Verkehr zu drosseln. Ein Fachmann wird erkennen, dass TCP einen existierenden Satz von Verkehrsflusssteuermechanismen bezogen auf Puffergrößen hat, sodass der Verkehr nicht schneller über ein Netzwerk gesendet wird, als es der Empfänger handhaben kann. Wie vorher angemerkt können in einer Ausführungsform künstlich kleine Puffergrößen gemeldet werden, sodass der Sender den Fluss von eingehendem Verkehr verlangsamt oder stoppt.

4. Ausgehende Netzwerkaktivität: In einer Ausführungsform wird Netzwerkzugriff vermittelt über die Verwendung von 2 Warteschlangen; eine für Aufgaben (Jobs), welche mit Vordergrundprozessen assoziiert sind, und eine für Aufgaben, welche mit Hintergrundprozessen assoziiert sind. Es ist als nützlich befunden worden, Vordergrundwarteschlangen Priorität über die Hintergrundwarteschlange zu geben. Beispielfhaft kann der Vordergrundwarteschlange 100% Priorität über die Hintergrundwarteschlange gegeben werden. Bei dieser Einstellung, ist der einzige Zeitpunkt, an dem eine Aufgabe in der Hintergrundwarteschlange bedient wird, wenn die Vordergrundwarteschlange leer ist. In einer anderen Ausführungsform können Aufgaben in der Vordergrundwarteschlange bedient werden in irgendeiner gewünschten Rate zu jenen der Hintergrundwarteschlange (z. B. 90/10, 80/20, 75/25 oder 50/50). In noch einer anderen Ausführungsform, sobald eine Aufgabe in eine Warteschlange gesetzt wurde (z. B. die Vordergrundwarteschlange wird nicht bewegt, wenn der assoziierte Prozess der Aufgabe anschließend in einen anderen Operationszustand gesetzt wird (z. B. Hintergrund). In noch einer anderen Ausführungsform wird die Warteschlangenposition einer Aufgabe aktualisiert, um den Operationsstatus seines assoziierten Prozesses widerzuspiegeln.

5. Graphische Operationen: In einer Ausführungsform wie z. B. dem iPhone®-Betriebssystem, gibt es mindestens zwei Mechanismen, durch welche eine Anwendung auf den Anzeigebildschirm schreiben oder malen kann (IPHONE ist eine registrierte Marke der Apple Inc.). Ein erster Mechanismus ist die Verwendung der spezialisierten Hardware, wie beispielsweise eine graphische Verarbeitungseinheit (GPU). Ein zweiter Mechanismus ist die Verwendung einer oder mehrerer der zentralen Verarbeitungseinheiten (CPU's) der Plattform. Gemäß einer Ausführungsform, wird es Hintergrundprozessen in der iPhone-Operationsumgebung nicht erlaubt auf spezialisierte graphische Hardware zuzugreifen; z. B. kann irgendein Hintergrundprozess, welcher versucht auf die GPU zuzugreifen, beendet werden. Durch ein anderes Beispiel, werden Hintergrundprozesse, welche versuchen sichtbare Graphiken über die CPU zu erzeugen, ignoriert – d. h. irgendein Versuch Befehle auszuführen, welche auf die Verwendung der CPU gerichtet sind, um auf einen Zeichnungspuffer/-speicher zu schreiben, werden ignoriert. In einigen Ausführungsformen kann ein Zeichnungsspeicher, welcher einem Prozess/einer Anwendung, welche/r in den Hintergrundzustand gesetzt wurde, assoziiert ist, markiert werden als „mobilisierbar“. Daher, wenn ein Vordergrundprozess mehr Speicher benötigt, kann er (durch Standard-Speicherzuordnungstechniken) Anzeigenspeicher verwenden, welcher ursprünglich einem Prozess gewährt wurde, welcher nun in dem Hintergrundzustand ist.

6. Benachrichtigungen: Wenn ein Prozess von dem Hintergrundzustand zu dem Vordergrundzustand übergeht, kann es ihm erlaubt werden 1 Benachrichtigung für jeden Typ von Benachrichtigung zu empfangen. Insbesondere, in einer Ausführungsform können mehrere Benachrichtigungsereignisse zusammengefasst werden. Z. B. wenn ein Nutzer seine persönliche elektronische Vorrichtung dreht (z. B. das iPhone) von der Porträt-darstellung zu Landschaftsdarstellung zu Porträt-darstellung und dann zurück zur Landschaftsdarstellung können die resultierenden 3 Orientierungsbenachrichtigungsereignisse in einer einzelnen Benachrichtigung vereinigt werden – den letzten Zustand, Landschaft. Dies kann die Antwort einer Anwendung verbessern, weil sie nicht mehreren Ereignissen antworten muss. Veranschaulichende Benachrichtigungstypen umfassen Orientierungsänderungsbenachrichtigungen (z. B. von einer Beschleunigungsmesser- oder Gyroskop-Sensoreingabe), Adressbuchdatenbankbenachrichtigungen (tritt auf, wenn ein Adressbucheintrag modifiziert wird) und Kamerarollbenachrichtigungen (camera roll notifications) (tritt auf, wenn ein Bild in einer geteilten Ressourcenbibliothek geändert worden ist).

7. Sperren: In einer Ausführungsform kann irgendein Prozess, welcher eine Sperre auf einer geteilten Ressource hat, welche in den ausgesetzten Zustand geht, beendet werden (anstatt ausgesetzt zu werden). Veranschaulichende geteilte Ressourcen umfassen spezialisierte graphische Hardware und Dateideskriptoren. In einer anderen Ausführungsform wird eine geteilte Ressource „weggenommen“ von einer Nutzeranwendung, wenn sie in den Hintergrundzustand geht. Z. B., wenn eine Anwendung (Prozess) eine Sperre auf Kamera-Hardware hat, wird diese Sperre entfernt, wenn die Nutzeranwendung in den Hintergrundzustand gesetzt wird. Wie hierin verwendet, ist der „Ausgesetzt-Zustand“ ein Zustand, in welcher es einem Prozess nicht erlaubt wird Threads zur Ausführung zu planen, aber dessen Zustand in dem Hauptspeicher behalten wird.

8. Hardwarebeschränkungen: In einer Ausführungsform werden Hintergrundprozesse daran gehindert Zugriff zu erhalten auf bestimmte Systemhardwareressourcen. Veranschaulichende Hardware, welche nicht verfügbar ist für Hintergrundprozesse, umfasst, aber ist nicht beschränkt auf: Kamera, GPU, Beschleunigungsmesser, Gyroskop, Näherungssensor und Mikrofon.

9. Speicherverwaltung: Um ausreichend Speicher zu behalten, um ausgeführte Prozesse unterzubringen, weil der Kernspeicher der persönlichen Vorrichtung einen spezifischen „kritischen“ Punkt erreicht, können Prozesse aus dem Speicher entfernt werden. In einer Ausführungsform kann ein Prozess, welcher so entfernt wird, in den Ruhezustand versetzt werden. In einer anderen Ausführungsform kann ein so entfernter Prozess beendet werden. Wie hierin verwendet bezeichnet der Ausdruck „Kernspeicher“ oder „Hauptspeicher“ einen Hauptspeicher mit wahlfreiem Zugriff der persönlichen elektronischen Vorrichtung, von welchem Nutzeranwendungen ausgeführt werden. Ein „in den Ruhezustand versetzter Prozess“ ist ein Pro-

zess, dessen operationeller Zustand auf nicht-flüchtigen Speicher geschrieben worden ist, sodass (1) er nicht Threads zur Ausführung planen kann und (2) sein Hauptspeicher-„Fußabdruck“ im Wesentlichen null ist; und ein „beendeter Prozess“ ist ein Prozess, dessen operationeller Zustand gelöscht worden ist und welcher, daher nicht ausgeführt werden kann, bis er erneut gestartet wird. In einer Ausführungsform, während der Ruhezustandshandlung, wird der Speicher, welcher erneut erzeugt werden kann (mobilisierbar) oder erneut geladen werden kann (sauber (clean)), auf Anforderung entsorgt; nur der Speicher, welcher modifiziert worden ist oder anderweitig nicht rekonstruierbar ist, wird auf den nicht-flüchtigen Speicher geschrieben. In einer anderen Ausführungsform wird eine spezifizierte Region des nicht-flüchtigen Speichers bei der Hochfahrzeit für die Verwendung als ein „Ruhezustands“-Speicher beiseitegelegt. Wenn ein in den Ruhezustand-gesetzter Prozess neu belebt wird, werden seine Daten in den nicht-flüchtigen Speicher zurück in den Hauptspeicher gebracht, wodurch Speicherplatz in dem nicht-flüchtigen Speicher frei wird. In einer Ausführungsform können Anwendungen auf welche nicht für mehr als eine spezifizierte Zeit zugegriffen wird, „in den Ruhezustand versetzt werden“.

**[0022]** Irgendeine gewünschte Herangehensweise für die Auswahl, welche Anwendungen aus dem Hauptspeicher zu entfernen sind, kann gemacht werden. Z. B. kann die am längsten nicht verwendete Anwendung als erstes entfernt werden. Alternativ kann die größte derzeit nicht ausgeführte Anwendung entfernt werden. In einer Ausführungsform können Anwendungen, welche für die Entfernung ausgewählt werden, welche größer sind als eine spezifizierte Größe (z. B. 8 MB, 16 MB oder 32 MB) beendet werden, während Anwendungen mit weniger als der spezifizierten Größe in den Ruhezustand versetzt werden können. In noch einer anderen Ausführungsform können Hintergrundaudionutzeranwendungen nur beendet werden, nachdem ausgesetzte Nutzeranwendungen beendet worden sind, aber vor Vordergrundnutzeranwendungen.

**[0023]** In ähnlicher Weise kann der „kritische“ Punkt irgendein Wert sein, welchen der Entwickler wünscht. Veranschaulichende kritische Punkte umfassen 50%, 75% und 90%. Wenn 75% als ein beispielhafter kritischer Punkt verwendet wird, werden Anwendung zur Entfernung aus dem Hauptspeicher ausgewählt werden, sobald der Hauptspeicher 75% seiner Kapazität erreicht – z. B. wenn ein 256 Megabyte(MB)-Hauptspeicher einen Füllstand von 192 Megabyte erreicht.

**[0024]** Operationszustände: Als eine Hilfe beim Verstehen, wie die obigen Beschränkungen auf die verschiedenen Dienste, welche unten beschrieben werden, angewendet werden, identifiziert und beschreibt die Tabelle 1 fünf (5) Operationszustände für einen Nutzerprozess, welcher auf einer persönlichen elektronischen Vorrichtung ausgeführt wird.

