

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第5784816号
(P5784816)

(45) 発行日 平成27年9月24日 (2015. 9. 24)

(24) 登録日 平成27年7月31日 (2015. 7. 31)

(51) Int. Cl.	F I
HO 4W 76/04 (2009. 01)	HO 4W 76/04
HO 4M 1/00 (2006. 01)	HO 4M 1/00 R
HO 4M 11/00 (2006. 01)	HO 4M 11/00 3 O 2
HO 4W 52/02 (2009. 01)	HO 4W 52/02

請求項の数 36 (全 39 頁)

(21) 出願番号	特願2014-501056 (P2014-501056)	(73) 特許権者	595020643
(86) (22) 出願日	平成23年11月4日 (2011. 11. 4)		クゥアルコム・インコーポレイテッド
(65) 公表番号	特表2014-514813 (P2014-514813A)		QUALCOMM INCORPORATED
(43) 公表日	平成26年6月19日 (2014. 6. 19)		ED
(86) 国際出願番号	PCT/US2011/059439		アメリカ合衆国、カリフォルニア州 92
(87) 国際公開番号	W02012/128792		121-1714、サン・ディエゴ、モア
(87) 国際公開日	平成24年9月27日 (2012. 9. 27)		ハウス・ドライブ 5775
審査請求日	平成25年11月18日 (2013. 11. 18)	(74) 代理人	100108855
(31) 優先権主張番号	61/454, 457		弁理士 蔵田 昌俊
(32) 優先日	平成23年3月18日 (2011. 3. 18)	(74) 代理人	100109830
(33) 優先権主張国	米国 (US)		弁理士 福原 淑弘
(31) 優先権主張番号	13/288, 933	(74) 代理人	100088683
(32) 優先日	平成23年11月3日 (2011. 11. 3)		弁理士 中村 誠
(33) 優先権主張国	米国 (US)	(74) 代理人	100103034
			弁理士 野河 信久

最終頁に続く

(54) 【発明の名称】 ネットワークアクセス要求の管理

(57) 【特許請求の範囲】

【請求項 1】

モバイルデバイスにおけるワイヤレス通信のための方法であって、

前記モバイルデバイス上の第1のアプリケーションからの要求をインターセプトすることであって、前記要求が、前記モバイルデバイスのための通信を行うための通信チャネルを確立するための要求である、インターセプトすることと、

前記モバイルデバイスがバックグラウンドモードにある場合、前記モバイルデバイス上で実行しているオペレーティングシステムの伝送制御プロトコル／インターネットプロトコル (TCP/IP) スタックに前記要求が到着することを遅延させることと、

トリガリングイベントを検出すると、前記オペレーティングシステムに対して前記通信チャネルを確立するための前記要求をリリースすることと、ここにおいて前記トリガリングイベントは前記モバイルデバイスがアクティブモードに入ることである、

を備える、方法。

【請求項 2】

前記要求を、前記モバイルデバイスのための通信を行うための他のインターセプトされた要求とアグリゲートすること

をさらに備える、請求項 1 に記載の方法。

【請求項 3】

前記第1のアプリケーションからの前記要求を前記インターセプトすることと、前記他の要求を前記インターセプトすることとが、異なる時間に行われる、請求項 2 に記載の方

10

20

法。

【請求項 4】

ラッパーのための命令を実行することであって、前記実行されたラッパーが、前記第 1 のアプリケーションからの前記要求を前記インターセプトすることを行う、実行することをさらに備える、請求項 1 に記載の方法。

【請求項 5】

前記ラッパーが、前記モバイルデバイスのアプリケーションレイヤと前記オペレーティングシステムのソケットレイヤとの間に位置する、請求項 4 に記載の方法。

【請求項 6】

前記第 1 のアプリケーションを、要求が遅延する種類のアプリケーションとして識別すること
10
をさらに備える、請求項 1 に記載の方法。

【請求項 7】

アプリケーションをクリティカルアプリケーションまたは非クリティカルアプリケーションとして識別することと、

非クリティカルアプリケーションからの要求のみを遅延させることと、
をさらに備える、請求項 6 に記載の方法。

【請求項 8】

前記第 1 のアプリケーションの遅延許容差を判断することと、

前記判断された遅延許容差に基づいて前記第 1 のアプリケーションにコールバック関数
20
を与えることであって、前記コールバック関数が、通信リソースに接続するように前記第 1 のアプリケーションに命令する、与えることと、
をさらに備える、請求項 1 に記載の方法。

【請求項 9】

前記第 1 のアプリケーションに関連する第 1 のタイマーの満了時間を判断することと、
第 2 のアプリケーションに関連する許容差と第 2 のタイマーの満了時間とを判断することと、

前記第 1 のタイマーの前記満了時間と、前記許容差と、前記第 2 のタイマーの前記満了時間とに基づいて、前記第 2 のタイマーを満了させることと、

前記モバイルデバイスのための通信を行うための前記第 1 のアプリケーションからの前
30
記要求と前記第 2 のアプリケーションからのインターセプトされた要求とをリリースすることと、
をさらに備える、請求項 1 に記載の方法。

【請求項 10】

前記第 1 のアプリケーションからデッドラインを受信することと、

前記デッドラインの前まで前記要求を遅延させることと、

前記デッドラインより前に前記通信リソースに接続するための前記要求をリリースすることと、
をさらに備える、請求項 1 に記載の方法。

【請求項 11】

前記要求が、前記モバイルデバイスのための通信チャネルを確立するためのシステムコールを備える、請求項 1 に記載の方法。

【請求項 12】

前記トリガリングイベントを検出すると、前記オペレーティングシステムのソケットレイヤに対して前記要求をリリースすること

をさらに備える、請求項 1 に記載の方法。

【請求項 13】

前記要求を前記リリースすることがどのくらいの頻度で行われるかに関係する間隔についての指示を受信すること

をさらに備える、請求項 1 に記載の方法。

10

20

30

40

50

【請求項 14】

前記間隔が、ネットワーク中のステートフルインターネットプロトコル（IP）ミドルボックスにおけるタイムアウト値よりも小さい、請求項13に記載の方法。

【請求項 15】

ワイヤレス通信のために構成されたモバイルデバイスであって、
プロセッサと、

前記プロセッサと電子通信しているメモリであって、実行可能な命令を記憶しており、オペレーティングシステムを備える、メモリとを備え、

前記プロセッサが、

前記モバイルデバイス上の第1のアプリケーションからの要求をインターセプトすることであって、前記要求が、前記モバイルデバイスのための通信を行うための通信チャネルを確立するための要求である、インターセプトすることと、

前記モバイルデバイスがバックグラウンドモードにある場合、前記モバイルデバイス上で実行しているオペレーティングシステムのTCP/IPスタックに前記要求が到着することを遅延させることと、

トリガリングイベントを検出すると、前記オペレーティングシステムに対して前記通信チャネルを確立するための前記要求をリリースすることと、ここにおいて前記トリガリングイベントは前記モバイルデバイスがアクティブモードに入ることである、
を行うための命令を実行するように構成された、モバイルデバイス。

【請求項 16】

前記要求を、前記モバイルデバイスのための通信を行うための他のインターセプトされた要求とアグリゲートすること

をさらに備える、請求項15に記載のモバイルデバイス。

【請求項 17】

前記第1のアプリケーションからの前記要求を前記インターセプトすることと、前記他の要求を前記インターセプトすることとが、異なる時間に行われる、請求項16に記載のモバイルデバイス。

【請求項 18】

前記メモリがラッパーをさらに備え、前記プロセッサが、前記ラッパーに対して命令を実行するようにさらに構成され、前記命令が実行されたとき、前記ラッパーが、前記第1のアプリケーションからの前記要求をインターセプトするように構成された、請求項15に記載のモバイルデバイス。

【請求項 19】

前記ラッパーが、前記モバイルデバイスのアプリケーションレイヤと前記オペレーティングシステムのソケットレイヤとの間に位置する、請求項18に記載のモバイルデバイス。

【請求項 20】

前記プロセッサは、

前記第1のアプリケーションを、要求が遅延する種類のアプリケーションとして識別すること

を行うための命令を実行するようにさらに構成された、請求項15に記載のモバイルデバイス。

【請求項 21】

前記プロセッサが、

アプリケーションをクリティカルアプリケーションまたは非クリティカルアプリケーションとして識別することと、

非クリティカルアプリケーションからの要求のみを遅延させることと、
を行うための命令を実行するようにさらに構成された、請求項20に記載のモバイルデバイス。

【請求項 2 2】

前記プロセッサは、

前記第 1 のアプリケーションの遅延許容差を判断することと、

前記判断された遅延許容差に基づいて前記第 1 のアプリケーションにコールバック関数を与えることであって、前記コールバック関数が、前記通信リソースに接続するように前記第 1 のアプリケーションに命令する、与えることと、

を行うための命令を実行するようにさらに構成された、請求項 1 5 に記載のモバイルデバイス。

【請求項 2 3】

前記プロセッサが、

前記第 1 のアプリケーションに関連する第 1 のタイマーの満了時間を判断することと、

第 2 のアプリケーションに関連する許容差と第 2 のタイマーの満了時間とを判断することと、

前記第 1 のタイマーの前記満了時間と、前記許容差と、前記第 2 のタイマーの前記満了時間とに基づいて、前記第 2 のタイマーを満了させることと、

前記モバイルデバイスのための通信を行うための前記第 1 のアプリケーションからの前記要求と前記第 2 のアプリケーションからのインターセプトされた要求とをリリースすることと、

を行うための命令を実行するようにさらに構成された、請求項 1 5 に記載のモバイルデバイス。

【請求項 2 4】

前記プロセッサが、

前記第 1 のアプリケーションからデッドラインを受信することと、

前記デッドラインの前まで前記要求を保留することと、

前記デッドラインより前に前記通信チャネルを確立するための前記要求をリリースすることと、

を行うための命令を実行するようにさらに構成された、請求項 1 5 に記載のモバイルデバイス。

【請求項 2 5】

前記要求が、通信チャネルを確立するためのシステムコールを備える、請求項 1 5 に記載のモバイルデバイス。

【請求項 2 6】

前記プロセッサが、

前記トリガリングイベントを検出すると、前記オペレーティングシステムのソケットレイヤに対して前記要求をリリースすること

を行うための命令を実行するようにさらに構成された、請求項 1 5 に記載のモバイルデバイス。

【請求項 2 7】

前記プロセッサは、

前記要求を前記リリースすることがどのくらいの頻度で行われるかに関係する間隔についての指示を受信すること

を行うための命令を実行するようにさらに構成された、請求項 1 5 に記載のモバイルデバイス。

【請求項 2 8】

前記間隔が、ネットワーク中のステートフルインターネットプロトコル (IP) ミドルボックスにおけるタイムアウト値よりも小さい、請求項 2 7 に記載のモバイルデバイス。

【請求項 2 9】

モバイルデバイス上のアプリケーションからのネットワークアクセスの要求を管理するように構成された装置であって、

前記モバイルデバイス上の第 1 のアプリケーションからの要求をインターセプトするた

10

20

30

40

50

め的手段であって、前記要求が、前記モバイルデバイスのための通信を行うための通信チャンネルを確立するための要求である、インターセプトするための手段と、

前記モバイルデバイスがバックグラウンドモードにある場合、前記モバイルデバイス上で実行しているオペレーティングシステムのTCP/IPスタックに前記要求が到着することを遅延させるための手段と、

トリガリングイベントを検出すると、前記オペレーティングシステムに対して前記通信チャンネルを確立するための前記要求をリリースするための手段と、ここにおいて前記トリガリングイベントは前記モバイルデバイスがアクティブモードに入ることである、
を備える、装置。

【請求項30】

前記要求を、前記モバイルデバイスのための通信を行うための他のインターセプトされた要求とアグリゲートするための手段
をさらに備える、請求項29に記載の装置。

【請求項31】

前記第1のアプリケーションからの前記要求を前記インターセプトすることと、前記他の要求を前記インターセプトすることとが、異なる時間に行われる、請求項29に記載の装置。

【請求項32】

ラッパーを実行するための手段であって、前記実行されたラッパーが、前記第1のアプリケーションからの前記要求をインターセプトするように構成された、実行するための手段
をさらに備える、請求項29に記載の装置。

【請求項33】

前記ラッパーが、前記モバイルデバイスのアプリケーションレイヤと前記オペレーティングシステムのソケットレイヤとの間に位置する、請求項32に記載の装置。

【請求項34】

前記第1のアプリケーションを、要求が遅延する種類のアプリケーションとして識別するための手段
をさらに備える、請求項29に記載の装置。

【請求項35】

アプリケーションをクリティカルアプリケーションまたは非クリティカルアプリケーションとして識別するための手段と、

非クリティカルアプリケーションからの要求のみを遅延させるための手段と、
をさらに備える、請求項34に記載の装置。

【請求項36】

モバイルデバイス上のアプリケーションからのネットワークアクセスの要求を管理するように構成されたコンピュータプログラムであって、前記コンピュータプログラムが、コンピュータに、前記モバイルデバイス上の第1のアプリケーションからの要求をインターセプトさせるためのコードであって、前記要求が、前記モバイルデバイスのための通信を行うための通信チャンネルを確立するための要求である、インターセプトさせるためのコードと、

前記コンピュータに、前記モバイルデバイスがバックグラウンドモードにある場合、前記モバイルデバイス上で実行しているオペレーティングシステムのTCP/IPスタックに前記要求が到着することを遅延させるためのコードと、

前記コンピュータに、トリガリングイベントを検出すると、前記オペレーティングシステムに対して前記通信チャンネルを確立するための前記要求をリリースさせるためのコードと、ここにおいて前記トリガリングイベントは前記モバイルデバイスがアクティブモードに入ることである、
を備える、コンピュータプログラム。

【発明の詳細な説明】

10

20

30

40

50

【技術分野】

【0001】

関連出願の相互参照

本出願は、その開示の全体が参照により本明細書に明確に組み込まれる、2011年3月18日に出願された、代理人整理番号第111354P1号を有する、Giarettaらによる「CONNECTIVITY MANAGEMENT FOR APPLICATIONS ON A USER DEVICE」と題する米国仮特許出願第61/454,457号に対する米国特許法第119条(e)項に基づく利益を主張する。

【背景技術】

【0002】

10

ワイヤレスデバイスに広範囲のアドオンサービスおよび特徴を与えるように動作するアプリケーションまたはデバイスアプレットが現在利用可能である。たとえば、ワイヤレスデバイスは、ショッピング、検索、位置特定、運転ナビゲーション、または一連の他の機能など、付加価値機能を実行するために、デバイスアプレットをダウンロードし、起動することが現在可能である。ネットワークおよびアプリケーションプロバイダは、概して、追加料金でデバイスユーザにこれらのデバイスアプレットを提供する。したがって、デバイスアプレットの使用は、ワイヤレスデバイスの機能性およびユーザビリティを高め得、デバイス自体では元来利用可能でない特徴および利便性をデバイスユーザに提供する。

【0003】

一般に、ワイヤレスデバイスは、いくつかの無線のいずれかを使用して、1つまたは複数の通信ネットワークとインターフェースする。たとえば、ワイヤレスデバイスは、セルラー、Wi-Fi、Bluetooth（登録商標）、または他のタイプの無線アクセス技術を使用して通信を提供する様々な無線を含み得る。したがって、ワイヤレスデバイス上で実行しているアプリケーションは、通信チャネルを確立するために無線とインターフェースし、チャネルは、適切なネットワークと通信するためにアプリケーションによって使用される。

20

【0004】

アプリケーションは、デバイスがバックグラウンドモードにあるときでも、通信チャネルを確立するためにワイヤレスデバイス上の無線とインターフェースし続け得る。デバイス上にインストールされたアプリケーションの数が増加するにつれて、デバイスがアクティブでない間のネットワーク通信の反復確立により、デバイスのバッテリー電力が不必要に消費され得る。さらに、スマートフォンなどのワイヤレスデバイスの使用が増加するにつれて、通信チャネルのセットアップに関連するネットワークシグナリングによってデータネットワークに過負荷がかかるようになり得る。

30

【発明の概要】

【0005】

ネットワークとモバイルデバイス上で実行しているアプリケーションとの間の接続性を管理するための方法、システム、およびデバイスについて説明する。一例では、デバイス上で実行しているアプリケーションからのネットワークアクセスの要求がインターセプトされ得る。たとえば、要求をインターセプトするために、モバイルデバイスのアプリケーションとオペレーティングシステムとの間にラッパーが配置され得る。要求をインターセプトすると、要求は、オペレーティングシステムの伝送制御プロトコル/インターネットプロトコル(TCP/IP)スタックに到着することを保留され得る。一例では、要求は、トリガリングイベントが発生したときに、オペレーティングシステムに対してリリースされ得る。要求のキャプチャ、保留、およびリリースは、モバイルデバイスがバックグラウンドモードにあるときに行われ得る。

