

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6134074号
(P6134074)

(45) 発行日 平成29年5月24日 (2017.5.24)

(24) 登録日 平成29年4月28日 (2017.4.28)

(51) Int.Cl. F I
G O 6 F 9/48 (2006.01) G O 6 F 9/46 4 5 2 H

請求項の数 19 (全 27 頁)

(21) 出願番号	特願2016-544717 (P2016-544717)	(73) 特許権者	513309030
(86) (22) 出願日	平成27年1月22日 (2015.1.22)		シャオミ・インコーポレイテッド
(65) 公表番号	特表2016-538663 (P2016-538663A)		中華人民共和国・100085・ベイジン
(43) 公表日	平成28年12月8日 (2016.12.8)		・ハイディアン・ディストリクト・キンヘ
(86) 国際出願番号	PCT/CN2015/071367		・ミドル・ストリート・ナンバー・68・
(87) 国際公開番号	W02016/029642		レインボー・シティ・ショッピング・モー
(87) 国際公開日	平成28年3月3日 (2016.3.3)		ル・2・オブ・チャイナ・リソーシズ・フ
審査請求日	平成27年4月30日 (2015.4.30)		ロア・13
(31) 優先権主張番号	201410436455.8	(74) 代理人	100168583
(32) 優先日	平成26年8月29日 (2014.8.29)		弁理士 前井 宏之
(33) 優先権主張国	中国 (CN)		

最終頁に続く

(54) 【発明の名称】 バックグラウンドアプリケーションを制御する方法および装置ならびに端末装置

(57) 【特許請求の範囲】

【請求項1】

端末装置において使用されるバックグラウンドアプリケーションを制御する方法であって、

前記端末装置が、オペレーティングシステムで実行中のアプリケーションにしたがって、前記オペレーティングシステムで実行中の前記アプリケーションの識別情報を少なくとも含むアプリケーションリストを作成するステップと、

前記端末装置が、前記アプリケーションリスト内の識別情報を走査するステップと、

前記端末装置が、走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションであるかどうかを判定するステップと、

前記端末装置が、前記走査中の識別情報に対応する前記アプリケーションがバックグラウンドアプリケーションである場合、ユーザにより指定したバックグラウンドアプリケーションの識別情報を含む所定のホワイトリストに前記走査中の識別情報が含まれるかどうかと、前記アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えているかどうかとを判定するステップと、

前記端末装置が、前記所定のホワイトリストに前記走査中の識別情報が含まれており、前記数が前記所定の閾値を超えている場合、前記アプリケーションリストから1つのバックグラウンドアプリケーションに対応する識別情報を選択し、前記選択した識別情報に対応する前記バックグラウンドアプリケーションを終了するステップと、

前記端末装置が、前記所定のホワイトリストに前記走査中の識別情報が含まれておらず

、前記数が前記所定の閾値を超えている場合、前記走査中の識別情報に対応する前記アプリケーションを終了するステップと
を含む、バックグラウンドアプリケーションを制御する方法。

【請求項 2】

前記オペレーティングシステムで実行中のアプリケーションにしたがってアプリケーションリストを前記端末装置が作成するステップは、

前記端末装置が、前記オペレーティングシステムで実行中の前記アプリケーションの実行状態の優先度を算出するステップであって、前記実行状態の優先度は、高いものから低いものへ、順に、フォアグラウンド実行状態、可視実行状態、可感知実行状態、バックグラウンド実行状態となっている、ステップと、

前記端末装置が、前記実行中のアプリケーションの前記識別情報を、前記実行状態の優先度にしたがって高いものから低いものへ順に配列して前記アプリケーションリストを得るステップと

を含む、請求項 1 に記載の方法。

【請求項 3】

前記走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションであるかどうかを前記端末装置が判定するステップは、

前記端末装置が、前記走査中の識別情報に対応する前記アプリケーションの実行状態の優先度を取得するステップと、

前記端末装置が、前記走査中の識別情報に対応する前記アプリケーションの前記実行状態の優先度がバックグラウンドである場合に、前記走査中の識別情報に対応する前記アプリケーションがバックグラウンドアプリケーションであると判定するステップと

を含む、請求項 2 に記載の方法。

【請求項 4】

前記アプリケーションリストから 1 つのバックグラウンドアプリケーションに対応する識別情報を前記端末装置が選択するステップは、

前記端末装置が、前記所定のホワイトリストには含まれておらず、前記アプリケーションリストにおいて前記走査中の識別情報に先行する 1 つ以上の識別情報を見つけるステップと、

前記端末装置が、前記所定のホワイトリストには含まれておらず、前記アプリケーションリストにおいて前記走査中の識別情報に先行する少なくとも 1 つの識別情報が見つかった場合、前記 1 つ以上の識別情報のうちの 1 つを選択するか、前記 1 つ以上の識別情報から前記走査中の識別情報に最も近い識別情報を選択するステップと、

前記端末装置が、前記所定のホワイトリストには含まれておらず、前記アプリケーションリストにおいて前記走査中の識別情報に先行する識別情報が見つからなかった場合、前記走査中の識別情報を選択するステップと

を含む、請求項 2 または 3 に記載の方法。

【請求項 5】

前記端末装置が、前記所定のホワイトリストに前記走査中の識別情報が含まれており、前記数が前記所定の閾値を超えていない場合、前記走査中の識別情報に対応する前記アプリケーションの休止時間を取得するステップと、

前記端末装置が、前記休止時間が第 1 所定時間を超えているかどうかを判定するステップと、

前記端末装置が、前記休止時間が前記第 1 所定時間を超えている場合、前記走査中の識別情報に対応する前記アプリケーションを終了するステップと

をさらに含む、請求項 1 に記載の方法。

【請求項 6】

前記端末装置が、前記休止時間が前記第 1 所定時間を超えていない場合に、前記休止時間が第 2 所定時間を超えているかどうかを検知するステップと、

前記端末装置が、前記休止時間が前記第 2 所定時間を超えている場合に、前記走査中の

10

20

30

40

50

識別情報に対応する前記アプリケーションのユーザインターフェースが占有しているメモリを解放するステップと

をさらに含み、前記第 2 所定時間は前記第 1 所定時間よりも短い、請求項 5 に記載の方法。

【請求項 7】

前記端末装置が、前記所定のホワイトリストに前記走査中の識別情報が含まれておらず、前記数が前記所定の閾値を超えていない場合、前記走査中の識別情報に対応する前記アプリケーションの休止時間を取得するステップと、

前記端末装置が、前記休止時間が第 3 所定時間を超えているかどうかを判定するステップと、

前記端末装置が、前記休止時間が前記第 3 所定時間を超えている場合、前記走査中の識別情報に対応する前記アプリケーションのユーザインターフェースが占有しているメモリを解放するステップと

をさらに含む、請求項 1 に記載の方法。

【請求項 8】

前記走査中の識別情報に対応する前記アプリケーションの休止時間を前記端末装置が取得するステップは、

前記端末装置が、前記走査中の識別情報に対応する前記アプリケーションの処理が、前記オペレーティングシステムの他の処理と相互作用しない時間を計測するステップと、

前記端末装置が、前記時間を前記休止時間とするステップと

を含む、請求項 5 ~ 7 のいずれかに記載の方法。

【請求項 9】

オペレーティングシステムで実行中のアプリケーションにしたがって、前記オペレーティングシステムで実行中の前記アプリケーションの識別情報を少なくとも含むアプリケーションリストを作成するように構成された作成ユニットと、

前記アプリケーションリスト内の前記識別情報を走査するように構成された走査ユニットと、

走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションであるかどうかを判定するように構成された第 1 判定ユニットと、

前記走査中の識別情報に対応する前記アプリケーションがバックグラウンドアプリケーションである場合、ユーザが指定したバックグラウンドアプリケーションの識別情報を含む所定のホワイトリストに前記走査中の識別情報が含まれるかどうかと、前記アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えているかどうかとを判定するように構成された第 2 判定ユニットと、

第 1 終了ユニットであって、

前記所定のホワイトリストに前記走査中の識別情報が含まれており、前記数が前記所定の閾値を超えている場合、前記アプリケーションリストから 1 つのバックグラウンドアプリケーションに対応する識別情報を選択し、前記選択した識別情報に対応する前記バックグラウンドアプリケーションを終了し、

前記所定のホワイトリストに前記走査中の識別情報が含まれておらず、前記数が前記所定の閾値を超えている場合、前記走査中の識別情報に対応する前記アプリケーションを終了するように構成された第 1 終了ユニットと

を備える、バックグラウンドアプリケーションを制御する装置。

【請求項 10】

前記作成ユニットは、

前記オペレーティングシステムで実行中の前記アプリケーションの実行状態の優先度を算出するように構成された算出サブユニットであって、前記実行状態の優先度は、高いものから低いものへ、順に、フォアグラウンド実行状態、可視実行状態、可感知実行状態、バックグラウンド実行状態となっている、算出サブユニットと、

前記実行中のアプリケーションの前記識別情報を、前記実行状態の優先度にしたがって

10

20

30

40

50

高いものから低いものへ順に配列して前記アプリケーションリストを得るように構成された配列サブユニットと
を備える、請求項 9 に記載の装置。

【請求項 1 1】

前記第 1 判定ユニットは、

前記走査中の識別情報に対応する前記アプリケーションの実行状態の優先度を取得するように構成された取得サブユニットと、

前記走査中の識別情報に対応する前記アプリケーションの前記実行状態の優先度がバックグラウンドである場合に、前記走査中の識別情報に対応する前記アプリケーションがバックグラウンドアプリケーションであると判定するように構成された第 1 判定サブユニットと