Tabelle 1: Operationszustände

Modus oder Zustand	Beschreibung	Beschränkungen
Vordergrund	Ein Prozess in diesem Zustand hat Zugriff auf Systemressourcen und präsentiert derzeit eine Nutzerschnittstelle einem Nutzer	Keine.
Hintergrund	Ein Prozess in diesem Zustand hat Zugriff auf Systemressourcen, aber präsentiert derzeit nicht eine Nutzerschnittstelle einem Nutzer	Echtzeitprozesse unterliegen Echtzeitbeschränkungen (d. h. sie sind degradiert auf Nicht-Echtzeitprioritäten, wenn sie in den Hintergrund gesetzt sind – außer für Audio- und VOIP-Netzwerkkommunikationsanwendungen). Alle anderen Prozesse unterliegen allen anderen Beschränkungen.
Gesperrt	Ein Prozess in diesem Zustand hat seinen operationellen Zustand im Hauptspeicher und könnte zur Ausführung geplant werden, außer der Tatsache, dass er ausgesetzt worden ist.	Nicht berechtigt zur Ausführung

In den Ruhezustand versetzt	Ein Prozess in diesem Zustand hatte seinen Operationszustand auf den nicht-flüchtigen Speicher geschrieben. Nur jene Informationen werden in dem Hauptspeicher behalten, welche notwendig sind zum Identifizieren, Orten und Neubeleben des Prozesses.	Nicht berechtigt zur Ausführung.
Beendet	Ein Prozess, dessen Operationszustand nicht länger im Hauptspeicher zugreifbar ist.	Nicht berechtigt zur Ausführung.

**[0025]** Wie hierin verwendet beschreibt der Ausdruck „nicht-ausgeführter Zustand“ einen der folgenden Zustände: ausgesetzt, Ruhezustand oder beendet.

**[0026]** Beispielhafte Dienste: Im Folgenden ist die „Umgebung“, innerhalb welcher die beschriebenen Dienste operieren, das iPhone-Betriebssystem. Dies ist nur veranschaulichend, die offenbarten Dienste können gleichwertig anwendbar sein auf irgendeiner persönlichen elektronischen Vorrichtung.

1. Schneller Aufgabenwechseldienst. In der Vergangenheit, wenn eine Anwendung den Vordergrundzustand in dem iPhone-Betriebssystem verlassen hat, wurde sie einfach beendet. Der schnelle Aufgabenwechseldienst erlaubt einer Anwendung, eine Benachrichtigung zu empfangen, dass sie den Vordergrundzustand verlässt und, zu jeder Zeit, wird ihr eine Gelegenheit gegeben, einiges an Verarbeitung zu machen (zum Beispiel einige Sekunden). Nach dieser Zeit wird die Anwendung ausgesetzt (d. h. in den ausgesetzten Zustand gesetzt). In einer Ausführungsform kann eine operierende Umgebung gemäß dieser Offenbarung schnellen Aufgabenwechsel auf alle ausführenden Prozesse anwenden (außer sie verwendet einen anderen Dienst wie unten beschrieben). In einer anderen Ausführungsform muss jede Anwendung programmatisch spezifizieren, dass sie ausgesetzt werden kann.

**[0027]** Während ausgesetzt, kann ein Prozess/eine Anwendung keine Anweisungen ausführen. In einer Betriebsumgebung mit Threads, wie beispielsweise Unix oder Mach, bedeutet dies, dass ein Prozess nicht einen Thread zur Ausführung planen kann. Jedoch in einigen Ausführungsformen können Systemkern-Bestrebungen, welche seitens eines Prozesses initiiert werden, fortsetzen auszuführen, selbst nachdem der Prozess selbst ausgesetzt worden ist (oder in den Ruhezustand versetzt worden ist oder beendet worden ist). Beispielhafte „Systemkern-Bestrebungen“ umfassen laufende I/O-Operationen und virtuelle Speicher- und Netzwerkpuffer-Operationen. Die Fertigstellung solcher Operationen kann oder kann nicht die Neubelebung des ausgesetzten Prozesses auslösen.

**[0028]** Es wird angemerkt, dass die Prozessaussetzung des Typs, welcher hier diskutiert wird, unterschiedlich ist zu den System-Ebenen-Operationen, wie beispielsweise Stand der Technik-„Schlaf-“ oder „Ruhezustands“-Aktionen – wovon beide „systemweite“ Operationen sind. Gemäß dem Stand der Technik kann kein Prozess ausgeführt werden, während er in dem Schlaf- oder Ruhezustands-Modus ist. Im Gegensatz dazu lässt sich der hierin beschriebene Aussetzmodus auf individuelle Prozesse anwenden; die Aussetzung eines ersten Prozesses beeinträchtigt nicht die Ausführung eines zweiten Prozesses.

2. Aufgabefertigstellungsdienst. Der Aufgabefertigstellungsdienst stellt einen Prozess mit der beschränkten Fähigkeit bereit, eine Aufgabe, welche in dem Vordergrundzustand begonnen wurde, fertigzustellen, während sie in dem Hintergrundzustand ist. In einer Ausführungsform werden Hintergrundprozesse (d. h. Prozesse, welche in dem Hintergrundzustand ausgeführt werden) gedrosselt, um das Beeinträchtigen von Vordergrundprozessen zu vermeiden. Beispielhafte Aufgaben, welche gut geeignet sind diesen Dienst zu verwenden, sind Netzwerkhochlade- und -herunterlade-Operationen, Plattenspeicher- und -abrufoperationen und Bildverarbeitungsoperationen (z. B. jene Operationen, die keine spezialisierte grafische Hardware, wie beispielsweise GPU, erfordern). In einer Ausführungsform wird es einem Hintergrundprozess erlaubt, die Verarbeitung fortzusetzen (d. h. Empfangen der CPU-Zeit gemäß seiner Planungspriorität wie oben diskutiert), bis ein spezifiziertes Ereignis auftritt. Beispielhafte Ereignisse umfassen einen System-Timer (welcher dem Prozess eine spezifizierte Zeitmenge gibt, um die Verarbeitung fertig zu stellen, z. B. 5 bis 30 Minuten der „Taktzeit (clock time)“ oder 5 Millisekunden der CPU-Zeit), Verlust der externen Leistung für die Vorrichtung, oder Aktivierung einer primären oder sekundären Anzeige.

**[0029]** Bezug nehmend auf [Fig. 2](#) beginnt die beispielhafte Aufgabefertigstellungsoperation **200**, wenn die Meldung, dass eine Nichtvordergrundanwendung (ZWEITE APP) in den Vordergrund zu bringen ist (Block **205**). Dies kann zum Beispiel aus einer Anfrage eines Nutzers resultieren, eine zweite Nutzeranwendung zu

starten, zu einer Systemanwendung überzugehen, oder einem eingehenden Aufruf an eine VOIP-Nutzeranwendung. Wenn die derzeitige Vordergrundnutzeranwendung (ERSTE APP) sich nicht selbst als eine Anwendung, welche fähig ist, im Hintergrund ausgeführt zu werden, bestimmt hat (das "NEIN"-Glied von Block **210**), wird die ERSTE APP in einen nicht-ausgeführten Zustand gesetzt (Block **215**) und die ZWEITE APP wird in den Vordergrund gebracht (Block **220**). Beispielfhaft kann die ERSTE APP ausgesetzt werden (d. h. in den ausgesetzten Zustand gesetzt werden). Wenn die ERSTE APP sich selbst als eine Anwendung, welche fähig ist, im Hintergrund ausgeführt zu werden, bestimmt hat, wie beispielsweise einen p-Listeneintrag (p list entry) (das "JA"-Glied von Block **210**), wird die ERSTE APP in den Hintergrundzustand gesetzt und ihr eine spezifizierte Menge an Zeit gegeben, ihre derzeitige Operation fertigzustellen (Block **225**). Die ZWEITE APP kann dann in den Vordergrund gebracht werden und ihr erlaubt werden, ausgeführt zu werden (Block **230**). Sobald im Hintergrund, kann die ERSTE APP geplant werden, um im notwendigen Verlauf ausgeführt zu werden (siehe die obige Beschreibung bezüglich Prozessor-Planung) (Block **235**). Der ERSTEN APP wird es erlaubt, im Hintergrund ausgeführt zu werden, bis ein spezifiziertes Ereignis auftritt und es seine Aufgabe nicht fertiggestellt hat. Sobald entweder das Ereignis auftritt oder der Prozess seine Aufgabe fertigstellt hat (das "JA"-Glied der Blöcke **240** und **245**), wird die ERSTE APP in einen nicht-ausgeführten Zustand gesetzt; zum Beispiel den ausgesetzten oder beendeten Zustand. Wenn keines dieser Ereignisse auftritt (die "NEIN"-Glieder der Blöcke **240** und **245**), setzt die ERSTE APP das Ausführen fort (Block **235**).

**[0030]** In einer Ausführungsform können nicht-aktive Vordergrundprozesse in ähnlicher Weise in ihrer Fähigkeit auf volle Systemressourcen zuzugreifen beschränkt werden. Zum Beispiel können ein oder mehrere nicht-aktive Vordergrundprozesse (d. h. ein Vordergrundprozess, welcher derzeit keine Nutzereingabe empfängt) ihren Zugriff auf Speicherplatten (Beschränkung 2), eingehende und ausgehende Netzwerkpuffer (Beschränkungen 3 und 4) und/oder ihre Fähigkeit, mit spezifizierter Hardware zu interagieren (Beschränkung 8), beschränkt in der gleichen Weise wie hier beschrieben für Hintergrundprozesse, haben. In ähnlicher Weise können bei Leistungsverlust alle Vordergrundprozesse außer dem aktiven Vordergrundprozess ausgesetzt oder in den Hintergrund "gepusht" werden, wo ihr Verhalten danach gemäß der Hintergrundverarbeitungsbehandlungsweise, welche hierin beschrieben wird, gesteuert wird.

**[0031]** In einer Ausführungsform muss ein Prozess programmatisch deklarieren, dass er sich des Aufgabefertigstellungsdienstes bedienen will (z. B. während der Handlungen gemäß Block **210**). Dies kann zum Beispiel gemacht werden durch statisches Deklarieren, dass die Nutzeranwendung sich des Aufgabefertigstellungsdienstes über einen API-Aufruf, einen p-Listeneintrag oder eine änderbare Einstellung, welche durch den Nutzer zur Installationszeit oder zur ersten Laufzeit spezifiziert wird, bedienen kann. In einer anderen Ausführungsform wird die Aufgabefertigstellung basierend auf dem Deklarieren einer Anwendung, zur Laufzeit und dem Code, dass sie eine Operation beginnt (dann anschließend deklarierend, dass sie jene Operation beendet hat), gewährt. Wenn solche Aufgaben noch ausstehend sind, wenn die Anwendung den Vordergrundzustand verlässt, kann der Anwendung einige Zeit, um diese zu beenden, gewährt werden. Bei Auftritt des spezifizierten Ereignisses, oder eher, wenn die Aufgabe/die Aufgaben fertiggestellt wird/werden, kann die Anwendung ausgesetzt werden. In noch einem Ausführungsprozess kann sich irgendein Prozess oder eine Nutzeranwendung, welche/r eine spezifizierte Zahl von berechtigten Aufgaben ausführt, sich des Aufgabefertigstellungsdienstes bedienen (z. B. einer der vorher genannten Operationen). Zum Beispiel könnte irgendein Prozess, welcher eine Dateihochlade- oder -herunterlade-Operation ausführt, in den Hintergrund gemäß den Aufgabefertigstellungsdienst-Beschränkungen gesetzt werden.

