

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2013-257873

(P2013-257873A)

(43) 公開日 平成25年12月26日 (2013. 12. 26)

(51) Int. Cl.	F I	テーマコード (参考)
G06F 9/48 (2006.01)	G06F 9/46 452B	
G06F 9/54 (2006.01)	G06F 9/46 480Z	

審査請求 有 請求項の数 20 O L 外国語出願 (全 36 頁)

(21) 出願番号 特願2013-120767 (P2013-120767)
 (22) 出願日 平成25年6月7日 (2013. 6. 7)
 (31) 優先権主張番号 61/657, 611
 (32) 優先日 平成24年6月8日 (2012. 6. 8)
 (33) 優先権主張国 米国 (US)
 (31) 優先権主張番号 13/656, 506
 (32) 優先日 平成24年10月19日 (2012. 10. 19)
 (33) 優先権主張国 米国 (US)

(特許庁注：以下のものは登録商標)

1. FIREWIRE

(71) 出願人 503260918
 アップル インコーポレイテッド
 アメリカ合衆国 95014 カリフォル
 ニア州 クパチーノ インフィニット ル
 ープ 1
 (74) 代理人 100092093
 弁理士 辻居 幸一
 (74) 代理人 100082005
 弁理士 熊倉 禎男
 (74) 代理人 100067013
 弁理士 大塚 文昭
 (74) 代理人 100086771
 弁理士 西島 孝喜
 (74) 代理人 100121979
 弁理士 岩崎 吉信

最終頁に続く

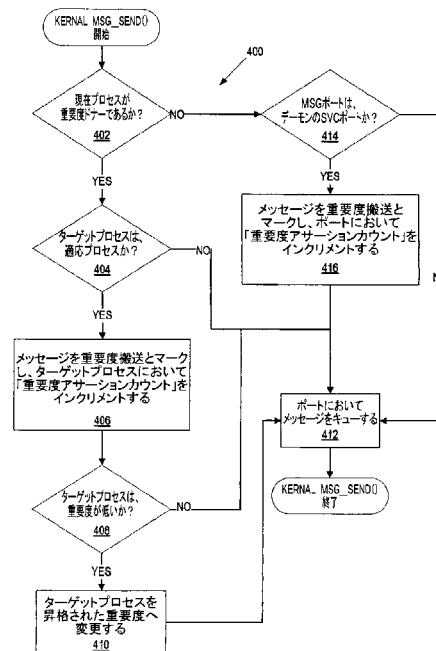
(54) 【発明の名称】 適応プロセス重要度

(57) 【要約】

【課題】受信したメッセージに基づいてプロセスの重要度を変更する技術を提供する。

【解決手段】デーモンプロセスの重要度を変更する方法及び装置が開示される。規範的な実施形態において、装置は、デーモンプロセスに向けられたユーザプロセスからのメッセージを受信し、デーモンプロセスは、ユーザプロセスとは独立して実行され、そして第1のデーモンプロセスは、他の実行プロセスとメッセージを通信する。更に、装置は、第1のデーモンプロセスの重要度を変更できることを第1のメッセージが指示するかどうか決定する。第1のデーモンプロセスの重要度を変更できることを第1のメッセージが指示する場合には、装置が第1のデーモンプロセスの重要度を変更する。装置は、更に、第1のメッセージを第1のデーモンプロセスに転送する。

【選択図】 図4



【特許請求の範囲】**【請求項 1】**

1つ以上の処理ユニットが第1のデーモンプロセスの重要度を変更する方法を遂行するようにさせる実行可能なインストラクションを有する非一時的マシン読み取り可能な媒体において、前記方法は、

前記第1デーモンプロセスに向けられたユーザプロセスからの第1メッセージを受け取り、前記第1デーモンプロセスは、前記ユーザプロセスとは独立して実行され、前記第1デーモンプロセスは、他の実行プロセスとメッセージを通信し、そのメッセージは、前記ユーザプロセスから前記第1デーモンプロセスへのサービス要求であり、

前記第1デーモンプロセスの重要度を変更できることを前記第1メッセージが指示するかどうか決定し、

前記第1デーモンプロセスの重要度を変更できることを前記第1メッセージが指示する場合には、前記第1デーモンプロセスの重要度を変更し、及び

前記第1メッセージを前記第1デーモンプロセスへ転送する、ことを含むものである、非一時的マシン読み取り可能な媒体。

【請求項 2】

前記ユーザプロセスは、前記第1デーモンプロセスよりも高い高重要度で実行される、請求項1に記載の非一時的マシン読み取り可能な媒体。

【請求項 3】

前記第1デーモンプロセスの重要度の変更は、前記第1デーモンプロセスの重要度を、前記ユーザプロセスと同じ重要度に変更する、請求項1に記載の非一時的マシン読み取り可能な媒体。

【請求項 4】

前記決定は、第1デーモンプロセスの重要度を変更できるメッセージが第1デーモンプロセスに向けられるかどうか決定することを含む、請求項1に記載の非一時的マシン読み取り可能な媒体。

【請求項 5】

前記メッセージの転送は、前記第1デーモンプロセスに対しポートキューにおいて前記メッセージをエンキューすることを含む、請求項1に記載の非一時的マシン読み取り可能な媒体。

【請求項 6】

前記第1デーモンプロセスに対するアサーションのカウン트를維持し、

前記第1デーモンプロセスに対するドロップアサーションを受け取り、

前記第1デーモンプロセスに対するアサーションのカウン트를デクリメントし、及び

前記第1デーモンプロセスに対するアサーションのカウン트가ゼロである場合には、前記第1デーモンプロセスの重要度をデフォルト重要度へ変更する、ことを更に含む、請求項1に記載の非一時的マシン読み取り可能な媒体。

【請求項 7】

第2デーモンプロセスに向けられたユーザプロセスからの第2メッセージを受け取り、前記第2デーモンプロセスは、実行されており、前記第2デーモンプロセスは、前記第1デーモンプロセスとは独立して実行され、前記第2デーモンプロセスは、他の実行プロセスとメッセージを通信し、

前記第2デーモンプロセスの重要度を変更できることを前記第2メッセージが指示するかどうか決定し、

前記第2デーモンプロセスの重要度を変更できることを前記第2メッセージが指示する場合には、前記第2デーモンプロセスの重要度を変更し、及び

前記第2メッセージを前記第2デーモンプロセスへ転送する、ことを更に含む、請求項1に記載の非一時的マシン読み取り可能な媒体。

【請求項 8】

前記第2デーモンプロセスの重要度の変更は、変更される前記第2デーモンプロセスの

10

20

30

40

50

重要度を、前記ユーザプロセスと同じ重要度に変更する、請求項 7 に記載の非一時的マシン読み取り可能な媒体。

【請求項 9】

前記第 1 メッセージを、重要度を搬送するものとしてマークすることを更に含む、請求項 1 に記載の非一時的マシン読み取り可能な媒体。

【請求項 10】

前記第 1 メッセージを前記第 1 デモンへ転送することは、前記第 1 デモンプロセスの重要度の変更に応答して行われる、請求項 1 に記載の非一時的マシン読み取り可能な媒体。

【請求項 11】

前記ユーザプロセスは、前景において実行されるユーザアプリケーションのためのプロセスである、請求項 1 に記載の非一時的マシン読み取り可能な媒体。

【請求項 12】

前記第 1 デモンプロセスは、背景において実行される、請求項 1 に記載の非一時的マシン読み取り可能な媒体。

【請求項 13】

プロセス重要度は、ユーザに代わってアクションが必要であることを指示する、請求項 1 に記載の非一時的マシン読み取り可能な媒体。

【請求項 14】

1 つ以上の処理ユニットが適応デモンプロセスの重要度を変更する方法を遂行するようにさせる実行可能なインストラクションを有する非一時的マシン読み取り可能な媒体において、前記方法は、

適応デモンに向けられた重要度ドナープロセスからメッセージを受け取り、前記適応デモンは、起動されるべきものであり、

前記適応デモンプロセスの重要度を変更できることを前記メッセージが指示するかどうか決定し、

前記適応デモンプロセスの重要度を変更できることを前記メッセージが指示する場合には、前記適応デモンプロセスを、異なる重要度、即ち前記適応デモンプロセスに関連したデフォルト重要度で起動し、及び

前記起動された適応デモンプロセスへ前記メッセージを転送する、

ことを含む、非一時的マシン読み取り可能な媒体。

【請求項 15】

前記適応デモンプロセスは、変更可能なプロセス重要度をもつデモンプロセスである、請求項 14 に記載の非一時的マシン読み取り可能な媒体。

【請求項 16】

前記重要度ドナープロセスは、別のプロセスに重要度を寄贈できるプロセスである、請求項 14 に記載の非一時的マシン読み取り可能な媒体。

【請求項 17】

前記重要度ドナープロセスは、ユーザアプリケーションプロセス、及び重要度が昇格された別の適応デモンプロセスより成るグループから選択される、請求項 16 に記載の非一時的マシン読み取り可能な媒体。

【請求項 18】

前記適応デモンプロセスの重要度は、重要度ドナーの重要度へ昇格される、請求項 14 に記載の非一時的マシン読み取り可能な媒体。

【請求項 19】

前記メッセージは、前記重要度ドナープロセスから前記適応デモンプロセスへのサービス要求である、請求項 14 に記載の非一時的マシン読み取り可能な媒体。

【請求項 20】

第 1 デモンプロセスの重要度を変更するための装置において、

第 1 デモンプロセスの重要度を変更できることを第 1 メッセージが指示するかどうか

10

20

30

40

50

決定する手段と、

前記第1デーモンプロセスの重要度を変更できることを前記第1メッセージが指示する場合には、前記第1デーモンプロセスの重要度を変更する手段と、

前記第1メッセージを前記第1デーモンプロセスへ転送する手段と、
を備えた装置。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、一般的に、プロセス管理に関するもので、より詳細には、受信したメッセージに基づいてプロセスの重要度を変更することに関する。

10

【0002】

関連出願：本願は、2012年6月8日に提出された同時係争中の米国特許出願第61/657,611号の優先権の利益を主張するもので、これは、参考として、ここにそのまま援用される。

【背景技術】

【0003】

移動装置は、ユーザと相互作用し且つその移動装置における多数のサービスを遂行するのに使用される複数の実行プロセスを有する装置である。例えば、移動装置は、前景において実行される1つのユーザアプリケーションプロセスを有し、ユーザは、このプロセスと相互作用することができる。他の実行プロセスは、ユーザアプリケーションプロセス又は他の実行プロセスにサービスを提供するデーモンプロセスである。これら他のプロセスは、中央処理ユニット(CPU)、メモリ、入力/出力、等の、移動装置のハードウェアリソースを消費する。これら消費されるリソースは、ユーザアプリケーション作用に必要なリソースと競合し、その結果、ユーザアプリケーションプロセスに適用されるリソースが非常に僅かになるので、ユーザインターフェイスの異常、装置の動きがのろいと感じること、タッチに対する反応が遅いこと、及び/又は装置がロックすることが生じる。

20

【発明の概要】

【発明が解決しようとする課題】

【0004】

この問題を軽減する1つの仕方は、デーモン及び/又は他の背景プロセスをユーザアプリケーションプロセスより低い重要度で実行させることである。これらの他のプロセスをユーザアプリケーションプロセスより低い重要度で実行させることで、ユーザアプリケーションプロセスは、より多くの移動装置リソースを消費することができ、従って、移動装置は、ユーザにとって、より良好な応答性をもつように見える。このシナリオに伴う問題は、ユーザアプリケーションプロセスがサービス又は情報についてそれらのデーモン又は他の背景プロセスの1つに要求を發するとき、それらのデーモン又は背景プロセスが低い重要度で要求に応じることである。これは、サービス要求を満足させるために、ユーザアプリケーションプロセスを待機させることになる。