40

【0006】

一構成では、要求は、モバイルデバイスのための通信を行うための他のインターセプトされた要求とアグリゲートされ得る。アプリケーションからの要求のインターセプションと、他の要求のインターセプションとは、異なる時間に行われ得る。

50

【0007】

一例では、ラッパーのための命令が実行され得る。実行されたラッパーは、第1のアプリケーションからの要求のインターセプションを実行し得る。一構成では、ラッパーは、モバイルデバイスのアプリケーションレイヤとオペレーティングシステムのソケットレイヤとの間に位置し得る。

【0008】

一構成では、第1のアプリケーションは、要求が保留される種類のアプリケーションとして識別され得る。アプリケーションは、クリティカルアプリケーションまたは非クリティカルアプリケーションとして識別され得る。非クリティカルアプリケーションからの要求のみが保留され得る。

10

【0009】

一実施形態では、トリガリングイベントは、タイマーの満了、ディスプレイのステータス変更、マイクロフォンのステータス変更、スピーカーのステータス変更、モバイルデバイスの全地球測位システム（GPS）センサーのステータス変更、ユニバーサルシリアルバスポートが使用中であるという指示、モバイルデバイスにオーディオ機器が接続されているという指示、モバイルデバイスにビデオ機器が接続されているという指示、Wi-Fiタイプのネットワークへの接続が利用可能であるという指示、またはセルラータイプのネットワークへの無線接続が開いているという指示のうちの少なくとも1つを含み得る。

【0010】

さらに、一例では、第1のアプリケーションの遅延許容差が判断され得る。さらに、判断された遅延許容差に基づいて第1のアプリケーションにコールバック関数が与えられ得る。コールバック関数は、通信リソースに接続するように第1のアプリケーションに命令し得る。

20

【0011】

一構成では、第1のアプリケーションに関連する第1のタイマーの満了時間が判断され得る。また、第2のアプリケーションに関連する許容差と第2のタイマーの満了時間とが判断され得る。第2のタイマーは、第1のタイマーの満了時間と、許容差と、第2のタイマーの満了時間とに基づいて満了させられ得る。モバイルデバイスのための通信を行うための第1のアプリケーションからの要求と第2のアプリケーションからのインターセプトされた要求とがリリースされ得る。

30

【0012】

一例では、アプリケーションからデッドラインが受信され得る。要求は、デッドラインの前まで保留され得る。通信リソースに接続するための要求は、デッドラインより前にリリースされ得る。一構成では、要求は、モバイルデバイスのための通信チャネルを確立するためのシステムコールを含み得る。要求は、トリガリングイベントを検出すると、オペレーティングシステムのソケットレイヤに対してリリースされ得る。

【0013】

一実施形態では、要求をリリースすることがどのくらいの頻度で行われるかに関する間隔についての指示が受信され得る。間隔は、ネットワーク中のステートフルインターネットプロトコル（IP）ミドルボックスにおけるタイムアウト値よりも小さくなり得る。

40

【0014】

また、ワイヤレス通信のために構成されたモバイルデバイスについて説明する。本デバイスは、プロセッサと、プロセッサと電子通信しているメモリとを含み得る。メモリはオペレーティングシステムを含み得る。プロセッサは接続性エンジンを含み得る。エンジンは、モバイルデバイス上の第1のアプリケーションからの要求をインターセプトするようにとの命令を実行するようにさらに構成され得る。要求は、モバイルデバイスのための通信を行うための要求であり得る。エンジンは、モバイルデバイス上で実行しているオペレーティングシステムのTCP/IPスタックに要求が到着することを保留することと、トリガリングイベントを検出すると、オペレーティングシステムに対して要求をリリースすることとを行うようにさらに構成され得る。

50

【0015】

また、モバイルデバイス上のアプリケーションからのネットワークアクセスの要求を管理するように構成された装置について説明する。本装置は、モバイルデバイス上のアプリケーションからの要求をインターセプトするための手段を含む。要求は、モバイルデバイスのための通信を行うための要求であり得る。本装置は、モバイルデバイス上で実行しているオペレーティングシステムのTCP/IPスタックに要求が到着することを保留するための手段と、トリガリングイベントを検出すると、オペレーティングシステムに対して要求をリリースするための手段とをさらに含み得る。

【0016】

また、モバイルデバイス上のアプリケーションからのネットワークアクセスの要求を管理するように構成されたコンピュータプログラム製品について説明する。本製品は非一時的コンピュータ可読媒体を含み得る。媒体は、モバイルデバイス上のアプリケーションからの要求をインターセプトするためのコードを含み得る。要求は、モバイルデバイスのための通信を行うための要求であり得る。媒体は、モバイルデバイス上で実行しているオペレーティングシステムのTCP/IPスタックに要求が到着することを保留するためのコードと、トリガリングイベントを検出すると、オペレーティングシステムに対して要求をリリースするためのコードとをさらに含み得る。

【0017】

上記では、開示による例の特徴および技術的態様についてやや広く概説した。以下で、追加の特徴について説明する。開示する概念および特定の例は、本開示の同じ目的を実行するための他の構造を変更または設計するための基礎として容易に利用され得る。そのような等価な構成は、添付の特許請求の範囲の趣旨および範囲から逸脱しない。本明細書で開示する概念の編成と動作の方法の両方に関して、それらの概念を特徴づけると考えられる特徴は、添付の図に関連して検討すると以下の説明からより良く理解されよう。図の各々は、例示および説明のみの目的で与えられ、特許請求の範囲の限界を定めるものではない。

【0018】

以下の図面を参照すれば、本発明の性質のさらなる理解が得られ得る。添付の図において、同様の構成要素または特徴は同じ参照ラベルを有し得る。さらに、同じタイプの様々なコンポーネントは、参照ラベルの後に、ダッシュと、それらの同様のコンポーネント同士を区別する第2のラベルとを続けることによって区別され得る。第1の参照ラベルのみが明細書において使用される場合、その説明は、第2の参照ラベルにかかわらず、同じ第1の参照ラベルを有する同様のコンポーネントのうちのいずれか1つに適用可能である。

【図面の簡単な説明】

【0019】

【図1】 ネットワーク環境のブロック図。

【図2】 モバイルデバイスのアーキテクチャを示すブロック図。

【図3】 ネットワークアクセスの要求の遅延を提供するモバイルデバイスのブロック図。

【図4】 ネットワークアクセスの要求を遅延させるためのモバイルデバイス上のアーキテクチャの例示的なブロック図。

【図5】 ネットワークアクセスの要求をアグリゲートするための例示的なタイミング図。

【図6】 モバイルデバイス上に実装されたアーキテクチャの一例を示す図。

【図7】 ネットワークアクセスの要求を遅延させるための方法の一例を示すフローチャート。

【図8】 アプリケーションの分類に基づいてネットワークアクセスの要求を遅延させるための方法の一例を示すフローチャート。

【図9】 いくつかのモバイルデバイスから受信されたネットワークアクセスの要求をアグリゲートするための方法の一例を示すフローチャート。

【図10】 データ接続要求を同期させるための方法の一例を示すフローチャート。

【図11】 3つのアプリケーションが接続要求を周期的に開始するタイミング図。

【図 1 2】 接続要求のいくつかが同期された、図 1 1 のタイミング図。

【図 1 3】 3つのアプリケーションが接続要求を周期的に開始するタイミング図。

【図 1 4】 接続要求のいくつかが同期された、図 1 3 のタイミング図。

【詳細な説明】

【0020】

モバイルデバイス上にインストールされたアプリケーションからの要求をインターセプトするための方法、システム、およびデバイスについて説明する。要求は、モバイルデバイスのための通信チャネルを確立するシステムコールであり得る。「要求」および「システムコール」という用語は互換的に使用され得る。要求は、キャプチャされ、モバイルデバイス上で実行しているオペレーティングシステムのTCP/IPスタックに到着することを保留され得る。インターセプトされた要求は、他のインターセプトされた要求とアグリゲートされ得る。アグリゲートされた要求は、一緒にバンドルされ、モバイルデバイス上でトリガリングイベントを検出するのとはほぼ同時にオペレーティングシステムに対してリリースされ得る。アプリケーションからの要求のキャプチャ、保留、およびアグリゲーションは、モバイルデバイスがバックグラウンドモードにあるときに行われ得る。

10

【0021】

スマートフォン、携帯情報端末などのモバイルデバイスでは、ソフトウェアアプリケーションは、ユーザがデバイスをアクティブに使用していなくても動作し続け得る。ソーシャルネットワーキングアプリケーション、電子メールまたは他の通信アプリケーション、データフィードなどのアプリケーション（普及している例には、Facebook（登録商標）、Gmail（登録商標）、Twitter（登録商標）などがある）は、ユーザがデバイスを使用していなくても、データを送信および受信し続け得る。非アクティビティ動作モードの下でさえ、デバイスが表面上は使用中でないときでも動作し続けるアプリケーションにより、アクティビティにおける電力消費およびスパイク(spikes)が発生し得る。これらのアプリケーションによるアクティビティは、外部ネットワークによって与えられるような通信リソースを利用し得る。

20

【0022】

アプリケーションは、バックグラウンドモードから接続モードへのモバイルデバイスによる頻繁な遷移をトリガし得るか、あるいはアプリケーションは、場合によっては、デバイスがバックグラウンドモードに入ることまたは間欠受信(discontinuous reception) (DRX)などの他の代替接続モードに入ること干渉し得る。ユーザがデバイスをアクティブに操作していないときの、アプリケーションによる無線アクティビティのこれらの高いレベルは、バッテリー寿命の早期の消耗、無線ネットワークの負荷の望ましくない増加、または他の望ましくない影響をもたらし得る。

30

【0023】

モバイルデバイスは、デバイスのいくつかの入力が使用可能でないときまたはスリープ状態にあるとき、バックグラウンドモードにあり得る。言い換えれば、デバイスは、ユーザがデバイスを使用していないとき、バックグラウンドモードにあり得る。たとえば、（マイクロフォンなどの）オーディオ入力がオフであるとき、デバイスはバックグラウンドモードにあると見なされ得る。さらに、（デバイスのディスプレイなどの）ビジュアル入力がオフであるとき、デバイスはバックグラウンドモードにあると判断され得る。以下で説明するように、モバイルデバイスがバックグラウンドモードにあるか否かを判断するために追加の入力が使用され得る。

40

【0024】

ネットワークとモバイルデバイス上で実行しているアプリケーションとの間の接続性の管理について説明する。モバイルデバイス上にインストールされたいくつかのアプリケーションが、デバイスがバックグラウンドモードにあるときにネットワークへのアクセスを要求したとき、不必要な量のネットワークシグナリングが発生し得る。たとえば、第1のアプリケーションが、通信チャネルを確立するためにシステムコールを開始し得、次いで、データが送信／受信された後、チャネルは中止され得る。次いで、第2のアプリケーシ

50

ョンが、データを送信／受信するための通信チャネルを同じく確立するためにシステムコールを開始し得る。通信チャネルが確立されるたびに、ネットワークの利用可能な帯域幅が減少するように、ネットワークシグナリングの量は増加し得る。さらに、いくつかのアプリケーションが、デバイスがバックグラウンドモードにある間にネットワークへのアクセスを要求したとき、不必要な量のバッテリー電力が消費され得る。通信チャネルが確立されるたびに、モバイルデバイスがアクティブモードに入ったとき、利用可能な電力がより少なくなるように、バッテリー電力は減少し得る。したがって、本システムおよび方法は、ネットワークシグナリングを低減し、バッテリー電力を温存するために、ネットワークアクセスの要求を保留し、アグリゲートし得る。前述のように、これは、デバイスがアクティブでないときに行われ得る。さらに、システムコールの保留およびアグリゲーションは、デバイスのバッテリー電力があるしきい値量を下回るときに行われ得る。トリガリングイベントが発生した（たとえば、デバイスがアクティブモードに入った）とき、アグリゲートされた要求は、ネットワークシグナリングの量を低減するため、ならびに各別個の要求に関連するバッテリー電力の消費を低減するために一緒にリリースされ得る。

【0025】

要求の保留およびアグリゲーションは、ユーザによるデバイスの使用に干渉しないように、モバイルデバイスが非アクティブモードにあるときに実行され得る。一例では、ユーザデバイス上のアプリケーションからのネットワークアクセスの要求がインターセプトされ得る。たとえば、要求をインターセプトするために、モバイルデバイスのアプリケーションレイヤとデバイスのオペレーティングシステムレイヤとの間にラッパーが配置され得る。一例では、ラッパーは、要求をインターセプトするソフトウェアエンティティであり得る。ラッパーは、アプリケーションレイヤ中のアプリケーションならびにオペレーティングシステムレイヤ中のオペレーティングシステムに対して透過的(transparent)であり得る。要求をインターセプトすると、要求は、オペレーティングシステムに到着することを保留または遅延され得る。一構成では、要求は、アプリケーションレイヤ中の追加のアプリケーションから受信された他のインターセプトされた要求とアグリゲートされ得る。トリガリングイベントが検出されたとき、アグリゲートされた要求はオペレーティングシステムに対してリリースされ得る。したがって、ラッパーは、要求を透過的にインターセプトし得、アグリゲートし得、次いで、追加の処理が完了したとき、アグリゲートされた要求を中継し得る。

【0026】

さらに、保留された要求がデバイスのオペレーティングシステムに対してどのくらいの頻度でリリースされるべきかを示す間隔が判断され得る。間隔は、以下で説明するミドルボックスの状態を維持するように判断され得る。一例では、インターネットプロトコル(IP)ホストはステートフルミドルボックスによって分離され得る。ステートフルミドルボックスは、ファイアウォールおよびネットワークアドレス変換(NAT)機能を実行し得る。ファイアウォールの機能は、デバイスのどのインバウンド／アウトバウンドポートが開いているかまたは利用可能であるかを判断することであり得る。NAT機能は、セルラーネットワーク上で常に展開されるとは限らないが、LAN/WLAN上で連続的に展開され得る。モバイルデバイス上で実行しているアプリケーションは、セルラーネットワークとWi-Fiネットワークとを区別しないことがあり、したがって、アプリケーションは、セルラーネットワーク上でNAT機能を動作可能にしておくための「キープアライブ」要求を発信するためにタイマーを使用し得る。ミドルボックスの状態は、タイマーが満了するまで維持され得る。長寿命(long-lived)接続(TCPまたはUDP)が必要とされる場合、ミドルボックスは、それらの状態を接続全体にわたって保ち得る。モバイルデバイス(たとえば、スマートフォン)上で実行しているアプリケーションは、(Wi-Fiネットワークとは対照的に)セルラーネットワークには適合されないことがある。したがって、これらのアプリケーションは、間隔がセルラーネットワークのためのシグナリングのピークを生じるかどうかにかかわらず、どこでも働くキープアライブ／再接続間隔を選択し得る。したがって、以下で、デバイスがバックグラウンドモードにある間、ネッ

10

20

30

40

50

トワークアクセスの要求を保留することと、あるトリガリングイベントが発生したときに、または特定のネットワークによって判断された間隔において要求をリリースすることとによって、ネットワークアクセスのためのシステムコールの数を低減して、電力を温存し、シグナリングを低減するためのシステムおよび方法について説明する。

【0027】

以下の説明は、例を与えるものであり、特許請求の範囲において記載される範囲、適用性、または構成を限定するものではない。本開示の趣旨および範囲から逸脱することなく、説明する要素の機能および構成において変更が行われ得る。様々な実施形態は、適宜に様々なプロシージャまたはコンポーネントを省略し得、置換し得、または追加し得る。たとえば、説明する方法は、説明する順序とは異なる順序で実行され得、様々なステップが追加され得、省略され得、または組み合わせられ得る。また、いくつかの実施形態に関して説明する特徴は、他の実施形態において組み合わせられ得る。

10

【0028】

次に図1を参照すると、ブロック図にワイヤレスネットワーク環境100の例が示されている。ネットワーク環境100は、モバイルデバイス102と通信ネットワーク115とを含み得る。デバイス102は、いくつかの無線チャネル110-aを使用してネットワーク115と通信し得る。たとえば、制御チャネル110-a-1は、デバイス105とネットワーク115との間に確立され得る。さらに、他のタイプのチャネル110-a-2~110-a-nも確立され得る。これらの他のタイプのチャネルは、データチャネル、ボイスチャネルなどを含み得る。