3. Audiodienst. Der Audiodienst erlaubt es einem Prozess oder einer Anwendung Audio abzuspielen oder aufzuzeichnen, während in dem Hintergrundzustand. In einer Ausführungsform verwenden Audioprozesse Echtzeit-Threads und werden entsprechend geplant (siehe [Fig. 1](#)). Während im Allgemeinen Echtzeitprozesse degradiert werden auf eine Nicht-Echtzeit-Priorität, wenn sie in den Hintergrund gesetzt werden, ist dies nicht zutreffend für eine Anwendung, welche den Audiodienst verwendet. Weil die beschriebenen Ausführungsformen Echtzeit-Threads für Audioanwendungen verwenden, um Beeinträchtigungen mit Vordergrundprozessen zu minimieren, kann nur ein Echtzeit-Audioprozess zu einer Zeit in dem Hintergrundzustand erlaubt werden. (Es ist klar, dass diese Zahl lediglich beispielhaft ist. Wenn die Verarbeitungsleistung der persönlichen elektronischen Vorrichtung ausreichend ist, können mehrere Audioprozesse in dem Hintergrund zu der gleichen Zeit aktiv sein.)

**[0032]** Bezug nehmend auf [Fig. 3](#) beginnt die beispielhafte Audiodienstoperation **300**, wenn eine Vordergrundnutzeranwendung, welche eine Audiooperation ausführt (z. B. Musikwiedergabe oder Stimmaufzeichnungsoperationen), eine Mitteilung empfängt, dass eine andere Anwendung, welche nicht derzeit in dem Vordergrund ist (ZWEITE APP), in den Vordergrund zu bringen ist (Block **305**). Die Audioanwendung des Nutzers wird dann in den Hintergrund gesetzt (Block **310**) und die ZWEITE APP wird in den Vordergrund gebracht (Block **315**). In einer Ausführungsform können Echtzeithintergrund-Threads beschränkt werden, um von einem

spezifischen Satz von Bibliotheken ausgeführt zu werden. Zum Beispiel können Echtzeithintergrund-Threads daran gehindert werden, Drittanbieter-Bibliotheken zu verwenden (z. B. Bibliotheken, welche durch einen Anwendungsentwickler bereitgestellt werden). Irgendwann in der Zukunft kann der Nutzer einen "Pause"-Befehl an seine Hintergrundaudioanwendung erteilen (Block **320**). In Antwort wird die Hintergrundaudioanwendung des Nutzers von dem Hintergrundzustand zu dem ausgesetzten Zustand bewegt (Block **325**). Die Audioanwendung wird in dem ausgesetzten Zustand gehalten, bis der Nutzer einen "Fortsetzen"-Befehl erteilt (das "JA"-Glieder von Block **330**), nachdem die Audioanwendung zurück in den Hintergrundzustand gebracht wird, wo sie mit der Ausführung ihrer Audiooperation fortsetzen kann (Block **335**). In einer anderen Ausführungsform kann die Audioanwendung in den Ruhezustandsmodus anstelle des ausgesetzten Modus gesetzt werden.

**[0033]** In einigen Ausführungsformen kann ein Prozess statisch erklären, dass er sich des Audiodienstes über einen API-Aufruf, einen p-Listeneintrag oder eine änderbare Einstellung, welche durch den Nutzer zur Installationszeit oder zur ersten Laufzeit spezifiziert wird, bedienen kann. In anderen Ausführungsformen kann irgendein Prozess, welcher entweder Audio abspielt oder aufzeichnet, den Audiodienst auf sich angewendet haben. Beim Ausführen von Code innerhalb dieser Anwendungen könnte der assoziierte Prozess nicht während der Audiooperationen gemäß der Schnellaufgabenwechsel- oder Aufgabenfertigstellungsdienste ausgesetzt werden.

4. Benachrichtigungsdienst. Der Benachrichtigungsdienst erlaubt es einem Prozess, in welchem Zustand auch immer (Vordergrund, Hintergrund, ausgesetzt oder Ruhezustand), Nachrichten zu empfangen und zu beantworten. Diese Nachrichten können von einer externen Quelle (einer "Push"-Benachrichtigung) oder dem empfangenen Prozess/der empfangenen Anwendung selbst ("lokale" Benachrichtigung) stammen.

**[0034]** Allgemein gesprochen, bei Nachrichtenempfang identifiziert ein Betriebssystemmodul (z. B. die "Springboard"-Anwendung in dem iPhone-Betriebssystem) die Zielanwendung (d. h. jene Anwendung, auf welche die Benachrichtigungsnachricht gerichtet ist), startet jene Anwendung im Hintergrund und gibt die Nachricht an die nun neu belebte Anwendung weiter. In einer anderen Ausführungsform, bei Nachrichtenempfang, erzeugt das Betriebssystem (z. B. Springboard) eine Dialogbox, über welche der Nutzer wählen kann, zu der Zielanwendung zu wechseln (d. h. die Anwendung, auf welche die Benachrichtigungsnachricht gerichtet ist). Wenn so gewählt, wird die Zielanwendung in den Vordergrund gebracht. (Es ist klar, dass, wenn die Zielanwendung bereits in dem Hintergrund ist, es nicht neu belebt werden muss.) Der Anwendung wird danach eine Gelegenheit gegeben, auf die Nachricht zu antworten, einschließlich des Präsentierens eines Dialogs an einen Nutzer über zum Beispiel eine Betriebssystemanfrage. Die Steuerung des empfangenden Prozesses wird dann bereitgestellt gemäß dem relevanten Dienst (z. B. Schnellaufgabenwechsel, Aufgabenfertigstellung, Audio, VOIP oder Ort und Geofencing). In einer anderen Ausführungsform können Benachrichtigungen, welche für einen Hintergrundprozess bestimmt sind, sich anstellen und zu einer späteren Zeit geliefert werden (z. B. alle 5 Minuten), statt im Wesentlichen sofort nach dem Benachrichtigungsempfang geliefert zu werden.

5. Netzwerkdienst. Der Netzwerkdienst erlaubt es einer ausgesetzten Netzwerkanwendung, ihre Kommunikationssocket(s) aufrechtzuerhalten (d. h. am Leben zu erhalten). Wie hierin verwendet ist eine Netzwerkanwendung eine Nutzeranwendung, welche Netzwerkkommunikations-Sockets verwendet. Eine beispielhafte Netzwerkanwendung ist eine Nutzer-VOIP-Anwendung. Im Stand der Technik, wenn eine Netzwerkanwendung/ein Netzwerkprozess eines Nutzers ausgesetzt wird, verliert sie ihre Fähigkeit, ihre Kommunikationssocket(s) aufrechtzuerhalten und als ein Ergebnis kann sie nicht den Empfang von Kommunikationsdaten fortsetzen (z. B. einen eingehenden VOIP-Telefonanruf).

**[0035]** Bezugnehmend auf [Fig. 4](#) wird eine beispielhafte Netzwerkdienstoperation durch die VOIP-Dienstoperation **400** demonstriert, welche mit einer VOIP-Anwendung des Nutzers beginnt, welche in den ausgesetzten Zustand überführt wird (Block **405**). In einer Ausführungsform, während ein VOIP-Prozess in den ausgesetzten Zustand überführt wird, spezifiziert er an das Betriebssystem (zum Beispiel über einen Netzwerk-Daemon) welches Zeitintervall er benötigt zum Senden von "Heartbeat"-Nachrichten (**410**). Dieser Daemon ist verantwortlich für das Neubeleben des Prozesses, wenn Daten bei den mit den Prozessen assoziierten Socket(s) oder bei Ablauf der spezifizierten Zeit empfangen werden – das Letztere für Heartbeat-Aufrechterhaltung. Für weitere Informationen über diesen Aspekt, siehe US-Patent-Veröffentlichung 20090305732 mit dem Titel Verwalten von Benachrichtigungsdienstverbindungen und Anzeigen von Icon-Badges (Managing Notification Service Connections and Displaying Icon Badges). Diese Anmeldung wird hierin in ihrer Gesamtheit aufgenommen.

**[0036]** Nach einer Weile empfängt der Netzwerk-Daemon eine Nachricht (Block **415**). Die Zielanwendung wird dann identifiziert (Block **420**), neu belebt in den Hintergrundzustand (Block **425**) und ihr wird die Nachricht weitergegeben (Block **430**). Die Hintergrund-VOIP-Anwendung kann dann auf die Nachricht antworten (zum Beispiel einen eingehenden VOIP-Anruf oder eine "Heartbeat erforderlich"-Nachricht) (Block **435**). Wenn es die erforderliche Aufgabe war, einen Heartbeat zu erstellen, kann die VOIP-Anwendung zu dem ausgesetzten

Zustand zurückgebracht werden. Wenn die erforderliche Aufgabe war, auf einen eingehenden VOIP-Anruf zu antworten, kann die VOIP-Anwendung das Betriebssystem anfragen, einen UI zu erstellen, sodass ein Nutzer wählen kann, ob er den Anruf annehmen oder ignorieren wird. Wenn der Anruf ignoriert wird, kann die VOIP-Anwendung zu dem ausgesetzten Zustand zurückgebracht werden. Wenn der Anruf zu beantworten ist, kann die VOIP-Anwendung in den Vordergrund gebracht werden (zum Beispiel durch das Wählen des geeigneten Elements in einer UI, welche durch das Betriebssystem präsentiert wird, durch Mittel des Nutzers). Es ist klar, dass Betriebssystem-Daemons wohl bekannt sind in der Technik, ist die Aufrechterhaltung der Netzwerkkommunikation (zum Beispiel VOIP und kabellose Netzwerkverbindung (Wireless Fidelity) oder WiFi) – Socket(s) für einen ausgesetzten Prozess in einer persönlichen elektronischen Vorrichtung des Typs, welcher hierin beschrieben wird, neu.

6. Orts- und Geofencing-Dienst. Der Orts- und Geofencing-Dienst erlaubt es einem Hintergrund- oder ausgesetzten Prozess, eine Mitteilung zu empfangen (zum Beispiel eine Nachricht von einem Betriebssystem-Daemon), wenn ein spezifiziertes Ortsereignis auftritt. Beispielhafte Ortsereignisse umfassen, sind aber nicht beschränkt auf das Ankommen an einem spezifizierten Ort, das Verlassen eines spezifizierten Ortes, das Betreten einer spezifizierten Region, das Verlassen einer spezifizierten Region, oder das sich eine "signifikante" Entfernung bewegen (wobei, was "signifikant" darstellt, spezifiziert werden kann durch den Nutzer oder die Nutzeranwendung). In einer Ausführungsform muss ein Prozess/eine Anwendung programmatisch deklarieren, dass er/sie sich des Orts- und Geofencing-Operationsdienstes bedienen will (zum Beispiel über einen p-Listenparameter). In einer anderen Ausführungsform erfordern die Geofencing-Operationen nicht die Verwendung eines p-Listeneintrags – alle Geofencing-Anwendungen können sich des Geofencing-Dienstes bedienen.