30

【課題を解決するための手段】

【0005】

デーモンプロセスの重要度を変更する方法及び装置について説明する。規範的な実施形態において、装置は、デーモンプロセスに向けられたユーザプロセスからのメッセージを受信し、デーモンプロセスは、ユーザプロセスとは独立して実行され、そして第1のデーモンプロセスは、他の実行プロセスとメッセージを通信する。更に、装置は、第1のデーモンプロセスの重要度を変更できることを第1のメッセージが指示するかどうか決定する。第1のデーモンプロセスの重要度を変更できることを第1のメッセージが指示する場合には、装置が第1のデーモンプロセスの重要度を変更する。装置は、更に、第1のメッセージを第1のデーモンプロセスに転送する。

40

【0006】

更に別の実施形態において、装置は、適応デーモンに向けられた重要度ドナー(donor)

50

プロセスからのメッセージを受信し、適応デーモンは、まだ起動されねばならないものである。更に、装置は、適応デーモンプロセスの重要度を変更できることをメッセージが指示するかどうか決定する。適応デーモンプロセスの重要度を変更できることをメッセージが指示する場合には、装置は、異なる重要度、即ち適応デーモンプロセスに関連したデフォルト重要度で適応デーモンプロセスを起動する。起動された適応デーモンプロセスには、更にメッセージが転送される。

【0007】

他の方法及び装置についても説明する。

【0008】

本発明は、同様の要素が同じ参照番号で示された添付図面に一例として示されるが、これに限定されるものではない。

10

【図面の簡単な説明】

【0009】

【図1】ユーザアプリケーションプロセスが低い重要度で実行されるデーモンプロセスと通信する移動装置の1つの実施形態のブロック図である。

【図2】ユーザアプリケーションプロセスが、低い重要度及び高い重要度の混合状態で実行されるデーモンプロセスと通信する移動装置の1つの実施形態のブロック図である。

【図3A】ユーザアプリケーションプロセスがデーモンプロセスと通信しそしてデーモンプロセスの重要度が昇格される移動装置の1つの実施形態のブロック図である。

【図3B】ユーザアプリケーションプロセスがデーモンプロセスと通信しそしてデーモンプロセスの重要度が昇格される移動装置の1つの実施形態のブロック図である。

20

【図4】デーモンに向けられたメッセージに基づいてデーモンの重要度を昇格するのに使用されるメッセージを送信するためのカーネルのプロセスの一実施形態のフローチャートである。

【図5】デーモンに向けられたメッセージに基づいてデーモンの重要度を昇格するのに使用されるメッセージを受信するためのカーネルのプロセスの一実施形態のフローチャートである。

【図6】デーモンの重要度を昇格するのに使用されるメッセージを受信するための適応デーモンのプロセスの一実施形態のフローチャートである。

【図7】重要度アサーションをドロップするためのカーネルのプロセスの一実施形態のフローチャートである。

30

【図8】オンデマンドでサービスを起動するためのプロセスの一実施形態のフローチャートである。

【図9】適応デーモンプロセスの重要度を昇格及び降格するシステムの一実施形態のブロック図である。

【図10】第1のデーモンにより受信されたメッセージに基づき第2のデーモンの重要度を昇格するためのプロセスの一実施形態のフローチャートである。

【図11】第1のデーモンプロセスが第2のデーモンプロセスと通信しそして第2のデーモンプロセスの重要度が昇格される移動装置の一実施形態のブロック図である。

【図12】プロセスによるメッセージの送信及び受信を容易にするためのメッセージサービスモジュールである。

40

【図13】デーモンにより受信されたメッセージに基づきデーモンの重要度を昇格するのに使用されるメッセージを送信するためのメッセージ送信モジュールである。

【図14】デーモンのデーモン重要度を昇格するのに使用されるメッセージを受信するためのメッセージ受信モジュールである。

【図15】デーモンにより受信されたメッセージに基づきデーモンの重要度を昇格するのに使用されるメッセージを受信するためのユーザ受信モジュールである。

【図16】重要度アサーションをドロップするためのドロップアサーション重要度モジュールである。

【図17】オンデマンドでサービスを起動するための起動サービスモジュールである。

50

【図18】第1のデーモンにより受信されたメッセージに基づき第2のデーモンの重要度を昇格するためのメッセージ処理モジュールである。

【図19】ここに述べる実施形態に関連して使用される典型的なコンピュータシステムの一例を示す。

【図20】本発明の1つの実施形態に使用されるデータ処理システムの一例を示す。

【発明を実施するための形態】

【0010】

デーモンプロセスの重要度を変更する方法及び装置について説明する。以下の記述では、本発明の実施形態を完全に説明するために多数の特定の細部について述べる。しかしながら、当業者であれば、それらの特定の細部がなくても、本発明の実施形態が具現化されることが明らかであろう。他の点について、この説明の理解を不明瞭にしないために良く知られたコンポーネント、構造及び技術は、詳細に示さない。

10

【0011】

明細書において「1つの実施形態」又は「一実施形態」を言及するときには、その実施形態に関して述べる特定の特徵、構造又は特性が本発明の少なくとも1つの実施形態に含まれることを意味する。明細書の種々の場所に「1つの実施形態」又は「一実施形態」が現れるときには、必ずしも全てが同じ実施形態を指していない。

【0012】

以下の説明及び特許請求の範囲において、「結合(coupled)」及び「接続(connected)」並びにそれから派生する語が使用される。これらの語は、互いに同意語として意図されていないことを理解されたい。「結合」は、2つ以上の要素が互いに直接物理的又は電氣的に接触してもしなくてもよいが、互いに協働し又は相互作用することを指示するのに使用される。「接続」は、互いに結合された2つ以上の要素間での通信の確立を指示するのに使用される。

20

【0013】

添付図面に示すプロセスは、ハードウェア(例えば、回路、専用ロジック、等)、ソフトウェア(例えば、汎用コンピュータシステム又は専用マシンで実行される)、又は両方の組み合わせを含む処理ロジックにより遂行される。これらのプロセスは、幾つかの順次のオペレーションに関して以下に述べるが、それらのオペレーションの幾つかは、異なる順序で遂行されてもよいことが明らかであろう。更に、幾つかのオペレーションは、順次ではなく、並列に遂行されてもよい。

30

【0014】

「サーバー」「クライアント」及び「装置」という語は、サーバー、クライアント及び/又は装置の特定フォームファクタを特別に指すのではなく、データ処理システムを一般的に指すことを意図している。

【0015】

デーモンプロセスの重要度を変更する方法及び装置について説明する。1つの実施形態において、装置は、重要度の高いプロセスから適応デーモンプロセスへ送信されるメッセージを装置のカーネルで検出させることにより、デーモンプロセスの重要度を変更することができる。適応デーモンプロセスとは、その適応デーモンプロセスが実行される環境に基づき重要度を上げ下げすることのできるデーモンプロセスである。この実施形態では、カーネルは、このメッセージを検出し、そしてメッセージを重要度でマークする。更に、カーネルは、このデーモンプロセスが実行される場合には適応デーモンプロセスの重要度を昇格し、或いは適応プロセスがカーネルからのメッセージを受信する前に、適応デーモンを高い重要度で起動することができる。適応デーモンプロセスがメッセージを受信する前に適応デーモンプロセスの重要度を昇格させることで、適応デーモンプロセスは、メッセージを受信して、高い重要度のプロセスとして処理することができる。更に、カーネルは、適応デーモンプロセスに対するアサーションを行う。

40

【0016】

適応デーモンプロセスは、それがメッセージを受信した後、そのメッセージを処理し、

50

そしてメッセージに対する応答を初期メッセージの発信者へ送信する。更に、適応デーモンプロセスは、適応デーモンプロセスにおけるアサーションをドロップする。カーネルは、適応デーモンプロセスにおけるペンディング中アサーションの数をデクリメントする。適応デーモンプロセスに対するペンディング中アサーションの数がゼロである場合には、カーネルは、適応デーモンプロセスの重要度をその適応デーモンプロセスのデフォルト重要度へ降格する。1つの実施形態では、デーモンのデフォルト重要度とは、昇格が存在しないか又は昇格された重要度での起動が存在しないときにデーモンが有する重要度である。

【0017】

図1は、ユーザアプリケーションプロセスが低重要度で実行されるデーモンプロセスと通信する移動装置100の1つの実施形態のブロック図である。1つの実施形態において、移動装置100は、スマートフォン、ラップトップ、パーソナルデジタルアシスタント、音楽再生装置、ゲーム装置、等である。装置は、ここでは移動装置として述べるが、ここに述べる本発明は、パーソナルコンピュータ、サーバー、又は複数のプロセスを実行する他の装置にも使用することができる。図1において、移動装置100は、コンピュータハードウェアリソースを管理するのに使用されるソフトウェアのセットであり且つアプリケーションプログラムのような他の実行コンピュータプログラムのための共通のサービスを提供するオペレーティングシステム102を備えている。1つの実施形態において、オペレーティングシステム102は、移動装置100のプロセッサが各実行プロセスを実行するときにタイムスケジューリングにより異なる実行プロセスを管理する。1つの実施形態において、プロセスは、実行されているコンピュータプログラムのインスタンスである。この実施形態では、プロセスは、ユーザ入力の結果として実行されるユーザアプリケーションである。プロセスの別の例は、ユーザアプリケーション、別のデーモン、他のプロセス、等に1つ以上のサービスを提供するデーモンプロセスである。例えば、1つの実施形態では、デーモンプロセスは、ワイヤレスホットスポットサービスの状態、インストールされたアプリケーションのリストを与え、サーチを容易にし、装置電力設定を監視及び調整し、等を行う。1つの実施形態において、デーモンプロセスは、移動装置の特定のハードウェアリソースに結び付けられていない。

【0018】

1つの実施形態において、あるプロセスは、他のプロセスよりも重要度が高いか低い。1つの実施形態において、プロセスの重要度は、装置と現在相互作用しているユーザに代わってアクションが必要であることを指示するためにタスク/プロセスに関連したプロパティの集合である。それは、ユーザがそれらアクションの完了を待機していることを指示し、このタスク/プロセスは、緊急に処理されるべきものである。プロセスの重要度は、プロセスに対する中央処理ユニットのスケジュールプライオリティ、入力/出力スケジュール、ネットワーク帯域巾プライオリティ、メモリ管理、グラフィックリソースの使用、等、及び/又はその組み合わせを含む。プロセスプライオリティとは、実行する必要がある多数のタスクがあるときにタスク又はスレッドを動作(又は実行)のために取り上げる緊急性を指示するプロパティである。このプロセスプライオリティは、プロセスを実行すべきときをスケジュールするためにオペレーティングシステムにより使用される。1つの実施形態では、他のプロセスより重要度が高いプロセスは、多量の移動装置リソース(例えば、プロセッサリソース、等)を消費する。この実施形態では、重要度の高いプロセスは、重要度の低いプロセスより高速で実行するように見える。1つの実施形態では、重要度は、最低重要度から最高重要度まである範囲の数字に基づいて指定される。例えば、1つの実施形態では、重要度の範囲は、1から10まで、0-127、低/高、又は値の別の範囲である。

【0019】

例えば、1つの実施形態では、次の移動装置リソース使用がプロセス重要度により影響される。1. プロセススケジューリング: 上述したように、高及び低重要度プロセスには、低重要度のプロセスが高重要度のプロセスを妨げないように異なるプロセススケジュー

10

20

30

40

50

リング「プライオリティ」が指定される。

【 0 0 2 0 】

2. ディスクスケジューリング：1つの実施形態では、システム記憶ユニット（例えば、磁気ハードドライブ及びソリッドステートハードディスクユニット）にアクセスするときに、高重要度のプロセスには、低重要度プロセスに勝るプライオリティが与えられる。更に、低重要度プロセス（即ち、背景状態のプロセスに関連したスレッド）は、システム記憶ユニットのアクセスにおいてレート制限される。

【 0 0 2 1 】

3. 着信ネットワークアクティビティ：1つの実施形態において、着信ネットワークトラフィックは、低重要度プロセスに向けられた幾つかの又は全てのパケットをドロップすることにより緩和される。更に、人為的に小さなバッファサイズが遠方のネットワークサイトに報告され、着信するTCP（送信コントロールプロトコル）トラフィックを絞る。当業者であれば、TCPは、バッファサイズに関連したトラフィックフローコントロールメカニズムの既存セットを有し、受取者が取り扱える以上に高速でトラフィックがネットワーク上に送られないようにする。上述したように、1つの実施形態では、送信者が着信トラフィックのフローを低速化し又は停止するように、人為的に小さなバッファサイズが報告される。