10

を備える、請求項 1 0 に記載の装置。

【請求項 1 2】

前記第 1 終了ユニットは、

前記所定のホワイトリストには含まれておらず、前記アプリケーションリストにおいて前記走査中の識別情報に先行する 1 つ以上の識別情報を見つけるように構成された発見サブユニットと、

選択サブユニットであって、

前記所定のホワイトリストには含まれておらず、前記アプリケーションリストにおいて前記走査中の識別情報に先行する少なくとも 1 つの識別情報が見つかった場合、前記 1 つ以上の識別情報のうちの 1 つを選択するか、前記 1 つ以上の識別情報から前記走査中の識別情報に最も近い識別情報を選択し、

20

前記所定のホワイトリストには含まれておらず、前記アプリケーションリストにおいて前記走査中の識別情報に先行する識別情報が見つからなかった場合、前記走査中の識別情報を選択するように構成された選択サブユニットと

を備える、請求項 1 0 または 1 1 に記載の装置。

【請求項 1 3】

前記所定のホワイトリストに前記走査中の識別情報が含まれており、前記数が前記所定の閾値を超えていない場合、前記走査中の識別情報に対応する前記アプリケーションの休止時間を取得するように構成された第 1 取得ユニットと、

30

前記休止時間が第 1 所定時間を超えているかどうかを判定するように構成された第 3 判定ユニットと、

前記休止時間が前記第 1 所定時間を超えている場合、前記走査中の識別情報に対応する前記アプリケーションを終了するように構成された第 2 終了ユニットと

をさらに備える、請求項 9 に記載の装置。

【請求項 1 4】

前記休止時間が前記第 1 所定時間を超えていない場合に、前記休止時間が第 2 所定時間を超えているかどうかを検知するように構成された検知ユニットと、

前記休止時間が前記第 2 所定時間を超えている場合に、前記走査中の識別情報に対応する前記アプリケーションのユーザインターフェースが占有しているメモリを解放する第 1 解放ユニットと

40

をさらに備え、前記第 2 所定時間は前記第 1 所定時間よりも短い、請求項 1 3 に記載の装置。

【請求項 1 5】

前記所定のホワイトリストに前記走査中の識別情報が含まれておらず、前記数が前記所定の閾値を超えていない場合、前記走査中の識別情報に対応する前記アプリケーションの休止時間を取得するように構成された第 2 取得ユニットと、

前記休止時間が第 3 所定時間を超えているかどうかを判定するように構成された第 4 判定ユニットと、

前記休止時間が前記第 3 所定時間を超えている場合、前記走査中の識別情報に対応する

50

前記アプリケーションのユーザインターフェースが占有しているメモリを解放するように構成された第2解放ユニットと
をさらに備える、請求項9に記載の装置。

【請求項16】

前記第1取得ユニットは、

前記走査中の識別情報に対応する前記アプリケーションの処理が前記オペレーティングシステムの他の処理と相互作用しない時間を計測するように構成された計測サブユニットと、

前記時間を前記休止時間とするように構成された第2判定サブユニットとを備える、請求項13又は14に記載の装置。

10

【請求項17】

前記第2取得ユニットは、

前記走査中の識別情報に対応する前記アプリケーションの処理が前記オペレーティングシステムの他の処理と相互作用しない時間を計測するように構成された計測サブユニットと、

前記時間を前記休止時間とするように構成された第2判定サブユニットとを備える、請求項15に記載の装置。

【請求項18】

端末装置のコンピュータで実行されると、前記端末装置に、請求項1～8のいずれかに記載のバックグラウンドアプリケーションを制御する方法を実行させるコンピュータプログラム。

20

【請求項19】

請求項18に記載のコンピュータプログラムを記憶したコンピュータ読取可能記憶媒体。

【発明の詳細な説明】

【技術分野】

【0001】

[関連出願の相互参照]

本出願は、2014年8月29日に出願された中国特許出願第201410436455.8号に基づく優先権を主張し、当該出願に記載された全ての記載内容を援用する。

30

【0002】

本開示は、端末装置の分野に関し、特にバックグラウンドアプリケーションを制御する方法、バックグラウンドアプリケーションを制御する装置、および端末装置に関する。

【背景技術】

【0003】

端末装置のインテリジェント化が進むにつれ、アプリケーション(A P P)の数が増加し、ユーザは端末装置のインテリジェント化による利便性を享受している。ユーザが端末装置を使用する際、常習的に使用頻度の高いアプリケーションをバックグラウンド実行状態にしていることがある。使用頻度が高いアプリケーションとしては、ツイッター、WeChat、QQなどが挙げられる。端末装置でバックグラウンド実行状態にあるアプリケーションをバックグラウンドアプリケーションと呼ぶことがある。バックグラウンドアプリケーションの数が少ない場合、端末装置において占有されるメモリリソースは少ない。バックグラウンドアプリケーションの数が多き場合は、端末装置において占有されるメモリリソースが多くなり、端末装置の処理速度に影響が出る。したがって、端末装置におけるバックグラウンドアプリケーションの制御の仕方を把握することが重要である。

40

【0004】

関連技術においては、以下のようなバックグラウンドアプリケーション制御方法がある。最大閾値を設定し、端末装置によってバックグラウンドアプリケーションの数が最大閾値を超えるかどうかを監視する。バックグラウンドアプリケーションの数が最大閾値以下の場合、各バックグラウンドアプリケーションをバックグラウンド実行状態にしておくこ

50

とができる。バックグラウンドアプリケーションの数が最大閾値を超える場合、最大メモリを占有しているバックグラウンドアプリケーションを終了することができる。たとえば、最大閾値が25であり、バックグラウンドアプリケーションの数が24である場合、各バックグラウンドアプリケーションをバックグラウンド実行状態にすることができる。バックグラウンドアプリケーションの数が26の場合、最大メモリを占有しているバックグラウンドアプリケーションを終了して、そのメモリを開放することができる。

【発明の概要】

【発明が解決しようとする課題】

【0005】

上記の方法では、バックグラウンドアプリケーションの数が最大閾値を超えているかどうかを判定することによって、バックグラウンドアプリケーションが終了されたかどうか判定される。バックグラウンドアプリケーションの数が最大閾値を超えている場合、最大メモリを占有しているバックグラウンドアプリケーションを終了することができる。ユーザが最大メモリを占有しているバックグラウンドアプリケーションを常にバックグラウンド実行状態にしておきたいときは、そのバックグラウンドアプリケーションが終了してしまうと、ユーザエクスペリエンスに影響を及ぼす可能性がある。したがって、上記の方法では、端末装置のバックグラウンドアプリケーションが適切に制御できているとは言えず、ユーザエクスペリエンスに影響が出てしまう。

【0006】

本開示は、上記関連技術における課題を解決するべく、バックグラウンドアプリケーションを制御する方法、バックグラウンドアプリケーションを制御する装置、および端末装置を提供する。

【課題を解決するための手段】

【0007】

本開示の実施形態の第1の側面によると、
 オペレーティングシステムで実行中のアプリケーションにしたがって、前記オペレーティングシステムで実行中の前記アプリケーションの識別情報を少なくとも含むアプリケーションリストを作成するステップと、
 前記アプリケーションリスト内の識別情報を走査するステップと、
 走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションであるかどうかを判定するステップと、
 前記走査中の識別情報に対応する前記アプリケーションがバックグラウンドアプリケーションである場合、ユーザが指定したバックグラウンドアプリケーションの識別情報を含む所定のホワイトリストに前記走査中の識別情報が含まれるかどうかと、前記アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えているかどうかとを判定するステップと、
 前記所定のホワイトリストに前記走査中の識別情報が含まれており、前記数が前記所定の閾値を超えている場合、前記アプリケーションリストから1つのバックグラウンドアプリケーションに対応する識別情報を選択し、前記選択した識別情報に対応する前記バックグラウンドアプリケーションを終了するステップと、
 前記所定のホワイトリストに前記走査中の識別情報が含まれておらず、前記数が前記所定の閾値を超えている場合、前記走査中の識別情報に対応する前記アプリケーションを終了するステップと
 を含む、バックグラウンドアプリケーションを制御する方法が提供される。

【0008】

第1の側面と組み合わせて、第1の側面の第1の実現可能な具体化態様では、前記オペレーティングシステムで実行中のアプリケーションにしたがってアプリケーションリストを作成するステップは、

前記オペレーティングシステムで実行中の前記アプリケーションの実行状態の優先度を算出するステップであって、前記実行状態の優先度は、高いものから低いものへ、順に、

10

20

30

40

50

フォアグラウンド、可視、可感知、バックグラウンドとなっている、ステップと、

前記実行中のアプリケーションの前記識別情報を、前記実行状態の優先度にしたがって高いものから低いものへ順に配列して前記アプリケーションリストを得るステップとを含む。

【 0 0 0 9 】

第 1 の側面の第 1 の実現可能な具体化態様と組み合わせて、第 1 の側面の第 2 の実現可能な具体化態様では、走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションであるかどうかを判定するステップは、

前記走査中の識別情報に対応する前記アプリケーションの実行状態の優先度を取得するステップと、

10

前記走査中の識別情報に対応する前記アプリケーションの前記実行状態の優先度がバックグラウンドである場合に、前記走査中の識別情報に対応する前記アプリケーションがバックグラウンドアプリケーションであると判定するステップとを含む。

【 0 0 1 0 】

第 1 の側面の第 1 の実現可能な具体化態様または第 2 の実現可能な具体化態様と組み合わせて、第 1 の側面の第 3 の実現可能な具体化態様では、アプリケーションリストからバックグラウンドアプリケーションに対応する識別情報を選択するステップは、