**[0037]** In Ortsoperationen, solange eine Ortsverfolgungsoperation im Gange ist, kann die Anwendung im Hintergrund gehalten werden – verarbeitungszeiterhaltend gemäß den Systemplanungsmechanismen (siehe obige Diskussion). In einer Ausführungsform, wenn eine Verfolgungsoperation entweder beendet wird oder anderweitig angehalten wird, kann der Prozess ausgesetzt werden (das heißt in den ausgesetzten Zustand gesetzt werden; der Prozess könnte auch beendet oder in den Ruhezustand versetzt werden). Sobald der Prozess auf die Nachricht antwortet, kann er stillgelegt werden (in den ausgesetzten Zustand zurückkehren) oder in dem Hintergrundzustand blockieren, auf die nächste Nachricht wartend. In einer Ausführungsform bezieht sich die Zahl der Nutzeranwendungen, welche Ortsdienste in dem Hintergrundzustand verwenden, auf die Verwendung anderer Systemressourcen; zum Beispiel, wenn System Speicher knapp wird, können Nutzeranwendungen, welche derzeit Ortsverfolgung ausführen, beendet werden. In einer anderen Ausführungsform sammelt ein zentralisierter Daemon alle der Ortsanfragen (insbesondere mit Bezug auf die gewünschte Genauigkeit, die angefragt wird), und startet geeignete Hardware für den aggressivsten Modus, welcher angefragt wird. Zum Beispiel, wenn eine Nutzeranwendung den Ort des Nutzers auf innerhalb von drei Kilometer anfragt (in welchem Fall Mobiltelefonzell-basierte Positionierung verwendet werden kann), fragt eine zweite Nutzeranwendung den Ort des Nutzers auf innerhalb von 500 Meter an (in welchem Fall WiFi-Scanning verwendet werden kann), und eine dritte Anwendung fragt den Ort des Nutzers auf innerhalb von 100 Metern an (für welche GPS notwendig ist), die GPS-Positionierung kann verwendet werden und alle Anwendungen, welche Orte in dem Hintergrund verfolgen, können Vorteil ziehen aus der genaueren Positionierung.

**[0038]** In Geofencing-Operationen, immer wenn eine spezifizierte Regionsgrenze durchquert wird, empfängt der Prozess eine Nachricht von dem Betriebssystem (zum Beispiel über einen Daemon). Wenn der Prozess ausgeführt wird (zum Beispiel während seiner zugeordneten Zeit, während im Hintergrund), würde er die Gelegenheit bekommen, die Nachricht zu verarbeiten. Wenn die Anwendung nicht verarbeiten würde (zum Beispiel sie war im Hintergrund oder nicht verarbeitend oder sie wurde ausgesetzt oder in den Ruhezustand versetzt), würde der Prozess gleich die Anwendung neu beleben, nachdem es ihr erlaubt würde, das Ereignis zu verarbeiten (im Hintergrund). Zusätzlich, wenn der Prozess beendet worden war, kann er in dem Hintergrundzustand auf eine Weise, ähnlich dem Benachrichtigungsdienst, welcher vorher beschrieben wurde, neu belebt werden. Siehe die verwandte Offenbarung mit dem Titel "Ortsbasierte Anwendungsprogrammverwaltung (Location Based Application Program Management)", Nummer 12/123,456. In einer Ausführungsform, wo Ortsverarbeitung einen separaten Prozessor nutzt als jenen, welcher das Betriebssystem für die Nutzeranwendungsausführung ausführt (in der Folge bezeichnet als "Anwendungsprozessor"), können die Orts- und Geofencing-Operationen des Typs, welcher hierin beschrieben wird, auftreten, selbst wenn der Anwendungsprozessor schläft. So wie mit Ortsdiensten bezieht sich die Zahl der Nutzeranwendungen, welche sich des Geofencing-Dienstes in dem Hintergrundzustand bedienen können, auf die Verwendung anderer Systemressourcen; zum Beispiel, wenn der System Speicher knapp wird, können Nutzeranwendungen, welche derzeit Ortsverfolgung ausführen, beendet werden.

**[0039]** Persönliche elektronische Vorrichtungensarchitektur: Bezugnehmend auf [Fig. 5](#), in einer Ausführungsform könnte die persönliche elektronische Vorrichtung **500** eine oder mehrere der obigen Dienste unter Verwendung einer oder mehrerer der oben beschriebenen Beschränkungen implementieren, um Multitasking-Operationen zu erlauben, wenn es vorteilhaft ist, ein Solches zu tun (zum Beispiel Opportunistisches Multitasking). Wie veranschaulicht, interagieren Nutzeranwendungen **505A** bis **505N** mit einem Anwendungsstarter **510** (zum Beispiel "Springboard" in dem iPhone-Betriebssystem) für Standardbetriebssystemaufrufe und Multitasking-Dienstbibliothek **515**, um auf die oben beschriebenen Dienste zuzugreifen und diese zu verwenden.

**[0040]** Bezugnehmend auf [Fig. 6](#) könnte System **600** gemäß einer anderen Ausführungsform Funkmittel **605** und Ortsverarbeitungsmittel **610** umfassen. Wie in [Fig. 7](#) gezeigt, können die Funkmittel **605** eine Empfängerschaltung **700**, eine Senderschaltung **705** und Speicher **710** umfassen. Empfänger- und Senderschaltungen **700** und **705** können verwendet werden, um zum Beispiel mit einem Mobiltelefonnetzwerk, einem Satellitenkommunikationsnetzwerk oder einem digitalen Computernetzwerk zu verbinden (zum Beispiel das Internet über eine lokale Bereichsnetzwerk-WiFi-Verbindung). Der Speicher **710** kann verwendet werden, um empfangene Daten, Daten, welche zu übertragen sind, und operationelle Parameterinformationen, wie in der Technik bekannt, zu speichern. Eine Schnittstelle **715** kann bereitgestellt werden, um Kommunikationen zwischen Funkmitteln **605** und Ortsmitteln **610** zu vereinfachen. Die Ortsmittel **610** können einen Prozessor **720**, einen Speicher **725** und eine Betriebssystemschnittstelle **730** umfassen. Im Allgemeinen empfangen die Ortsmittel **610** Anfragen für Ortsaktualisierungen und Geofencing-Beschränkungen (zum Beispiel von einem oder mehreren Nutzeranwendungen **505A** bis **505N** über Schnittstelle **730**), verarbeiten Ortsinformationen, welche von den Funkmitteln **605** empfangen werden (empfangen über Schnittstelle **715**) und geben Ortsinformationen zurück und aktualisieren an die anfragende(n) Anwendung(en) über die Schnittstelle **730**. Der Prozessor **720** kann einer oder mehrere handelsübliche Prozessoren, einer oder mehrere auf Kundenbedürfnisse zugeschnittene Prozessoren oder eine Kombination von handelsüblichen und auf Kundenbedürfnisse zugeschnittene Prozessoren sein. Der Speicher **725** kann irgendeine Form oder Kombination von flüchtigen und nicht-flüchtigen Festkörper-, magnetischen oder optischen Speichern sein.

**[0041]** In einer Ausführungsform können ein oder mehrere der beschriebenen Dienste bereitgestellt werden, basierend auf der Verfügbarkeit von Leistung. Zum Beispiel, wenn die persönliche elektronische Vorrichtung in eine Stromquelle gesteckt ist (zum Beispiel Wechselstrom), können Multitasking-Fähigkeiten bereitgestellt werden gemäß den standardmodernen Betriebssystemprozeduren (d. h. volles oder nicht beschränktes Multitasking). Wenn, auf der anderen Seite, die persönliche elektronische Vorrichtung nicht in eine Stromquelle gesteckt ist, können Multitasking-Fähigkeiten bereitgestellt werden gemäß dieser Offenbarung (zum Beispiel teilweises oder opportunistisches Multitasking).

**[0042]** Bezugnehmend auf [Fig. 8](#) kann eine persönliche elektronische Vorrichtung **800** verkörpert sein in einer physischen Vorrichtung, welche eine Verarbeitungseinheit **805** und Unterstützungsvorrichtungen **810** umfasst. Die Verarbeitungseinheit **805** kann einen Prozessor **815**, einen Speicher **820** und eine Eingabe-Ausgabe-Schaltung **825** umfassen. Die Vorrichtung **800** kann auch Funkmittel **605** und Ortsmittel **610**, wie gezeigt, umfassen. Die Eingabe-Ausgabe-Schaltung **825** koppelt den Prozessor **815** und den Speicher **820** an eine oder mehrere Anzeigeeinheiten **830** (zum Beispiel eine Berührungsbildschirmanzeige), eine oder mehrere Eingabe-Ausgabe-Vorrichtungen **835** (zum Beispiel Berührungsbildschirm, Tastatur, Maus, Joystick), eine oder mehrere Programmspeichervorrichtungen **840** (zum Beispiel magnetische, optische oder Festkörperspeicher) und andere Hardware **845** (zum Beispiel eine Netzwerkschnittstelle). Es ist klar, dass in der Architektur, welche in [Fig. 6](#), [Fig. 7](#) und [Fig. 8](#) präsentiert wird, Orts- und Geofencing-Verarbeitung über Funkmittel **605** und Ortsmittel **610** fortgesetzt werden kann, selbst wenn die Anwendung, welche die Orts- oder Geofencing-Anfrage macht (zum Beispiel die Anwendung **505A**), ausgesetzt worden ist, in den Ruhezustand versetzt worden ist oder beendet worden ist. Noch weiterhin können Orts- und Geofencing-Operationen fortlaufen, selbst wenn der Anwendungsprozessor **815** im Ruhezustand ist oder schläft. Das heißt, die Orts- und Geofencing-Operationen können fortgesetzt werden über Prozessor **720**, unabhängig davon, in welchem Zustand der Anwendungsprozessor **815** ist.

**[0043]** Verschiedene Änderungen in den Materialien, Komponenten, Schaltelementen sowie in den Details der beispielhaften operationellen Verfahren sind möglich ohne von dem Geltungsbereich der folgenden Ansprüche abzuweichen. Die Bereitstellung von Multitasking-Fähigkeiten gemäß dieser Offenbarung wird ausgeführt durch eine programmierbare Steuervorrichtung, welche Anweisungen ausführt, welche in einem oder mehreren Programmmodulen organisiert sind. Eine programmierbare Steuervorrichtung (zum Beispiel Prozessoren **720** und **215**) kann irgendeine programmierbare Steuervorrichtung umfassen, einschließlich zum Beispiel eines oder mehrerer Mitglieder der Intel Atom<sup>®</sup>-, Core<sup>®</sup>-, Pentium<sup>®</sup>- und Celeron<sup>®</sup>-Prozessorfamilien von Intel Corporation und der Cortex- und ARM-Prozessorfamilien von ARM oder auf Kundenbedürfnisse zugeschnittene

ne Zustandsmaschinen. (INTEL, INTEL ATOM, CORE, PENTIUM, und CELERON sind registrierte Marken der Intel Corporation. CORTEX ist eine registrierte Marke der ARM Limited Corporation. ARM ist eine registrierte Marke der ARM Limited Company.) Auf Kundenbedürfnisse zugeschnittene Zustandsmaschinen können verkörpert sein in einer Hardwarevorrichtung, wie beispielsweise einem integrierten Schaltkreis, einschließlich, aber nicht beschränkt auf anwendungsspezifische integrierte Schaltkreise ("ASICs") oder feldprogrammierbare Gate Arrays ("FPGAs").