【 0 0 2 2 】

4. 発信ネットワークアクティビティ：1つの実施形態において、ネットワークアクセスは、2つのキューを使用することで緩和され、その1つは、高重要度プロセスに関連したジョブに対するものであり、そしてもう1つは、低重要度プロセスに関連したジョブに対するものである。低重要度キューに勝る高重要度キュープライオリティを与えるのが有益であることが分かった。例えば、高重要度キューは、背景キューに勝る100%プライオリティが与えられる。この設定では、低重要度キューのジョブがサービスされる唯一の時間は、高重要度キューが空の場合である。別の実施形態では、高重要度キューのジョブは、低重要度キューからのジョブと希望の比率、例えば、90/10、80/20、75/27又は50/50でサービスされる。更に別の実施形態では、ジョブがキュー（例えば、高重要度キュー）に入れられると、ジョブ関連プロセスがその後異なる動作状態（例えば、低重要度）に入れられる場合には、それが移動されない。更に別の実施形態では、ジョブのキュー位置が、その関連プロセスの動作状態を反映するように更新される。

【 0 0 2 3 】

5. ハードウェア制約：1つの実施形態において、低重要度プロセスは、あるシステムハードウェアリソースへのアクセスを得ることが防止される。低重要度プロセスに利用できない例示的なハードウェアは、カメラ、GPU、加速度計、ジャイロスコープ、メモリ消費、接近センサ及びマイクロホンを含むが、これに限定されない。

【 0 0 2 4 】

1つの実施形態では、ユーザアプリケーションプロセス104は、前景において実行することができ、一方、1つ以上のデーモンプロセス106A-Nは、ユーザアプリケーションプロセス104より低い重要度で背景において実行される。前景で実行されるプロセスは、ユーザと相互作用するプロセスであり、そしてユーザと相互作用せずに実行されるプロセスは、背景のプロセスである。例えば、1つの実施形態では、移動装置100において前景で実行されるユーザアプリケーションプロセス104は、移動装置100に表示されるユーザアプリケーションである。この実施形態では、移動装置100は、移動装置に表示されずに他のプロセスと通信する1つ以上のデーモンプロセスを同時に実行することができる。これらのデーモンプロセスは、背景において実行され、ユーザには直接見えない。

【 0 0 2 5 】

1つの実施形態において、前景の低重要度プロセスは、異なるプライオリティで実行される。前景のプロセスは、ユーザと相互作用できるので、前景のプロセスは、低重要度プロセスであるデーモンプロセスより高い重要度で実行される。前景のプロセスを高い重要

10

20

30

40

50

度で実行させることで、前景のプロセスは、より多くのプロセッサリソースを得、そして前景のプロセスは高速で実行されるので前景のプロセスには良好なユーザ経験が与えられる。

【0026】

1つの実施形態では、オペレーティングシステム102は、ユーザアプリケーションプロセス104及びデーモンプロセス106A-Nを含む。この実施形態では、ユーザアプリケーションプロセス104は、高重要度プロセス108であり、そしてデーモンプロセス106A-Nは、低重要度プロセス110である。更に、ユーザアプリケーションプロセス104は、低重要度で実行される個々のデーモンプロセス106A-Nに比して移動装置100のリソースの高いシェアを得る。ユーザアプリケーションプロセス104を高い重要度で実行させることにより、ユーザアプリケーションプロセス104は、ユーザにとって高速で実行されるように見える。

10

【0027】

上述したように、この構成の1つの問題は、ユーザアプリケーションプロセス104が時々デーモンプロセス106A-Bから1つ以上のサービスを要求できることである。この実施形態では、デーモンプロセス106A-Bが低重要度で実行されるので、デーモンがその要求にサービスするには時間がかかる。ユーザアプリケーションプロセス104が低重要度で実行される1つ以上のデーモンプロセス106A-Bからサービスを要求することにより、ユーザアプリケーションプロセス104は、デーモン106A-Bが要求にサービスする時間中は、低重要度で実行されるように見える。例えば、1つの実施形態では、ユーザアプリケーションプロセス104は、デーモンから移動ホットスポットの状態を要求し、電力管理を調整及び報告し、サーチを遂行し、等々である。

20

【0028】

図2は、ユーザアプリケーションプロセスが、低重要度及び高重要度の混合状態で実行されるデーモンプロセスと通信する移動装置の1つの実施形態のブロック図である。図2及び1つの実施形態において、移動装置200のオペレーティングシステム202は、高重要度プロセス208の1つとして実行されるデーモンプロセス212を有する。更に、オペレーティングシステム202は、低重要度プロセス210として実行される他のデーモンプロセス206A-N、及びユーザアプリケーションプロセス204を含む。更に、ユーザアプリケーションプロセス204は、低重要度プロセスデーモン206A-B及び高重要度プロセスデーモン212へサービス要求を発する。

30

【0029】

デーモンプロセス212を高重要度プロセス208の1つとして実行させることで、ユーザアプリケーションプロセス204は、デーモンプロセス212にサービス要求を発することにより性能低下を経験しない。このデーモンプロセス212については、デーモンサービス要求を発することで、ユーザアプリケーションプロセス204に対する性能低下が軽減される。しかしながら、このデーモン212は、オペレーティングシステム202で実行される他のデーモン又は他のプロセスからのサービス要求も処理するので、別の問題が生じる。他のサービス要求を処理することにより、高重要度のデーモンプロセス212は、移動装置200のより多くのリソースを受け取り、ユーザアプリケーションプロセス204に利用できるリソースの量を減少する。この問題は、高重要度プロセス208として実行されるデーモンプロセスの数が増加するにつれて悪化する。更に、この状況は、低重要度のデーモンが低重要度のサービス要求を取り扱う問題からも生じる。

40

【0030】

図1及び2は、全てが低重要度プロセスとして実行されるか又は低及び高プライオリティ混合プロセスとして実行されるデーモンプロセスへサービス要求を発することによりユーザアプリケーションプロセスの性能が低下する問題を示している。1つの実施形態では、デーモンプロセスの重要度をユーザアプリケーションプロセスサービス要求に対して高重要度へ昇格させそして他のサービス要求については低重要度で実行させることにより、ユーザアプリケーションプロセスに優れた性能をもたせることができる。これは、ユーザ

50

アプリケーションプロセスがデーモンプロセスにサービス要求を発するとき、そのサービス要求が高重要度で処理されるからである。サービス要求を高プロセス重要度で処理させることで、ユーザアプリケーションプロセスは、更に迅速に実行されるように見える。他のサービス要求については、デーモンは、低重要度で実行され、そして他の機能についてユーザアプリケーションプロセスの性能を低下させない。

【 0 0 3 1 】

1つの実施形態では、プロセスは、1つ以上のスレッドを含み、スレッドとは、オペレーティングシステムによりスケジュールできる最小単位の処理である。同じプロセス内に複数のスレッドが存在して、プロセスリソース（例えば、メモリ、インストラクション、コンテキスト、等）をシェアしている。1つの実施形態では、プロセスの重要度が変更された場合に、全てのスレッドがその変更された重要度で実行される。この実施形態では、メッセージの受信又はメッセージ返事の送信に応答して重要度が昇格又は降格されたときに、そのプロセスの全ての現在実行スレッドが新たな重要度で実行される。別の実施形態では、幾つかのスレッドがその変更された重要度で実行される。

10

【 0 0 3 2 】

図3Aは、ユーザアプリケーションプロセス304がデーモンプロセス306Aと通信しそしてデーモンプロセス306Aの重要度が昇格される移動装置300の1つの実施形態のブロック図である。図3Aにおいて、移動装置300は、ユーザアプリケーションプロセス304及びデーモンプロセス306A-Nの実行を管理するオペレーティングシステム302を備えている。更に、ユーザアプリケーションプロセス304は、高重要度で実行され、そしてデーモンプロセスは、最初、低重要度で実行される。ユーザアプリケーションプロセス304がデーモンプロセスからのサービスを要求するとき、デーモンプロセスは、このサービス要求を処理するために高重要度へ昇格される。1つの実施形態では、このデーモンプロセス306Aは、このプロセスの重要度を、プロセスが作用する環境に適應できるので、適應デーモンである。例えば、1つの実施形態では、ユーザアプリケーションプロセス304は、メッセージ312を経てデーモンプロセス306Aにサービス要求を発する。1つの実施形態では、メッセージは、プロセス間通信（IPC）メッセージである。メッセージがデーモンに対してキューされるのに応答して、デーモンの重要度は、高重要度デーモン316へ昇格される。1つの実施形態では、昇格されたデーモン316の重要度が要求側ユーザアプリケーションプロセス304と同じであるが、別の実施形態では、デーモン316の昇格された重要度が要求側ユーザアプリケーションプロセス304の重要度より高くても低くてもよい。

20

30

【 0 0 3 3 】

更に、ユーザアプリケーションプロセス304は、メッセージをデーモンプロセス306Aへ送信した後に、そのメッセージの返事をデーモンプロセス306Aから受信するまで、実行を続ける。例えば、1つの実施形態では、ユーザアプリケーションプロセス304がデーモンプロセス306Aへメッセージを送信した後に、ユーザアプリケーションプロセス304がそのメッセージの返事を受信するまで、ユーザアプリケーションプロセス304は、ユーザから入力を受け取り、ユーザアプリケーションプロセス304の表示を更新し（例えば、スクロール、リサイズ、回転、等）、及び/又は他の機能を遂行する。別の例として、ユーザアプリケーションプロセス304は、デーモンプロセス306A/316から返事を待つ間に停止しない。

40

【 0 0 3 4 】

1つの実施形態において、デーモンプロセス306Aは、ユーザアプリケーションプロセス304がデーモンプロセス306Aへサービス要求メッセージを送信するときに実行される。この実施形態では、デーモンプロセス306Aは、ユーザアプリケーションプロセス304とは独立して実行され、そしてユーザアプリケーションプロセス304、他のユーザアプリケーションプロセス、他のデーモン、他のプロセス及び/又はその組み合わせからのサービス要求を取り扱う。更に、デーモンプロセス306Aは、ユーザアプリケーションプロセス304がスタートする前又はその後にスタートする。例えば、1つの実

50

施形態において、デーモンプロセス306Aは、移動装置がブートアップするときにスタートし、そしてユーザアプリケーションプロセス304は、移動装置がブートした後にスタートする。この例では、デーモンプロセス306Aは、ユーザアプリケーションプロセス304とは独立して移動装置で実行される1つ以上のプロセスからのサービス要求メッセージを処理する。

【0035】

別の実施形態では、デーモンプロセス306Aは、ユーザアプリケーションプロセス304がメッセージを送信するときに実行されない。この実施形態では、デーモンプロセス306Aは、このデーモンプロセス306Aより通常は高い重要度のオンデマンドで起動される。オンデマンドの起動は、図8を参照して以下に更に説明する。1つの実施形態では、起動サービス326は、デーモンのオンデマンド起動328を取り扱う。この実施形態では、サービス要求メッセージが起動サービス326へ転送され、起動サービス326は、デーモン306A/316を高重要度プロセス316として起動し、このデーモンは、通常、低重要度プロセス306Aとして起動するものである。

10

【0036】

図3A及び1つの実施形態において、オペレーティングシステムカーネル320は、デーモンプロセス重要度昇格を取り扱う。別の実施形態では、オペレーティングシステムカーネル320は、プロセス管理モジュール322を使用してデーモンプロセス重要度昇格を取り扱う。この実施形態では、プロセス管理モジュール322は、ユーザアプリケーションプロセス304とデーモンプロセス306A/316との間のメッセージの送信及び受信を取り扱う。例えば、1つの実施形態では、プロセス管理モジュール322は、このプロセス管理モジュール322がデーモンプロセス306A/316のためのメッセージをエンキューするときにデーモンプロセス306A/316を昇格させる。これら2つのプロセス間のメッセージ通過を取り扱うことにより、プロセス管理モジュール322は、デーモンプロセスの昇格を管理することができる。更に、デーモンプロセス306A/316は、受信したメッセージを処理するデーモンmsg_recvモジュール324を含む。受信したメッセージの処理は、図6において以下で更に説明する。