前記所定のホワイトリストには含まれておらず、前記アプリケーションリストにおいて前記走査中の識別情報に先行する 1 つ以上の識別情報を見つけるステップと、

20

前記所定のホワイトリストには含まれておらず、前記アプリケーションリストにおいて前記走査中の識別情報に先行する少なくとも 1 つの識別情報が見つかった場合、前記 1 つ以上の識別情報のうちの 1 つを選択するか、前記 1 つ以上の識別情報から前記走査中の識別情報に最も近い識別情報を選択するステップと、

前記所定のホワイトリストには含まれておらず、前記アプリケーションリストにおいて前記走査中の識別情報に先行する識別情報が見つからなかった場合、前記走査中の識別情報を選択するステップと

を含む。

【 0 0 1 1 】

第 1 の側面と組み合わせて、第 1 の側面の第 4 の実現可能な具体化態様では、前記方法は、

30

前記所定のホワイトリストに前記走査中の識別情報が含まれており、前記数が前記所定の閾値を超えていない場合、前記走査中の識別情報に対応する前記アプリケーションの休止時間を取得するステップと、

前記休止時間が第 1 所定時間を超えているかどうかを判定するステップと、

前記休止時間が前記第 1 所定時間を超えている場合、前記走査中の識別情報に対応する前記アプリケーションを終了するステップと

をさらに含む。

【 0 0 1 2 】

第 1 の側面の第 4 の実現可能な具体化態様と組み合わせて、第 1 の側面の第 5 の実現可能な具体化態様では、前記方法は、

40

前記休止時間が前記第 1 所定時間を超えていない場合に、前記休止時間が第 2 所定時間を超えているかどうかを検知するステップと、

前記休止時間が前記第 2 所定時間を超えている場合に、前記走査中の識別情報に対応する前記アプリケーションのユーザインターフェースが占有しているメモリを開放するステップと

をさらに含み、前記第 2 所定時間は前記第 1 所定時間よりも短い。

【 0 0 1 3 】

第 1 の側面と組み合わせて、第 1 の側面の第 6 の実現可能な具体化態様では、前記方法は、

50

前記所定のホワイトリストに前記走査中の識別情報が含まれておらず、前記数が前記所定の閾値を超えていない場合、前記走査中の識別情報に対応する前記アプリケーションの休止時間を取得するステップと、

前記休止時間が第3所定時間を超えているかどうかを判定するステップと、

前記休止時間が前記第3所定時間を超えている場合、前記走査中の識別情報に対応する前記アプリケーションのユーザインターフェースが占有しているメモリを開放するステップと

をさらに含む。

【0014】

第1の側面の第4の実現可能な具体化態様、第5の実現可能な具体化態様、または第6の実現可能な具体化態様と組み合わせて、第1の側面の第7の実現可能な具体化態様では、走査中の識別情報に対応するアプリケーションの休止時間を取得するステップは、

前記走査中の識別情報に対応する前記アプリケーションの処理が、前記オペレーティングシステムの他の処理と相互作用しない時間を計測するステップと、

前記時間を前記休止時間とするステップとを含む。

【0015】

本開示の実施形態の第2の側面によると、

オペレーティングシステムで実行中のアプリケーションにしたがって、前記オペレーティングシステムで実行中の前記アプリケーションの識別情報を少なくとも含むアプリケーションリストを作成するように構成された作成ユニットと、

前記アプリケーションリスト内の前記識別情報を走査するように構成された走査ユニットと、

走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションであるかどうかを判定するように構成された第1判定ユニットと、

前記走査中の識別情報に対応する前記アプリケーションがバックグラウンドアプリケーションである場合、ユーザが指定したバックグラウンドアプリケーションの識別情報を含む所定のホワイトリストに前記走査中の識別情報が含まれるかどうかと、前記アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えているかどうかとを判定するように構成された第2判定ユニットと、

第1終了ユニットであって、

前記所定のホワイトリストに前記走査中の識別情報が含まれており、前記数が前記所定の閾値を超えている場合、前記アプリケーションリストから1つのバックグラウンドアプリケーションに対応する識別情報を選択し、前記選択した識別情報に対応するバックグラウンドアプリケーションを終了し、

前記所定のホワイトリストに前記走査中の識別情報が含まれておらず、前記数が前記所定の閾値を超えている場合、前記走査中の識別情報に対応する前記アプリケーションを終了するように構成される第1終了ユニットと

を備える、バックグラウンドアプリケーションを制御する装置が提供される。

【0016】

第2の側面と組み合わせて、第2の側面の第1の実現可能な具体化態様では、前記作成ユニットは、

前記オペレーティングシステムで実行中の前記アプリケーションの実行状態の優先度を算出するように構成された算出サブユニットであって、前記実行状態の優先度は、高いものから低いものへ、順に、フォアグラウンド、可視、可感知、バックグラウンドとなっている、算出サブユニットと、

前記実行中のアプリケーションの前記識別情報を、前記実行状態の優先度にしたがって高いものから低いものへ順に配列して前記アプリケーションリストを得るように構成された配列サブユニットと

を備える。

10

20

30

40

50

【 0 0 1 7 】

第2の側面の第1の実現可能な具体化態様と組み合わせて、第2の側面の第2の実現可能な具体化態様では、前記第1判定ユニットは、

前記走査中の識別情報に対応する前記アプリケーションの実行状態の優先度を取得するように構成された取得サブユニットと、

前記走査中の識別情報に対応する前記アプリケーションの実行状態の優先度がバックグラウンドである場合に、前記走査中の識別情報に対応する前記アプリケーションがバックグラウンドアプリケーションであると判定するように構成された第1判定サブユニットとを備える。

【 0 0 1 8 】

第2の側面の第1の実現可能な具体化態様または第2の実現可能な具体化態様と組み合わせて、第2の側面の第3の実現可能な具体化態様では、前記第1終了ユニットは、

前記所定のホワイトリストには含まれておらず、前記アプリケーションリストにおいて前記走査中の識別情報に先行する1つ以上の識別情報を見つけるように構成された発見サブユニットと、

選択サブユニットであって、

前記所定のホワイトリストには含まれておらず、前記アプリケーションリストにおいて前記走査中の識別情報に先行する少なくとも1つの識別情報が見つかった場合、前記1つ以上の識別情報のうちの1つを選択するか、前記1つ以上の識別情報から前記走査中の識別情報に最も近い識別情報を選択し、

前記所定のホワイトリストには含まれておらず、前記アプリケーションリストにおいて前記走査中の識別情報に先行する識別情報が見つからなかった場合、前記走査中の識別情報を選択するように構成された選択サブユニットとを備える。

【 0 0 1 9 】

第2の側面と組み合わせて、第2の側面の第4の実現可能な具体化態様では、前記装置は、

前記所定のホワイトリストに前記走査中の識別情報が含まれており、前記数が前記所定の閾値を超えていない場合、前記走査中の識別情報に対応する前記アプリケーションの休止時間を取得するように構成された第1取得ユニットと

前記休止時間が第1所定時間を超えているかどうかを判定するように構成された第3判定ユニットと、

前記休止時間が前記第1所定時間を超えている場合、前記走査中の識別情報に対応する前記アプリケーションを終了するように構成された第2終了ユニットとをさらに備える。

【 0 0 2 0 】

第2の側面の第4の実現可能な具体化態様と組み合わせて、第2の側面の第5の実現可能な具体化態様では、前記装置は、

前記休止時間が前記第1所定時間を超えていない場合に、前記休止時間が第2所定時間を超えているかどうかを検知するように構成された検知ユニットと、

前記休止時間が前記第2所定時間を超えている場合に、前記走査中の識別情報に対応する前記アプリケーションのユーザインターフェースが占有しているメモリを開放するように構成された第1開放ユニットと

をさらに備え、前記第2所定時間は前記第1所定時間よりも短い。

【 0 0 2 1 】

第2の側面と組み合わせて、第2の側面の第6の実現可能な具体化態様では、前記装置は、

前記所定のホワイトリストに前記走査中の識別情報が含まれておらず、前記数が前記所定の閾値を超えていない場合、前記走査中の識別情報に対応する前記アプリケーションの休止時間を取得するように構成された第2取得ユニットと、

前記休止時間が第3所定時間を超えているかどうかを判定するように構成された第4判定ユニットと、

前記休止時間が前記第3所定時間を超えている場合、前記走査中の識別情報に対応する前記アプリケーションのユーザインターフェースが占有しているメモリを開放するように構成された第2開放ユニットとをさらに備える。

【0022】

第2の側面の第4の実現可能な具体化態様、第5の実現可能な具体化態様、または第6の実現可能な具体化態様と組み合わせて、第2の側面の第7の実現可能な具体化態様では、前記第1取得ユニットおよび前記第2取得ユニットは、

前記走査中の識別情報に対応する前記アプリケーションの処理が前記オペレーティングシステムの他の処理と相互作用しない時間を計測するように構成された計測サブユニットと、

前記時間を前記休止時間とするように構成された第2判定サブユニットとを備える。

【0023】

本開示の実施形態の第3の側面によると、プロセッサと、

前記プロセッサにより実行可能な命令を格納するように構成されたメモリとを備え、

前記プロセッサは、

オペレーティングシステムで実行中のアプリケーションにしたがって、前記オペレーティングシステムで実行中のアプリケーションの識別情報を少なくとも含むアプリケーションリストを作成し、

前記アプリケーションリスト内の識別情報を走査し、

走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションであるかどうかを判定し、

前記走査中の識別情報に対応する前記アプリケーションがバックグラウンドアプリケーションである場合、ユーザが指定したバックグラウンドアプリケーションの識別情報を含む所定のホワイトリストに前記走査中の識別情報が含まれるかどうかと、前記アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えているかどうかとを判定し、