**[0044]** Speichervorrichtungen, welche für konkretes Verkörpern von Programmanweisungen geeignet sind (zum Beispiel Speicher **710**, **725**, **220** und **820**) umfassen, sind aber nicht beschränkt auf: magnetische Platten (befestigte, Disketten und entfernbare) und Band; optische Medien, wie beispielsweise CD-ROMs und digitale Video Disks ("DVDs"); und Halbleiterspeichervorrichtungen wie beispielsweise elektrische programmierbare Nur-Lese-Speicher ("EPROM"), elektrisch löschbare programmierbare Nur-Lese-Speicher ("EEPROM"), programmierbare Gate Arrays und Flash Vorrichtungen.

**[0045]** Schließlich ist klar, dass die obige Beschreibung veranschaulichend und nicht beschränkend sein soll. Zum Beispiel können die oben beschriebenen Ausführungsformen in Kombination miteinander verwendet werden. Viele andere Ausführungsformen werden dem Fachmann nach Lesen der obigen Beschreibung offensichtlich werden. Der Geltungsbereich der Erfindung sollte daher mit Verweis auf die angehängten Ansprüche bestimmt werden, zusammen mit dem vollen Geltungsbereich der Äquivalente, für welche solche Ansprüche berechtigt sind. In den angehängten Ansprüchen werden die Ausdrücke "einschließlich" und "in welchen" verwendet als die einfachen englischen Äquivalente der entsprechenden Ausdrücke "aufweisend" und "wobei".

**ZITATE ENTHALTEN IN DER BESCHREIBUNG**

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

**Zitierte Patentliteratur**

- US 20090305732 [\[0035\]](#)

**Patentansprüche**

1. Multitasking-Verfahren, aufweisend:  
Identifizieren einer ersten Nutzeranwendung, welche in einem Vordergrundzustand ausgeführt wird;  
Empfangen einer Nutzereingabe, welche anweist, dass eine zweite Nutzeranwendung in den Vordergrundzustand zu versetzen ist;  
Setzen der ersten Nutzeranwendung in einen Hintergrundzustand in Antwort auf die empfangene Nutzereingabe, nur wenn die erste Nutzeranwendung sich selbst als eine Nutzeranwendung identifiziert hat, welche in dem Hintergrundzustand ausgeführt werden kann; und  
Setzen der zweiten Nutzeranwendung in den Vordergrundzustand.
2. Verfahren gemäß Anspruch 1, weiterhin aufweisend das Versetzen der ersten Nutzeranwendung in einen nicht-ausgeführten Zustand, wenn die erste Nutzeranwendung sich nicht als eine Nutzeranwendung identifiziert hat, welche in dem Hintergrundzustand ausgeführt werden kann.
3. Verfahren gemäß Anspruch 1, weiterhin aufweisend das Versetzen der ersten Nutzeranwendung in einen nicht-ausgeführten Zustand, nachdem sie in den Hintergrundzustand gesetzt worden ist, wenn die erste Nutzeranwendung versucht auf eine spezifizierte Hardwareressource zuzugreifen.
4. Verfahren gemäß Anspruch 3, wobei eine spezifizierte Hardwareressource eine Grafikverarbeitungseinheit aufweist.
5. Verfahren gemäß Anspruch 3, wobei eine spezifizierte Hardwareressource eine Kameravorrichtung aufweist.
6. Verfahren gemäß Anspruch 1, weiterhin aufweisend Bewegen der ersten Nutzeranwendung in einen ausgesetzten Zustand, nachdem der erste Nutzerprozess eine Aufgabe fertigstellt, wobei die Aufgabe die Aufgabe ist, welche die erste Nutzeranwendung ausführte, als sie in den Hintergrundzustand gesetzt wurde.
7. Verfahren gemäß Anspruch 1, weiterhin aufweisend Bewegen der ersten Nutzeranwendung in einen ausgesetzten Zustand, nachdem der erste Nutzerprozess eine spezifische Zeitmenge ausgeführt hat, während in dem Hintergrundzustand.
8. Verfahren gemäß Anspruch 1, wobei der erste Nutzerprozess zur Zeit, als der erste Nutzerprozess in den Hintergrundzustand bewegt wurde, eine Operation ausführt, welche ausgewählt wird aus der Gruppe bestehend aus: eine Netzwerkhochlade-Operation, eine Netzwerkherunterlade-Operation, eine Plattenspeicheroperation und eine Plattenabrufoperation.
9. Verfahren gemäß Anspruch 1, weiterhin aufweisend Bewegen der ersten Nutzeranwendung aus dem Hintergrundzustand in einen beendeten Zustand, wenn die erste Nutzeranwendung eine Sperre auf eine geteilte Systemressource erhält.
10. Verfahren gemäß Anspruch 9, wobei die geteilte Systemressource einen geteilten Dateideskriptor aufweist.
11. Programmspeichervorrichtung, lesbar durch eine programmierbare Steuervorrichtung, aufweisend darauf gespeicherte Anweisungen zum Veranlassen der programmierbaren Steuervorrichtung ein Verfahren gemäß Anspruch 1 auszuführen.
12. Persönliche elektronische Vorrichtung, aufweisend:  
einen Speicher;  
Computerprogrammanweisungen, welche in dem Speicher gespeichert werden; und  
eine programmierbare Steuervorrichtung, welche operativ mit dem Speicher gekoppelt ist und eingerichtet ist zum Ausführen der Computerprogrammanweisungen, um die programmierbare Steuervorrichtung zu veranlassen:  
eine erste Nutzeranwendung, welche in einem Vordergrundzustand ausgeführt wird, zu identifizieren;  
eine Nutzereingabe, welche anweist, dass eine zweite Nutzeranwendung in den Vordergrundzustand zu Versetzen ist, zu empfangen;

die erste Nutzeranwendung in einen Hintergrundzustand zu Versetzen in Antwort auf die empfangene Nutzereingabe, nur wenn die erste Nutzeranwendung sich selbst als eine Nutzeranwendung identifiziert hat, welche in dem Hintergrundzustand ausgeführt werden kann; und  
die zweite Nutzeranwendung in den Vordergrundzustand zu versetzen.

13. Persönliche elektronische Vorrichtung gemäß Anspruch 12, wobei die persönliche elektronische Vorrichtung eine persönliche elektronische Vorrichtung, ausgewählt aus der Gruppe bestehend aus: einem Mobiltelefon, einem persönlichen digitalen Assistenten, einer handgehaltenen Unterhaltungsvorrichtung, einem Pad-Computersystem und einer Set-Top-Box.

14. Multitasking-Verfahren für eine persönliche elektronische Vorrichtung, aufweisend:  
Identifizieren einer ersten Nutzeranwendung, welche eine Audiooperation in einem Vordergrundzustand auf einer persönlichen elektronischen Vorrichtung ausführt;  
Empfangen einer ersten Nutzereingabe von der persönlichen elektronischen Vorrichtung, welche identifiziert, dass eine zweite Nutzeranwendung in den Vordergrundzustand zu Versetzen ist;  
Versetzen der ersten Nutzeranwendung in einen Hintergrundzustand in Antwort auf die erste empfangene Nutzereingabe, wobei die erste Nutzeranwendung fortsetzt, die Audiooperation auszuführen;  
Versetzen der zweiten Nutzeranwendung in den Vordergrundzustand;  
Empfangen der zweiten Nutzereingabe, welche anweist, dass die erste Nutzeranwendung pausiert wird; und  
Versetzen der ersten Nutzeranwendung in einen nicht-ausgeführten Zustand in Antwort auf die zweite empfangene Nutzereingabe.

15. Verfahren gemäß Anspruch 14, wobei die Handlung des Versetzens der ersten Nutzeranwendung in einen nicht-ausgeführten Zustand das Versetzen der ersten Nutzeranwendung in einen ausgesetzten Zustand aufweist.

16. Verfahren gemäß Anspruch 14, weiterhin aufweisend:  
Empfangen einer dritten Nutzereingabe, welche anweist, dass die erste Nutzeranwendung fortzusetzen ist; und  
Bewegen der ersten Nutzeranwendung von dem nicht-ausgeführten Zustand in den Hintergrundzustand, um das Ausführen der Audiooperation fortzusetzen.

17. Verfahren gemäß Anspruch 14, wobei die Handlung des Ausführens einer Audiooperation das Erzeugen eines Audiosignals aufweist.

18. Verfahren gemäß Anspruch 14, wobei die Handlung des Ausführens einer Audiooperation das Aufzeichnen eines Audiosignals aufweist.

19. Programmspeichervorrichtung, lesbar durch eine programmierbare Steuervorrichtung, aufweisend darauf gespeicherte Anweisungen zum Veranlassen der programmierbaren Steuervorrichtung, ein Verfahren gemäß Anspruch 14 auszuführen.

20. Persönliche elektronische Vorrichtung, aufweisend:  
einen Speicher;  
Computerprogramminstruktionen, welche in dem Speicher gespeichert sind;  
eine programmierbare Steuervorrichtung, welche operativ mit dem Speicher gekoppelt ist, wobei die programmierbare Steuervorrichtung eingerichtet ist zum Ausführen von Computerprogrammanweisungen zum:  
Empfangen einer ersten Nutzereingabe von einer persönlichen elektronischen Vorrichtung, welche identifiziert, dass eine zweite Nutzeranwendung in den Vordergrundzustand zu versetzen ist;  
Versetzen der ersten Nutzeranwendung in einen Hintergrundzustand in Antwort auf die erste empfangene Nutzereingabe, wobei die erste Nutzeranwendung das Ausführen der Audiooperation fortsetzt;  
Versetzen der zweiten Nutzeranwendung in den Vordergrundzustand;  
Empfangen der zweiten Nutzereingabe, welche anweist, dass die erste Nutzeranwendung pausiert wird; und  
Versetzen der ersten Nutzeranwendung in einen nicht-ausgeführten Zustand in Antwort auf die zweite empfangene Nutzereingabe.

21. Persönliche elektronische Vorrichtung gemäß Anspruch 20, aufweisend eine persönliche elektronische Vorrichtung, welche ausgewählt wird aus der Gruppe bestehend aus: einem Mobiltelefon, einem persönlichen digitalen Assistenten, einer handgehaltenen Unterhaltungsvorrichtung, einem Pad-Computersystem und einer Set-Top-Box.

22. Multitasking-verfahren, aufweisend:

Empfangen, auf einer persönlichen elektronischen Vorrichtung, einer Nachricht für eine ausgesetzte Netzwerkkommunikationsanwendung;  
Identifizieren, auf der persönlichen elektronischen Vorrichtung, einer spezifischen Netzwerkkommunikationsanwendung;  
Bewegen der identifizierten Netzwerkkommunikationsanwendung in einen Hintergrundzustand; und  
Bereitstellen der Nachricht an die identifizierte Netzwerkkommunikationsanwendung in dem Hintergrund.