20

【0037】

1つの実施形態において、デーモンがメッセージの処理を完了した後、デーモンの重要度が最初のデーモン重要度に戻される。図3Bは、ユーザアプリケーションプロセス304が昇格されたデーモンプロセス316と通信しそしてデーモンプロセス316の重要度が降格される移動装置300の1つの実施形態のブロック図である。図3Bにおいて、移動装置300は、ユーザアプリケーションプロセス304、デーモンプロセス306B-N、及び昇格されたデーモンプロセス316の実行を管理するオペレーティングシステム302を含む。図3Aと同様に、ユーザアプリケーションプロセス304は、高重要度で実行され、デーモンプロセス306B-Nは、最初、低重要度で実行され、そしてデーモン306Aは、高重要度で実行されるように昇格され、これは、デーモンプロセス(昇格)316として示されている。1つの実施形態では、昇格されたデーモンプロセス316は、メッセージを処理し、そして返事をユーザアプリケーションプロセス304へ送信する。この実施形態では、メッセージの送信に回答して、昇格されたデーモンプロセス316を、その最初の重要度306Aへ降格することができる。更に、デーモンプロセス316の降格は、デーモンプロセス306Aの重要度が昇格されたが、その昇格された重要度で実行され又は起動される場合にも、行われる。プロセスの降格については、以下で更に述べるが、昇格されたデーモンプロセス316は、アサーションをドロップさせ、このプロセス316が降格されるように導く。例えば、プロセスの降格は、図6において以下に更に説明する。

30

40

【0038】

図3Bにおいて、1つの実施形態では、オペレーティングシステムカーネル320は、デーモンプロセスの降格を取り扱う。別の実施形態では、オペレーティングシステムカーネル320は、プロセス管理モジュール322を使用してデーモンプロセスの昇格を取り

50

扱う。この実施形態では、プロセス管理 3 2 2 は、ユーザアプリケーションプロセス 3 0 4 とデーモンプロセス 3 0 6 A / 3 1 6 との間のメッセージの送信及び受信を取り扱う。これら 2 つのプロセス間のメッセージ通過を取り扱うことで、プロセス管理モジュール 3 2 2 は、デーモンプロセスの降格を管理することができる。更に、デーモンプロセス 3 0 6 A / 3 1 6 は、受信したメッセージを処理するデーモン `msg_recv` モジュール 3 2 4 を含む。受信したメッセージの処理は、図 6 において以下に更に説明する。

【 0 0 3 9 】

1 つの実施形態において、昇格されたデーモンプロセス 3 1 6 がアサーションをドロップさせることでプロセスの降格が生じる。例えば、1 つの実施形態において、デーモンプロセスがカーネルへシステムコールを発することでプロセスアサーションドロップが生じる。このシステムコールは、デーモンプロセスがユーザアプリケーションプロセスに代わって仕事を完了したことをカーネルに指示する。この実施形態では、アサーションは、メッセージが処理されて、プロセスを降格できることをオペレーティングシステムに通知する。1 つの実施形態では、オペレーティングシステムは、このデーモンの昇格のためにペンディング中アサーションの数をデクリメントする。この実施形態では、デーモンプロセスの重要度を昇格すべきであることを指示する他のアサーションがない場合には、昇格されたデーモンプロセス 3 1 6 の重要度がデフォルト重要度へ降格される。或いは又、このデーモンに対する他のペンディング中アサーションがある場合には、オペレーティングシステムは、このデーモンの昇格に対するペンディング中アサーションの数をデクリメントする。1 つの実施形態では、デーモンプロセスの重要度が、デーモンプロセスのデフォルト重要度より低くすることも高くすることもできる。1 つの実施形態では、デフォルト重要度は、重要度が変更されないプロセスの重要度である。

10

20

【 0 0 4 0 】

例えば、1 つの実施形態において、いつ好みのアプリケーションを起動するか及び移動装置のネットワーク設定をチェックするようにユーザがナビゲートするかの一例について考える。この例では、移動インターネットシェアデーモンは、デーモンプロセス 3 0 6 A により表され、そして好みのアプリケーションは、ユーザアプリケーションプロセス 3 0 4 である。好みのアプリケーションは、パーソナルホットスポットサービスに関する状態情報を問合せするためにメッセージを移動インターネットシェアデーモン（例えば、デーモンプロセス 3 0 6 A）へ送信する。移動インターネットシェアデーモンは、このメッセージを受け取ると、そのキャリアに対してテザリングが利用できるかどうか及びユーザがその使用について許可されていることを決定する。移動インターネットシェアデーモンは、その情報を好みのアプリケーションへ返送し、このアプリケーションは、パーソナルホットスポットサービスに対する適切な状態を表示する。

30

【 0 0 4 1 】

1 つの実施形態において、移動インターネットシェアデーモンは、好みのアプリケーションがこのデーモンにメッセージを送信するときに実行される。この実施形態では、このデーモンのためのサービスポートは、移動インターネットシェアデーモンを伴う。好みのアプリケーションからメッセージが送信される際には、カーネルは、ポートへメッセージをキューする部分として、メッセージがユーザアプリケーションからのものでありそして適応デーモンへ向けられることを通知する。カーネルは、アサーションをテークし、そしてこのデーモンが低重要度（例えば、背景重要度）で実行される場合には移動インターネットシェアデーモンを高重要度へ昇格させる。更に、メッセージは、そのメッセージがカーネルにアサーションを保持させたことを指示するためにマークされる。移動インターネットシェアデーモンは、このメッセージを受信すると、その特定のメッセージがカーネルによりアサーションを生じさせたかどうかチェックし、もしそうであれば、処理及び返事の後、デーモンは、アサーションをドロップすることをカーネルに告げる。アサーションがドロップすると、他のアサーションがペンディングでない状態では、カーネルは、移動インターネットシェアデーモンをこのプロセスのデフォルト重要度（例えば、背景

40

50

重要度)へ降格させる。

【0042】

別の実施形態では、移動インターネットシェアデーモンが実行されず、このデーモンプロセスは、デーモンの1つのデフォルト重要度とは異なる重要度でオンデマンド起動される。例えば、1つの実施形態では、デーモン、上述した移動インターネットシェアデーモンについて考える。移動インターネットシェアデーモンが現在実行中でない場合には、移動インターネットシェアデーモンのサービスポートが起動デーモンと共に記憶される。このポートがメッセージ(例えば、イベントメッセージ)を有するとの通知を起動デーモンが得ると、起動デーモンは、スポン(spawn)、即ち適応デーモンである移動インターネットシェアデーモンをスタートするプロセスを呼び出す。起動のための引数の1つとして、起動デーモンは、カーネルへの当該ポートにハンドルを与える。オペレーティングシステムカーネルは、ペンディング中メッセージがユーザアプリケーションプロセスからのデーモンサービスポートを含み、更に、デーモンプロセスを昇格させるアサーションを搬送するかどうか決定する。この形式のメッセージが存在する場合には、カーネルは、移動インターネットシェアデーモンの起動を昇格された重要度でスタートさせる。1つの実施形態では、昇格された重要度は、デフォルトである移動インターネットシェアデーモンの重要度より高い重要度であって、ユーザアプリケーションプロセスの重要度と同じ、それより高い又はそれより低い、の1つである。この実施形態では、カーネルは、そのときのプロセスにアサーションを保持する。デーモンのメッセージがアサーションを含まない場合には、移動インターネットシェアデーモンは、このプロセスのデフォルト重要度で起動される。例えば、1つの実施形態では、移動インターネットシェアデーモンは、背景重要度で起動される。

10

20

【0043】

移動インターネットシェアデーモンプロセスの起動が完了すると、移動インターネットシェアデーモンは、起動デーモンとランデブーして、移動インターネットシェアデーモンへ転送されるサービスポートを得る。移動インターネットシェアデーモンは、メッセージを読み取り、そしてそのメッセージがアサーションを引き起こしたかどうかチェックし、アサーションは、カーネルによりマークされる。メッセージがアサーションを生じた場合には、移動インターネットシェアデーモンがメッセージの処理を完了しそしてメッセージに対する返事をするときに、移動インターネットシェアデーモンは、アサーションをドロップする。ペンディング中アサーションがプロセスにおいてドロップされると、移動インターネットシェアデーモンは、デフォルトプロセス重要度(例えば、背景重要度)へ戻る。

30

【0044】

別の例として、ユーザが移動装置を使用してスタートすると、アプリケーションランチャーアプリケーションは、アイドルスリープを防止するためにパワーアサーションを保持するメッセージをパワーデーモンへ送信する。1つの実施形態では、パワーデーモンは、移動装置のバックライトがオンになるまでこのアサーションを保持する。この実施形態では、ユーザが装置をロックし、背景オーディオが再生されないときに、アプリケーションランチャーアプリケーションがパワーアサーションをドロップさせる。装置がある期間中使用されなかった後に、移動装置のバックライトがターンオフされる。アプリケーションランチャーアプリケーションは、パワーアサーションをドロップさせ、そしてカーネルが装置をアイドルスリープできるようにする。

40

【0045】

この例では、パワーデーモンが適応デーモンであり、これは、アプリケーションランチャーアプリケーションのような高重要度プロセスからのメッセージに応答して昇格できるデフォルト重要度をパワーデーモンが有することを意味する。これは、前景アプリケーションアクティビティへの影響を減少することができる。例えば、1つの実施形態では、アプリケーションランチャーアプリケーションからのメッセージがパワーデーモンに到着するときに、パワーデーモンは、高重要度へブーストされ、そしてパワーデーモンによる

50

メッセージの処理が完了すると、パワーデーモンは、パワーデーモンのデフォルト重要度（例えば、背景重要度）へドロップされる。この例は、適応デーモンがメッセージ処理時間中に通常の重要度で実行されて、前景アプリケーションに不必要に影響を及ぼさない場合を示している。この例は、更に、入力/出力、処理ユニット、等の著しく多くのリソースを消費したり競合したりせずに前景アプリケーションの応答性を改善するために適応デーモンがどのように役立つかも示す。更に、別のデーモンが、パワーデーモンの重要度を昇格することなくパワーデーモンへ要求を発することができる。このような他のデーモン要求は、パワーデーモンのデフォルト重要性で処理される。

【 0 0 4 6 】

更に別の例及び実施形態では、カーネルは、通信チャンネルの受信端（ポート）の所有権をプロセス間で移送する能力を有する。例えば、1つの実施形態では、この移送は、適応デーモンが起動中デーモンからサービスポートの所有権を要求するときに生じる。別の例及び別の実施形態では、適応デーモンAは、ポートPに対する受信権利を有する。この例では、ポートPにおいてエンキューされるN個の重要度寄贈(donating)メッセージがあり、従って、デーモンは、N個の重要度寄贈/ブースト基準を有する。適応デーモンがN個のメッセージを受信し、それらを処理し、そして各1つの後にアサーションをドロップする場合には、適応デーモンは、通常の適応重要度に復帰する。しかし、このケース/例では、N個のメッセージ全部を受信するのではなく、適応デーモンは、異なるポートQに向けられるメッセージのコンテンツにおいてポートPの受信端を送信することによりそのポートの受信端の所有権を別のプロセスBへ移行する。このケースでは、適応デーモンAは、N個のブーストを失い、そしてプロセスB（これも適応性であれば）は、受信権移送メッセージがエンキューされたときにN個のブーストを得る。

10

20

【 0 0 4 7 】

次いで、適応デーモンBが、ポートPの受信端を含むメッセージを受け取る前に、ポートQの受信端を他のプロセスへ移送する場合には、適応デーモンBは、ポートQにおける他のメッセージに対する重要度寄贈アサーションの数、及びポートPに関連したN個の重要度アサーションを移送しなければならない。このように、重要度寄贈アカウンティングは、これらポートのいずれかにおいてエンキューされるいずれかのメッセージの受信権利コンテンツに基づいて繰り返される。