前記所定のホワイトリストに前記走査中の識別情報が含まれており、前記数が前記所定の閾値を超えている場合、前記アプリケーションリストから1つのバックグラウンドアプリケーションに対応する識別情報を選択し、前記選択した識別情報に対応する前記バックグラウンドアプリケーションを終了し、

前記所定のホワイトリストに前記走査中の識別情報が含まれておらず、前記数が前記所定の閾値を超えている場合、前記走査中の識別情報に対応する前記アプリケーションを終了するように構成される端末装置が提供される。

【発明の効果】

【0024】

本開示の実施形態で開示する技術的解決手段は、以下の有利な効果を有すると考えられる。終了するバックグラウンドアプリケーションを端末装置によって決定するために、バックグラウンドアプリケーションの数が所定の閾値を超えているかどうかと、所定のホワイトリストに終了するバックグラウンドアプリケーションの識別情報が含まれているかどうかとをいずれも検討する必要がある。所定のホワイトリストにはユーザが指定したバックグラウンドアプリケーションの識別情報が含まれる。すなわち、所定のホワイトリスト内の識別情報に対応するバックグラウンドアプリケーションのいずれについても、ユーザは終了を望まない。走査中の識別情報について、所定のホワイトリストがその走査中の識別情報を含み、バックグラウンドアプリケーションの数が所定の閾値を超えている場合、

10

20

30

40

50

バックグラウンドアプリケーションが1つ選択され終了される。所定のホワイトリストに走査中の識別情報が含まれておらず、バックグラウンドアプリケーションの数が所定の閾値を超えている場合、その走査中の識別情報に対応するアプリケーションを終了する。ユーザが指定したバックグラウンドアプリケーションが軽率に終了されてしまうことがないため、最大メモリを占有している終了対象のアプリケーションが、ユーザが指定したバックグラウンドアプリケーションになってしまう可能性がある関連技術の方法に比べて、バックグラウンドアプリケーションを適切に制御し、ユーザエクスペリエンスを改善することができる。

【0025】

上記の一般的な説明および下記の詳細な説明は、いずれも説明を目的とした例示的なものに過ぎず、請求の範囲に記載した本開示を限定するものではないことが理解されるべきである。

【図面の簡単な説明】

【0026】

本明細書に参照され、また、本明細書の一部を構成する添付の図面は、本開示と整合する実施形態を示し、以下の記述と共に本開示の要旨を説明するものである。

【図1】図1は、本開示の例示的な実施形態に係るバックグラウンドアプリケーションの制御方法を示すフローチャートである。

【図2】図2は、本開示の例示的な実施形態に係るバックグラウンドアプリケーションの別の制御方法を示すフローチャートである。

【図3】図3は、本開示の例示的な実施形態に係るバックグラウンドアプリケーションを制御する装置を示すブロック図である。

【図4】図4は、本開示の例示的な実施形態に係る端末装置を示すブロック図である。

【発明を実施するための形態】

【0027】

以下に、例示的な実施形態を詳細に説明する。添付の図面は、実施形態の例を示す。以下の説明で言及する図面中で、複数の図面において用いられている同一の符号は、特に記載しない限り、同一あるいは類似の要素を示す。以下の例示的な実施形態の説明に記載する具体化態様は、本開示と整合するすべての具体化態様を示すものではなく、請求項に記載した本開示に関連する側面に従った装置および方法の例示に過ぎない。

【0028】

図1は、本開示の例示的な実施形態に係るバックグラウンドアプリケーションの制御方法を示すフローチャートである。図1に示すとおり、端末装置において使用される本方法は以下のステップを含む。

【0029】

ステップ11において、オペレーティングシステムで実行中のアプリケーションにしたがってアプリケーションリストを作成する。

【0030】

一般に、端末装置にはオペレーティングシステムがインストールされている。よく使われるオペレーティングシステムとしては、アンドロイドシステム、ウィンドウズシステム、IOSシステムなどが挙げられる。ユーザの様々な要求を満たすために、端末装置には多数のアプリケーションがインストールされていることがある。また、ユーザが必要なアプリケーションをセットアップすることもある。これらのアプリケーションは、端末装置のオペレーティングシステムでしか実行することができない。したがって、バックグラウンドアプリケーションを制御するためにオペレーティングシステムで実行中のアプリケーションにしたがってアプリケーションリストを作成する。アプリケーションリストには、オペレーティングシステムで実行中のアプリケーションの識別情報が含まれる。アプリケーションの識別情報には、アプリケーションの名前やアイコンなどがあるが、これらに限定されない。

【0031】

アプリケーションリストには、オペレーティングシステムで実行中のアプリケーションの識別情報、最終起動時間、および実行状態の優先度などの情報が含まれる。アプリケーションリストには、オペレーティングシステムで実行中のアプリケーションの識別情報が少なくとも含まれる。

【0032】

ステップ12において、アプリケーションリストの識別情報を走査する。

【0033】

ステップ13において、走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションであるかどうかを判定する。

【0034】

アプリケーションリスト内の識別情報が走査されるが、走査対象になっている識別情報を走査中の識別情報とする。本発明は、バックグラウンドアプリケーションを制御することを狙いとするため、走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションであるかどうかをまず判定する。

【0035】

走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションである場合、ステップ14において、所定のホワイトリストに走査中の識別情報が含まれるかどうかと、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えているかどうかとを判定する。

【0036】

走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションではない場合、ステップ12を実行する。

【0037】

所定のホワイトリストには、ユーザが指定したバックグラウンドアプリケーションの識別情報、すなわち、ユーザが終了したくないバックグラウンドアプリケーションの識別情報が含まれる。たとえば、ユーザが指定するバックグラウンドアプリケーションとしては、QQ、WeChat、Fetion、Baidu mapなどが挙げられる。所定のホワイトリストには、QQ、WeChat、Fetion、Baidu mapなどのアプリケーションの識別情報が含むことができる。

【0038】

所定の閾値は、端末装置のハードウェア構成および実用上の要求に応じて設定することができる。たとえば、所定の閾値は、10、20、25などに設定することができる。

【0039】

所定のホワイトリストに走査中の識別情報が含まれ、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えている場合、ステップ15において、アプリケーションリストから1つのバックグラウンドアプリケーションに対応する識別情報を選択し、選択した識別情報に対応するアプリケーションを終了する。

【0040】

所定のホワイトリストに走査中の識別情報が含まれ、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えている場合、オペレーティングシステムを正常に実行させ、ユーザエクスペリエンスを良いものにするため、バックグラウンドアプリケーションを1つ選択して終了してもよい。

【0041】

所定のホワイトリストに走査中の識別情報が含まれておらず、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えている場合、ステップ16において、走査中の識別情報に対応するアプリケーションを終了する。

【0042】

所定のホワイトリストに走査中の識別情報が含まれておらず、アプリケーションリスト

10

20

30

40

50

内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えている場合、走査中の識別情報に対応するアプリケーションを直ちに終了することができる。所定のホワイトリストには走査中の識別情報が含まれていないため、すなわち、走査中の識別情報に対応するアプリケーションはユーザが指定したバックグラウンドアプリケーションではないため、走査中の識別情報に対応するアプリケーションを直ちに終了し、メモリリソースを節約し、ユーザエクスペリエンスを改善することができる。

【0043】

所定のホワイトリストに走査中の識別情報が含まれており、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えていない場合、または、所定のホワイトリストに走査中の識別情報が含まれておらず、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えていない場合、ステップ12を実行する。

10

【0044】

本開示の実施形態に係るバックグラウンドアプリケーションを制御する方法では、終了するバックグラウンドアプリケーションを端末装置によって決定するために、バックグラウンドアプリケーションの数が所定の閾値を超えているかどうかと、所定のホワイトリストに終了するバックグラウンドアプリケーションの識別情報が含まれているかどうかとをいずれも検討する必要がある。ホワイトリストにはユーザが指定したバックグラウンドアプリケーションの識別情報が含まれる。すなわち、ユーザは、所定のホワイトリスト内の識別情報に対応するバックグラウンドアプリケーションのいずれについても終了を望まない。走査中の識別情報について、所定のホワイトリストにその走査中の識別情報が含まれており、バックグラウンドアプリケーションの数が所定の閾値を超えている場合、バックグラウンドアプリケーションを1つ選択して終了する。所定のホワイトリストに走査中の識別情報が含まれておらず、バックグラウンドアプリケーションの数が所定の閾値を超えている場合、走査中の識別情報に対応するアプリケーションを終了する。ユーザが指定したバックグラウンドアプリケーションが軽率に終了されてしまうことがないため、最大メモリを占有している終了対象のバックグラウンドアプリケーションが、ユーザが指定したアプリケーションになってしまう可能性がある関連技術の方法に比べて、バックグラウンドアプリケーションを適切に制御することができ、ユーザエクスペリエンスを改善することができる。

20

30

【0045】

あるいは、ステップ11において、オペレーティングシステムで実行中のアプリケーションにしたがってアプリケーションリストを作成するステップは、

オペレーティングシステムで実行中のアプリケーションの実行状態の優先度を算出するステップと、

実行中のアプリケーションの識別情報を実行状態の優先度にしたがって高いものから低いものへ順に配列してアプリケーションリストを得るステップとを含む。

【0046】

一般に、実行状態の優先度は、高いものから低いものへ、順に、フォアグラウンド、可視、可感知、バックグラウンドとすることができる。このように、オペレーティングシステムで実行中のアプリケーションの実行状態の優先度が端末装置によって算出された後、該実行中のアプリケーションの識別情報が実行状態の優先度にしたがって高いものから低いものへ順に配列され、アプリケーションリストが得られてもよい。