20

【0029】

動作中、デバイス102は、いくつかの無線のいずれかを使用してネットワーク115とインターフェースし得るアプリケーションを実行し得る。たとえば、実行アプリケーションは、ネットワーク115との通信を確立するための要求を発行し得る。一例では、要求は、ソケットレイヤコールなどのネットワーキングシステムコールであり得る。要求は、デバイス105上のオペレーティングシステムのソケットレイヤに宛てられ(be intended)得る。従来のデバイスは、一般に、これらのタイプの要求が、処理されるためにオペレーティングシステムに直接進むことを可能にする。要求を受信すると、従来のデバイスは、データ接続セットアッププロシージャを通して制御チャネル110-a-1を確立するためにネットワークシグナリングプロセスを開始する。データ接続セットアッププロシージャがモバイルデバイス105上で実行されるとき、バッテリー電力が消費され、ネットワークにわたるシグナリングのレベルが増加し得る。これは、モバイルデバイス105およびネットワーク115の効率を低下させ得る。

30

【0030】

一構成では、デバイス105は、オペレーティングシステムに対する要求のリリースを遅延させるためのアーキテクチャを含み得る。このアーキテクチャは、アプリケーションからのネットワークアクセスの要求をインターセプトし得る。要求をインターセプトすると、アーキテクチャは、要求がオペレーティングシステムのTCP/IPスタックに到着することを保留するかまたは遅延させ得る。TCP/IPスタックは、オペレーティングシステムに組み込まれ得る通信プロトコルを含み得、オペレーティングシステムに、ネットワークを介してデータを送信するための規格を与え得る。インターセプトされた要求は、追加のアプリケーションから受信されたネットワークアクセスの他のインターセプトされた要求とアグリゲートされ得る。アグリゲートされた要求は、一緒にバンドルされ得、ネットワークアクセスの単一の要求としてリリースされ得る。別の例では、アグリゲートされた要求は、特定のイベント（たとえば、モバイルデバイスがアクティブになる）の発生時にリリースされ得る。一構成では、インターセプトすることと、保留することと、アグリゲーションとに關係する上記で説明したアーキテクチャは、デバイス105が非アクティブモードにあるときに使用され得る。

40

【0031】

図2に、図1のモバイルデバイス105の一例であり得るモバイルデバイス105-a

50

のアーキテクチャ 200 の一例を示す。デバイス 105-a のアーキテクチャ 200 は接続性エンジン 225 を含み得る。接続性エンジン 225 は、デバイス 105-a 上のアプリケーションレイヤ 220 中で実行しているアプリケーションが、図 1 のネットワーク 115 などのネットワークにいつアクセスし得るかを管理し得る。アプリケーションレイヤ 220 は、様々な機能を与えることと、無線ユニット 245 の 1 つまたは複数の無線 250-a を使用して、ネットワーク 115 などの外部ネットワークと通信することとを実行し得るアプリケーションを含み得る。

【0032】

一構成では、接続性エンジン 225 はラッパー 230 を実行し得る。一例では、ラッパー 230 は、アプリケーションレイヤ 220 中のアプリケーションから発生したネットワークアクセスのためのシステムコールをインターセプトし得る。ラッパー 230 は、デバイス 105-a 上で実行しているオペレーティングシステム 235 に要求が到着することを保留し得る。ラッパー 230 はまた、インターセプトされたシステムコールを追加のアプリケーションからインターセプトされた他のシステムコールとアグリゲートし得る。ラッパー 230 は、アグリゲートされたシステムコールがオペレーティングシステム 235 のソケットレイヤ 240 に到着することを保留し得る。ネットワークアクセスのためのシステムコールがソケットレイヤ 240 に到着したとき、無線 250-a のうちの 1 つまたは複数を使用して通信チャネルを確立するためのプロセスが開始され得る。ソケットレイヤ 240 は、要求を処理し得、要求を開始したアプリケーションとネットワーク 115 との間の接続を確立するために接続セットアッププロシーダを開始するように特定の無線に通知し得る。たとえば、ソケットレイヤ 240 は、特定のアプリケーションと無線、たとえば無線 1250-a-1 との間のバインディングを確立するためにコール（または要求）を発行し得る。無線 1250-a-1 は、図 1 の制御チャネル 110-a-1 の一例であり得る制御チャネルを確立することによって、接続セットアッププロシーダを開始するための信号をネットワーク 115 に送信することを開始し得る。

【0033】

アプリケーションがネットワークアクセスのためのシステムコールを与えるたびにセットアッププロシーダを開始するのではなく、アグリゲートされた要求がソケットレイヤ 240 に対して一緒にリリースされたときに、要求を送ったアプリケーションと特定の無線 250-a との間の接続を確立するためにソケットレイヤ関数が 1 回開始され得る。選択された無線は、次いで、ネットワーク 115 と要求を発信したアプリケーションとのデータ接続を確立するために、ネットワークシグナリングを開始し得る。

【0034】

したがって、デバイスアーキテクチャ 200 は、デバイス 105-a 上で実行しているアプリケーションによる、ネットワークにアクセスするためのシステムコールのアグリゲーションを提供する。アグリゲーションは、ソケットレイヤ 235 に対していくつかのシステムコールをバンドルとしてリリースすることによって、バッテリー消費量とネットワークシグナリングとを低減するのに役立ち得る。

【0035】

図 3 に、ネットワークアクセスの要求の保留とアグリゲーションとを実装するモバイルデバイス 105-b のブロック図 300 を示す。デバイス 105-b は、図 1 または図 2 のデバイス 105 の一例であり得る。デバイス 105-b は、通信バス 314 を使用して通信するようにすべて結合されたプロセッサ 360、メモリ 355、アプリケーションレイヤ 220、ラッパー 230、接続性エンジン 225、オペレーティングシステム 235、および無線ユニット 245 を含み得る。メモリ 355 は、アプリケーションレイヤ 220 と、ラッパー 230 と、オペレーティングシステム 235 とを記憶し得る。プロセッサ 360 は接続性エンジン 225 を含み得る。接続性エンジン 225 は、汎用プロセッサ、デジタル信号プロセッサ (DSP)、特定用途向け集積回路 (ASIC)、フィールドプログラマブルゲートアレイ (FPGA) または他のプログラマブル論理デバイス、個別ゲートまたはトランジスタ論理、個別ハードウェアコンポーネント、あるいは本明細書で説

10

20

30

40

50

明する機能を実行するように設計されたそれらの任意の組合せとして実装され得る。接続性エンジン 225 は、モバイルデバイス上のアプリケーションからの要求をインターセプトするための手段と、要求がモバイルデバイス上のオペレーティングシステムに到着することを保留するための手段と、トリガリングイベントを検出すると、オペレーティングシステムに対して要求をリリースするための手段とを含み得る。さらに、接続性エンジン 225 は、要求を、モバイルデバイス 105 のための通信を行うための他のインターセプトされた要求とアグリゲートするための手段を含み得る。接続性エンジン 225 はまた、図 2、図 3、または図 4 のラッパー 230 を実行するための手段を含み得る。実行されたラッパーは、アプリケーションからの要求をインターセプトし得る。さらに、エンジン 225 は、アプリケーションを、要求が保留される種類のアプリケーションとして識別するための手段を含み得る。さらに、接続性エンジン 225 は、アプリケーションをクリティカルアプリケーションまたは非クリティカルアプリケーションとして識別するための手段と、非クリティカルアプリケーションからの要求のみを保留するための手段とを含み得る。デバイス 105-b は一実装形態にすぎず、他の実装形態が可能であることに留意されたい。

【0036】

一態様では、プロセッサ 360 は、中央処理ユニット（CPU）、プロセッサ、ゲートアレイ、ハードウェア論理、メモリ要素、および／またはソフトウェアを実行するハードウェアのうちの少なくとも 1 つを含む。プロセッサ 360 は、アプリケーションレイヤ 220 において実行しているアプリケーションによって開始されたネットワークアクセスのためのシステムコールが、オペレーティングシステム 235 に到着することを保留され得、他のシステムコールとアグリゲートされ得るように、デバイス 105-b の動作を制御するように動作する。一実装形態では、プロセッサ 360 は、いくつかの機能のいずれかを実行することに関係するコンピュータ可読命令を実行し得る。たとえば、プロセッサ 360 は、ネットワークアクセスの要求のインターセプションおよびアグリゲーションを実現するために、デバイス 105-b から受信または通信された情報を分析するように動作し得る。別の態様では、プロセッサ 360 は、いくつかのアプリケーションからのネットワークアクセスのためのシステムコールのアグリゲーションを実現するために、メモリ 355、無線ユニット 245、アプリケーションレイヤ 220、ラッパー 230、オペレーティングシステム 235、および／または接続性エンジン 225 によって利用され得る情報を生成するように動作し得る。

【0037】

無線ユニット 245 は、デバイス 105-b を、いくつかのチャネル 110-a を使用する外部通信ネットワークなど、いくつかの外部エンティティにインターフェースするために使用され得るいくつかの無線／インターフェースを与え得る、ハードウェアおよび／またはソフトウェアを実行するプロセッサを含み得る。たとえば、無線ユニット 245 は、セルラー、Wi-Fi、Bluetooth、またはチャネル 110-a を使用して通信ネットワークと通信するための他の技術を使用して通信するための無線／インターフェースを与え得る。

【0038】

アプリケーションレイヤ 220 は、デバイス 105-b 上の 1 つまたは複数のアプリケーションを記憶および／または実行し得る、ハードウェアおよび／またはソフトウェアを実行するプロセッサを含み得る。一実装形態では、アプリケーションレイヤ 220 は、アプリケーションが、外部ネットワークまたはシステムと通信する目的で無線／インターフェースへの接続を要求するなど、ネットワークングサービスを要求するためにネットワークング関数呼出しを開始することを可能にし得る。

【0039】

オペレーティングシステム 235 はソケットレイヤを含み得る。ソケットレイヤは、ソケットレイヤ関数を実行し得る、ハードウェアおよび／またはソフトウェアを実行するプロセッサを含み得る。一実装形態では、ソケットレイヤ関数は、Connect()、B

10

20

30

40

50

ind ()、および Setsockopt () などの関数を含み得る。Connect () 関数は、アプリケーションと特定の無線／インターフェースとの間の接続を確立するように動作する。たとえば、特定の無線／インターフェースは、無線ユニット 245 によって与えられたいくつかの候補無線から選択され得る。一態様では、ソケットレイヤは、様々なソケットレイヤ関数またはコマンドを実行し得る。

【0040】

接続性エンジン 225 は、モバイルデバイス 105-b 上で実行しているアプリケーションからのネットワークアクセスの要求をラッパーにインターセプトさせるためにラッパー 230 を実行し得る、ハードウェアおよび／またはソフトウェアを実行するプロセッサを含み得る。ラッパー 230 はまた、インターセプトされた要求がオペレーティングシステム 235 に到着することを保留し得る。さらに、ラッパー 230 は、インターセプトされた要求を他のインターセプトされた要求とアグリゲートし得る。アグリゲートされた要求は、トリガリングイベントが発生した（たとえば、モバイルデバイス 105-b がアクティブモードに入った）ときにオペレーティングシステムのソケットレイヤに対してリリースされ得る。

10

【0041】

接続性エンジン 225 は、ラッパー 230 に、様々な方法でネットワークアクセスの要求をキャプチャさせ得、保留させ得、アグリゲートさせ得る。たとえば、デバイス 105-b がバックグラウンドモードにあるとき、（ラッパー 230 を介して）接続性エンジン 225 は、デバイス 105-a 上で実行しているいくつかのアプリケーションからのいくつかの要求をインターセプトし得る。インターセプトされた要求は、一緒にアグリゲートされ得、あるトリガリングイベントが発生するまで保留され得る。たとえば、要求は、モバイルデバイス 105-b がアクティブ状態に入ったときにリリースされ得る。一構成では、現在保留されているアグリゲートされた要求は一緒にリリースされ得る。たとえば、アグリゲートされた要求は、バンドルされ得、単一のシステムコールとしてソケットレイヤに対して一緒にリリースされ得る。ソケットレイヤは、ネットワーク 115 とのデータ接続のために通信チャネルを確立するためのプロシージャを開始し得る。

20

【0042】

メモリ 355 は、RAM、ROM、EEPROM、または情報がデバイス 105-b において記憶され、取り出されることを可能にするように動作する他のタイプのメモリデバイスを含み得る。一実装形態では、メモリ 355 は、プロセッサ 360 によって実行されるコンピュータ可読命令を記憶し得る。メモリ 355 はまた、プロセッサ 360、無線ユニット 245、アプリケーションレイヤ 220、ラッパー 230、オペレーティングシステム 235、および／または接続性エンジン 225 のいずれかによって生成されたデータを含む、いくつかの他のタイプのデータのいずれかを記憶し得る。メモリ 355 は、ランダムアクセスメモリ、バッテリーバックの (battery-backed) メモリ、ハードディスク、磁気テープなどを含む、いくつかの異なる構成を含み得る。また、圧縮および自動バックアップなど、様々な特徴がメモリ 355 上に実装され得る。

30

【0043】

様々な実装形態では、デバイス 105-b は、非一時的コンピュータ可読媒体上に記憶されるかまたは非一時的コンピュータ可読媒体上で実施される 1 つまたは複数のプログラム命令（「命令」）または「コード」のセットを有するコンピュータプログラム製品を含み得る。少なくとも 1 つのプロセッサ、たとえば、プロセッサ 360 および／または接続性エンジン 225 によってコードが実行されたとき、それらの実行は、プロセッサ 360 および／または接続性エンジン 225 に、本明細書で説明するアグリゲーションアーキテクチャの機能を与えるようにデバイス 105-b を制御させ得る。たとえば、非一時的コンピュータ可読媒体は、フロッピー（登録商標）ディスク、CDROM、メモリカード、フラッシュメモリデバイス、RAM、ROM、あるいはデバイス 105-b とインターフェースする他のタイプのメモリデバイスまたはコンピュータ可読媒体であり得る。別の態様では、コードのセットは、外部デバイスまたは通信ネットワークリソースからデバイス

40

50

105-bにダウンロードされ得る。コードのセットは、実行されたとき、本明細書で説明するシステムコールアグリゲーションアーキテクチャの態様を与えるように動作する。

【0044】

図4に、上記で説明したように、ネットワークアクセスの要求をインターセプトし、アグリゲートするために有用なモバイルデバイス105-c上のアーキテクチャの例示的なブロック図400を示す。モバイルデバイス105-cは、図1、図2、または図3のモバイルデバイス105の一例であり得る。

【0045】

図示のように、ブロックは、アプリケーションプロセッサ490とモデムプロセッサ495との間で分割されるが、様々な機能は、図4の例とは別様に編成され得る。アプリケーションレイヤ220は、アプリケーション接続エンジン(A P P C n E)475およびソケットレイヤ240と対話し得る。アプリケーション接続エンジン475はモデム接続エンジン(モデムC n E)485と通信し得る。モデム接続エンジンは、無線ユニット245およびその中のいくつかの無線250-aなど、通信リソースを管理し得る。ラッパー230は、アプリケーションプロセッサ490においてアプリケーションレイヤ220とオペレーティングシステムのソケットレイヤ240との間で実行され得る。ラッパー230は、アプリケーションレイヤ220とソケットレイヤ240との間で受け渡されるデータをキャプチャし得る。たとえば、ラッパー230は、アプリケーションレイヤ220から送られ、ソケットレイヤ240に宛てられたシステムコールをインターセプトするために、アプリケーション220とソケットレイヤ240との間に配置され得る。一構成では、ラッパー230は、デバイス105-cによる非アクティビティ期間中にアプリケーションレイヤ220からのシステムコールをインターセプトし得、ラッパーは、ソケットレイヤ240に対してシステムコールをリリースする前に、トリガリングイベントが発生するまでインターセプトされたコールを保留し得る。システムコールは、無線ユニット245内の無線250を使用して通信チャネルを確立するための要求であり得る。

【0046】

別の例では、ラッパー230は、デバイス105-cによる非アクティビティ期間中にアプリケーションレイヤ220からインターセプトされたシステムコールをアグリゲートし得る。ラッパー230は、動作/送信のためにソケットレイヤ240および最終的に無線ユニット245に対してアグリゲートされたシステムコールをリリースする前に、特定のイベントが発生するまで、インターセプトされ、アグリゲートされたコールを保留し得る。