【 0 0 4 8 】

別の例及び実施形態では、適応デーモンAは、それがアイドル状態であることを決定すると、より多くのシステムリソースをセーブするために退出することを選択する。それ又はこのデーモンはクラッシュするだけかもしれない。いずれの場合も、ポートPの受信権は、起動中デーモンへ移送されて戻される（例えば、オペレーティングシステムのプロセス間通信システムにおける内蔵バックアップシステムを経て）。この退出/クラッシュの直前に重要度寄贈メッセージがエンキューされた場合には、適応デーモンAは、実行を停止するときに重要度寄贈をもつ。この例では、カーネルは、適応デーモンのブーストが永久的に失われるのを希望しない。従って、起動中デーモンがポートPの受信端を得ると、起動中デーモンは、エンキューされたメッセージが既にあることを検出し、そして適応デーモンAの次のインスタンスを再起動する（例えば、A2と称される）。適応デーモンA2は、ポートが起動中デーモンへ返送された後に到達する重要度寄贈メッセージの量によるだけでなく、そのポートに既にエンキューされたメッセージによってもブーストされる必要がある。この「ポート移動による重要度更新」メカニズムは、これを生じさせる。

30

40

【 0 0 4 9 】

図4は、デーモンにより受信されるメッセージに基づきデーモンプロセス重要度を昇格するのに使用されるメッセージを送信するためのカーネルのプロセス400の一実施形態のフローチャートである。このプロセスは、ハードウェア（例えば、回路、専用ロジック、プログラム可能なロジック、マイクロコード、等）、ソフトウェア（例えば、ハードウェアシミュレーションを遂行するために処理装置で実行されるインストラクション）、又はその組み合わせを含む処理ロジックにより遂行される。1つの実施形態では、プロセス

50

400は、図3A又は3Bのカーネル320のようなカーネルによって遂行される。

【0050】

図4において、プロセス400は、ブロック402において、メッセージを送信する現在プロセスが重要度ドナーであるかどうか決定することにより開始する。1つの実施形態では、重要度ドナーは、ユーザアプリケーションプロセスであるプロセスで、ユーザは、アクションの完了を待機する。別の実施形態では、重要度ドナーは、ユーザアプリケーションプロセスに代わって仕事を行うデーモン又はサービスである。この実施形態では、デーモンは、昇格された重要度を有する。1つの実施形態では、重要度ドナーは、前景において実行されるユーザアプリケーションプロセスである。例えば、1つの実施形態において、重要度プロセスは、図3Aにおいて上述したユーザアプリケーションプロセス304

10

【0051】

現在プロセスが重要度ドナーである場合には、ブロック404において、プロセス400は、メッセージを受信するターゲットプロセスが適応デーモンプロセスであるかどうか決定する。ターゲットプロセスが適応デーモンプロセスでない場合には、実行がブロック412へ進む。ターゲットプロセスが重要度受取者である場合には、プロセス400は、ブロック406において、メッセージを重要度搬送とマークし、そしてターゲットプロセスにおいて「重要度アサーションアカウント」をインクリメントする。1つの実施形態では、「重要度アサーションアカウント」をインクリメントすることにより、プロセス400は、ターゲットプロセスにおいてアサーションをテークする。このアサーションアカウントは、アサーションの重要度を降格するかどうか決定するために後でカーネルにより使用される。この実施形態では、アサーションは、ターゲットプロセスによるメッセージの処理に関連して後でドロップされる。アサーションをドロップすることは、図6において以下で更に述べる。例えば、1つの実施形態において、ユーザがネットワーク状態を知りたいときに、ユーザ好みアプリケーションプロセスは、ネットワークホットスポットサービスの状態に関するメッセージを移動インターネットシェアデーモンへ送信する。以上の説明について、好みのアプリケーションは、ユーザアプリケーションプロセスであり、そして移動インターネットシェアデーモンは、適応デーモンであり、ここで、移動インターネットシェアデーモンへのメッセージは、重要度搬送とマークされ、そして移動インターネットシェアデーモンの重要度アサーションアカウントがインクリメントされる。

20

30

【0052】

ブロック408において、プロセス400は、ターゲットプロセスが低い重要度を有するかどうか決定する。1つの実施形態において、プロセス400は、ターゲットプロセスが現在プロセスより重要度が低いかどうか決定する。例えば、1つの実施形態において、プロセス400は、現在プロセス及びターゲットプロセスの重要度を比較する。

【0053】

プロセス400は、ブロック410において、ターゲットプロセスを昇格された重要度に変更させる。1つの実施形態において、プロセス400は、ターゲットプロセスを現在プロセスの重要度へ変更させる。この実施形態では、プロセス400は、ポートキューにおいてメッセージをエンキューするとき又はその前であって且つターゲットプロセスがメッセージ処理のためのメッセージをデキューする前にターゲットプロセス重要度を昇格させる。ターゲットプロセスがメッセージをデキューする前にターゲットプロセスを昇格させることにより、ターゲットプロセスは、より高い重要度でメッセージをデキューし処理することができる。例えば、1つの実施形態では、現在プロセスは、前景重要度で実行することができる。そしてプロセス400は、ターゲットプロセス重要度を前景重要度へ変更する。別の実施形態では、プロセス400は、ターゲットプロセス重要度を、現在プロセスの重要度とは異なる値に変更する。1つの実施形態では、プロセス400は、図3Aにおいて上述したターゲットプロセスの重要度を変更する。実行は、ブロック412へと続く。

40

【0054】

50

前記ブロック402で決定された現在プロセスが重要度ドナーでない場合には、プロセスは、ブロック414において、メッセージポートがサービスポートであるかどうか決定する。1つの実施形態において、現在プロセスが重要度ドナーでない場合には、現在プロセスは、オンデマンドで起動されるべきデーモンに対するメッセージを取り扱う起動デーモンである。この実施形態では、サービスポートは、そのサービスのためのターゲットプロセスへメッセージを通信するのに使用されるポートである。メッセージポートがサービスポートである場合には、ブロック416において、プロセス400は、メッセージを、重要度を搬送するものとしてマークし、そしてサービスポートにおいて「重要度アサーションアカウント」をインクリメントする。1つの実施形態では、サービスポートにおいて「重要度アサーションアカウント」をインクリメントすることにより、プロセス400は、ポートが適応デーモン又は別の重要度ドナーによりオンデマンドで起動されたことをマークする。実行は、ブロック412へ進む。メッセージポートがサービスポートでない場合には、実行がブロック412へ進む。

10

【0055】

ブロック412において、プロセス400はポート上でメッセージをキューする。1つの実施形態では、ポートは、ターゲットプロセスに使用されるポートである。1つの実施形態では、M a c hプロセス間通信ポートがサポートされ、他の実施形態では、他の形式のポート（例えば、ローカルドメインソケット、パイプ、等）、他の形式のIPC、等がサポートされる。

20

【0056】

図4は、デーモンプロセスの重要度を昇格するのに使用できるメッセージを送信するためにカーネルにより使用されるプロセスを示している。図5は、デーモンに向けられたメッセージに基づいてデーモンの重要度を昇格するのに使用されるメッセージを受信するためのカーネルのプロセス500の一実施形態のフローチャートである。このプロセスは、ハードウェア（例えば、回路、専用ロジック、プログラム可能なロジック、マイクロコード、等）、ソフトウェア（例えば、ハードウェアシミュレーションを遂行するために処理装置で実行されるインストラクション）、又はその組み合わせにより遂行される。1つの実施形態では、プロセス500は、図3A又は3Bのカーネル320のようなカーネルにより遂行される。

30

【0057】

図5では、プロセス500は、ブロック502において、現在プロセスに対してキューされたメッセージがあるかどうか決定することにより開始する。1つの実施形態では、プロセス500は、図3Aについて上述したデーモンプロセスのような現在プロセスに関連したポートに対してメッセージがキューされるかどうか決定する。1つの実施形態では、現在プロセスは、適応デーモンである。キューされたメッセージがない場合には、実行プロセス500が終了となる。

【0058】

メッセージがある場合には、プロセス500は、ブロック504において、ポートからメッセージをデキューする。1つの実施形態では、プロセス500は、そのメッセージをポートキューから除去する。プロセス500は、ブロック506において、メッセージが重要度についてマークされているかどうか決定する。1つの実施形態では、プロセス500は、メッセージが重要度についてマークされているかどうか決定するためにメッセージを検査する。例えば、1つの実施形態では、メッセージは、重要度ビットを含み、この重要度ビットがセットされた場合には、メッセージが重要度についてマークされている。メッセージが重要度についてマークされていない場合には、実行がブロック512へ進む。

40


【0059】

メッセージが重要度についてマークされている場合には、プロセス500は、ブロック508において、現在プロセスが適応デーモンプロセスであるかどうか決定する。現在プロセスが適応デーモンプロセスでない場合には、プロセス500は、ブロック510において、メッセージが重要度を保持することを指示するフラグを除去する。1つの実施形態

50





では、プロセス500は、メッセージの重要度ビットをアンセットする。更に、プロセス500は、ブロック510において、ポート重要度カウンタをデクリメントする。この実施形態では、ポート重要度カウンタは、重要度ドナープロセスからの未処理メッセージの数を指示するカウンタである。実行は、ブロック512へ進む。現在プロセスが適応プロセスである場合には、実行は、ブロック512へ進む。

【0060】

ブロック512において、プロセス500は、メッセージをユーザバッファにコピーする。1つの実施形態では、メッセージをユーザバッファにコピーすることにより、現在プロセスは、メッセージを処理することができる。1つの実施形態では、現在プロセスは、6について述べるようにメッセージを処理するためのプロセス600を遂行する。

10

【0061】

6は、デーモンの重要度を、そのデーモンにより受け取られるメッセージに基づいて昇格するのに使用されるメッセージを受信するための適応デーモンのプロセス600の一実施形態のフローチャートである。このプロセスは、ハードウェア（例えば、回路、専用ロジック、プログラム可能なロジック、マイクロコード、等）、ソフトウェア（例えば、ハードウェアシミュレーションを遂行するために処理装置で実行されるインストラクション）、又はその組み合わせを含む処理ロジックにより遂行される。1つの実施形態において、プロセス600は、3A又は3Bのデーモン306/316のような適応デーモンにより遂行される。6において、プロセス600は、メッセージが受信されたかどうか決定することにより開始する。1つの実施形態では、プロセス600は、デーモンプロセスにより使用されるユーザバッファにメッセージがあるかどうか決定する。プロセス600で処理すべきメッセージがない場合には、実行が終了となる。


20

【0062】

プロセス600で処理すべきメッセージがある場合には、プロセス600は、ブロック604において、メッセージが重要度についてマークされているかどうか決定する。1つの実施形態において、プロセス600は、上述した重要度ビットのような重要度フラグがセットされたかどうかチェックする。メッセージが重要度についてマークされていない場合には、実行は、ブロック608へ進む。メッセージが重要度についてマークされている場合には、プロセス600は、ブロック606において、ローカル変数を非ゼロ値にセットし、ここで、非ゼロ値は、メッセージが重要であるかどうか指示する。

30

【0063】



プロセス600は、ブロック608において、メッセージを処理する。1つの実施形態では、プロセス600は、要求された情報を収集するためのアクションを遂行する。例えば、1つの実施形態では、プロセス600は、移動インターネットシェアサービスの状態を決定し、電力管理を遂行し、サーチを遂行し、インストール済アプリケーションのリストを返送し、等を行うことができる。プロセス600は、ブロック610において、要求された情報で送信者に返事を送信する。1つの実施形態では、プロセス600は、3Bについて上述したようにユーザアプリケーションプロセスに返事を送信する。

【0064】

ブロック612において、プロセス600は、メッセージが重要度についてマークされたかどうか決定する。1つの実施形態では、プロセス600は、重要度フラグが真にセットされたかどうかチェックする。メッセージが重要度についてマークされた場合には、ブロック614において、プロセス600は、重要度アサーションをドロップさせるためにカーネルコールを発信する。重要度フラグがセットされず、メッセージが重要度についてマークされた場合には、実行が終了となる。