40

【0047】

以上、アプリケーションリストを作成する方法を記載した。アプリケーションリストを作成する方法は他にもいくつかある。たとえば、実行中のアプリケーションの識別情報を、実行状態の優先度にしたがって低いものから高いものへ順に配列してアプリケーションリストを得てもよい。アプリケーションリストのその他の作成方法については、ここでは詳述しない。

50

【 0 0 4 8 】

オペレーティングシステムで実行中のアプリケーションは、短時間ではそれほど変化しないため、リソースを節約するために、端末装置はオペレーティングシステムで実行中のアプリケーションの実行状態の優先度を設定時間に応じて算出してもよい。設定時間は、実用上の要求に応じて設定することができる。たとえば、設定時間は30秒、45秒、60秒などにすることができる。

【 0 0 4 9 】

あるいは、ステップ13において、走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションであるかどうかを判定するステップは、

走査中の識別情報に対応するアプリケーションの実行状態の優先度を取得するステップと、

走査中の識別情報に対応するアプリケーションの実行状態の優先度がバックグラウンドである場合に、走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションであると判定するステップと

を含む。

【 0 0 5 0 】

オペレーティングシステムで実行中のアプリケーションの実行状態の優先度を端末装置によって算出することができるため、走査中の識別情報に対応するアプリケーションの実行状態の優先度にしたがってバックグラウンドアプリケーションを判定することができる。すなわち、実行状態の優先度がバックグラウンドであるアプリケーションをバックグラウンドアプリケーションであると判定することができる。

【 0 0 5 1 】

あるいは、ステップ15において、アプリケーションリストからバックグラウンドアプリケーションに対応する識別情報を選択するステップは、

所定のホワイトリストには含まれておらず、アプリケーションリストにおいて走査中の識別情報に先行する1つ以上の識別情報を見つけるステップと、

所定のホワイトリストには含まれておらず、アプリケーションリストにおいて走査中の識別情報に先行する少なくとも1つの識別情報が見つかった場合、該1つ以上の識別情報のうちの1つを選択するか、該1つ以上の識別情報から走査中の識別情報に最も近い識別情報を選択するステップと、

所定のホワイトリストには含まれておらず、アプリケーションリストにおいて走査中の識別情報に先行する識別情報が見つからなかった場合、走査中の識別情報を選択するステップと

を含む。

【 0 0 5 2 】

アプリケーションリストからバックグラウンドアプリケーションに対応する識別情報を選択するために、まずは、所定のホワイトリストには含まれておらず、アプリケーションリストにおいて走査中の識別情報に先行する1つ以上の識別情報を見つけてもよい。すなわち、まずはユーザが指定していないバックグラウンドアプリケーションを見つけてもよい。少なくとも1つの識別情報が見つかった場合、その1つ以上の識別情報のうちの1つ、または、走査中の識別情報に最も近い識別情報を選択することができる。識別情報が見つからなかった場合、すなわち、走査中の識別情報に先行する上記1つ以上の識別情報に対応するバックグラウンドアプリケーションがいずれもユーザが指定したバックグラウンドアプリケーションである場合、走査中の識別情報を選択することができる。

【 0 0 5 3 】

あるいは、上記方法は、

所定のホワイトリストに走査中の識別情報が含まれており、上記数が所定の閾値を超えていない場合、走査中の識別情報に対応するアプリケーションの休止時間を取得するステップと、

該休止時間が第1所定時間を超えているかどうかを判定するステップと、

10

20

30

40

50

上記休止時間が第1所定時間を超えている場合、走査中の識別情報に対応するアプリケーションを終了するステップとさらに含む。

【0054】

メモリリソースの浪費を避けるため、所定のホワイトリストが走査中の識別情報を含んでおり、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えていない場合、走査中の識別情報に対応するアプリケーションの休止時間を取得してもよい。休止時間が第1所定時間を超えている場合、走査中の識別情報に対応するアプリケーションを直ちに終了することができる。すなわち、所定のホワイトリストに走査中の識別情報が含まれておらず、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えていなくても、走査中の識別情報に対応するアプリケーションが長時間休止していれば、そのアプリケーションを終了することができる。第1所定時間は、実用上の要求に応じて決定することができる。たとえば、第1所定時間は3分、5分、10分などに設定することができる。

10

【0055】

たとえばQQについて、ユーザが使用していない場合は、QQをバックグラウンド実行状態にしておくことができる。QQの休止時間が10分を超えている場合、所定のホワイトリストにQQの識別情報が含まれていても、QQを終了してメモリリソースを節約することができる。

【0056】

あるいは、上記方法は、

20

休止時間が第1所定時間を超えていない場合に、休止時間が第2所定時間を超えているかどうかを検知するステップと、

休止時間が第2所定時間を超えている場合に、走査中の識別情報に対応するアプリケーションのユーザインターフェースが占有しているメモリを開放するステップとをさらに含む。ここで、第2所定時間は第1所定時間よりも短い。

【0057】

上記例にしたがって、休止時間が第1所定時間を超えていない場合、休止時間が第2所定時間を超えているかどうかを検知してもよい。第2所定時間は第1所定時間よりも短い。休止時間が第2所定時間を超えている場合、走査中の識別情報に対応するアプリケーションのユーザインターフェースが占有しているメモリを開放することができる。休止時間が第2所定時間を超えている場合、アプリケーションリストにおいて次の識別情報を走査することができる。第2所定時間は実用上の要求に応じて設定することができる。たとえば、第2所定時間を3分、5分、10分などに設定することができる。

30

【0058】

たとえばQQについて、ユーザが使用していない場合は、QQをバックグラウンド実行状態にしておくことができる。QQの休止時間が5分を超えている場合、所定のホワイトリストにQQの識別情報が含まれていても、QQのユーザインターフェースが占有しているメモリを開放してメモリリソースを節約することができる。

【0059】

一般に、アプリケーションは論理コードおよびユーザインターフェース(UI)の2つの部分を含んでもよい。UIには、画像やアニメーションなどの要素が含まれる。通常、これらの2つの部分はメモリに記憶することができる。アンドロイドシステムにおいて、UIはアクティビティであってもよい。アクティビティは、アプリケーションのUIの記憶に用いるコンテナである。論理コードが占有するメモリは少量であるため、休止時間が第2所定時間を超えるアプリケーションの論理コードを壊す代わりに、このアプリケーションのUIが占有するメモリを開放すると、大量のメモリリソースを節約することができる。アプリケーションの実行状態がフォアグラウンドになった場合、このアプリケーションの論理コードは保存されているため、UIを再構築してユーザに影響が出ないようにすることができる。

40

【0060】

50

あるいは、上記方法は、

所定のホワイトリストに走査中の識別情報が含まれておらず、上記数が所定の閾値を超えていない場合、走査中の識別情報に対応するアプリケーションの休止時間を取得するステップと、

該休止時間が第3所定時間を超えているかどうかを判定するステップと、

上記休止時間が第3所定時間を超えている場合、走査中の識別情報に対応するアプリケーションのユーザインターフェースが占有しているメモリを開放するステップとをさらに含む。

【0061】

したがって、メモリリソースを節約するため、所定のホワイトリストに走査中の識別情報を含まれておらず、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えていない場合は、走査中の識別情報に対応するアプリケーションの休止時間を取得することができる。休止時間が第3所定時間を超えている場合、走査中の識別情報に対応するアプリケーションのユーザインターフェースが占有しているメモリを開放することができる。

【0062】

第3所定時間は、実用上の要求に応じて設定することができる。たとえば、第3所定時間は3分、5分、10分などに設定することができる。

【0063】

たとえば電卓について、ユーザが使用していない場合は、電卓をバックグラウンド実行状態にしておくことができる。電卓の休止時間が5分を超えている場合、所定のホワイトリストにはこの電卓の識別情報が含まれていないため、この電卓のユーザインターフェースが占有しているメモリを開放してメモリリソースを節約することができる。

【0064】

あるいは、走査中の識別情報に対応するアプリケーションの休止時間を取得するステップは、

走査中の識別情報に対応するアプリケーションの処理がオペレーティングシステムの他の処理と相互作用しない時間を計測するステップと、

該時間を休止時間とするステップと

を含む。

【0065】

アプリケーションの休止状態とは、アプリケーションの処理がオペレーティングシステムの他の処理と相互作用しない状態のことを言う。相互作用にはデータ要求の送信、データ要求への応答、他の処理が送信する通信要求などがあってもよい。走査中の識別情報に対応するアプリケーションの処理がオペレーティングシステムの他の処理と相互作用しない時間を計測することによって、走査中の識別情報に対応するアプリケーションの休止時間を取得することができる。

【0066】

図2は、本開示の例示的な実施形態に係るバックグラウンドアプリケーションの別の制御方法を示すフローチャートである。図2に示すとおり、端末装置において使用される本方法は以下のステップを含む。

【0067】

ステップ21において、オペレーティングシステムで実行中のアプリケーションの実行状態の優先度が算出される。

【0068】

一般に、実行中のアプリケーションの実行状態の優先度は、高いものから低いものへ、順に、フォアグラウンド、可視、可感知、バックグラウンドとすることができる。実行中のアプリケーションの実行状態の優先度を算出する方法は、用いられるオペレーティングシステムごとに異なってもよい。アンドロイドシステムを例に取ると、実行中のアプリケーションの実行状態の優先度は、アクティビティマネージャサービスによって算出するこ