【0047】

一構成では、ラッパー230は、アプリケーションが、それらの要求がソケットレイヤ240に到着することを保留されていることに気づかないように、アプリケーションレイヤ220にあるアプリケーションにとって不可視であり得る。ラッパー230は、別個のソフトウェアコンポーネントであり得るか、あるいは接続性エンジン225またはアプリケーション接続エンジン475などの別のコンポーネントに組み込まれ得る。

【0048】

図5に、第1のアプリケーションおよび第2のアプリケーションなど、いくつかのアプリケーションのためのタイミング図500を示す。アプリケーションは、モバイルデバイス105のアプリケーションレイヤ220中に設置され得る。タイミング図500は、図2または図3の接続性エンジン225の実装の結果であり得る。一構成では、第1の要求505-a-1が、時間 t_0 において第1のアプリケーションから送られ得る。第1の要求505-a-1はC o n n e c t () システムコールであり得る。第1の要求505-a-1は、モバイルデバイス上で実行しているオペレーティングシステム235に到着することを保留され得る。たとえば、第1の要求は、オペレーティングシステムのTCP/IPスタックに到着することを保留され得る。要求が保留される時間は H_0 として表され得る。第1の要求505-a-1は、時間 t_2 においてオペレーティングシステム235に対してリリースされ得る。

10

20

30

40

50

【0049】

一例では、第2の要求505-a-2が、時間 t_1 において第2のアプリケーションから送られ得る。時間 t_1 は時間 t_0 の後であり得る。第2の要求505-a-2もConnect()システムコールであり得る。第2の要求505-a-2は、 H_1 として表されるある時間期間の間、オペレーティングシステム235に到着することを保留され得る。たとえば、第2の要求も、オペレーティングシステムのTCP/IPスタックに到着することを保留され得る。第2の要求505-a-2は、時間 t_2 においてリリースされ得る。したがって、第1の要求505-a-1および第2の要求505-a-2は、一緒にまたは同時（すなわち時間 t_2 ）にリリースされ得る。時間期間 H_1 は、時間期間 H_0 よりも少なくなり得る。言い換えれば、第2の要求505-a-2は、第1の要求505-a-1よりも少ない時間の間保留され得る。

10

【0050】

一構成では、時間期間 H_0 中に、第2の要求505-a-2が時間 t_1 においてインターセプトされたとき、第1の要求は第2の要求505-a-2とアグリゲートされ得る。要求のアグリゲーションは、両方の要求が実質的に同じ時間（すなわち、時間 t_2 ）にリリースされることを可能にする。したがって、タイミング図500は、異なる時間（ t_0 および t_1 ）に送られた要求が、異なる時間期間（ H_0 および H_1 ）の間保留され得、次いで、同時（ t_2 ）にリリースされ得ることを示す。

【0051】

図6に、モバイルデバイス105-d上で実装され得るアグリゲーションアーキテクチャの一例を示す。モバイルデバイス105-dは、図1、図2、図3、および図4のモバイルデバイス105の一例であり得る。図示のように、デバイス105-dは、アプリケーションレイヤ220-aと、ラッパー230-bと、オペレーティングシステム235とを含み得る。オペレーティングシステム235はソケットレイヤ240を含み得る。図2または図3の接続性エンジン225は、ラッパー230-bソフトウェアを実行するようにとの命令を実行し得る。一構成では、デバイス105-dはバックグラウンドモードにあり得る。モバイルデバイス105-dは、たとえば、デバイス105-dのスクリーンまたはディスプレイがオフであるとき、デバイス105-dのマイクロフォン、スピーカー、または他のオーディオ出力がオフであるとき、デバイス105-d中の全地球測位システム（GPS）機構が、デバイス105-dは静止していると判断したとき、デバイスのバッテリーレベルがあるしきい値レベルを下回るときなど、バックグラウンドモードにあると見なされ得る。

20

30

【0052】

一例では、アプリケーションレイヤ220-aにおいて実行しているいくつかのアプリケーション605-aは、Connect()システムコールなど、ネットワークアクセスのためのシステムコール505-aを送り得る。システムコール505-aは、異なる時間に各アプリケーションから送られ得る。ラッパー230-bは、これらの要求をキャプチャし得、オペレーティングシステム235に到着することを保留し得る。特に、コールは、オペレーティングシステム235のソケットレイヤ240に到着することを保留され得る。一構成では、アグリゲーションモジュール610は、インターセプトされたシステムコール505-aをアグリゲートし得る。アグリゲートされた要求は、ラッパー230-a-1からオペレーティングシステム235のソケットレイヤ240に対して一緒に（または実質的に同じ時間に）リリースされ得る。アグリゲートされた要求を受信すると、ソケットレイヤ240は、続いてネットワーク115との接続を確立し得る。プロシージャは、制御チャネル110-a-1を確立するために、モバイルデバイス105-dとネットワーク115との間でシグナリングメッセージを送信することを含み得る。

40

【0053】

したがって、ネットワークアクセスの要求のインターセプション、保留、およびアグリゲーションは、複数のシステムコールが、異なる時間にソケットレイヤ240において実行されないの、モバイルデバイス105-dの電力消費量を低減し得る。代わりに、複

50

数の要求と一緒にバンドルされ、ほぼ同時にソケットレイヤ240に対してリリースされる。さらに、要求のアグリゲーションは、ネットワーク115との接続セットアッププロシージャの頻度を低減し、したがって、ネットワークトラフィックを低減し得る。

【0054】

アプリケーション605-aからの要求の保留およびアグリゲーションは、選択的に行われ得る（すなわち、ユーザが中断され得ないように実装され得る）。通信チャネルを確立するためのアプリケーション605-aからの要求をいつ保留し、アグリゲートすべきかを判断するために、様々なファクタが採用され得る。たとえば、要求をインターセプトするという判断は、スクリーンがオフである、またはオーディオ出力がオフであるなど、モバイルデバイス105-dのいくつかの特性に基づいて行われ得る。要求を保留することは、無線がロードされないとき、さもなければモバイルデバイス105-dが使用中でない（通話、オーディオストリーミングなどが無い）ときに、そのような遅延を処理することが可能であるとわかっているアプリケーションのみに対して実装され得る。アプリケーションからのシステムコールのインターセプションおよび保留は、ユーザが、アプリケーションによって与えられるサービスに登録する(subscribes)かどうかに基づいて実装され得る。ユーザがサービスに登録する場合、アプリケーションからの要求は、オペレーティングシステムに到着することを保留されないことがある。代わりに、このサブスクリプションベースのアプリケーションからの要求は、ソケットレイヤに直ちに受け渡すことが可能にされ得る。一例では、モバイルデバイス105-d上で実行しているアプリケーションは分類され得る。たとえば、第1のアプリケーション605-a-1は非クリティカルアプリケーションとして分類され得、第2のアプリケーション605-a-2はクリティカルアプリケーションとして分類され得る。非クリティカルアプリケーションは、ある遅延許容差をもつアプリケーションであり得る。言い換えれば、通信チャネルを確立するための非クリティカルアプリケーションからのシステムコールは遅延され得る。しかしながら、クリティカルアプリケーションは、遅延許容差がほとんどまたはまったくないアプリケーションであり得る。クリティカルアプリケーションの例には、限定はしないが、子供追跡アプリケーション、緊急ベースのアプリケーション、サブスクリプションベースのアプリケーションなどがあり得る。一構成では、要求の保留およびアグリゲーションは、非クリティカルアプリケーションから発生した要求に対して行われ得る。クリティカルアプリケーションから送られた要求は、保留（またはアグリゲート）されないことがあり、代わりに、オペレーティングシステムのソケットレイヤに直接進み得る。保留およびアグリゲーションはまた、上記のファクタまたは他のファクタの組合せを利用して実装され得る。

【0055】

さらに、アグリゲートされた要求をいつリリースし、アプリケーション接続性を許可すべきかを判断するために、様々なファクタが採用され得る。たとえば、データ接続セットアッププロシージャを確立するためのトリガ（遅延され得ない緊急アプリケーションなどのクリティカルアプリケーションからのシステムコールを受信することなど）がある場合、通信チャネルが緊急アプリケーションに関連して確立され得、バックグラウンド状態と接続状態との間の遷移の数を低減し得るように、保留された要求がソケットレイヤ240に対してリリースされ得る。別の例は、より望ましい無線（たとえば、ワイドローカルエリアネットワーク(WLAN)無線）がアクティブ化されるかまたはデフォルトとして選択されたとき、アグリゲートされた要求がリリースされ得ることである。要求はまた、無線チャネルが極めて良好である場合（たとえば、高い信号強度、SNR、または他の望ましいパフォーマンスメトリック）、リリースされ得る。要求はまた、あらかじめ決定されたように、またはモバイルデバイス105-dによって選択的に判断されたように、周期的にリリースされ得る。知られずに(incognito)動作するために要求をリリースするための別のヒューリスティック(heuristic)は、（ユーザがスクリーンをオンにする前に）ユーザがデバイスに接近したときであり得る。この例では、加速度計が、ユーザが電話をつかんだことを検出し得るか、またはユーザ近接度センサーが、ユーザが接近していること

を示し得る。別の態様では、バッテリーで稼動している間、要求は、スクリーンがアンロックされるとすぐに（たとえば、PINが正しく入力された後に）リリースされ得る。この態様では、ランダムなボタンが押された（デバイス105-dが財布またはポケットの中にある）ときには要求のリリースは行われ得ない。

【0056】

一例では、保留された要求をリリースさせるトリガリングイベントは、タイマーの満了であり得る。イベントはまた、ディスプレイのステータス変更であり得る。たとえば、ディスプレイは、「オフ」状態から「オン」状態に変化し得る。マイクロフォンのステータス変更（オフからオン）もトリガイイベントになり得る。さらに、GPSセンサーのステータス変更がトリガリングイベントになり得る。たとえば、センサーがモバイルデバイスの移動を検出したとき、センサーは状態を変更し得る。要求をリリースするための追加のトリガリングイベントは、ユニバーサルシリアルバスポートが使用中であるという指示、またはオーディオ機器がデバイスに接続されているという指示を含み得る。さらに、ビデオ機器がモバイルデバイスに接続されているという指示も、モバイルデバイスのオペレーティングシステムに対して保留された要求をリリースするためのトリガリングイベントとして働き得る。さらに、あるネットワークへの接続が利用可能であるという指示が、要求のリリースをトリガし得る。たとえば、Wi-Fiタイプのネットワークへの接続の指示が、要求をリリースさせ得る。同様に、セルラータイプのネットワークへの無線接続がすでに開いているという指示も、デバイスのオペレーティングシステムに対する要求のリリースをトリガし得る。さらに別の態様では、要求は、上記または他のファクタの何らかの組合せに従ってリリースされ得る。上記の説明は、APIアーキテクチャに関するものであるが、概念は、ハードウェア、ファームウェア、またはハードウェアおよびソフトウェアの任意の組合せにおいて等しく適用され得る。

【0057】

一構成では、アプリケーションはタイマーに関連付けられ得る。タイマーの満了前の時間期間は、関連するアプリケーションの許容差レベルを示し得る。たとえば、許容差を受け入れないタイマーは「ハードタイマー」と呼ばれることがある。ハードタイマーは、比較的固定された時点において満了するように意図されたタイマーであり得る。逆に、許容差値を受け入れるタイマーは「ソフトタイマー」になり得る。ソフトタイマーは、意図された満了時間において満了し得るが、指定された許容差範囲内の満了をも可能にし得る。一例として、電子メール更新サービスなどのいくつかのアプリケーションは、接続要求が明確な固定時間に行われることを必要としないことがある。したがって、そのようなアプリケーションのためのタイマーは広い許容差を与えられ得、ソフトタイマーが生成され、そのようなアプリケーションに関連付けられ得る。逆に、株式トレーダーによって使用される株式プログラムは、株価情報の精度を保証するために、固定時間における一貫した更新を必要とし得る。そのようなアプリケーションは、許容差をほとんどまたはまったく受け入れないことがあり、したがってハードタイマーに関連付けられるであろう。

【0058】

一例では、ソフトタイマーまたはハードタイマーのうちの1つなどのタイマーが満了しようとしていることがある。満了時間が識別されると、たとえば、図2または図3の接続性エンジン225は、満了時間が様々なアプリケーションのいずれかのソフトタイマーの許容差内に入るかどうかを判断し得る。許容差が満了時間内に入るすべてのソフトタイマーに対して、接続性エンジンは、それらのタイマーを早期に満了させるようにとの命令を実行し得る。タイマーが満了したアプリケーションからのインターセプトされたネットワークアクセス要求は、モバイルデバイスのオペレーティングシステムに対してリリースされ得る。いくつかの構成では、タイマーが満了したアプリケーションの各々が、それらの必要な通信アクティビティを完了するまで、ネットワーク接続は開いたままになり得る。タイマーが満了したときに通信要求が行われるので、アプリケーションによる通信システムの共用の結果として、リソース管理がより効率的になる。

【0059】

上述のように、要求がどのくらいの頻度でリリースされるべきかを示す間隔またはリフレッシュレートが判断され得る。間隔は、ファイアウォールおよびNAT機能を実行するミドルボックスの状態を維持するように判断され得る。ミドルボックスの状態は、アプリケーションがキープアライブメッセージまたは一連のより短い接続要求を発行することによって維持され得る。一構成では、ネットワークは、モバイルデバイス上のミドルボックスの状態を維持するための最小リフレッシュレートに関する情報を与え得る。リフレッシュレートは、UDP対TCP接続について与えられ得る。ネットワークは、キープアライブ／再接続メッセージの量に起因して発生した無線負荷に基づいて、ミドルボックスにおけるリフレッシュレートを適応させ得る。たとえば、キープアライブメッセージの数があるしきい値を上回る場合、ミドルボックスにおける状態は、より長い時間期間の間維持され得、モバイルデバイスのためのリフレッシュレートはスローダウンされ得る。間隔（またはリフレッシュレート）は、ステートフルミドルボックスにおけるタイムアウト値よりも小さくなり得る。したがって、モバイルデバイスは、ネットワークによって示されたリフレッシュレート（または間隔）に従って、アップリンクに対してゲートを開き得る（すなわち、要求をリリースし得る）。ゲートは、デバイスがバックグラウンドモードにないときには開かれ得、モバイルデバイスがバックグラウンドモードにあるときには閉じられ得る。

10

【0060】

図7は、ネットワークアクセスの要求を保留するための方法700の一例を示すフローチャートである。明快のために、方法700について、図1、図2、図3、または図4に示したモバイルデバイス105に関して以下で説明する。一実装形態では、プロセッサ360は、以下で説明する機能を実行するためにデバイス105の機能要素を制御するためのコードの1つまたは複数のセットを実行し得る。一構成では、以下で説明する方法700は、デバイス105がバックグラウンドモードにあるときに実装され得る。

20

【0061】

ブロック705において、モバイルデバイス105上のアプリケーションからの要求をインターセプトする。要求は、モバイルデバイス105のための通信チャネルを確立するなど、モバイルデバイスのための通信を行うための要求であり得る。要求は、モバイルデバイス105のアプリケーションレイヤ220において実行しているアプリケーションから送られ得る。一例では、要求は、アプリケーションが、ネットワーク115などの外部ネットワークとインターフェースすることを可能にするために、データ接続セットアッププロシージャを開始するための要求であり得る。たとえば、要求は、モバイルデバイス105上のオペレーティングシステム235のソケットレイヤ240に対するシステムコールであり得る。

30

【0062】

ブロック710において、モバイルデバイス105上で実行しているオペレーティングシステム235に要求が到着することを保留され得る。たとえば、要求は、オペレーティングシステム235のソケットレイヤ240に到着することを保留され得る。一構成では、ラッパー230が要求をインターセプトし得、保留し得る。

【0063】

ブロック715において、トリガリングイベントを検出すると、オペレーティングシステムに対して要求がリリースされ得る。たとえば、デバイス105は、上記で説明したようにアクティブモードに入り得る。

40

【0064】

したがって、方法700は、モバイルデバイス105のアプリケーションレイヤ220において実行しているアプリケーションによってサブミットされたネットワークアクセスの要求のインターセプションおよび保留を提供し得る。方法700は一実装形態にすぎないこと、および方法700の動作は、他の実装形態が可能であるように、並べ替えられるかまたは場合によっては変更され得ることに留意されたい。