40

【0065】

6において、プロセス600は、重要度アサーションをドロップさせるコールを発信する。重要度アサーションをドロップさせることは、デーモンプロセスの重要度をデーモンプロセスのデフォルト重要度に戻すべきかどうか決定するためにカーネルにより使用される。7は、重要度アサーションをドロップさせるためのカーネルのプロセス700

50

の一実施形態のフローチャートである。このプロセスは、ハードウェア（例えば、回路、専用ロジック、プログラム可能なロジック、マイクロコード、等）、ソフトウェア（例えば、ハードウェアシミュレーションを遂行するために処理装置で実行されるインストラクション）、又はその組み合わせを含む処理ロジックにより遂行される。1つの実施形態において、プロセス700は、図3A又は3Bのカーネル320のようなカーネルにより遂行される。図7では、プロセス700は、ブロック702において、現在プロセスが適応デーモンであるかどうか決定することにより開始する。現在プロセスが適応デーモンでない場合には、実行が終了となる。

【0066】

プロセスが適応デーモンである場合には、プロセス700は、ブロック704において現在プロセスにおける「重要度アサーションカウント」をデクリメントする。上述したように、重要度アサーションカウントは、このプロセスに対して上昇された現在アサーションの数を追跡する。プロセス700は、ブロック706において、重要度アサーションカウントがゼロであるかどうか決定する。重要度アサーションカウントがゼロであることは、現在プロセスに対して上昇された現在アサーションがないことを意味する。重要度アサーションカウントがゼロである場合には、プロセス700は、ブロック708において、現在プロセスをデフォルト重要度に降格させる。ブロック706において、重要度カウントがゼロでない場合には、実行が終了となる。

10

【0067】

図8は、オンデマンドでサービスを起動させるプロセス800の一実施形態のフローチャートである。このプロセスは、ハードウェア（例えば、回路、専用ロジック、プログラム可能なロジック、マイクロコード、等）、ソフトウェア（例えば、ハードウェアシミュレーションを遂行するために処理装置で実行されるインストラクション）、又はその組み合わせを含む処理ロジックにより遂行される。1つの実施形態において、プロセス800は、図3A又は3Bの起動サービス326のような起動サービスにより遂行される。図8では、プロセス800は、ブロック802において、プロセス800で処理すべきペンディングイベントがあるかどうか決定することにより開始する。ペンディングイベントがない場合には、プロセス800は、ブロック802に戻り、他のペンディング事象を待機する。

20

【0068】

ペンディングイベントがある場合には、ブロック804において、プロセス800は、メッセージサービスを生成すべきことをイベントが指示するかどうか決定する。1つの実施形態では、メッセージサービスは、デーモン及びユーザアプリケーションの両方により使用されて、メッセージ経路/接続の確立、メッセージの交換、及びそれらの間の重要度寄贈プロパティを管理するためのサービスのセット（例えば、ライブラリー）である。これは、ユーザアプリケーションとデーモンとの間で情報を交換するためのアブストラクションを与え、そして更に、重要度寄贈をシームレスに取り扱う。例えば、1つの実施形態では、メッセージサービスは、サービス要求を取り扱い及び満足させるためにメッセージを送信及び受信することのできるサーチデーモンのようなデーモンである。メッセージサービスを生成すべきであることをイベントが指示する場合には、ブロック806において、プロセス800は、メッセージサービスのためのサービスポートを生成し、そしてそのサービスポートをウォッチリストに追加する。1つの実施形態では、サービスポートを生成し、そしてそのポートをウォッチリストに追加することにより、プロセス800は、生成されたポートにおいてこのサービスに対してメッセージが到達するかどうか決定することができる。実行は、ブロック802へ進み、プロセス800は、処理すべき別の事象を待機する。

30

40

【0069】

イベントが、メッセージサービスを生成すべきことを示すものでない場合には、ブロック808において、プロセス800は、起動すべきデーモンに対してサービスポートにメッセージがあるかどうか決定する。そのようなメッセージがある場合には、ブロック81

50

0において、プロセス800は、デーモンを起動して、そのデーモンにサービスポートを関連付けるためにカーネルにコールする。1つの実施形態では、プロセス800は、このサービスポートにとってどちらが適当なデーモンであるか決定し、そのデーモンを起動するためにカーネルにコールする。更に、プロセス800は、その起動されたデーモンに、メッセージサービスポートを関連させる。この実施形態では、カーネルは、サービスポートを検査し、そしてそのポートが重要度メッセージを有するかどうか決定する。重要度メッセージがあることをカーネルが決定すると、カーネルは、デーモンを高い重要度で起動する。サービスポートに重要度メッセージがない場合には、カーネルは、そのデーモンのデフォルト重要度でデーモンを起動する。更に、カーネルは、デーモン/サービスポートの関連性を知っているので、デーモンが通常の/低い重要度で起動され、そしてその後、重要度メッセージがこのサービスポートにおいてキューされた場合には、カーネルは、重要度メッセージをキューするときに、デーモンを高重要度に昇格させることができる。実行は、ブロック802へ進み、プロセス800は、処理すべき別の事象を待機する。

10

20

30

40

50

【0070】

ブロック808において、サービスポートにメッセージがない場合には、プロセス800は、ブロック812において、イベントが適応デーモンの起動後にそのデーモンのチェックインに対するものであるかどうか決定する。もしそうであれば、ブロック814において、プロセス800は、サービスポートを適応デーモンへ転送する。1つの実施形態では、これは、適応デーモンがそのポートにおいてメッセージを送信及び受信できるようにする。実行は、ブロック802へ進み、プロセス800は、処理すべき別の事象を待機する。イベントが適応デーモンチェックインでない場合には、ブロック816において、プロセス800は、他のイベントを処理する。

【0071】

上述したように、デーモンプロセスの重要度は、そのデーモンに対してエンキューされる重要度ドナーからのメッセージ、又は昇格された重要度で起動されるデーモンをトリガーする重要度ドナーからのメッセージに基づき、実行中デーモンに対して昇格することができる。デーモンの重要度が昇格されたとき、カーネルは、そのデーモンに対するアサーションをテークする。このデーモンプロセスは、ペンディングアサーションがゼロにドロップするときその重要度を降格させる。アサーションは、重要度でマークされたメッセージをデーモンが処理するのに応答してドロップされる。図9は、適応デーモンプロセスの重要度を昇格及び降格するシステム900の一実施形態のブロック図である。図9において、システム900は、ユーザアプリケーションプロセス902、適応デーモンプロセス904、及びカーネル906を含む。1つの実施形態では、ユーザアプリケーションプロセス902は、メッセージサービス940を使用して、適応デーモンプロセス904とでメッセージを送信及び受信する。この実施形態では、メッセージサービス940は、ユーザアプリケーションプロセス902と適応デーモンプロセス906との間でのメッセージの通信を管理するためにユーザアプリケーションプロセス902及び適応デーモンプロセス906にわたって作用するサービスである。1つの実施形態では、メッセージサービス940は、同期及び非同期メッセージ通信をサポートする。

【0072】

1つの実施形態において、ユーザアプリケーションプロセス904は、メッセージサービス940を使用して、メッセージを適応デーモンプロセス906へ送信する。例えば、1つの実施形態では、ユーザアプリケーションプロセス902は、パワーアサーションを保持してアイドルスリープを防止し、サーチを遂行し、等々のために、パーソナルホットスポットサービスの状態を要求する。この実施形態では、ユーザアプリケーションプロセス902は、ユーザアプリケーション接続918を使用してメッセージ920を生成することによりメッセージ922を送信する。このメッセージ920は、カーネル906へ送られ、カーネルは、`kernel_msg_send`を使用して、適応デーモンプロセス904に対してポートキュー912にメッセージ920をエンキューする。1つの実施形態では、ポートキューは、適応デーモンプロセス904と通信されるメッセージをキュー

するためにカーネル 906 により使用されるキューのセットである。1つの実施形態において、カーネル 906 の `kernel_msg_send_914` 機能は、メッセージに対して重要度 908 をセットし、そして適応デーモンプロセス 904 に対してアサーション 910 をテークする。例えば、1つの実施形態では、カーネル 906 は、図 4 に示すように、メッセージ重要度のセット及びアサーションのテークを遂行する。

【0073】

ポートキュー 912 においてメッセージがエンキューされた状態で、カーネル 906 は、機能 `kernel_msg_recv_916` を遂行して、ポートキュー 912 からメッセージをデキューし、そのメッセージを適応デーモンプロセス 904 のためのユーザバッファへコピーする。1つの実施形態では、カーネル 906 は、図 5 に示すように、メッセージをデキューし、コピーする。

10

【0074】

適応デーモンプロセス 904 は、デーモン接続 932 を経てメッセージ 928 をメッセージオブジェクト 938 として受信する。1つの実施形態では、適応デーモンプロセス 904 は、メッセージ 928 に対して応答しない。この実施形態では、メッセージサービス 940 は、メッセージオブジェクト 938 を破壊する。メッセージオブジェクトの破壊 934 に応答して、メッセージサービス 940 は、カーネルへのアサーション 926 をドロップさせる。アサーションをドロップさせることで、カーネル 906 は、適応デーモンプロセス 904 が適応デーモンプロセス 904 の重要度を降格すべきかどうか決定する。1つの実施形態では、カーネル 906 は、図 6 について上述したように適応デーモンプロセス 904 が降格されるべきかどうか決定する。

20

【0075】

別の実施形態では、メッセージサービスは、受信したメッセージ 928 に対する返事を準備する。この実施形態では、メッセージサービス 940 は、メッセージオブジェクト 938 に対する返事オブジェクト 936 を生成する。返事オブジェクト 936 を生成する際に、メッセージオブジェクト 938 に含まれた重要度が返事オブジェクト 936 へ転送される。メッセージサービス 940 は、デーモン接続 932 及びメッセージ 928 を経て、カーネル 906 を通して、ユーザアプリケーションプロセス 904 へ返事を送信する。更に、返事を送信することで、メッセージサービス 940 は、返事オブジェクト 930 を破壊する。返事オブジェクトは、重要度セットを有しているので、返事オブジェクトを破壊すると、メッセージサービス 940 がカーネルへのアサーション 926 をドロップさせる。アサーションをドロップすることで、カーネル 906 は、適応デーモンプロセス 904 が適応デーモンプロセス 904 の重要度を降格すべきかどうか決定する。1つの実施形態では、カーネル 906 は、図 6 について上述したように、適応デーモンプロセス 904 が降格されるべきかどうか決定する。更に、返事は、メッセージ 828 を使用し、ポートキュー 912 を経て、メッセージサービス 940 の `msg_recv` 機能を使用して、ユーザアプリケーションプロセス 902 へ送られる。この実施形態では、メッセージサービスは、メッセージ 920 及びユーザアプリケーション接続 918 を使用して、ユーザアプリケーションプロセス 902 へ返事を配送する。

30

【0076】

上述したように、デーモンプロセスの重要度は、ユーザアプリケーションプロセスからのメッセージの受信に基づいて変更することができる。別の実施形態では、第 2 のデーモンプロセスは、別のデーモンがユーザアプリケーションプロセスからメッセージを送信するのに応答して重要度を変化させることができる。図 10 は、第 1 デーモンにより受信されたメッセージに基づき第 2 デーモンの重要度を昇格させるプロセス 1000 の 1つの実施形態のフローチャートである。このプロセスは、ハードウェア（例えば、回路、専用ロジック、プログラム可能なロジック、マイクロコード、等）、ソフトウェア（例えば、ハードウェアシミュレーションを遂行するために処理装置で実行されるインストラクション）、又はその組み合わせを含む処理ロジックにより遂行される。1つの実施形態では、図 11 のカーネル 1132 のようなカーネルがプロセス 1000 を遂行する。1つの実施形