10

20

30

40

50

とができる。

【0069】

ステップ22において、実行中のアプリケーションの識別情報を実行状態の優先度にしたがって高いものから低いものへ順に配列してアプリケーションリストが取得される。

【0070】

以上、アプリケーションリストを作成する方法を記載した。アプリケーションリストを作成する方法は他にもいくつかある。たとえば、実行中のアプリケーションの識別情報を、実行状態の優先度にしたがって低いものから高いものへ順に配列してアプリケーションリストを得てもよい。アプリケーションリストのその他の作成方法については、ここでは詳述しない。

10

【0071】

アプリケーションの識別情報には、アプリケーションの名前やアイコンなどがあるが、これらに限定されない。

【0072】

ステップ23において、アプリケーションリスト内の識別情報を走査する。

【0073】

走査される個々の識別情報について、バックグラウンドアプリケーションの制御手順は同一である。以下、走査中の識別情報を例に取る。

【0074】

ステップ24において、走査中の識別情報に対応するアプリケーションの実行状態の優先度を取得する。

20

【0075】

走査中の識別情報に対応するアプリケーションの実行状態の優先度がバックグラウンドである場合、ステップ25において、走査中の識別情報に対応するアプリケーションはバックグラウンドアプリケーションであるとして、ステップ26に進む。

【0076】

走査中の識別情報に対応するアプリケーションの実行状態の優先度がバックグラウンドではない場合、ステップ40に進む。

【0077】

ステップ26において、所定のホワイトリストに走査中の識別情報が含まれているかどうかを判定する。

30

【0078】

所定のホワイトリストには、ユーザが指定したバックグラウンドアプリケーションの識別情報、すなわち、ユーザが終了を望まないバックグラウンドアプリケーションの識別情報が含まれる。たとえば、ユーザが指定するバックグラウンドアプリケーションとしては、QQ、WeChat、Fetion、Baidu mapなどが挙げられる。所定のホワイトリストには、QQ、WeChat、Fetion、Baidu mapなどのアプリケーションの識別情報を含むことができる。

【0079】

所定のホワイトリストに走査中の識別情報が含まれている場合、ステップ27において、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えているかどうかを判定する。

40

【0080】

所定の閾値は、端末装置のハードウェア構成および実用上の要求に応じて設定することができる。たとえば、所定の閾値は、10、20、25などに設定することができる。

【0081】

アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えている場合、ステップ28において、所定のホワイトリストには含まれておらず、アプリケーションリストにおいて走査中の識別情報に先行する1つ以上の識別情報を見つける。

50

【 0 0 8 2 】

少なくとも1つの識別情報が見つかった場合、ステップ29において、その1つ以上の識別情報のうちの1つ、または、走査中の識別情報に最も近い識別情報を選択し、ステップ31に進む。

【 0 0 8 3 】

識別情報が見つからなかった場合、ステップ30において、走査中の識別情報を選択し、ステップ31に進む。

【 0 0 8 4 】

ステップ31において、選択した識別情報に対応するアプリケーションを終了し、ステップ40に進む。

10

【 0 0 8 5 】

アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えていない場合、ステップ32において、走査中の識別情報に対応するアプリケーションの休止時間を取得し、該休止時間が第1所定時間を超えているかどうかを判定する。

【 0 0 8 6 】

上記休止時間が第1所定時間を超えている場合、ステップ33において、走査中の識別情報に対応するアプリケーションを終了し、ステップ40に進む。

【 0 0 8 7 】

メモリリソースの浪費を避けるため、所定のホワイトリストが走査中の識別情報を含んでおり、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えていない場合、走査中の識別情報に対応するアプリケーションの休止時間を取得することができる。休止時間が第1所定時間を超えている場合、走査中の識別情報に対応するアプリケーションを直ちに終了することができる。すなわち、所定のホワイトリストに走査中の識別情報が含まれており、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えていなくても、走査中の識別情報に対応するアプリケーションが長時間休止していれば、そのアプリケーションを終了することができる。第1所定時間は、実用上の要求に応じて設定することができる。たとえば、第1所定時間は3分、5分、10分などに設定することができる。

20

30

【 0 0 8 8 】

たとえばQQについて、ユーザが使用していない場合は、QQをバックグラウンド実行状態にしておくことができる。QQの休止時間が10分を超える場合、所定のホワイトリストにQQの識別情報が含まれていても、QQを終了してメモリリソースを節約することができる。

【 0 0 8 9 】

休止時間が第1所定時間を超えていない場合、ステップ34において、休止時間が第2所定時間を超えているかどうかを検知する。

【 0 0 9 0 】

休止時間が第2所定時間を超えている場合、ステップ35において、走査中の識別情報に対応するアプリケーションのユーザインターフェースが占有しているメモリを開放し、ステップ40に進む。

40

【 0 0 9 1 】

休止時間が第2所定時間を超えていない場合、ステップ40に進む。

【 0 0 9 2 】

上記例にしたがって、休止時間が第1所定時間を超えていない場合、休止時間が第2所定時間を超えているかどうかを検知することができる。第2所定時間は第1所定時間よりも短い。休止時間が第2所定時間を超えている場合、走査中の識別情報に対応するアプリケーションのユーザインターフェースが占有しているメモリを開放することができる。休止時間が第2所定時間を超えていない場合、アプリケーションリストの次の識別情報を走

50

査することができる。第2所定時間は実用上の要求に応じて設定することができる。たとえば、第2所定時間は、3分、5分、10分などに設定することができる。

【0093】

たとえばQQについて、ユーザが使用していない場合は、QQをバックグラウンド実行状態にしておくことができる。QQの休止時間が5分を超える場合、所定のホワイトリストにQQの識別情報が含まれていても、QQのユーザインターフェースが占有しているメモリを開放してメモリリソースを節約することができる。

【0094】

休止時間が第2所定時間を超えるアプリケーションのUIが占有しているメモリを開放するため、大量のメモリリソースを節約することができる。アプリケーションの実行状態がフォアグラウンドになった場合、UIを再構築してユーザに影響が出ないようにすることができる。

10

【0095】

所定のホワイトリストに走査中の識別情報が含まれていない場合、ステップ36において、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えているかどうかを判定する。

【0096】

アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えている場合、ステップ37において、走査中の識別情報に対応するアプリケーションを終了し、ステップ40に進む。

20

【0097】

所定のホワイトリストに走査中の識別情報が含まれておらず、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えている場合、走査中の識別情報に対応するアプリケーションを直ちに終了することができる。所定のホワイトリストには走査中の識別情報が含まれていないため、すなわち、走査中の識別情報に対応するアプリケーションはユーザが指定したバックグラウンドアプリケーションではないため、走査中の識別情報に対応するアプリケーションを直ちに終了してメモリリソースを節約し、ユーザエクスペリエンスを改善することができる。

【0098】

アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えていない場合、ステップ38において、走査中の識別情報に対応するアプリケーションの休止時間を取得し、該休止時間が第3所定時間を超えているかどうかを判定する。

30

【0099】

休止時間が第3所定時間を超えている場合、ステップ39において、走査中の識別情報に対応するアプリケーションのユーザインターフェースが占有しているメモリを開放し、ステップ40に進む。

【0100】

休止時間が第3所定時間を超えていない場合、ステップ40に進む。

【0101】

したがって、メモリリソースを節約するため、所定のホワイトリストに走査中の識別情報が含まれておらず、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えていない場合は、走査中の識別情報に対応するアプリケーションの休止時間を取得することができる。休止時間が第3所定時間を超えている場合、走査中の識別情報に対応するアプリケーションのユーザインターフェースが占有しているメモリを開放することができる。

40

【0102】

第3所定時間は、実用上の要求に応じて設定することができる。たとえば、第3所定時間は3分、5分、10分などに設定することができる。

【0103】

50

たとえば電卓について、ユーザが使用していない場合は、電卓をバックグラウンド実行状態にしておくことができる。電卓の休止時間が5分を超えている場合、所定のホワイトリストにこの電卓の識別情報が含まれていないため、この電卓のユーザインターフェースが占有しているメモリを開放してメモリリソースを節約することができる。

【0104】

ステップ40でフローは終了する。

【0105】

図3は、本開示の例示的な実施形態に係るバックグラウンドアプリケーションを制御する装置を示すブロック図である。図3に示すとおり、装置は作成ユニット31、走査ユニット32、第1判定ユニット33、第2判定ユニット34、および第1終了ユニット35を備える。

10

【0106】

作成ユニット31は、オペレーティングシステムで実行中のアプリケーションにしたがって、オペレーティングシステムで実行中のアプリケーションの識別情報を少なくとも含むアプリケーションリストを作成するように構成される。

【0107】

走査ユニット32は、アプリケーションリストの識別情報を走査するように構成される。

【0108】

第1判定ユニット33は、走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションであるかどうかを判定するように構成される。

20

【0109】

第2判定ユニット34は、走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションである場合、ユーザが指定したバックグラウンドアプリケーションの識別情報を含む所定のホワイトリストに走査中の識別情報が含まれているかどうかと、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えているかどうかとを判定するように構成される。

【0110】

第1終了ユニット35は、

所定のホワイトリストに走査中の識別情報が含まれ、上記数が所定の閾値を超えている場合、アプリケーションリストから1つのバックグラウンドアプリケーションに対応する識別情報を選択し、選択した識別情報に対応するバックグラウンドアプリケーションを終了し、

30

所定のホワイトリストに走査中の識別情報が含まれておらず、上記数が所定の閾値を超えている場合、走査中の識別情報に対応するアプリケーションを終了するように構成される。