【0065】

50

図8は、モバイルデバイス上で実行している非クリティカルアプリケーションからのネットワークアクセスの要求をインターセプトするための方法800の一例を示すフローチャートである。明快のために、方法800について、図1、図2、図3、または図4に示したデバイス105に関して以下で説明する。一実装形態では、プロセッサ360は、以下で説明する機能を実行するためにデバイス105の機能要素を制御するためのコードの1つまたは複数のセットを実行し得る。

【0066】

ブロック805において、ネットワークアクセスの要求がインターセプトされる。要求は、モバイルデバイス105のアプリケーションレイヤ220において実行しているアプリケーションから送られ得る。一例では、要求は、ネットワーク115などの外部ネットワークとの通信チャネルを確立するための要求であり得る。要求は、デバイス105上のオペレーティングシステム235のソケットレイヤ240に対するシステムコールであり得る。要求を受信すると、ソケットレイヤ240は、通信チャネルを確立するためのプロシージャを開始し得、チャネルが確立されたとき、アプリケーションにコールバック関数を与え得る。

10

【0067】

ブロック810において、デバイス105がバックグラウンドモードにあるかどうかに関する判断が行われ得る。たとえば、デバイス105が電源切断されているかどうか、スリープモードにあるかどうかなどに関する判断が行われ得る。デバイス105はまた、たとえば、デバイス105のディスプレイが非アクティブである場合、オーディオ出力が非アクティブである場合など、バックグラウンドモードにあると判断され得る。デバイス105-aが非アクティブであると判断された場合、ブロック815において、システムコールを開始したアプリケーションが非クリティカルアプリケーションであるかどうかを判断するための第2の判断が行われ得る。クリティカルアプリケーションは、ネットワークアクセスのための優先権をもつ緊急アプリケーション、サブスクリプションベースのアプリケーション、遅延についての低い許容差をもつアプリケーションなどであり得る。

20

【0068】

デバイス105がアクティブであること、またはアプリケーションがクリティカルアプリケーションであることが判断された場合、要求は、オペレーティングシステム235のソケットレイヤ240に対してリリースされ得る。言い換えれば、要求は、ソケットレイヤが、ネットワーク115との通信チャネルを確立するためのプロシージャを開始することを可能にするためにリリースされ得る。デバイスがバックグラウンドモードにあること、およびアプリケーションが非クリティカルアプリケーションとして分類されることが判断された場合、ブロック820において、要求がオペレーティングシステムに到着することを保留し得、したがって、通信チャネルをセットアップするためのプロシージャの開始を遅延させる。

30

【0069】

ブロック825において、要求が他のインターセプトされた要求とアグリゲートされ得る。他の要求は、モバイルデバイス105上で実行している追加のアプリケーションによって開始され得る。ブロック830において、上記で説明したように、トリガリングイベントが検出されたかどうかに関する判断が行われ得る。トリガリングイベントが検出されないと判断された場合、方法800は、インターセプトされたネットワークアクセスの要求をアグリゲートし続けるために戻り得る。しかしながら、トリガリングイベントが検出されたと判断された場合、ブロック835において、オペレーティングシステム240のソケットレイヤ240に対してアグリゲートされた要求がリリースされ得る。言い換えれば、いくつかのアプリケーションからのシステムコールが保留され得、一緒にバンドルされ得、次いで、ソケットレイヤ240に対して単一のシステムコールとしてリリースされ得る。

40

【0070】

したがって、方法800は、モバイルデバイス105上で実行している非クリティカル

50

アプリケーションからのネットワークアクセスの要求をインターセプトすることと、保留することと、アグリゲートすることとを提供し得る。要求を保留し、アグリゲートすることによって、いくつかのシステムコールが、一緒にバンドルされ得、単一のシステムコールとしてリリースされ得る。これにより、通信チャネルをセットアップするためのプロシージャを開始するためのシステムコールの量が低減され得るので、モバイルデバイス105のためのバッテリー電力が節約され得、ならびにネットワークシグナリングが低減され得る。方法800は一実装形態にすぎないこと、および方法800の動作は、他の実装形態が可能であるように、並べ替えられるかまたは場合によっては変更され得ることに留意されたい。

【0071】

10

図9は、モバイルデバイス上で実行しているいくつかのアプリケーションからのネットワークアクセスの要求をインターセプトするための方法900の一構成を示すフローチャートである。明快のために、方法900について、図1、図2、図3、または図4に示したデバイス105に関して以下で説明する。一実装形態では、プロセッサ360は、以下で説明する機能を実行するためにデバイス105の機能要素を制御するためのコードの1つまたは複数のセットを実行し得る。

【0072】

ブロック905において、第1の時間において第1のアプリケーションからのネットワークアクセスの第1の要求がインターセプトされる。一例では、ブロック910において、第2の時間において第2のアプリケーションからの第2の要求がインターセプトされ得る。第2の時間は第1の時間とは異なり得る。ブロック915において、第3の時間において第3のアプリケーションからの第3の要求がインターセプトされ得る。一構成では、第3の時間は、第1の時間および第2の時間とは異なり得る。インターセプトされた要求は、ネットワークアクセスのために通信チャネルを確立するためのシステムコールであり得る。アプリケーションは、モバイルデバイス105上で実行していることがある。

20

【0073】

要求がインターセプトされたとき、ブロック920において、モバイルデバイス105がバックグラウンドモードにあるかどうかに関する判断が行われ得る。デバイス105がバックグラウンドモードにあると判断された場合、ブロック925、930、および935において、要求がオペレーティングシステムに到着することを保留し得る。デバイス105がアクティブモードにある場合、ブロック950において、モバイルデバイス105上のオペレーティングシステム235のソケットレイヤ240に対して要求がリリースされ得る。

30

【0074】

一構成では、ブロック940において、第1、第2、および第3の要求と一緒にアグリゲートされ得またはバンドルされ得る。ブロック945において、トリガリングイベントが発生したかどうかに関する判断が行われ得る。たとえば、デバイスがアクティブモードに入ったかどうか、デバイス上のディスプレイがアクティブ化されたかどうか、デバイスがロケーションを変更したかどうか、ユーザがデバイスの近くにいるかどうかなどに関する判断が行われ得る。トリガリングイベントが検出されない場合、方法900は、トリガリングイベントの検出のための監視を続けるために戻り得る。トリガリングイベントが検出された場合、ブロック950において、オペレーティングシステムのソケットレイヤに対してアグリゲートされた要求がリリースされ得る。要求を受信すると、ソケットレイヤは、ネットワーク115などの外部ネットワークとの通信チャネルを確立するためのプロシージャを開始し得る。

40

【0075】

したがって、方法900は、モバイルデバイス105上で実行している複数のアプリケーションからのネットワークアクセスの要求をインターセプトすることと、保留することと、アグリゲートすることとを提供し得る。したがって、いくつかのシステムコールが、一緒にバンドルされ得、単一のシステムコールとしてリリースされ得る。これにより、シ

50

システムコールの量が低減され得るので、モバイルデバイス105のためのバッテリー消費量が低減され、ならびにネットワークシグナリングが低減され得る。方法900は一実装形態にすぎないこと、および方法900の動作は、他の実装形態が可能であるように、並べ替えられるかまたは場合によっては変更され得ることに留意されたい。

【0076】

図10に、ネットワークアクセス要求などのアプリケーション接続要求の間の同期を改善するための、いくつかの実施形態において実装される1つの可能なプロセス1000を示す。以下の例で説明するように、いくつかの実施形態がモバイルデバイス上のプロセス1000を実装し得る。明快のために、方法1000について、図1、図2、図3、または図4に示したデバイス105に関して以下で説明する。一実装形態では、図2または図3のプロセッサ360または接続性エンジン225は、以下で説明する機能を実行するためにデバイス105の機能要素を制御するためのコードの1つまたは複数のセットを実行し得る。

10

【0077】

本方法は、ブロック1005において、ソフトタイマーまたはハードタイマーのうちの1つなどの第1のタイマーが満了しようとしている可能性があることを識別することによって開始する。これは、様々なアプリケーションのタイマーをポーリングする中央システムによって達成され得る。代替的に、各アプリケーションは、それ自体のタイマーを個々に監視し、タイマーの満了時に通知を与え得る。

【0078】

20

満了時間が識別されると、ブロック1010において、同じく満了し得る追加のタイマーが識別されるかどうかに関する判断を行う。たとえば、第1のタイマーの満了時間がソフトタイマーの許容差内に入るかどうかに関する判断が行われ得る。許容差が満了時間内に入るすべてのソフトタイマーに対して、ブロック1015において、追加のタイマーを早期に満了させ得る。ブロック1020において、満了タイマーに関連するアプリケーションのためのデータが同期され得る。たとえば、接続を形成することによって、タイマーが満了したアプリケーションにチャネルアクセスが与えられ得る。いくつかの実施形態では、次いで、タイマーが満了したアプリケーションの各々がそれらの必要な通信アクティビティを完了するまで、接続は開いたままになり得る。タイマーが満了したときに通信要求が行われるので、アプリケーションによる通信システムの共用の結果として、リソース管理がより効率的になる。方法1000は、ブロック1005において、タイマーが満了するのを再び待つことによって無期限に実行され得るか、または本方法は終了し得る。

30

【0079】

ブロック1005において、意図された時間において最初に満了した1つまたは複数のタイマーは、いくつかの実施形態では、「マスタタイマー」と呼ばれ得るか、または互換的に「トリガイイベント」と呼ばれ得る。すなわち、これらのタイマーは、他のタイマー（「ソフトタイマー」）がそれらのそれぞれの許容差に基づいていつ満了し得るかを規定する。したがって、ソフトタイマーまたはハードタイマーは、マスタタイマーとして働き得る。しかしながら、（ソフトタイマーのみが許容差を所有するので）ソフトタイマーのみがマスタタイマーによって影響を及ぼされ得る。

40

【0080】

図11に、3つのアプリケーション（App1、App2、およびApp3）の各々の1つの可能な一連の接続要求を示す。これらのアプリケーションは、図1、図2、図3、または図4のモバイルデバイス105などのモバイルデバイス上で実行され得る。特に、図11は、様々なアプリケーションからの通信要求に基づいて接続を形成するための技法を表す。この例では、接続要求は協調されない。したがって、各アプリケーションは、他のアプリケーションからの非協調間隔において接続を要求する。協調のこの欠如は、アプリケーションが接続を共有する機会を無視するので、モバイルデバイスのリソースの使用が非効率的になる。タイマー満了は、各時点における縦の矢印によって表され、結果として生じる要求期間は矩形によって示される。各接続期間は、30秒間の幅をもつ要求期間

50

矩形中のウィンドウによって表される（これは、かなり恣意的な持続時間として認識され、はるかに短いかまたは長い持続時間が各接続について発生し得る）。この幅は説明のためのものにすぎず、実際のデバイスでは任意の幅が存在し得る。図11の数値範囲ならびに後続の図は、説明のために選択されたものにすぎない。これら図は、必ずしも、いずれかのシステムまたは実施形態の実際の実装を表すものとして解釈すべきではない。

【0081】

図11では、App1は、5分ごとに通信要求を開始し得る（時間1、6、11、16などにおける接続）。同様に、App2は、5分ごとに、ただしApp1から1分だけオフセットされて接続要求を開始し得る。すなわち、App2の要求は、App1の要求と同じ期間を共有するが、同じ位相は共有しない。最後に、App3は、10分ごとに接続を実行し得、同様にApp1から4分だけずれている。App1、App2、およびApp3の非同期接続要求は、接続が常に起動および停止されるので、バッテリー電力および通信帯域幅の使用が非効率的になる。この例では、通信システムとの（モバイルデバイスの通信要素の再アクティブ化を必要とする）14個の別個の接続が、時間6と時間32との間に行われる。

【0082】

本実施形態のうちのいくつかは、より効率的な接続パターンが生じるように接続要求を協調させ得るシステムを提供することを企図する。たとえば、モバイルデバイス105が、1つのアプリケーションのためにネットワークとの接続をすでに有する場合、モバイルデバイス105は、その接続をティアダウンすること(tearing down)、および別の接続を形成することなしに、別のアプリケーションのために同じ接続を使用し得る。したがって、アプリケーションが接続を共有するようにアプリケーション接続要求間のタイミングを協調させることは、形成される必要がある接続の数を低減し得る。これらの実施形態のいくつかは、アプリケーション開発者が、モバイルデバイス105上で動作するプラットフォームのためのアプリケーションを実装するときに使用し得るプログラミングモジュールを含み得る。これらの実施形態のいくつかでは、モバイルデバイス105は、スケジューリングについての情報をネットワークに送信し得る。多くの実施形態では、モバイルデバイス105上で実行しているアプリケーションは、「タイマー」、すなわち、特定の「満了時間」がいつ発生したかを判断するように構成されたソフトウェア、ファームウェア、またはハードウェアモジュールを含み得る。タイマーは、図2または図3の接続性エンジン225によって実行され得る。タイマーは、接続性エンジン225の一部として実装され得る。これらのタイマーは、アプリケーションがその間に接続を必要とし得る時間または時間間隔を示す。ネットワークからの周期的更新を必要とする（または、ネットワークに情報を周期的に送信する）アプリケーションは、それらが接続をいつ要求すべきかを判断するために、これらのタイマーに依拠し得る。アプリケーションはまた、データを非周期的に必要とし得、タイマーによって定義されたある時間内にデータを要求し得る。いくつかのアプリケーションは、それらがどのくらいの頻度で接続を必要とするかにおいてフレキシブルであり得る。本実施形態のいくつかは、各アプリケーションに関連するフレキシビリティまたはその欠如に適応するために、異なるタイプまたは構成のタイマーを企図する。以下の説明では、説明しやすいように、「タイマー」を有するものとしてアプリケーションに言及するが、当業者は、アプリケーションが、1つまたは複数のタイマーにそれら自体で個々に関連し得る多くのコンポーネントを備えることを認識されよう。

【0083】

図12は、図11からのアプリケーションのためのタイミング図を表すが、ここで、アプリケーションは、図10の方法などのプロセスに関連してソフトタイマーまたはハードタイマーのいずれかを採用する。上記で説明したように、アプリケーションは、図1、図2、図3、または図4のモバイルデバイス105などのモバイルデバイス上で実行し得る。さらに、図2または図3の接続性エンジン225が、スケジューリングを協調させるためにプロセス1000を実装し得る。図12の例では、アプリケーション（App1、App2、およびApp3）の各々は、2分間の許容差をもつソフトタイマーを含む。プロ

セス1000などのプロセスがない場合に、図11で発生したであろうタイマー満了が、破線の矢印によって示されている。実線の矢印は、プロセス1000などのプロセスの管理の下で発生するタイマー満了を表す。いくつかの実施形態では、プロセス1000は、各時間間隔（すなわち、分1、2、3など）において実行され得る。

【0084】

たとえば、時間7において、App2のソフトタイマーは、通常満了したであろう。しかしながら、App1のタイマーが、App2のソフトタイマーの2分間の許容差内にある時間6において満了したので、App2のソフトタイマーは、時間6において早期に満了した。同様に、時間14におけるApp3のクロックの満了は、App1とApp2の両方のためのクロックの2分間の許容差内に入る。時間19までに、App1、App2、およびApp3のためのクロックの各々は、等しいかまたは互いの高調波(harmonics)である期間（それぞれ5分、5分、および10分）を有するので、クロックは正確に同相になり（通信要求は、その後、時間19、24、および29において行われ、マスタクロックは、他のアプリケーションに対する影響を有しない）、通信リソースをはるかに効率的に利用することになる。概して、タイマーが等しい期間を共有する場合、またはそれらの期間が互いの高調波（すなわち、倍数）である場合、それらは、他の「マスタ」タイマーの満了からの干渉を除外して、永続的に互いに同相のままであり得る（もちろん、同期はまた、アプリケーションのタイマーのいずれかの周期が変化した場合、中断され得る）。したがって、この例は、プロセスが適用されないときのタイミング図11と比較して、分6と分33との間に、14個の接続要求ではなく、6つの接続要求が行われることを示す。

【0085】

別の例として、図13に、アプリケーション（App1、App2、およびApp3）が同じ期間を所有しない（それぞれ、8分、5分、および10分）、別の状況を示す。図11のタイミング図の場合のように、図13の接続要求は、大部分のアプリケーションによって採用される従来の非協調技法を反映する。協調のこの欠如により、モバイルデバイスのリソースの使用が非効率的になり、分7と分34との間にモバイルデバイスの通信リソースを12回再アクティブ化することになる。