23. Verfahren gemäß Anspruch 22, wobei die Netzwerkkommunikationsanwendung eine Voice Over Internet Protocol(VOIP)-Anwendung aufweist.

24. Verfahren gemäß Anspruch 22, wobei die Netzwerkkommunikationsanwendung eine Wireless Fidelity (WiFi)-Anwendung aufweist.

25. Verfahren gemäß Anspruch 22, weiterhin aufweisend Bewegen der spezifischen Netzwerkkommunikationsanwendung von dem Hintergrundzustand zu einem ausgesetzten Zustand, nachdem die spezifische Netzwerkkommunikationsanwendung auf die Nachricht geantwortet hat.

26. Programmspeichervorrichtung, lesbar durch eine programmierbare Steuervorrichtung, aufweisend darauf gespeicherte Instruktionen zum Veranlassen der programmierbaren Steuervorrichtung, ein Verfahren gemäß Anspruch 22 auszuführen.

27. Persönliche elektronische Vorrichtung, aufweisend:

einen Speicher;

einen programmierbaren Steuerprozessor operativ gekoppelt mit dem Speicher, wobei der Prozessor eingerichtet ist zum Ausführen der Anweisungen in dem Speicher zum Veranlassen des programmierbaren Steuerprozessors zum:

Empfangen, auf einer persönlichen elektronischen Vorrichtung, einer Nachricht für eine ausgesetzte Netzwerkkommunikationsanwendung;  
Identifizieren, auf der persönlichen elektronischen Vorrichtung, einer spezifischen Netzwerkkommunikationsanwendung;  
Bewegen der identifizierten Netzwerkkommunikationsanwendung in einen Hintergrundzustand; und  
Bereitstellen der Nachricht an die identifizierte Netzwerkkommunikationsanwendung in dem Hintergrund.

28. Persönliche elektronische Vorrichtung gemäß Anspruch 27, aufweisend eine persönliche elektronische Vorrichtung, welche ausgewählt wird aus der Gruppe bestehend aus: einem Mobiltelefon, einem persönlichen digitalen Assistenten, einer handgehaltenen Unterhaltungsvorrichtung, einem Pad-Computersystem und einer Set Top Box.

29. Multitasking-Verfahren, aufweisend:

Identifizieren einer ersten Nutzeranwendung, welche in einem Vordergrundzustand ausgeführt wird;

Empfangen einer Nutzereingabe, welche anweist, dass eine zweite Anwendung in den Vordergrundzustand zu versetzen ist;

Beenden der ersten Nutzeranwendung, wenn die erste Nutzeranwendung mindestens eine Sperre auf einer geteilten Systemressource hat, sonst Versetzen der ersten Nutzeranwendung in einen Hintergrundzustand in Antwort auf die empfangene Nutzereingabe, nur wenn die erste Nutzeranwendung sich selbst als eine Nutzeranwendung identifiziert hat, welche in dem Hintergrundzustand ausgeführt werden kann; und  
Versetzen der zweiten Anwendung in den Vordergrundzustand.

30. Verfahren gemäß Anspruch 29, weiterhin aufweisend Planen der ersten Nutzeranwendung, nachdem die erste Nutzeranwendung in den Hintergrundzustand gesetzt worden ist, um einen Prozessor gemäß einer Prioritätsebene zu verwenden, wobei die Prioritätsebene aus einem Bereich von Prioritätsebenen, welche mit einigen, aber nicht allen der Prioritätsebenen, welche an die Vordergrundprozesse zugewiesen sind, ausgewählt wird.

31. Verfahren gemäß Anspruch 29, weiterhin aufweisend Beenden der ersten Nutzeranwendung, nachdem die erste Nutzeranwendung in den Hintergrundzustand gesetzt worden ist, wenn die erste Nutzeranwendung versucht, Anweisungen auszuführen, welche auf eine spezifizierte Hardwareressource zielen.

32. Verfahren gemäß Anspruch 31, wobei die spezifizierten Hardwareressourcen eine grafische Verarbeitungseinheit aufweisen.

33. Verfahren gemäß Anspruch 29, weiterhin aufweisend Nicht-Ausführen von Instruktionen für die erste Nutzeranwendung, sobald die erste Nutzeranwendung in dem Hintergrundzustand ist, wenn die Anweisungen auf das Ändern eines Anzeigespeichers gerichtet sind.

34. Verfahren gemäß Anspruch 29, wobei die geteilten Systemressourcen einen geteilten Dateideskriptor aufweisen.

35. Verfahren gemäß Anspruch 29, wobei die Handlung des Versetzens der zweiten Anwendung in den Vordergrundzustand ein Versetzen einer Betriebssystemanwendung in den Vordergrundzustand aufweist.

36. Programmspeichervorrichtung, lesbar durch eine programmierbare Steuervorrichtung, aufweisend darauf gespeicherte Anweisungen, zum Veranlassen der programmierbaren Steuervorrichtung ein Verfahren gemäß Anspruch 29 auszuführen.

37. Persönliche elektronische Vorrichtung, aufweisend:

einen Speicher;

eine programmierbare Steuervorrichtung operativ gekoppelt mit dem Speicher, wobei die programmierbare Steuervorrichtung eingerichtet ist zum Ausführen von Instruktionen, welche in dem Speicher gespeichert sind, um:

eine erste Nutzeranwendung, welche in einem Vordergrundzustand ausgeführt wird, zu identifizieren;

eine Nutzereingabe, welche anweist, dass eine zweite Anwendung in den Vordergrundzustand zu versetzen ist, zu empfangen;

die erste Nutzeranwendung zu beenden, wenn die erste Nutzeranwendung mindestens eine Sperre auf einer geteilten Systemressource hat, sonst Versetzen der ersten Nutzeranwendung in einen Hintergrundzustand in Antwort auf die empfangene Nutzereingabe, nur wenn die erste Nutzeranwendung sich selbst als eine Nutzeranwendung identifiziert hat, welche in dem Hintergrundzustand ausgeführt werden kann; und die zweite Anwendung in den Vordergrundzustand zu versetzen.

38. Persönliche elektronische Vorrichtung gemäß Anspruch 37, aufweisend eine persönliche elektronische Vorrichtung, welche ausgewählt wird aus der Gruppe bestehend aus: einem Mobiltelefon, einem persönlichen digitalen Assistenten, einer handgehaltenen Unterhaltungsvorrichtung, einem Pad-Computersystem und einer Set Top Box.

39. Persönliche elektronische Vorrichtung gemäß Anspruch 37, wobei die zweite Anwendung eine zweite Nutzeranwendung aufweist.

40. Multitasking-Verfahren, aufweisend:

Initiieren einer ersten Nutzer-Voice Over Internet Protocol (VOIP)-Anwendung zu einer ersten Zeit;

Instanzieren einer oder mehrerer Kommunikations-Sockets für die erste Nutzer-VOIP-Anwendung;

Versetzen der ersten Nutzer-VOIP-Anwendung in einen nicht-ausgeführten Zustand zu einer zweiten Zeit;

Empfangen einer Benachrichtigung, welche mit der einen oder den mehreren Kommunikations-Sockets assoziiert ist, zu einer dritten Zeit;

Versetzen der ersten Nutzer-VOIP-Anwendung in einen Hintergrundzustand in Antwort auf die empfangene Benachrichtigung; und

Zurückbringen der ersten Nutzer-VOIP-Anwendung in den nicht-ausgeführten Zustand, nachdem die erste Nutzer-VOIP-Anwendung auf die Benachrichtigung geantwortet hat, wobei die erste Zeit der zweiten Zeit vorangeht und die zweite Zeit der dritten Zeit vorangeht.

41. Verfahren gemäß Anspruch 40, wobei der nicht-ausgeführte Zustand ausgewählt wird aus der Gruppe bestehend aus: ausgesetzter Zustand und Ruhezustand.

42. Verfahren gemäß Anspruch 40, wobei die Handlung des Versetzens der ersten Nutzer-VOIP-Anwendung in einen nicht-ausgeführten Zustand das Versetzen der ersten Nutzer-VOIP-Anwendung in einen ausgesetzten Zustand aufweist.

43. Programmspeichervorrichtung, lesbar durch eine programmierbare Steuervorrichtung, aufweisend darauf gespeicherte Anweisungen zum Veranlassen der programmierbaren Steuervorrichtung, das Verfahren gemäß Anspruch 40 auszuführen.

44. Persönliche handgehaltene elektronische Vorrichtung, aufweisend:  
einen Speicher;  
einen Empfänger, welcher operativ mit dem Speicher gekoppelt ist;  
einen Sender, welcher operativ mit dem Speicher gekoppelt ist; und  
eine programmierbare Steuervorrichtung, welche operativ mit dem Speicher gekoppelt ist und eingerichtet ist zum Ausführen der darauf gespeicherten Anweisungen, wobei die Anweisungen die programmierbare Steuervorrichtung veranlassen zum:  
Initiieren einer ersten Nutzer-Voice Over Internet Protocol (VOIP)-Anwendung zu einer ersten Zeit;  
Instanzieren einer oder mehrerer Kommunikations-Sockets für die erste Nutzer VOIP-Anwendung;  
Versetzen der ersten Nutzer-VOIP-Anwendung in einen nicht-ausgeführten Zustand zu einer zweiten Zeit;  
Empfangen einer Benachrichtigung, welche mit der einen oder den mehreren Kommunikations-Sockets assoziiert ist, zu einer dritten Zeit;  
Versetzen der ersten Nutzer-VOIP-Anwendung in einen Hintergrundzustand in Antwort auf die empfangene Benachrichtigung; und  
Zurückbringen der ersten Nutzer-VOIP-Anwendung in den nicht-ausgeführten Zustand, nachdem die erste Nutzer-VOIP-Anwendung auf die Benachrichtigung geantwortet hat, wobei die erste Zeit der zweiten Zeit vorangeht und die zweite Zeit der dritten Zeit vorangeht.

45. Handgehaltene digitale Vorrichtung-Multitasking-Verfahren, aufweisend:  
Instanzieren mindestens eines Voice Over Internet Protocol (VOIP)-Sockets für eine erste Nutzeranwendung zu einer ersten Zeit;  
Versetzen der ersten Nutzeranwendung in einen ausgesetzten Zustand zu einer zweiten Zeit;  
Empfangen einer Nachricht für die erste Nutzeranwendung zu einer dritten Zeit;  
Versetzen der Nutzeranwendung in einen Hintergrundzustand zu einer vierten Zeit;  
Liefen der Nachricht an die erste Nutzeranwendung zu einer fünften Zeit;  
Empfangen einer Benachrichtigung von der ersten Nutzeranwendung, dass sie auf die Nachricht geantwortet hat, zu einer sechsten Zeit; und  
Versetzen der ersten Nutzeranwendung in den ausgesetzten Zustand zu einer siebten Zeit.