40

50

態では、ユーザアプリケーションプロセスから要求を受け取りそして別のデーモンから他の情報を要求することのできるデーモンがプロセス1000を遂行する。例えば、1つの実施形態では、プロセス1000は、デーモン1106A/1116のメッセージ処理モジュール1128により遂行される。図10では、プロセス1000は、ブロック1002において、第1メッセージを受信することにより開始する。1つの実施形態において、このメッセージは、ユーザアプリケーションプロセスからのメッセージのような、重要度を保持するメッセージである。別の実施形態では、メッセージは、重要度フラグセットでマークされる。例えば、1つの実施形態では、メッセージは、図9について上述したように重要度ビットセットを有する。更に別の実施形態では、メッセージは、別のデーモンにより送信され、カーネルは、メッセージを、重要度を保持するものとしてマークする。この実施形態では、デーモンからのメッセージを、重要度を保持するものとしてマークすることにより、デーモンプロセスプライオリティのSTRINGを昇格及び降格することができる。1つの実施形態では、デーモンによりこのメッセージを受信することで、図3A、4、及び10について述べたように、このデーモンのプロセス重要度を昇格させる。

10

20

30

40

50

【0077】

ブロック1004において、プロセス1000は、このメッセージの返事が別のデーモンからの情報を使用するかどうか決定する。例えば、1つの実施形態では、サーチデーモンは、サーチ要求を受け取る。このサーチデーモンは、サーチ要求を満足させるために、移動装置にインストールされたアプリケーションのリストを決定する必要がある。サーチデーモンは、このリストを得るための要求をインストールデーモンに発信する。プロセス1000が別のデーモンからの情報を使用しない場合には、実行がブロック1010へ進む。

【0078】

プロセス1000が別のデーモンからの情報を使用する場合には、ブロック1006において、プロセス1000は、その別のデーモンへ第2のメッセージを送信し、カーネルは、そのメッセージを、重要度を保持するものとしてマークする。更に、カーネルは、他のデーモンの重要度を昇格させ、このデーモンのためのアサーションをテークし、そして前記他のデーモンについてこのメッセージをキューする。ブロック1008において、プロセス1000は、第2メッセージに対する返事を受け取る。プロセス1000は、ブロック1010において、第1メッセージに対する返事を準備し、そして重要度で返事メッセージをマークする。更に、プロセス1000は、ブロック1002において、第1メッセージを受け取ったデーモンのアサーションをドロップさせる。ブロック1012において、プロセス1000は、第1メッセージを送信したプロセスに返事を返送する。1つの実施形態では、このメッセージを送信することにより、プロセス1000を遂行するデーモンの昇格された重要度が降格される。

【0079】

図11は、第1のデーモンプロセス1106Aが第2のデーモンプロセス1106Bと通信しそして第2のデーモンプロセス1106Bの重要度が昇格される移動装置1100の一実施形態のブロック図である。図11において、移動装置1100は、ユーザアプリケーションプロセス1104及びデーモンプロセス1106A-Nの実行を管理するオペレーティングシステム1102を含む。更に、ユーザアプリケーションプロセス1104は、高重要度で実行され、そしてデーモンプロセス1106A-Nは、最初、低重要度で実行される。ユーザアプリケーションプロセス1104がデーモンプロセス(例えば、デーモンプロセス1106A)からのサービスを要求するとき、このデーモンプロセスは、このサービス要求を処理するために高い重要度へ昇格される。1つの実施形態では、このデーモン1106Aは、適応デーモンと称される。例えば、1つの実施形態では、ユーザアプリケーションプロセス1104は、メッセージ1112を経てデーモン1106Aにサービス要求を発する。デーモン1106Aがメッセージを受け取るのに応答して、デーモンの重要度が、より高い重要度、即ちデーモン1116へ昇格される。1つの実施形態において、昇格されたデーモン1116の重要度は、要求側ユーザアプリケーションプロ

セス 1 1 0 4 と同じであるが、別の実施形態では、デーモン 1 1 1 6 の昇格されたプロセスは、要求側ユーザアプリケーションプロセス 1 1 0 4 の重要度より高くても低くてもよい。

【 0 0 8 0 】

デーモン A のプロセス 1 1 1 6 がメッセージを受け取りそして重要度が昇格されるのに応答して、デーモン A のプロセスは、別のメッセージ 1 1 3 6 をデーモン B のプロセス 1 1 0 6 B へ送る。このメッセージ 1 1 3 6 は、デーモン A に送られる同じメッセージでもよいし、異なるメッセージでもよい。デーモン 1 1 0 6 B がメッセージ 1 1 3 6 を受け取るのに応答して、デーモン B の重要度は、より高い重要度、即ちデーモン B 1 1 2 0 に昇格される。1 つの実施形態では、昇格されたデーモン B 1 1 2 0 の重要度が、最初の要求側ユーザアプリケーションプロセス 1 1 0 4 と同じであるが、別の実施形態では、デーモン 1 1 2 0 の昇格されたプロセスが、要求側ユーザアプリケーションプロセス 1 1 0 4 の重要度より高くても低くてもよい。

10

【 0 0 8 1 】

この実施形態では、昇格されたデーモン B 1 1 2 0 は、第 2 のメッセージを処理し、そして返事をデーモン A 1 1 1 6 へ送る。更に、デーモン B プロセスの重要度は、デーモン B 1 1 0 6 B の重要度へ降格される。1 つの実施形態では、デーモンプロセス B 1 1 2 0 は、アサーションをテークし、そしてカーネルは、デーモン B プロセス 1 1 2 0 の重要度を降格させる。返事 1 1 2 2 を受け取るのに応答して、デーモン A は、初期メッセージ 1 1 1 2 に対する返事を形成し、そしてそのメッセージをユーザアプリケーションプロセス 1 1 0 4 へ送る。更に、デーモン A プロセス 1 1 1 6 の重要度は、このプロセスのデフォルト重要度へ降格される。1 つの実施形態では、オペレーティングシステムカーネル 1 1 3 2 は、デーモンプロセスの昇格及び降格を取り扱う。別の実施形態では、オペレーティングシステムカーネル 1 1 3 2 は、メッセージサービス 1 1 3 4 を使用してデーモンプロセスの昇格及び降格を取り扱う。この実施形態では、プロセスマネージメント 1 1 3 4 は、ユーザアプリケーションプロセス 1 1 0 4 と、デーモンプロセス 1 1 0 6 A / 1 1 1 6 と、デーモンプロセス 1 1 0 6 B / 1 1 2 0 との間でのメッセージの送信及び受信を取り扱う。これらのプロセス間でのメッセージの通過を取り扱うことにより、プロセスマネージメント 1 1 3 2 は、デーモンプロセスの昇格及び降格を管理することができる。

20

30

【 0 0 8 2 】

例えば、1 つの実施形態では、サーチデーモンを使用してサーチを行う例について考える。この例では、サーチデーモンは、ユーザアプリケーションプロセスからサーチ要求を受け取る結果としてサーチデーモンプロセスの重要度を昇格させる。サーチ要求メッセージを受け取るのに応答して、サーチデーモンは、移動装置におけるインストール済アプリケーションのリストを得るために第 2 のメッセージをインストールデーモンへ送る。重要度の昇格それ自体を取得したサーチデーモンは、インストール済アプリケーションのリストのための第 2 メッセージをインストールデーモンへ送ることによりこの重要度を寄贈する。この例では、インストールデーモンは、サーチデーモンから第 2 メッセージを受け取った際にその重要度を昇格させる。更に、インストールデーモンの重要度は、インストールデーモンがサーチデーモンに返事を送る際にインストールデーモンのデフォルト重要度へ降格される。1 つの実施形態では、メッセージの取り扱い及びデーモンプロセスの昇格 / 降格は、上述したメッセージサービス 1 0 3 4 のようなメッセージサービスによって取り扱われる。

40

【 0 0 8 3 】

図 1 2 は、プロセス間でのメッセージの送信及び受信を管理するためのプロセスマネージメントモジュール 3 2 2 である。1 つの実施形態において、プロセスマネージメントモジュール 3 2 2 は、図 3 A - B のカーネル 3 2 0 及び / 又は図 1 0 のカーネル 1 0 3 2 のような、送信及び受信メッセージを容易にするためのカーネルにより使用される。1 つの実施形態では、プロセスマネージメントモジュール 3 2 2 は、メッセージ送信モジュール

50

1202、メッセージ受信モジュール1204、及びドロップアサーション重要度モジュール1206を含む。この例では、メッセージ送信モジュール1202は、図4について上述したメッセージを送信する。メッセージ受信モジュール1204は、図5について上述したように受信を行う。ドロップアサーション重要度モジュール1206は、図6について上述したようにアサーションをドロップさせる。

【0084】

図13は、デーモンに向けられたメッセージに基づきデーモンの重要度を昇格するのに使用されるメッセージを送信するためのメッセージ送信モジュール1202である。1つの実施形態では、メッセージ送信モジュール1202は、プロセス重要度ドナーモジュール1302、プロセス重要度受信モジュール1304、マークメッセージモジュール1306、プロセス低重要度モジュール1308、昇格プロセスモジュール1310、及びキューメッセージモジュール1312を含む。1つの実施形態では、プロセス重要度ドナーモジュール1302は、図4のブロック402で上述したように、現在プロセスが重要度ドナーであるかどうか決定する。プロセス重要度受信モジュール1304は、図4のブロック404で上述したように、ターゲットプロセスが重要度受取者であるかどうか決定する。マークメッセージモジュール1306は、図4のブロック406で上述したように、メッセージをマークし、そして重要度アサーションカウントをインクリメントする。プロセス低重要度モジュール1308は、図4のブロック408で上述したように、ターゲットプロセスが低重要度をもつかどうか決定する。昇格プロセスモジュール1310は、図4のブロック410で上述したように、ターゲットプロセス重要度を変更する。キューメッセージモジュール1312は、図4のブロック413で上述したように、ポートにおいてメッセージをキューする。

10

20

【0085】

図14は、デーモンに向けられたメッセージに基づいてデーモン重要度を昇格するのに使用されるメッセージを受信するためのメッセージ受信モジュール1204である。1つの実施形態では、メッセージ受信モジュール1204は、キューメッセージモジュール1402、デキューメッセージモジュール1404、メッセージマークモジュール1406、プロセス重要度受信モジュール1408、除去フラグモジュール1410、及びコピーメッセージモジュール1412を備えている。1つの実施形態において、キューメッセージモジュール1402は、図5のブロック502で上述したように、キューメッセージがあるかどうか決定する。デキューメッセージモジュール1404は、図5のブロック504で上述したように、ポートからメッセージをデキューする。メッセージマークモジュール1406は、図5のブロック506で上述したように、メッセージが重要度についてマークされるかどうか決定する。プロセス重要度受信モジュール1408は、図5のブロック508で上述したように、現在プロセスが適応デーモンプロセスであるかどうか決定する。除去フラグモジュール1410は、図5のブロック510で上述したように、フラグを除去し、そして重要度カウントをデクリメントする。コピーメッセージモジュール1412は、図5のブロック512で上述したように、ユーザバッファへメッセージをコピーする。

30

【0086】

図15は、デーモン重要度を昇格するために使用されるメッセージを受け取るデーモンmsg_recvモジュール324である。1つの実施形態において、デーモンmsg_recvモジュール324は、メッセージ受信モジュール1502、メッセージマークモジュール1504、セットメッセージ重要度モジュール1506、プロセスメッセージモジュール1508、メッセージ返事モジュール、重要度メッセージモジュール、及びドロップ重要度モジュール1514を備えている。この実施形態では、メッセージ受信モジュール1502は、図6のブロック602で上述したように、メッセージが受信されるかどうか決定する。メッセージマークモジュール1504は、図6のブロック604で上述したように、重要度についてメッセージをマークする。セットメッセージ重要度モジュール1506は、図6のブロック606で上述したように、重要度を真としてセットする。プ

40

50

ロセスメッセージモジュール1508は、図6のブロック608で上述したように、メッセージを処理し、そして必要なアクションを遂行する。メッセージ返事モジュール1510は、図6のブロック610で上述したように、返事を送信する。重要度メッセージモジュール1512は、図6のブロック612で上述したように、メッセージが重要度としてマークされるかどうか決定する。ドロップ重要度モジュール1514は、図6のブロック614で上述したように、重要度アサーションをドロップさせる。