【0086】

図13のアプリケーションは、図13のApp1が時間敏感であるという点で図11とは異なる。本実施形態のいくつかの利益を採用することを望む開発者またはシステム設計者は、図13のApp1のために許容差をほとんどまたはまったく有しないタイマーを生成する可能性がある。したがって、ハードタイマーがApp1のために使用され得る。図13のApp2およびApp3は、対照的に時間敏感でなく、したがって、ソフトタイマーによって適応され得る。図13では、App2およびApp3は、それぞれ、2分間の許容差をもつソフトタイマーを与えられる。

【0087】

図14に、プロセス1000などのプロセスを、今回は、図13のアプリケーションに適用することの効果を示す。図14のApp1のタイマーは、ハードタイマーであるので、タイミング図13の図13のApp1と同じままである。しかしながら、図14のApp3は、App2の満了が、App3の意図された満了時間の2分以内であるので、時間12においてApp2のタイマーの満了に基づいて早期に満了する。その後、App2は、時間25においてApp1と同期され、App3は、時間30においてApp2と同期される。上記で説明したように、アプリケーションは、図1、図2、図3、または図4のモバイルデバイス105上で実行し得る。さらに、モバイルデバイス105は、スケジューリングを協調させるためにプロセス1000を実装し得る。この例は、あるクロックの前の同期（App1に基づく時間25におけるApp2）が、別のクロックの後続の同期（App2に基づく時間30におけるApp3）にどのように影響を及ぼし得るかを示す。アプリケーションクロックの期間は等しくないかまたは高調波でないので、クロックは、永久に正確に同期され得ない。それでも、この例は、プロセス1000などのプロセス

が適用されないときの図13のタイミング図と比較して、分6と分33との間に、12個の接続要求ではなく、8つの接続要求が行われることを示す。

【0088】

マスタクロックは、それらが関連付けられるアプリケーションに基づいて優先順位を付けられ得る。すなわち、多くの帯域幅またはバッテリー電力を消費するアプリケーションは、より少ない帯域幅またはバッテリー電力を消費するアプリケーションよりも、リソースを共有するためには好適でないことがある。したがって、いくつかの実施形態では、モバイルデバイス105は、プロセス1000が、適切な範囲内の許容差を有するソフトタイマーを早期に満了する前に、満了した「マスタ」タイマーの挙動および／または対応する優先順位を考慮するように調整され得る。一例として、プロセスは、許容差がタイマーの早期の満了を可能にするアプリケーションの帯域幅要件と比較して、タイマー（すなわちマスタタイマー）がブロック1005において満了したアプリケーションの各々のための帯域幅要件をさらに考慮し得る。マスタタイマーをもつアプリケーションの累積帯域幅要件が、他のアプリケーションとのチャンネルの共用に対して緩和する場合、モバイルデバイス105は適切なアクションをとり得る。たとえば、動作を実行することを可能にされた、マスタタイマーをもつアプリケーションにおいて、接続要求が行われ得る。次いで、ソフトタイマーは、マスタタイマーアプリケーションが過大な帯域幅をもはや消費しなくなると、アプリケーションが既存の接続を利用するように、（ソフトタイマーの許容差によって許可される場合、それらの意図された満了時間の後に可能な）適切な時間に満了させられ得る。代替的に、ソフトタイマーの満了は、より少ない帯域幅を採用する新しい接続要求が行われるまで、またはソフトタイマーの許容差の終了まで遅延され得る。

【0089】

図11および図13を参照しながら説明したような、上記で説明した実施形態のいくつかでは、タイマー満了は、通信ウィンドウの立上りエッジにおいて発生することが示される。いくつかの実施形態は、代わりに、データ接続の「立下りエッジ」において、すなわち、ノードとの通信が休眠状態に入ろうとしているときに分析を実行することを企図する。接続を開くことと閉じることの両方にコストがかかるので、モバイルデバイス105は、いずれかのアプリケーションが接続をオープンしようとしているかどうかと、現在の接続が閉じられると新しい要求を開始するのではなく、既存の接続を代わりに利用することができるかどうかとを確かめるために、接続を閉じる前にプロセス1000と同様のプロセスを実行し得る。

【0090】

本明細書で開示する技法および構造を採用して、モバイルデバイスは、アプリケーションからのシステムコールをキャプチャし、それらがオペレーティングシステムに到着することを保留するためのアプリケーションプログラムインターフェース（API）を与える、（説明のためにラッパーと呼ぶ）ソフトウェアレイヤを採用し得る。キャプチャされたコールはアグリゲートされ得るので、ユーザがモバイルデバイスをアクティブに使用していない期間中に、モバイルデバイスの頻繁な起動が低減され得、他の通信リソースが温存され得る。

【0091】

情報および信号は多種多様な技術および技法のいずれかを使用して表され得ることを、当業者は理解されよう。たとえば、上記の説明全体にわたって言及され得るデータ、命令、コマンド、情報、信号、ビット、シンボル、およびチップは、電圧、電流、電磁波、磁界または磁性粒子、光場または光学粒子、あるいはそれらの任意の組合せによって表され得る。

【0092】

さらに、本明細書で開示した実施形態に関して説明した様々な例示的な論理ブロック、モジュール、回路、およびアルゴリズムステップは、電子ハードウェア、コンピュータソフトウェア、または両方の組合せとして実装され得ることを、当業者は諒解されよう。ハードウェアとソフトウェアのこの互換性を明確に示すために、様々な例示的なコンポーネ

ント、ブロック、モジュール、回路、およびステップについて、上記では概してそれらの機能に関して説明した。そのような機能をハードウェアとして実装するか、ソフトウェアとして実装するかは、特定の適用例および全体的なシステムに課された設計制約に依存する。当業者は、説明した機能を特定の適用例ごとに様々な方法で実装し得るが、そのような実装の決定は、本発明の例示的な実施形態の範囲からの逸脱を生じるものと解釈すべきではない。

【0093】

本明細書で開示した実施形態に関連して説明した様々な例示的な論理ブロック、モジュール、および回路は、汎用プロセッサ、デジタル信号プロセッサ(DSP)、特定用途向け集積回路(ASIC)、フィールドプログラマブルゲートアレイ(FPGA)または他のプログラマブル論理デバイス、個別ゲートまたはトランジスタロジック、個別ハードウェアコンポーネント、あるいは本明細書で説明した機能を実行するように設計されたそれらの任意の組合せを用いて実装または実行し得る。汎用プロセッサはマイクロプロセッサであり得るが、代替として、プロセッサは、任意の従来のプロセッサ、コントローラ、マイクロコントローラ、または状態機械であり得る。プロセッサはまた、コンピューティングデバイスの組合せ、たとえば、DSPとマイクロプロセッサとの組合せ、複数のマイクロプロセッサ、DSPコアと連携する1つまたは複数のマイクロプロセッサ、あるいは任意の他のそのような構成として実装され得る。

【0094】

本明細書で開示する実施形態に関して説明した方法またはアルゴリズムのステップは、直接ハードウェアで実施され得るか、プロセッサによって実行されるソフトウェアモジュールで実施され得るか、またはその2つの組合せで実施され得る。ソフトウェアモジュールは、ランダムアクセスメモリ(RAM)、フラッシュメモリ、読取り専用メモリ(ROM)、電氣的プログラマブルROM(EPROM)、電氣的消去可能プログラマブルROM(EEPROM)、レジスタ、ハードディスク、リムーバブルディスク、CD-ROM、または当技術分野で知られている任意の他の形態の記憶媒体中に常駐し得る。例示的な記憶媒体は、プロセッサが記憶媒体から情報を読み取り、記憶媒体に情報を書き込むことができるように、プロセッサに結合される。代替として、記憶媒体はプロセッサに一体化され得る。プロセッサおよび記憶媒体はASIC中に常駐し得る。ASICはユーザ端末中に常駐し得る。代替として、プロセッサおよび記憶媒体は、ユーザ端末中に個別構成要素として常駐し得る。

【0095】

1つまたは複数の例示的な実施形態では、説明した機能は、ハードウェア、ソフトウェア、ファームウェア、またはそれらの任意の組合せで実装され得る。ソフトウェアで実装した場合、機能は、1つまたは複数の命令またはコードとして非一時的コンピュータ可読媒体上に記憶され得るか、あるいは非一時的コンピュータ可読媒体を介して送信され得る。コンピュータ可読媒体は、ある場所から別の場所へのコンピュータプログラムの転送を可能にする任意の媒体を含む、コンピュータ記憶媒体とコンピュータ通信媒体の両方を含む。記憶媒体は、コンピュータによってアクセスされ得る任意の利用可能な媒体であり得る。限定ではなく例として、そのようなコンピュータ可読媒体は、RAM、ROM、EEPROM、CD-ROMまたは他の光ディスクストレージ、磁気ディスクストレージまたは他の磁気ストレージデバイス、あるいは命令またはデータ構造の形態の所望のプログラムコードを搬送または記憶するために使用され得、コンピュータによってアクセスされ得る、任意の他の媒体を備えることができる。また、いかなる接続もコンピュータ可読媒体と適切に呼ばれる。たとえば、ソフトウェアが、同軸ケーブル、光ファイバーケーブル、ツイストペア、デジタル加入者回線(DSL)、または赤外線、無線、およびマイクロ波などのワイヤレス技術を使用して、ウェブサイト、サーバ、または他のリモートソースから送信される場合、同軸ケーブル、光ファイバーケーブル、ツイストペア、DSL、または赤外線、無線、およびマイクロ波などのワイヤレス技術は、媒体の定義に含まれる。本明細書で使用するディスク(disk)およびディスク(disc)は、コンパクトディスク(di

sc) (CD)、レーザーディスク(登録商標)(disc)、光ディスク(disc)、デジタル
多用途ディスク(disc)(DVD)、フロッピー(登録商標)ディスク(disk)およびブ
ルーレイ(登録商標)ディスク(disc)を含み、ディスク(disk)は、通常、データを磁
氣的に再生し、ディスク(disc)は、データをレーザーで光学的に再生する。上記の組合
せもコンピュータ可読媒体の範囲内に含めるべきである。

【0096】

開示した例示的な実施形態の前述の説明は、当業者が本発明を実施または使用すること
ができるように与えたものである。これらの例示的な実施形態への様々な修正は当業者には
容易に明らかであり、本明細書で定義した一般原理は、本発明の趣旨または範囲から逸
脱することなく他の実施形態に適用され得る。したがって、本発明は、本明細書で示した
例示的な実施形態に限定されるものではなく、本明細書で開示した原理および新規の特徴
に一致する最も広い範囲を与えられるべきである。

以下に、本願出願の当初の特許請求の範囲に記載された発明を付記する。

〔1〕モバイルデバイスにおけるワイヤレス通信のための方法であって、

前記モバイルデバイス上の第1のアプリケーションからの要求をインターセプトすることであって、前記要求が、前記モバイルデバイスのための通信を行うための要求である、
インターセプトすることと、

前記モバイルデバイス上で実行しているオペレーティングシステムの伝送制御プロトコル／インターネットプロトコル(TCP／IP)スタックに前記要求が到着することを保留することと、

トリガリングイベントを検出すると、前記オペレーティングシステムに対して前記要求をリリースすることと、
を備える、方法。

〔2〕前記保留することは、前記モバイルデバイスがバックグラウンドモードにある期間中に行われる、〔1〕に記載の方法。

〔3〕前記要求を、前記モバイルデバイスのための通信を行うための他のインターセプトされた要求とアグリゲートすること
をさらに備える、〔1〕に記載の方法。

〔4〕前記アプリケーションからの前記要求を前記インターセプトすることと、前記他の要求を前記インターセプトすることとが、異なる時間に行われる、〔3〕に記載の方法。

〔5〕ラッパーのための命令を実行することであって、前記実行されたラッパーが、前記第1のアプリケーションからの前記要求を前記インターセプトすることを行う、実行すること
をさらに備える、〔1〕に記載の方法。

〔6〕前記ラッパーが、前記モバイルデバイスのアプリケーションレイヤと前記オペレーティングシステムのソケットレイヤとの間に位置する、〔5〕に記載の方法。

〔7〕前記第1のアプリケーションを、要求が保留される種類のアプリケーションとして識別すること

をさらに備える、〔1〕に記載の方法。

〔8〕アプリケーションをクリティカルアプリケーションまたは非クリティカルアプリケーションとして識別することと、

非クリティカルアプリケーションからの要求のみを保留することと、
をさらに備える、〔7〕に記載の方法。

〔9〕前記トリガリングイベントは、タイマーの満了、ディスプレイのステータス変更、マイクロフォンのステータス変更、スピーカーのステータス変更、前記モバイルデバイスの全地球測位システム(GPS)センサーのステータス変更、ユニバーサルシリアルバスポートが使用中であるという指示、前記モバイルデバイスにオーディオ機器が接続されているという指示、前記モバイルデバイスにビデオ機器が接続されているという指示、Wi-Fiタイプのネットワークへの接続が利用可能であるという指示、またはセルラータイプのネットワークへの無線接続が開いているという指示のうちの少なくとも1つを備える

10

20

30

40

50

、〔１〕に記載の方法。

〔１０〕前記第１のアプリケーションの遅延許容差を判断することと、

前記判断された遅延許容差に基づいて前記第１のアプリケーションにコールバック関数を与えることであって、前記コールバック関数が、通信リソースに接続するように前記第１のアプリケーションに命令する、与えることと、

をさらに備える、〔１〕に記載の方法。

〔１１〕前記第１のアプリケーションに関連する第１のタイマーの満了時間を判断することと、

第２のアプリケーションに関連する許容差と第２のタイマーの満了時間とを判断することと、

前記第１のタイマーの前記満了時間と、前記許容差と、前記第２のタイマーの前記満了時間とに基づいて、前記第２のタイマーを満了させることと、

前記モバイルデバイスのための通信を行うための前記第１のアプリケーションからの前記要求と前記第２のアプリケーションからのインターセプトされた要求とをリリースすることと、

をさらに備える、〔１〕に記載の方法。

〔１２〕前記アプリケーションからデッドラインを受信することと、

前記デッドラインの前まで前記要求を保留することと、

前記デッドラインより前に前記通信リソースに接続するための前記要求をリリースすることと、

をさらに備える、〔１〕に記載の方法。

〔１３〕前記要求が、前記モバイルデバイスのための通信チャネルを確立するためのシステムコールを備える、〔１〕に記載の方法。

〔１４〕前記トリガリングイベントを検出すると、前記オペレーティングシステムのソケットレイヤに対して前記要求をリリースすること

をさらに備える、〔１〕に記載の方法。

〔１５〕前記要求を前記リリースすることがどのくらいの頻度で行われるかに関係する間隔についての指示を受信すること

をさらに備える、〔１〕に記載の方法。

〔１６〕前記間隔が、ネットワーク中のステートフルインターネットプロトコル（ＩＰ）ミドルボックスにおけるタイムアウト値よりも小さい、〔１５〕に記載の方法。

〔１７〕ワイヤレス通信のために構成されたモバイルデバイスであって、

プロセッサと、

前記プロセッサと電子通信しているメモリであって、オペレーティングシステムを備える、メモリと

を備え、

前記プロセッサが接続性エンジンを備え、前記エンジンが、

前記モバイルデバイス上の第１のアプリケーションからの要求をインターセプトすることであって、前記要求が、前記モバイルデバイスのための通信を行うための要求である、インターセプトすることと、

前記モバイルデバイス上で実行しているオペレーティングシステムのＴＣＰ／ＩＰスタックに前記要求が到着することを保留することと、

トリガリングイベントを検出すると、前記オペレーティングシステムに対して前記要求をリリースすることと、

を行うための命令を実行するように構成された、モバイルデバイス。

〔１８〕前記保留することは、前記モバイルデバイスがバックグラウンドモードにある期間中に行われる、〔１７〕に記載のモバイルデバイス。

〔１９〕前記要求を、前記モバイルデバイスのための通信を行うための他のインターセプトされた要求とアグリゲートすること

をさらに備える、〔１７〕に記載のモバイルデバイス。

10

20

30

40

50

〔20〕前記第1のアプリケーションからの前記要求を前記インターセプトすることと、前記他の要求を前記インターセプトすることとが、異なる時間に行われる、〔19〕に記載のモバイルデバイス。

〔21〕前記メモリがラッパーをさらに備え、前記接続性エンジンが、前記ラッパーに対して命令を実行するようにさらに構成され、前記命令が実行されたとき、前記ラッパーが、前記アプリケーションからの前記要求をインターセプトするように構成された、〔17〕に記載のモバイルデバイス。

〔22〕前記ラッパーが、前記モバイルデバイスのアプリケーションレイヤと前記オペレーティングシステムのソケットレイヤとの間に位置する、〔21〕に記載のモバイルデバイス。