46. Verfahren gemäß Anspruch 45, wobei die Nachricht eine Heartbeat-erforderlich-Nachricht aufweist.

47. Verfahren gemäß Anspruch 45, wobei die Handlung des Empfangens, zu einer sechsten Zeit, einer Benachrichtigung von der ersten Nutzeranwendung, dass sie auf die Nachricht geantwortet hat, ein Anzeigen eines Nutzer-Schnittstellenelements für einen Nutzer aufweist, wobei das Nutzer-Schnittstellenelement dem Nutzer eine Option gibt, die erste Nutzeranwendung zu (i) verwenden oder (ii) ignorieren.

48. Verfahren gemäß Anspruch 47, wobei die Handlung des Versetzens, zu einer siebten Zeit, der ersten Nutzeranwendung in den ausgesetzten Zustand nur auftritt, wenn der Nutzer die Nutzer-Schnittstellenoption zum Ignorieren der ersten Nutzeranwendung auswählt, sonst wird die erste Nutzeranwendung in einen Vordergrundzustand versetzt.

49. Verfahren zum Planen der Verarbeitung in einem beschränkten Multitasking-Betriebsverfahren, aufweisend:  
Planen eines ersten Vordergrundprozesses mit einer Planungspriorität aus einem ersten Bereich von Prioritätsebenen; und  
Planen eines ersten Hintergrundprozesses mit einer Planungspriorität aus einem zweiten Bereich von Prioritätsebenen, wobei die Prioritätsebenen in dem ersten Bereich und dem zweiten Bereich durch eine, aber nicht alle Prioritätsebenen überlappen.

50. Verfahren gemäß Anspruch 49, weiterhin aufweisend das Planen eines dritten Prozesses mit einer Priorität aus einem Echtzeitprioritätsbereich, wobei alle Prioritätsebenen in dem Echtzeitprioritätsbereich größer sind als alle Prioritätsebenen in dem ersten Bereich.

51. Verfahren gemäß Anspruch 50, wobei die Handlung des Planens eines dritten Prozesses ein Planen eines Hintergrundprozesses aufweist.

52. Multitasking-Verfahren, aufweisend:

Identifizieren einer ersten Nutzeranwendung, welche in einem Vordergrundzustand ausgeführt wird und eine Nutzereingabe empfängt;

Identifizieren einer zweiten Nutzeranwendung, welche in einem Vordergrundzustand ausgeführt wird und keine Nutzereingabe empfängt;

Beschränken des Zugriffs der zweiten Vordergrundanwendung auf eine oder mehrere spezifizierte Systemressourcen, weil die zweite Vordergrundanwendung keine Nutzereingabe empfängt.

53. Verfahren gemäß Anspruch 52, wobei die eine oder mehreren spezifizierten Systemressourcen ausgewählt werden aus der Gruppe bestehend aus: Kamera, Grafikverarbeitungseinheit, Beschleunigungsmesser, Näherungssensor und Mikrofon.

54. Verfahren gemäß Anspruch 52, wobei die Handlung des Beschränkens des Zugriffs der zweiten Vordergrundanwendung auf eine oder mehrere spezifizierte Systemressourcen ein Berichten einer kleineren als der derzeitige eingehenden Netzwerkpuffergröße an eine entfernte Quelle beinhaltet.

55. Programmspeichervorrichtung, lesbar durch eine programmierbare Steuervorrichtung, aufweisend darauf gespeicherte Anweisungen zum Veranlassen der programmierbaren Steuervorrichtung, ein Verfahren gemäß Anspruch 52 auszuführen.

56. Multitasking-Verfahren, aufweisend:

Identifizieren einer ersten Nutzeranwendung, welche in einem Vordergrundzustand ausgeführt wird;

Empfangen einer Nutzereingabe, welche anweist, dass eine zweite Nutzeranwendung in den Vordergrundzustand zu versetzen ist;

Aussetzen nur der ersten Nutzeranwendung in Antwort auf die empfangene Nutzereingabe; und

Versetzen der zweiten Nutzeranwendung in den Vordergrundzustand.

57. Verfahren gemäß Anspruch 56, wobei die Handlung des Aussetzens nur ausgeführt wird, wenn die erste Nutzeranwendung sich nicht als eine Anwendung identifiziert hat, welche im Hintergrund ausgeführt werden kann.

58. Verfahren gemäß Anspruch 56, weiterhin aufweisend:

Erhalten einer Nachricht für die ausgesetzte erste Nutzeranwendung;

Bewegen der ausgesetzten ersten Nutzeranwendung in einen Hintergrundzustand; und

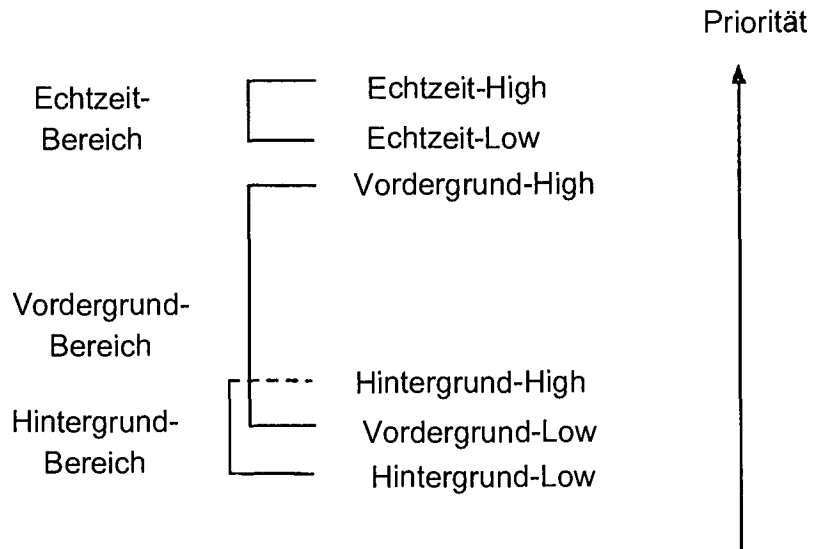
Bereitstellen der Nachricht an die erste Nutzeranwendung im Hintergrund.

59. Programmspeichervorrichtung, lesbar durch eine programmierbare Steuervorrichtung, aufweisend darauf gespeicherte Anweisungen zum Veranlassen der programmierbaren Steuervorrichtung, ein Verfahren gemäß Anspruch 56 auszuführen.