【0111】

本開示の実施形態に係るバックグラウンドアプリケーションを制御する装置では、終了するバックグラウンドアプリケーションを端末装置によって決定するために、バックグラウンドアプリケーションの数が所定の閾値を超えているかどうかと、所定のホワイトリストに終了するバックグラウンドアプリケーションの識別情報が含まれているかどうかとをいずれも検討する必要がある。所定のホワイトリストには、ユーザが指定したバックグラウンドアプリケーションの識別情報が含まれる。すなわち、ユーザは、所定のホワイトリスト内の識別情報に対応するバックグラウンドアプリケーションのいずれについても終了を望まない。走査中の識別情報について、所定のホワイトリストにその走査中の識別情報が含まれており、バックグラウンドアプリケーションの数が所定の閾値を超えている場合、バックグラウンドアプリケーションを1つ選択して終了する。所定のホワイトリストに走査中の識別情報が含まれておらず、バックグラウンドアプリケーションの数が所定の閾値を超えている場合、走査中の識別情報に対応するアプリケーションを終了する。ユーザが指定したバックグラウンドアプリケーションが軽率に終了されてしまうことがないため

40

50

、最大メモリを占有する終了対象のバックグラウンドアプリケーションが、ユーザが指定したバックグラウンドアプリケーションになってしまう可能性がある関連技術の方法に比べて、バックグラウンドアプリケーションを適切に制御することができ、ユーザエクスペリエンスを改善することができる。

【0112】

あるいは、作成ユニット31は、算出サブユニットおよび配列サブユニットを備える。

【0113】

算出サブユニットは、オペレーティングシステムで実行中のアプリケーションの実行状態の優先度を算出するように構成され、実行状態の優先度は、高いものから低いものへ、順に、フォアグラウンド、可視、可感知、バックグラウンドとなっている。

10

【0114】

配列サブユニットは、実行中のアプリケーションの識別情報を実行状態の優先度にしたがって高いものから低いものへ順に配列してアプリケーションリストを得るように構成される。

【0115】

あるいは、第1判定ユニット33は、取得サブユニットおよび第1判定サブユニットを備える。

【0116】

取得サブユニットは、走査中の識別情報に対応するアプリケーションの実行状態の優先度を取得するように構成される。

20

【0117】

第1判定サブユニットは、走査中の識別情報に対応するアプリケーションの実行状態の優先度がバックグラウンドである場合に、走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションであると判定するように構成される。

【0118】

あるいは、第1終了ユニット35は、発見サブユニットおよび選択サブユニットを備える。

【0119】

発見サブユニットは、所定のホワイトリストには含まれておらず、アプリケーションリストにおいて走査中の識別情報に先行する1つ以上の識別情報を見つけるように構成される。

30

【0120】

選択サブユニットは、

所定のホワイトリストには含まれておらず、アプリケーションリストにおいて走査中の識別情報に先行する少なくとも1つの識別情報が見つかった場合、該1つ以上の識別情報のうちの1つを選択するか、該1つ以上の識別情報から走査中の識別情報に最も近い識別情報を選択し、

所定のホワイトリストには含まれておらず、アプリケーションリストにおいて走査中の識別情報に先行する識別情報が見つからなかった場合、走査中の識別情報を選択するように構成される。

40

【0121】

あるいは、装置は、第1取得ユニット、第3判定ユニット、および第2終了ユニットをさらに備える。

【0122】

第1取得ユニットは、所定のホワイトリストに走査中の識別情報が含まれており、上記数が所定の閾値を超えていない場合、走査中の識別情報に対応するアプリケーションの休止時間を取得するように構成される。

【0123】

第3判定ユニットは、該休止時間が第1所定時間を超えているかどうかを判定するように構成される。

50

【 0 1 2 4 】

第 2 終了ユニットは、上記休止時間が第 1 所定時間を超えている場合、走査中の識別情報に対応するアプリケーションを終了するように構成される。

【 0 1 2 5 】

あるいは、装置は、検知ユニットおよび第 1 開放ユニットをさらに備える。

【 0 1 2 6 】

検知ユニットは、休止時間が第 1 所定時間を超えていない場合に、休止時間が第 2 所定時間を超えているかどうかを検知するように構成される。

【 0 1 2 7 】

第 1 開放ユニットは、休止時間が第 2 所定時間を超えている場合に、走査中の識別情報に対応するアプリケーションのユーザインターフェースが占有しているメモリを開放するように構成される。ここで、第 2 所定時間は第 1 所定時間よりも短い。

10

【 0 1 2 8 】

あるいは、装置は、第 2 取得ユニット、第 4 判定ユニット、および第 2 開放ユニットをさらに備える。

【 0 1 2 9 】

第 2 取得ユニットは、所定のホワイトリストに走査中の識別情報が含まれており、上記数が所定の閾値を超えていない場合、走査中の識別情報に対応する所定のアプリケーションの休止時間を取得するように構成される。

【 0 1 3 0 】

第 4 判定ユニットは、該休止時間が第 3 所定時間を超えているかどうかを判定するように構成される。

20

【 0 1 3 1 】

第 2 開放ユニットは、上記休止時間が第 3 所定時間を超えている場合、走査中の識別情報に対応するアプリケーションのユーザインターフェースが占有しているメモリを開放するように構成される。

【 0 1 3 2 】

あるいは、第 1 取得ユニットおよび第 2 取得ユニットは、計測サブユニットおよび第 2 判定サブユニットを備える。

【 0 1 3 3 】

計測サブユニットは、走査中の識別情報に対応するアプリケーションの処理がオペレーティングシステムの他の処理と相互作用しない時間を計測するように構成される。

30

【 0 1 3 4 】

第 2 判定サブユニットは、該時間を休止時間とするように構成される。

【 0 1 3 5 】

上述の実施形態に係る装置については、これら装置に含まれる個々のモジュールの動作の詳細については、バックグラウンドアプリケーションの制御方法に関する実施形態において詳細に説明したので、それらについては、本実施形態では詳述しない。

【 0 1 3 6 】

図 4 は、本開示の例示的な実施形態に係るバックグラウンドアプリケーションを制御する装置 800 を示す模式図である。たとえば、装置 800 は、携帯電話、コンピュータ、デジタル放送端末、メッセージ送受信機器、ゲームコントローラ、タブレット装置、医療機器、運動機器、PDA などであってもよい。

40

【 0 1 3 7 】

図 4 を参照し、装置 800 は、プロセッサ部 802、メモリ 804、電源部 806、マルチメディア部 808、音声部 810、入出力 (I/O) インターフェース 812、センサ部 814、および通信部 816 のうち 1 つ以上を備えていてもよい。

【 0 1 3 8 】

通常、プロセッサ部 802 は、表示、通話、データ通信、カメラ動作、記録動作に関連する動作などの装置 800 の動作全体を制御する。プロセッサ部 802 は、1 つ以上のプ

50

ロセッサ 820 を備えて上記方法のステップの全部または一部を実施する命令を実行してもよい。さらに、プロセッサ部 802 は、プロセッサ部 802 と他の構成要素との間の相互作用を支援するモジュールを 1 つ以上備えていてもよい。たとえば、プロセッサ部 802 は、マルチメディア部 808 とプロセッサ部 802 との間の相互作用を支援するマルチメディアモジュールを備えていてもよい。

【0139】

メモリ 804 は、各種データを格納して装置 800 の動作をサポートするように構成される。このようなデータとしては、たとえば、装置 800 で操作される任意のアプリケーションまたは方法に関する命令、連絡先データ、電話帳データ、メッセージ、写真、動画などが挙げられる。メモリ 804 は、揮発性あるいは不揮発性、またはその組み合わせなど、どの種類のメモリ装置を使って実装してもよい。たとえば、スタティック RAM (Static Random Access Memory)、EEPROM (Electrically Erasable Programmable Read-Only Memory)、EPROM (Erasable Programmable Read-Only Memory)、PROM (Programmable Read-Only Memory)、ROM (Read-Only Memory)、磁気メモリ、フラッシュメモリ、磁気ディスク、光ディスクなどである。

10

【0140】

電源部 806 は、装置 800 の各種構成要素に電力を供給する。電源部 806 は、電力管理システム、1 つ以上の電源、装置 800 における電力の生成、管理、配電に関連する他の構成要素を備えていてもよい。

20

【0141】

マルチメディア部 808 は、装置 800 とユーザとの間の出力インターフェースを提供する画面を備える。いくつかの実施形態では、該画面は液晶表示装置 (LCD) およびタッチパネル (TP) を含んでもよい。画面にタッチパネルが含まれる場合、該画面は、ユーザからの入力信号を受け付けるタッチスクリーンとして実装され得る。タッチパネルは、タッチパネル上のタッチ、スワイプ、ジェスチャーを感知するタッチセンサを 1 つ以上備える。タッチセンサは、タッチ操作またはスワイプ操作の範囲を感知することに加えて、タッチ操作またはスワイプ操作の時間および押圧力も感知する。いくつかの実施形態では、マルチメディア部 808 は、フロントカメラおよび/またはバックカメラを備える。フロントカメラおよびバックカメラは、装置 800 が撮影モードや動画モードなどの動作モードで稼働している間、外部からマルチメディアデータを受信できる。フロントカメラおよびバックカメラは、それぞれ固定光学レンズ系であってもよいし、フォーカス機能や光学ズーム機能を有していてもよい。

30

【0142】

音声部 810 は、音声信号を出力および/または入力するように構成される。たとえば、音声部 810 はマイク (「MIC」) を含む。マイクは、装置 800 が、通話モード、記録モード、音声認識モードなどの動作モードで稼働している際、外部からの音声信号を受信するように構成される。受信した音声信号は、さらにメモリ 804 に記憶されたり、通信部 816 を介して送信されたりしてもよい。いくつかの実施形態では、音声部 810 は音声信号を出力するためのスピーカーを備える。