〔23〕前記接続性エンジンは、
前記第1のアプリケーションを、要求が保留される種類のアプリケーションとして識別すること

を行うための命令を実行するようにさらに構成された、〔17〕に記載のモバイルデバイス。

〔24〕前記接続性エンジンが、
アプリケーションをクリティカルアプリケーションまたは非クリティカルアプリケーションとして識別することと、

非クリティカルアプリケーションからの要求のみを保留することと、
を行うための命令を実行するようにさらに構成された、〔23〕に記載のモバイルデバイス。

〔25〕前記トリガリングイベントは、タイマーの満了、ディスプレイのステータス変更、マイクロフォンのステータス変更、スピーカーのステータス変更、前記モバイルデバイスの全地球測位システム（GPS）センサーのステータス変更、ユニバーサルシリアルバスポートが使用中であるという指示、前記モバイルデバイスにオーディオ機器が接続されているという指示、前記モバイルデバイスにビデオ機器が接続されているという指示、Wi-Fiタイプのネットワークへの接続が利用可能であるという指示、またはセルラータイプのネットワークへの無線接続が開いているという指示のうちの少なくとも1つを備える、〔17〕に記載のモバイルデバイス。

〔26〕前記接続性エンジンは、
前記第1のアプリケーションの遅延許容差を判断することと、

前記判断された遅延許容差に基づいて前記第1のアプリケーションにコールバック関数を与えることであって、前記コールバック関数が、前記通信リソースに接続するように前記第1のアプリケーションに命令する、与えることと、
を行うための命令を実行するようにさらに構成された、〔17〕に記載のモバイルデバイス。

〔27〕前記接続性エンジンが、
前記第1のアプリケーションに関連する第1のタイマーの満了時間を判断することと、
第2のアプリケーションに関連する許容差と第2のタイマーの満了時間とを判断することと、

前記第1のタイマーの前記満了時間と、前記許容差と、前記第2のタイマーの前記満了時間とに基づいて、前記第2のタイマーを満了させることと、

前記モバイルデバイスのための通信を行うための前記第1のアプリケーションからの前記要求と前記第2のアプリケーションからのインターセプトされた要求とをリリースすることと、

を行うための命令を実行するようにさらに構成された、〔17〕に記載のモバイルデバイス。

〔28〕前記接続性エンジンが、
前記アプリケーションからデッドラインを受信することと、
前記デッドラインの前まで前記要求を保留することと、

10

20

30

40

50

前記デッドラインより前に前記通信リソースに接続するための前記要求をリリースすることと、
を行うための命令を実行するようにさらに構成された、[1 7] に記載のモバイルデバイス。

[2 9] 前記要求が、通信チャネルを確立するためのシステムコールを備える、[1 7] に記載のモバイルデバイス。

[3 0] 前記接続性エンジンが、
前記トリガリングイベントを検出すると、前記オペレーティングシステムのソケットレイヤに対して前記要求をリリースすること
を行うための命令を実行するようにさらに構成された、[1 7] に記載のモバイルデバイス。

[3 1] 前記接続性エンジンは、
前記要求を前記リリースすることがどのくらいの頻度で行われるかに関係する間隔についての指示を受信すること
を行うための命令を実行するようにさらに構成された、[1 7] に記載のモバイルデバイス。

[3 2] 前記間隔が、ネットワーク中のステートフルインターネットプロトコル (I P) ボックスにおけるタイムアウト値よりも小さい、[3 1] に記載のモバイルデバイス。

[3 3] モバイルデバイス上のアプリケーションからのネットワークアクセスの要求を管理するように構成された装置であって、

前記モバイルデバイス上のアプリケーションからの要求をインターセプトするための手段であって、前記要求が、前記モバイルデバイスのための通信を行うための要求である、インターセプトするための手段と、

前記モバイルデバイス上で実行しているオペレーティングシステムの T C P / I P スタックに前記要求が到着することを保留するための手段と、

トリガリングイベントを検出すると、前記オペレーティングシステムに対して前記要求をリリースするための手段と、
を備える、装置。

[3 4] 前記保留することは、前記モバイルデバイスがバックグラウンドモードにある期間中に行われる、[3 3] に記載の装置。

[3 5] 前記要求を、前記モバイルデバイスのための通信を行うための他のインターセプトされた要求とアグリゲートするための手段
をさらに備える、[3 3] に記載の装置。

[3 6] 前記アプリケーションからの前記要求を前記インターセプトすることと、前記他の要求を前記インターセプトすることとが、異なる時間に行われる、[3 3] に記載の装置。

[3 7] ラッパーを実行するための手段であって、前記実行されたラッパーが、前記第 1 のアプリケーションからの前記要求をインターセプトするように構成された、実行するための手段

をさらに備える、[3 3] に記載の装置。

[3 8] 前記ラッパーが、前記モバイルデバイスのアプリケーションレイヤと前記オペレーティングシステムのソケットレイヤとの間に位置する、[3 7] に記載の装置。

[3 9] 前記アプリケーションを、要求が保留される種類のアプリケーションとして識別するための手段

をさらに備える、[3 3] に記載の装置。

[4 0] アプリケーションをクリティカルアプリケーションまたは非クリティカルアプリケーションとして識別するための手段と、

非クリティカルアプリケーションからの要求のみを保留するための手段と、
をさらに備える、[3 9] に記載の装置。

[4 1] モバイルデバイス上のアプリケーションからのネットワークアクセスの要求を管

10

20

30

40

50

理するように構成されたコンピュータプログラム製品であって、前記製品が非一時的コンピュータ可読媒体を備え、前記媒体が、

前記モバイルデバイス上のアプリケーションからの要求をインターセプトするためのコードであって、前記要求が、前記モバイルデバイスのための通信を行うための要求である、インターセプトするためのコードと、