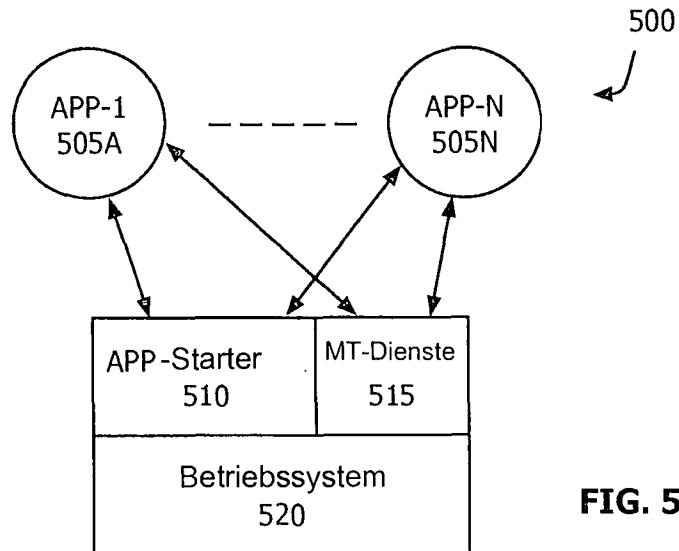
Es folgen 6 Blatt Zeichnungen

Anhängende Zeichnungen

Prozessor-Thread-Planungsstruktur



**FIG. 1**



**FIG. 5**

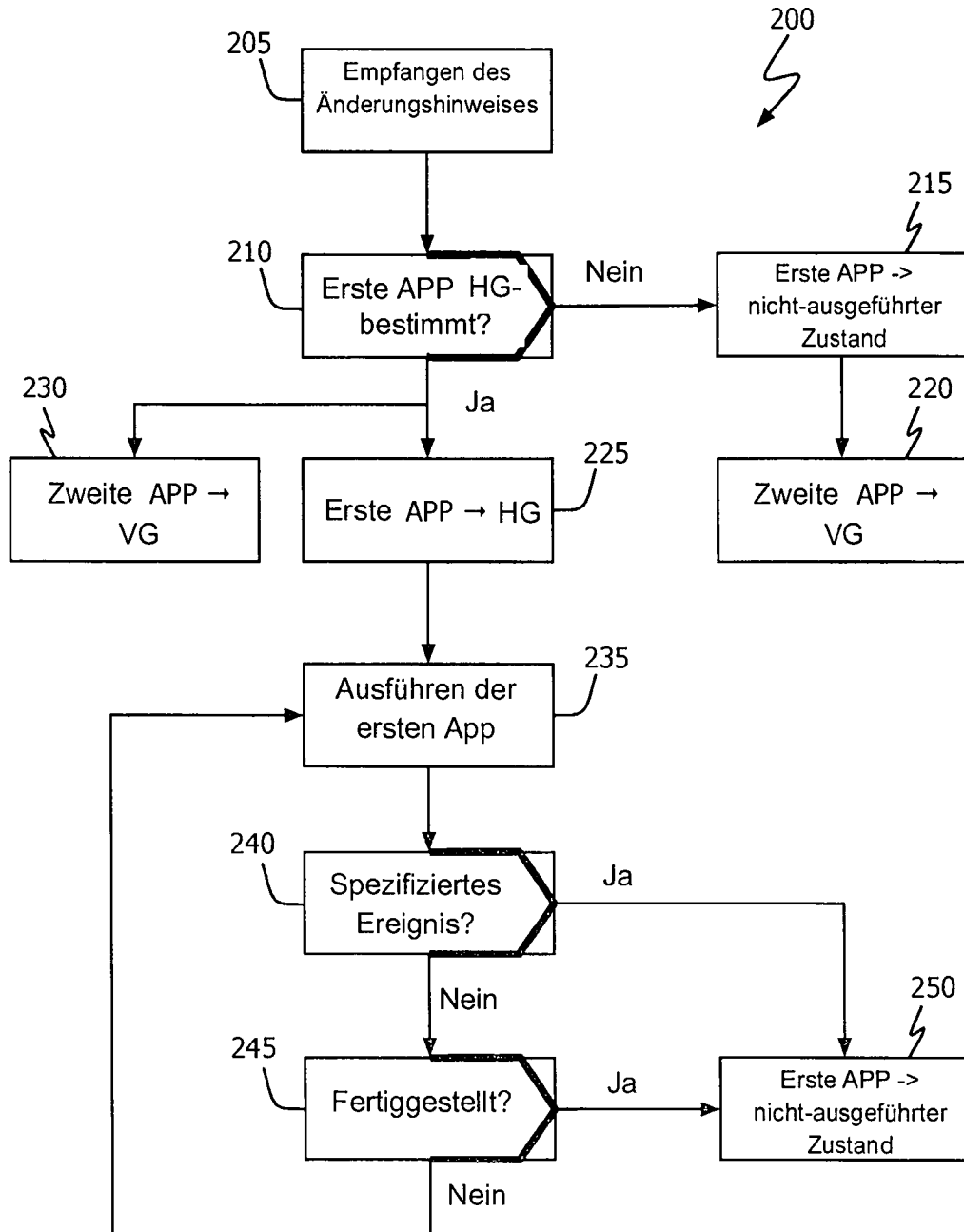
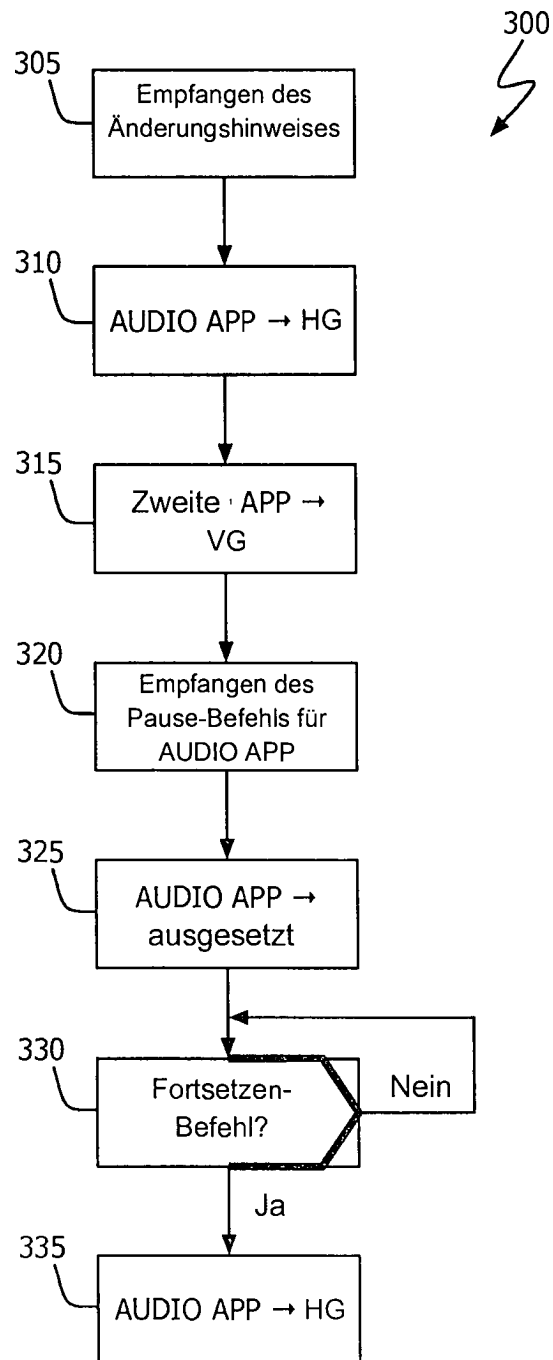
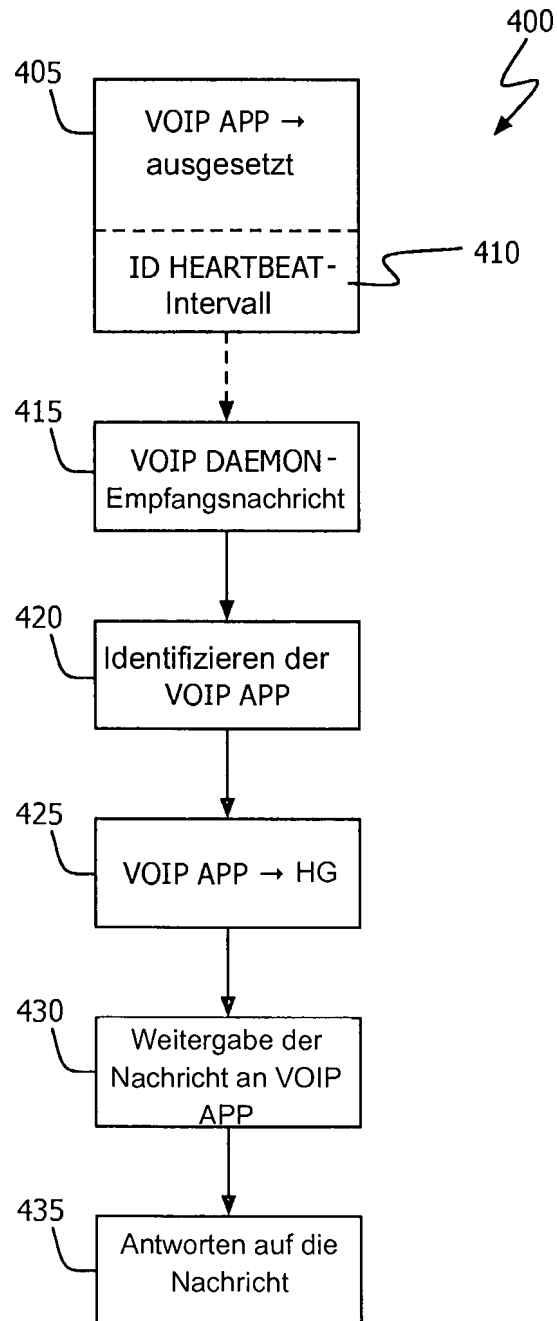


FIG. 2



**FIG. 3**



**FIG. 4**

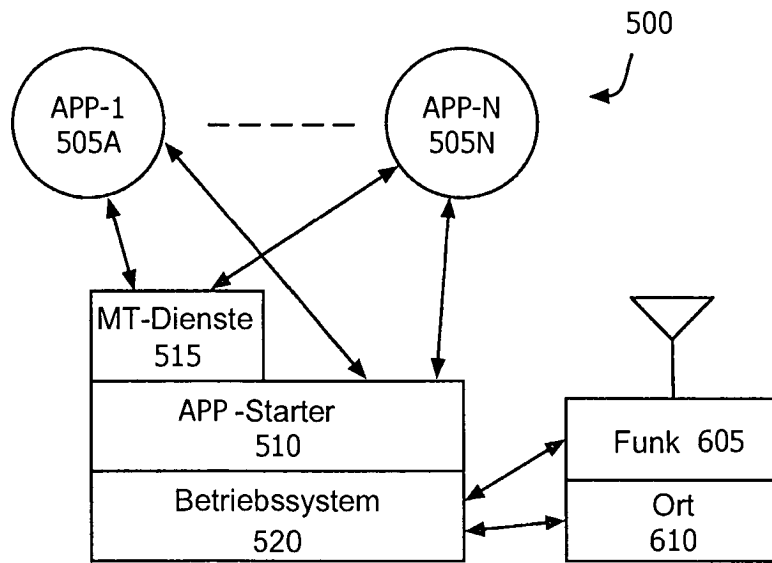


FIG. 6

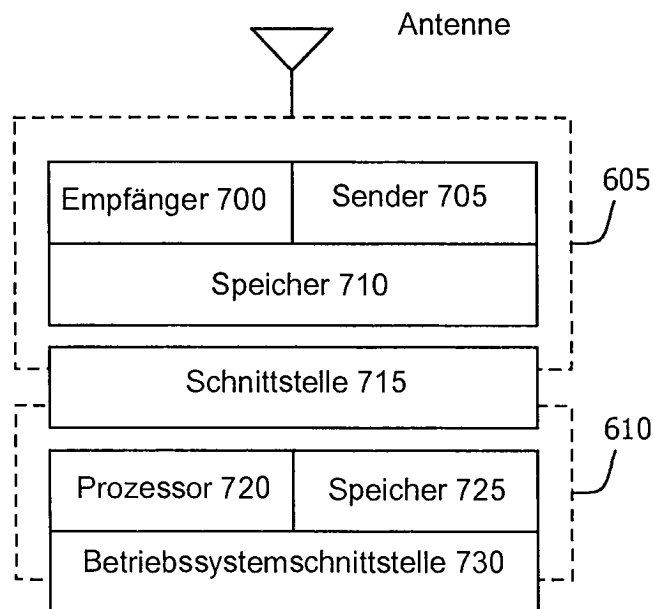


FIG. 7

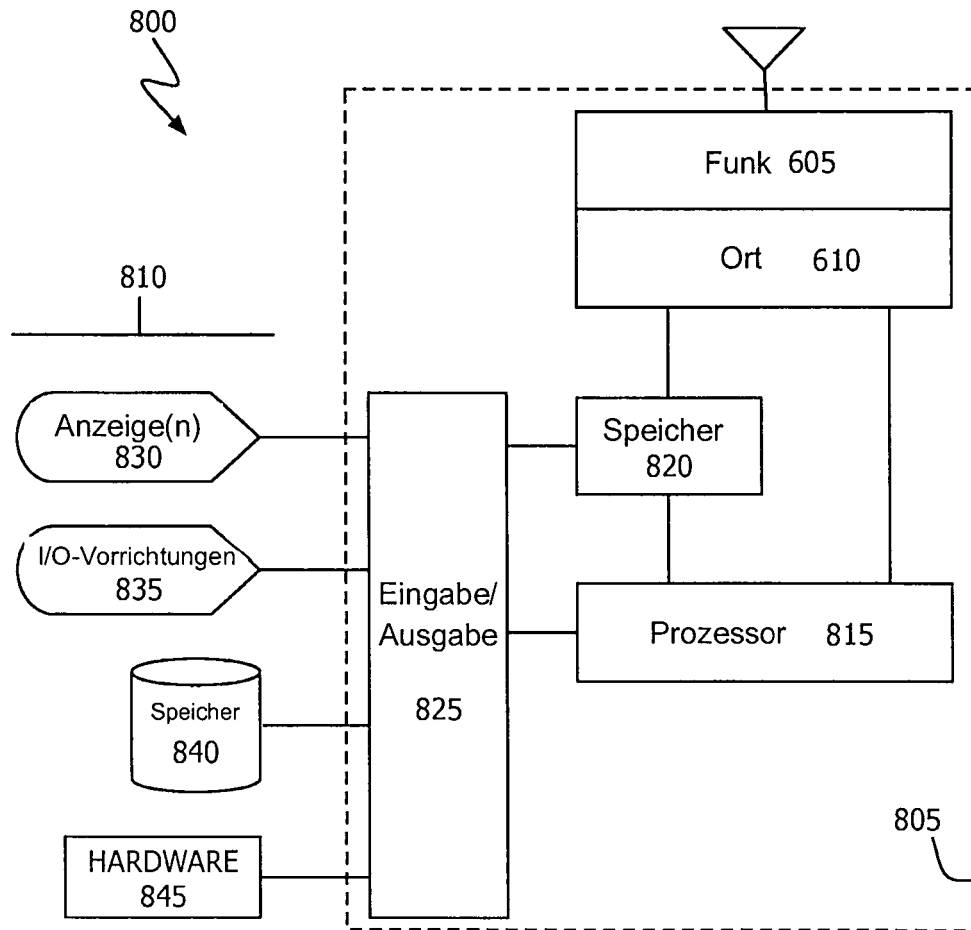


FIG. 8