40

【0143】

I/O インターフェース 812 は、プロセッサ部 802 と、キーボード、クリックホイール、ボタンなどの周辺機器インターフェースモジュールとの間のインターフェースを提供する。ボタンとしては、ホームボタン、ボリュームボタン、起動ボタン、ロックボタンが挙げられるが、これらに限定されない。

【0144】

センサ部 814 は、装置 800 の各種機能の状態を判断するためのセンサを 1 つ以上備える。たとえば、センサ部 814 は、装置 800 の開閉状態、装置 800 の構成要素間 (たとえば、ディスプレイとキーパッド) の相対位置、装置 800 または装置 800 の構成

50

要素の位置の変化、装置 800 に対するユーザの接触の有無、装置 800 の向きまたは加速 / 減速、および装置 800 の温度変化を検知することができる。センサ部 814 は、近くにある物体の存在を物理的に接触せずに検出するように構成された近接センサを備えていてもよい。また、センサ部 814 は、撮像アプリケーション用の CMOS イメージセンサや CCD イメージセンサなどの光センサを備えていてもよい。いくつかの実施形態では、センサ部 814 は、加速度センサ、ジャイロセンサ、磁気センサ、圧力センサ、または温度センサを備えていてもよい。

【0145】

通信部 816 は、装置 800 と他の装置との間の有線または無線の通信を支援するように構成される。装置 800 は Wi-Fi、2G、3G などの通信規格またはその組み合わせに基づいて、無線ネットワークにアクセスすることができる。例示的な一実施形態では、通信部 816 は、放送信号や放送に関連する情報を、外部の放送管理システムから放送チャネル経由で受信する。例示的な一実施形態では、通信部 816 は、近距離通信を支援する近距離無線通信 (NFC) モジュールをさらに含む。たとえば、NFC モジュールの実装には、RFID (Radio Frequency Identification) 技術、IrDA (Infrared Data Association) 技術、超広帯域無線 (UWB) 技術、Bluetooth (登録商標) (BT) 技術、その他の通信技術を用いることができる。

【0146】

ある例示的な実施形態では、装置 800 は、1つ以上の特定用途向け集積回路 (ASIC)、デジタル信号プロセッサ (DSP)、デジタル信号処理装置 (DSPD)、プログラム可能な論理素子 (PLD)、フィールドプログラマブルゲートアレイ (FPGA)、コントローラ、マイクロコントローラ、マイクロプロセッサ、その他の電子部品を用いて実装し、本開示の実施形態に係る上記方法を実施することができる。

【0147】

また、ある例示的な実施形態では、上記方法を実施するため、装置 800 のプロセッサ 820 によって実行可能な命令を格納するメモリ 804 のような、命令を格納する非一時的コンピュータ読取可能記憶媒体が提供される。非一時的コンピュータ読取可能記憶媒体としては、たとえば ROM、RAM (Random Access Memory)、CD-ROM、磁気テープ、フロッピー (登録商標) ディスク、光データ記憶装置、などが挙げられる。

【0148】

端末装置のプロセッサによって実行されると、バックグラウンドアプリケーションを制御する方法を端末装置に実行させる命令を記憶する非一時的コンピュータ読取可能記憶媒体が提供される。該方法は、

オペレーティングシステムで実行中のアプリケーションにしたがって、オペレーティングシステムで実行中のアプリケーションの識別情報を少なくとも含むアプリケーションリストを作成するステップと、

アプリケーションリストの識別情報を走査するステップと、

走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションであるかどうかを判定するステップと、

走査中の識別情報に対応するアプリケーションがバックグラウンドアプリケーションである場合、ユーザが指定したバックグラウンドアプリケーションの識別情報を含む所定のホワイトリストに走査中の識別情報が含まれるかどうかと、アプリケーションリスト内のバックグラウンドアプリケーションに対応する識別情報の数が所定の閾値を超えているかどうかとを判定するステップと、

所定のホワイトリストに走査中の識別情報が含まれており、上記数が所定の閾値を超えている場合、アプリケーションリストから1つのバックグラウンドアプリケーションに対応する識別情報を選択し、選択した識別情報に対応するバックグラウンドアプリケーションを終了するステップと、

10

20

30

40

50

所定のホワイトリストに走査中の識別情報が含まれておらず、上記数が所定の閾値を超えている場合、走査中の識別情報に対応するアプリケーションを終了するステップとを含む。

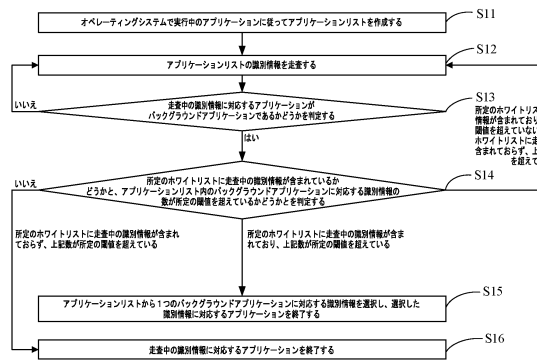
【0149】

本開示の他の実施形態は、本願明細書の記載や本開示の実施を通じて当業者には明らかであろう。本願は、本開示の原理に従ったあらゆる変形、使用、適合を包含することを意図しており、当該分野で公知または慣行の範疇にある逸脱も本願の範囲に含まれる。明細書の記載や例はあくまでも例示であって、本開示の真の範囲および精神は特許請求の範囲によって示す。

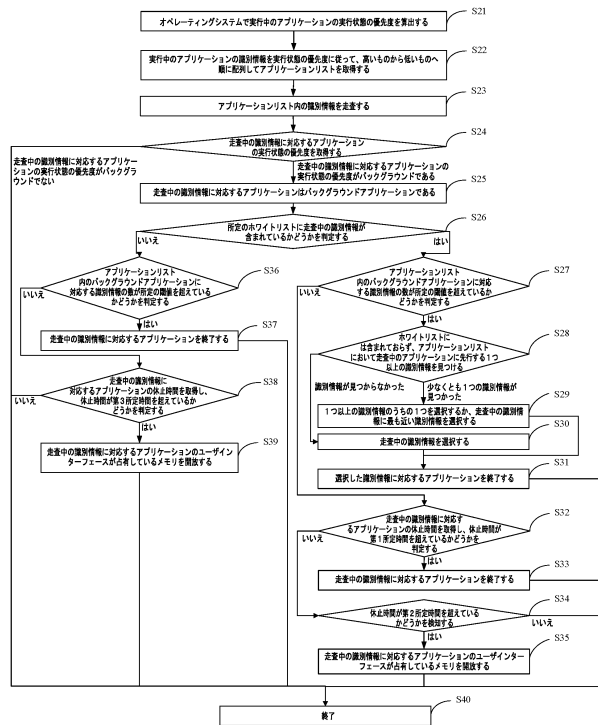
【0150】

本開示は、上述および添付図面に図示した特定の構成に何ら限定されるものではなく、本開示の範囲から逸脱することなく種々の変形、変更が可能であることが理解されるであろう。本開示の範囲は、特許請求の範囲によってのみ限定されるものである。

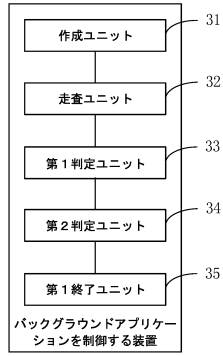
【図1】



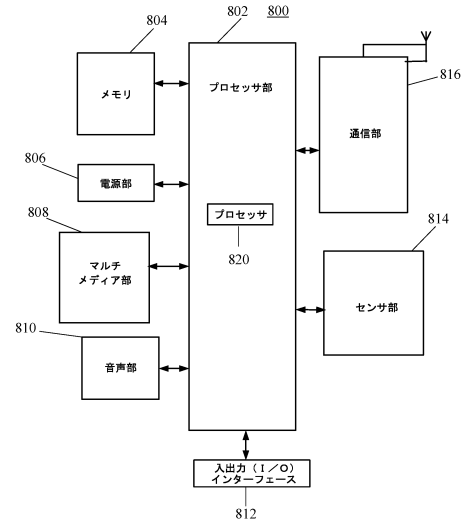
【図2】



【図3】



【図4】



フロントページの続き

- (72)発明者 ユエン, ジュン
中華人民共和国, 100085 ベイジン, ハイディアン ディストリクト, キンヘ ミドル ス
トリート, ナンバー 68, レインボー シティ ショッピング モール 2 オブ チャイナ
リゾーシズ, フロア 13
- (72)発明者 リー, ミンハオ
中華人民共和国, 100085 ベイジン, ハイディアン ディストリクト, キンヘ ミドル ス
トリート, ナンバー 68, レインボー シティ ショッピング モール 2 オブ チャイナ
リゾーシズ, フロア 13
- (72)発明者 リァン, ジンシァン
中華人民共和国, 100085 ベイジン, ハイディアン ディストリクト, キンヘ ミドル ス
トリート, ナンバー 68, レインボー シティ ショッピング モール 2 オブ チャイナ
リゾーシズ, フロア 13
- (72)発明者 タオ, ジェンタオ
中華人民共和国, 100085 ベイジン, ハイディアン ディストリクト, キンヘ ミドル ス
トリート, ナンバー 68, レインボー シティ ショッピング モール 2 オブ チャイナ
リゾーシズ, フロア 13

審査官 田中 幸雄

- (56)参考文献 米国特許出願公開第2014/0007106 (US, A1)
米国特許出願公開第2014/0074082 (US, A1)
国際公開第2012/127596 (WO, A1)
本田亮ほか, Androidソフトウェアプラットフォーム開発への取組み, FUJITSU, 日本,
富士通株式会社, 2012年 9月, Vol. 63 No. 5, 591-597ページ

- (58)調査した分野(Int.Cl., DB名)
G06F 9/48