前記モバイルデバイス上で実行しているオペレーティングシステムのTCP/IPスタックに前記要求が到着することを保留するためのコードと、

トリガリングイベントを検出すると、前記オペレーティングシステムに対して前記要求をリリースするためのコードと、

を備える、コンピュータプログラム製品。

10

【図 1】

図 1

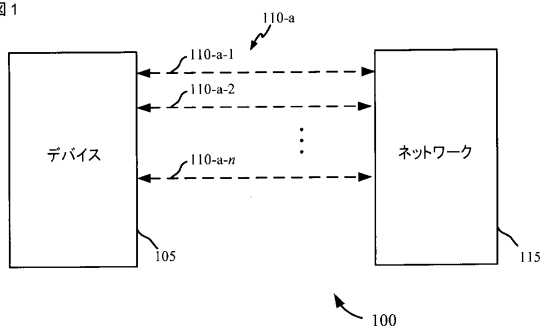


FIG. 1

【図 2】

図 2

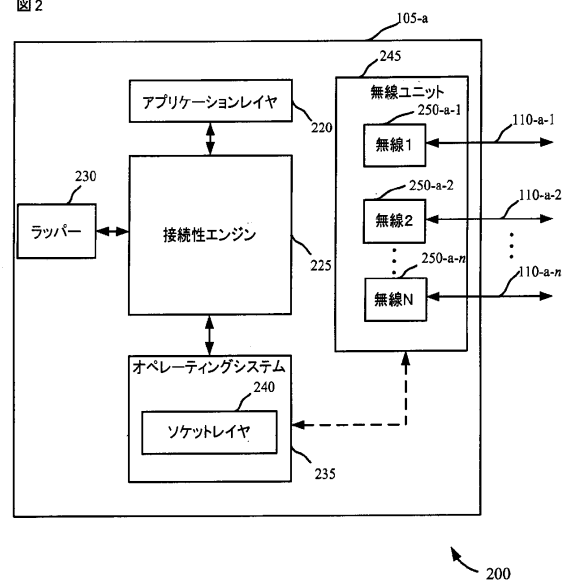


FIG. 2

【図 3】

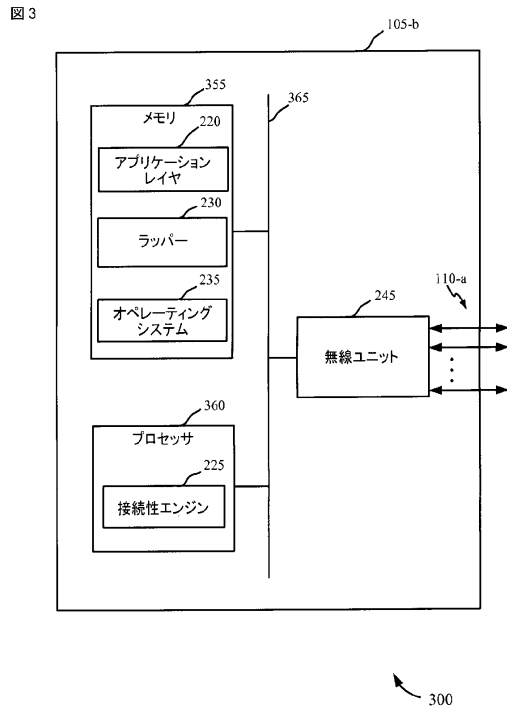


FIG. 3

【図 4】

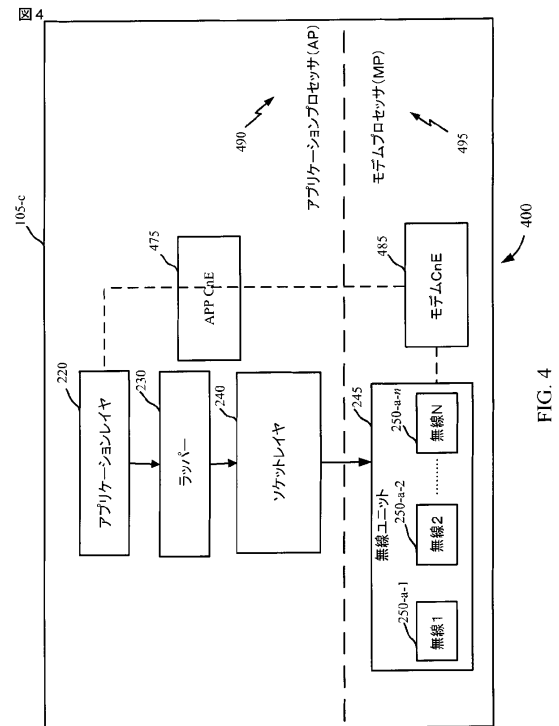


FIG. 4

【図 5】

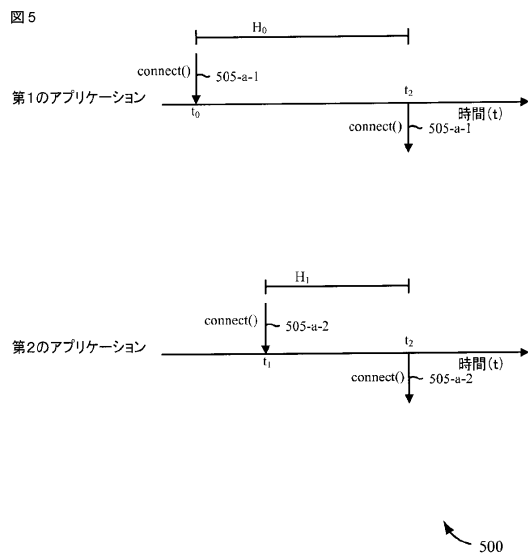


FIG. 5

【図 6】

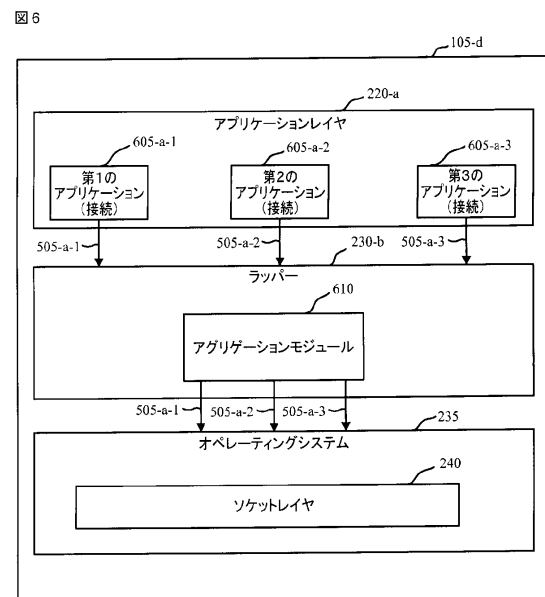


FIG. 6

【図 7】

図 7

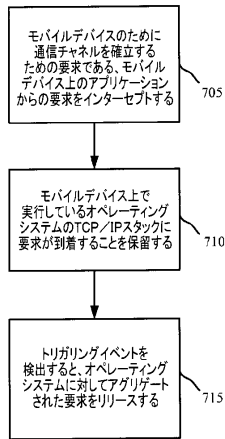


FIG. 7

【図 8】

図 8

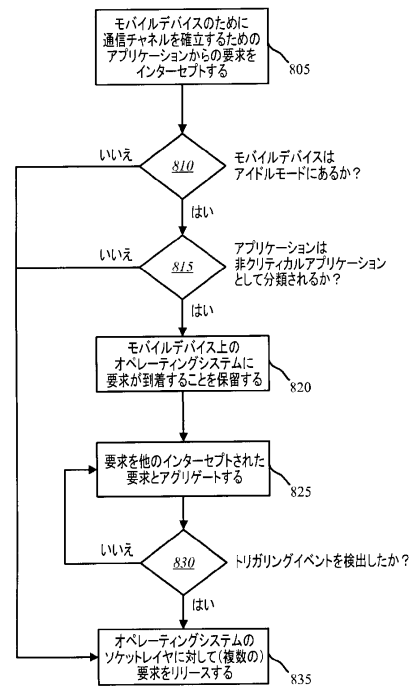


FIG. 8

【図 9】

図 9

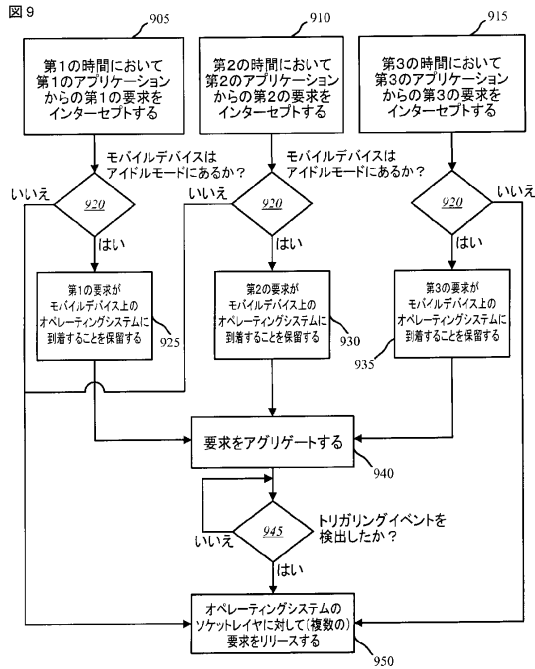


FIG. 9

【図 10】

図 10

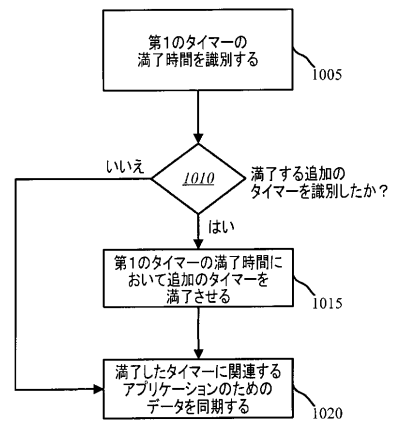
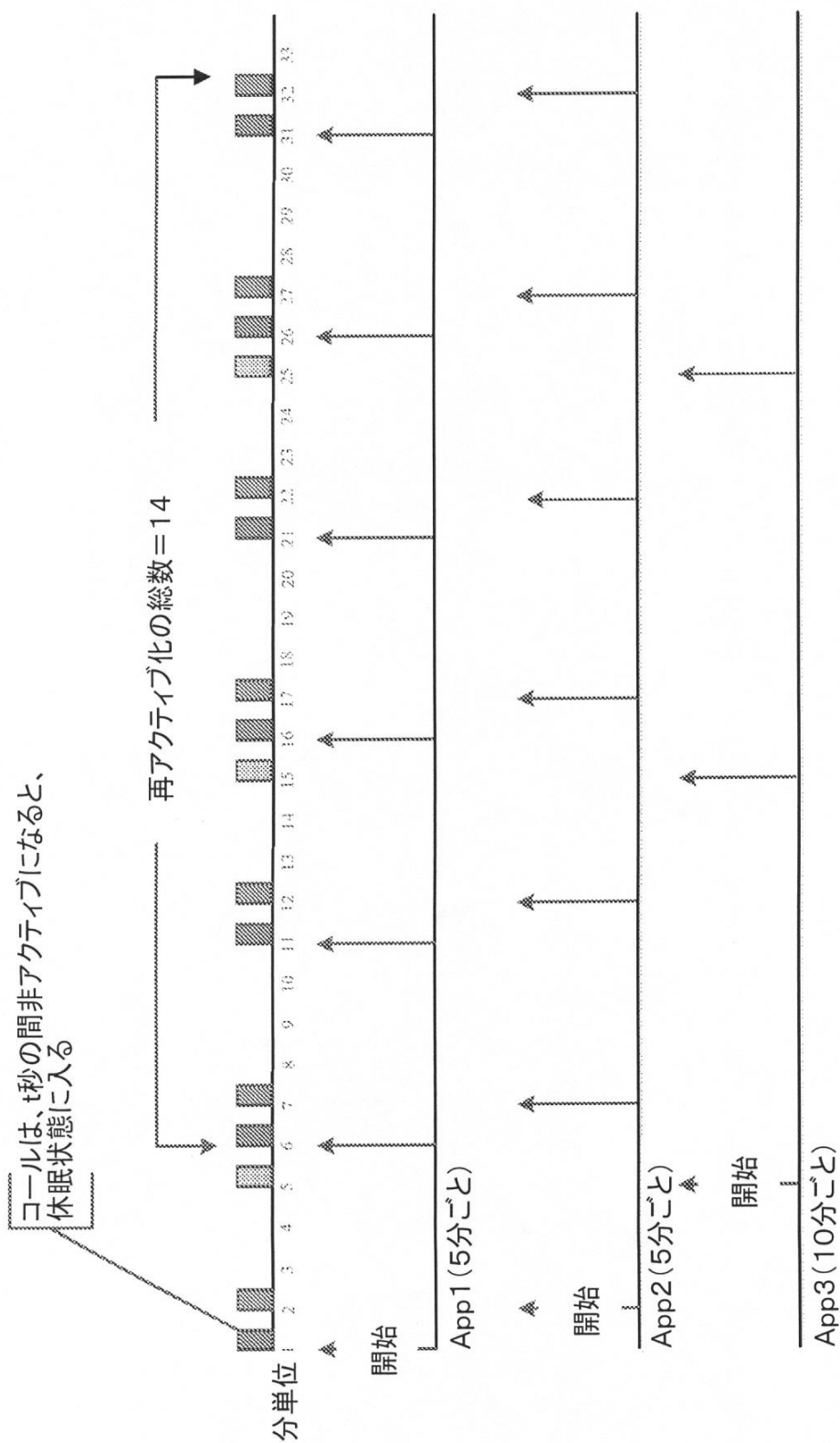


FIG. 10

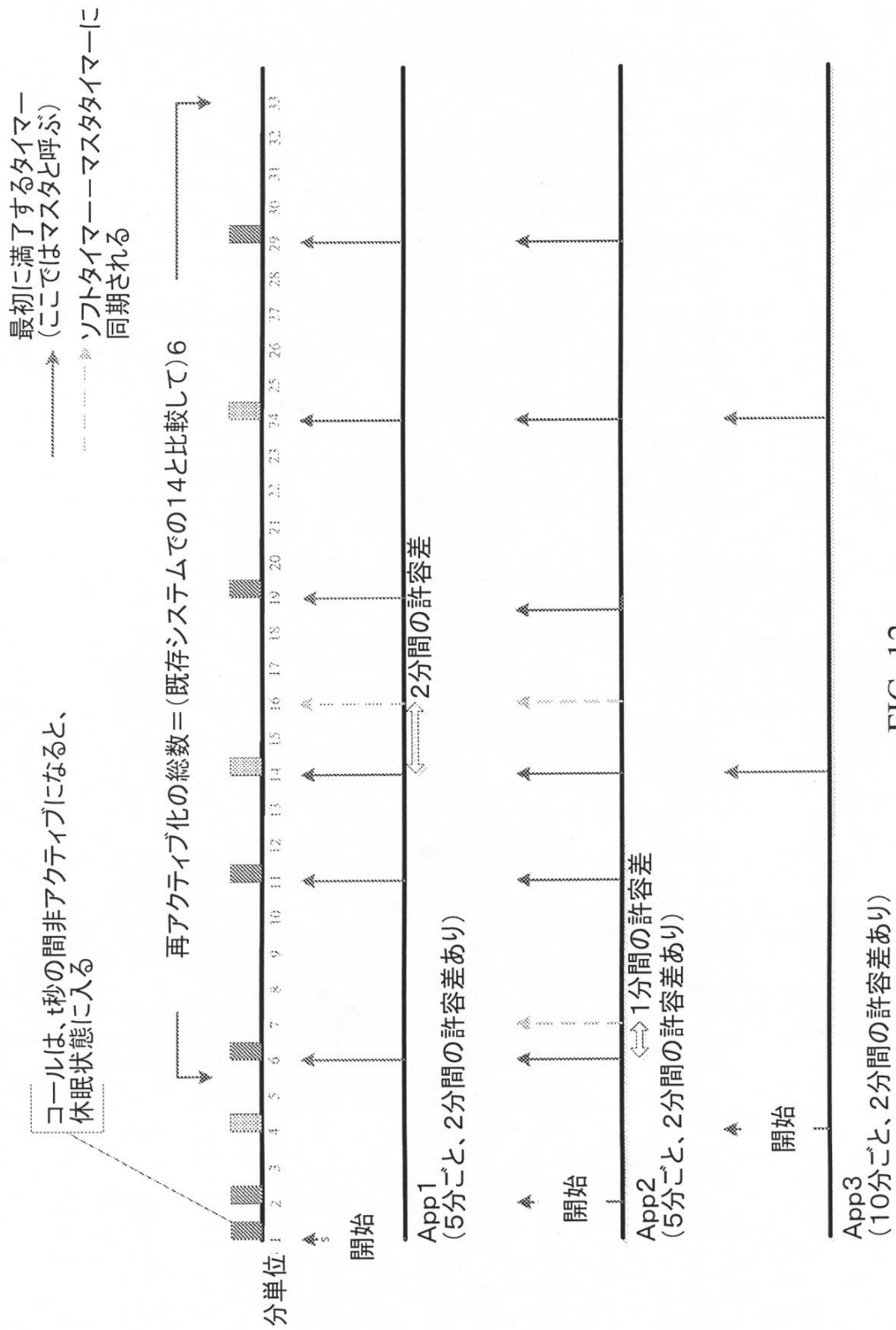
【図 1 1】

图 11



【図 12】

図 12



【図 1 3】

图 13

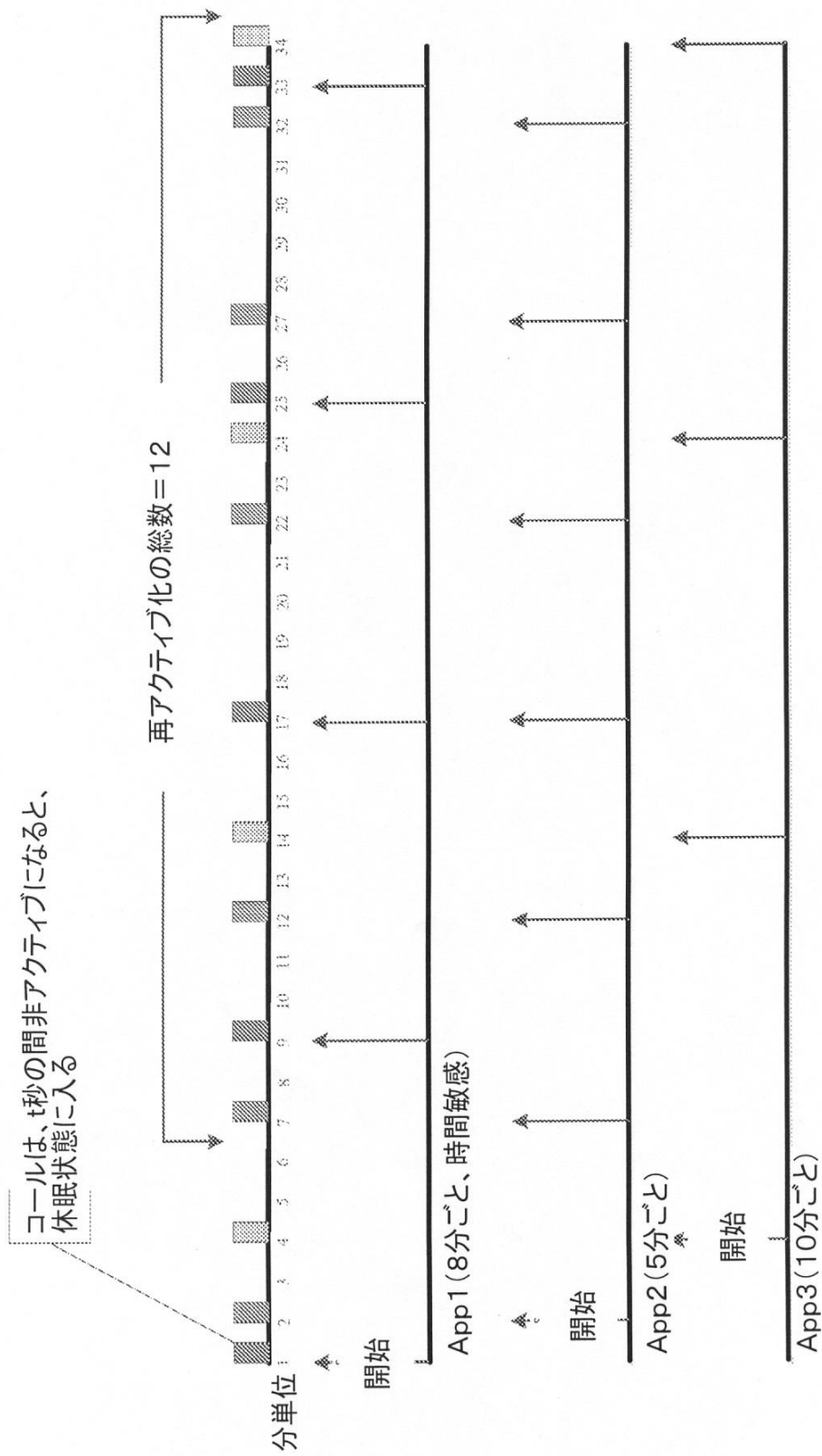


FIG. 13

【図 14】

図 14

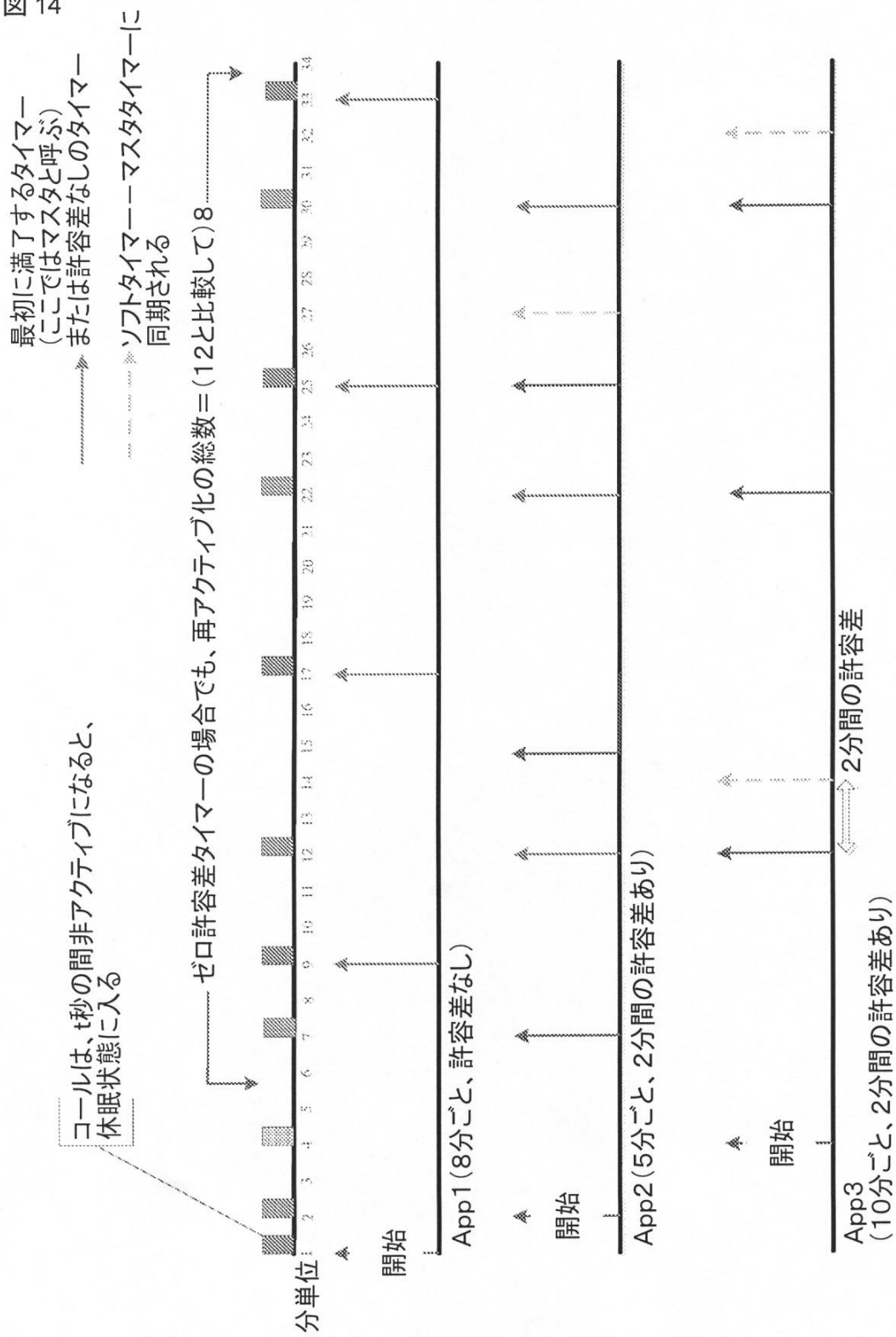


FIG. 14

フロントページの続き

- (74)代理人 100075672
弁理士 峰 隆司
- (74)代理人 100153051
弁理士 河野 直樹
- (74)代理人 100140176
弁理士 砂川 克
- (74)代理人 100158805
弁理士 井関 守三
- (74)代理人 100172580
弁理士 赤穂 隆雄
- (74)代理人 100179062
弁理士 井上 正
- (74)代理人 100124394
弁理士 佐藤 立志
- (74)代理人 100112807
弁理士 岡田 貴志
- (74)代理人 100111073
弁理士 堀内 美保子
- (72)発明者 ジアレッタ、ジェラルド
アメリカ合衆国、カリフォルニア州 9 2 1 2 1-1 7 1 4、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5
- (72)発明者 スンダララジャン、ジャイ・クマー
アメリカ合衆国、カリフォルニア州 9 2 1 2 1-1 7 1 4、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5
- (72)発明者 ゴエル、サタシュ
アメリカ合衆国、カリフォルニア州 9 2 1 2 1-1 7 1 4、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5
- (72)発明者 メイラン、アーナウド
アメリカ合衆国、カリフォルニア州 9 2 1 2 1-1 7 1 4、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5
- (72)発明者 サフ、デベシュ・クマー
アメリカ合衆国、カリフォルニア州 9 2 1 2 1-1 7 1 4、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5
- (72)発明者 ジャイン、サチン
アメリカ合衆国、カリフォルニア州 9 2 1 2 1-1 7 1 4、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5
- (72)発明者 ペイヤッピー、アジス・トム
アメリカ合衆国、カリフォルニア州 9 2 1 2 1、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5

審査官 齋藤 浩兵

- (56)参考文献 特開2009-021966 (JP, A)
国際公開第2011/023096 (WO, A1)
特開2008-072568 (JP, A)
特開2010-074818 (JP, A)
特表2009-540687 (JP, A)
特開2002-091841 (JP, A)

特開2001-339465 (JP, A)

米国特許出願公開第2009/0228602 (US, A1)

(58)調査した分野(Int.Cl., DB名)

H04B 7/24 - 7/26

H04W 4/00 - 99/00

H04M 1/00

H04M 11/00