【0087】

図16は、重要度アサーションをドロップするためのドロップアサーション重要度モジュール1206である。1つの実施形態において、ドロップ重要度モジュール1206は、現在プロセスモジュール1602、デクリメントアサーションカウントモジュール1604、アサーションカウントゼロモジュール1606、及びプロセス降格モジュール1608を備えている。この実施形態では、現在プロセスモジュール1602は、図7のブロック702で上述したように、現在プロセスが重要度プロセスであるかどうか決定する。デクリメントアサーションカウントモジュール1604は、図7のブロック704で上述したように、重要度アサーションカウントをデクリメントする。アサーションカウントゼロモジュール1606は、図7のブロック706で上述したように、重要度アサーションカウントを決定する。プロセス降格モジュール1608は、図7のブロック708で上述したように、現在プロセスを降格させる。

【0088】

図17は、オンデマンドでサービスを起動する起動サービス326である。1つの実施形態では、この起動サービスは、ペンディングイベントモジュール1702、メッセージサービスモジュール1704、生成サービスポートモジュール1706、メッセージサービスポートモジュール1708、適応デーモンモジュール1710、起動適応デーモン1712、起動デーモンモジュール1714、適応デーモンチェックインモジュール1716、転送サービスポートモジュール1718、及び他イベント処理モジュール1720を備えている。1つの実施形態では、ペンディングイベントモジュール1702は、図8のブロック802で上述したように、いずれかのペンディングイベントがあるかどうか決定する。メッセージサービスモジュール1704は、図8のブロック804で上述したように、生成されるべきメッセージサービスがあるかどうか決定する。生成サービスポートモジュール1706は、図8のブロック806で上述したように、サービスポートを生成し、そしてこのサービスポートをウォッチリストに追加する。メッセージサービスポートモジュール1708は、図8のブロック808で上述したように、サービスポートにメッセージがあるかどうか決定する。適応デーモンモジュール1710は、図8のブロック810で上述したように、メッセージが適応デーモンに対するものであるかどうか決定する。起動適応デーモン1712は、図8のブロック812で上述したように、適応デーモンを起動する。起動デーモンモジュール1714は、図8のブロック814で上述したように、デーモンを起動する。適応デーモンチェックインモジュール1716は、図8のブロック816で上述したように、起動後に適応デーモンがチェックインしたかどうか決定する。転送サービスポートモジュール1718は、図8のブロック818で上述したように、サービスポートを適応デーモンへ転送する。他イベント処理モジュール1720は、図8のブロック820で上述したように、他のイベントを処理する。

【0089】

図18は、第1デーモンにより受信されたメッセージに基づき第2デーモンの重要度を昇格するためのメッセージ処理モジュール1508である。1つの実施形態では、メッセージ処理モジュール1508は、受信メッセージモジュール1802、メッセージデーモンモジュール1804、送信メッセージデーモンモジュール1806、受信返事デーモンモジュール1808、準備返事モジュール1810、及び送信返事モジュール1812を備えている。この実施形態では、受信メッセージモジュール1802は、図9のブロック902で上述したように、第1メッセージを受信する。メッセージデーモンモジュール1804は、図9のブロック904で上述したように、第1メッセージの返事が別のデーモ

10

20

30

40

50

ンからの情報を要求するかどうか決定する。第2メッセージデーモンモジュール1806は、図9のブロック906で上述したように、他のデーモンへ第2のメッセージを送信する。受信返事デーモンモジュール1808は、図9のブロック908で上述したように、返事を受信する。準備返事モジュール1810は、図9のブロック910で上述したように、第1メッセージに対する返事メッセージを準備する。送信返事モジュール1812は、図9のブロック912で上述したように、返事を送信する。

【0090】

図19は、本発明の1つの実施形態に使用されるデータ処理システム1900の一例を示す。例えば、システム1900は、図3A-Bに示された装置300を含めて、具現化される。図19は、コンピュータシステムの種々のコンポーネントを示すが、コンポーネントを相互接続する特定のアーキテクチャー又は仕方を表わすことを意図していない。というのは、そのような詳細は、本発明と関係がないからである。より少数のコンポーネント又はおそらくより多数のコンポーネントを有するネットワークコンピュータ及び他のデータ処理システム又は他の消費者向け電子装置も、本発明に使用できることが明らかであろう。

10

【0091】

図19に示すように、一形式のデータ処理システムであるコンピュータシステム1900は、バス1903を備え、これは、マイクロプロセッサ1905、ROM(リードオンリメモリ)1907、揮発性RAM1909、及び不揮発性メモリ1911に結合される。マイクロプロセッサ1905は、メモリ1907、1909、1911からインストラクションを検索し、そしてそれらインストラクションを実行して、上述したオペレーションを遂行する。バス1903は、これら種々のコンポーネントと一緒に相互接続すると共に、これらのコンポーネント1905、1907、1909及び1911を、ディスプレイコントローラ・ディスプレイ装置1913、並びに入力/出力(I/O)装置のような周辺装置に相互接続し、そのI/O装置は、マウス、キーボード、モデム、ネットワークインターフェイス、プリンタ、及びこの分野で良く知られた他の装置である。典型的に、入力/出力装置1915は、入力/出力コントローラ1913を通してシステムに結合される。揮発性RAM(ランダムアクセスメモリ)1909は、典型的に、メモリ内のデータをリフレッシュ又は維持するために常時電力を必要とするダイナミックRAM(DRAM)として具現化される。

20

30

【0092】

大量記憶装置1911は、典型的に、システムから電力が取り去られた後もデータ(例えば、大量のデータ)を維持する磁気ハードディスク、磁気光学ドライブ、光学ドライブ又はDVD RAM、又はフラッシュメモリ、或いは他の形式のメモリシステムである。典型的に、大量記憶装置1911は、ランダムアクセスメモリでもあるが、これは、必要なことではない。図19は、大量記憶装置1911がデータ処理システムの残りのコンポーネントに直結されたローカル装置であることを示しているが、本発明は、システムから遠隔にある不揮発性メモリ、例えば、モデムのようなネットワークインターフェイス、イーサネット(登録商標)インターフェイス、又はワイヤレスネットワークを通してデータ処理システムに結合されたネットワーク記憶装置を使用してもよいことが明らかである。バス1903は、この分野で良く知られたように、種々のブリッジ、コントローラ、及び/又はアダプタを通して互いに接続される1つ以上のバスを含む。

40

【0093】

図20は、本発明の一実施形態に使用される別のデータ処理システム2000の一例を示す。例えば、システム2000は、図3A-Bに示す装置300として具現化される。図20に示すデータ処理システムは、1つ以上のマイクロプロセッサであるか又はシステムオンチップ集積回路である処理システム2011を備え、又、処理システムにより実行するためのデータ及びプログラムを記憶するメモリ2001も備えている。又、システム2000は、オーディオ入力/出力サブシステム2005も備え、これは、例えば、音楽を再生するためのマイクロホン及びスピーカを含むか、又はスピーカ及びマイクロホンを

50

通して電話機能を与える。

【0094】

ディスプレイコントローラ・ディスプレイ装置2009は、ユーザのための視覚ユーザインターフェイスを与え、このデジタルインターフェイスは、OS Xオペレーティングシステムソフトウェアを実行するときのMacintoshコンピュータ、又はiOSオペレーティングシステムを実行するときのApple iPhone、等に示されるものと同様のグラフィックユーザインターフェイスを含む。又、システム2000は、図20のシステム2000のような別のデータ処理システムと通信するための1つ以上のワイヤレストランシーバ2003も備えている。ワイヤレストランシーバは、WLANトランシーバ、赤外線トランシーバ、Bluetooth（登録商標）トランシーバ、及び/又はワイヤレスセルラー電話トランシーバである。ある実施形態では、図示されていない付加的なコンポーネントも、システム2000の一部であり、そしてある実施形態では、図20の示すものより少ないコンポーネントをデータ処理システムに使用してもよいことが明らかであろう。更に、システム2000は、図15のシステム1500のような別のデータ処理システムと通信するために1つ以上の通信ポート2017も備えている。通信ポートは、USBポート、Firewireポート、Bluetooth（登録商標）インターフェイス、等である。

10

【0095】

又、データ処理システム2000は、ユーザがシステムに入力を与えることができるようにする1つ以上の入力装置2013も備えている。それらの入力装置は、キーパッド、又はキーボード、又はタッチパネル、又はマルチタッチパネルである。又、データ処理システム2000は、ドックのためのコネクタである任意の入力/出力装置2015も備えている。当業者に明らかなように、種々のコンポーネントを相互接続するのに、図示されていない1つ以上のバスを使用できることが明らかであろう。図20に示すデータ処理システムは、ハンドヘルドコンピュータ、又はパーソナルデジタルアシスタント（PDA）、又はPDAのような機能をもつセルラー電話、又はセルラー電話を含むハンドヘルドコンピュータ、又はiPodのようなメディアプレーヤ、或いはそれら装置の態様又は機能を結合する装置、例えば、PDA及びセルラー電話を1つの装置に結合したメディアプレーヤ、又は埋め込み型装置、又は他の消費者向け電子装置である。他の実施形態では、データ処理システム2000は、ネットワークコンピュータ、又は別の装置内の埋め込み型処理装置、或いは図20に示すものより少数のコンポーネント又はおそらくそれより多数のコンポーネントを有する他の形式のデータ処理システムである。

20

30

【0096】

本発明の少なくとも幾つかの実施形態は、ポータブル音楽及び/又はビデオメディアプレーヤのようなデジタルメディアプレーヤの部分であり、これは、メディアを提示するためのメディア処理システム、メディアを記憶するための記憶装置を含むと共に、アンテナシステム及びメディア処理システムに結合された高周波（RF）トランシーバ（例えば、セルラー電話のためのRFトランシーバ）を含む。ある実施形態では、リモート記憶装置に記憶されるメディアは、RFトランシーバを通してメディアプレーヤへ送信される。メディアは、例えば、音楽又は他のオーディオ、静止画像、又は動画の1つ以上である。

40

【0097】

ポータブルメディアプレーヤは、カリフォルニア州クパチーノのアップル・インクからのiPod（登録商標）又はiPod Nano（登録商標）プレーヤにおけるクリックホイール入力、タッチスクリーン入力装置、プッシュボタン装置、可動ポインティング入力装置、又は他の入力装置のようなメディア選択装置を含む。このメディア選択装置は、記憶装置及び/又はリモート記憶装置に記憶されたメディアを選択するのに使用される。ポータブルメディアプレーヤは、少なくとも幾つかの実施形態では、入力装置を通して選択され、且つスピーカ又はイヤホンを通して、又はディスプレイ装置に、或いはディスプレイ装置及びスピーカ又はイヤホンの両方に提示されるメディアのタイトル又は他の指示子を表示するためにメディア処理システムに結合されるディスプレイ装置を含む。ポータ

50

ブルメディアプレーヤの例は、米国特許第 7, 345, 671 号及び米国特許出願公告第 2004/0224638 号に説明されており、これらは参考としてここに援用される。

【0098】

上述された部分は、専用ロジック回路のようなロジック回路、或いはプログラムコードインストラクションを実行するマイクロコントローラ又は他の形式の処理コアで具現化される。従って、前記説明により教示されるプロセスは、マシン実行可能なインストラクションを実行するマシンが幾つかの機能を遂行するようにさせるマシン実行可能なインストラクションのようなプログラムコードで遂行される。この点に関して、「マシン」とは、中間形態（又は「アブストラクト」）インストラクションを、プロセッサ特有のインストラクション（例えば、「バーチャルマシン」（例えば、Java（登録商標）バーチャルマシン）、インタープリター、コモン・ランゲージ・ランタイム、高レベル言語バーチャルマシン、等のようなアブストラクト実行環境）へ変換するマシン、及び/又は汎用プロセッサ及び/又は特殊目的プロセッサのような、インストラクションを実行するように設計された半導体チップ（例えば、トランジスタで具現化される「ロジック回路」）上に配置された電子回路である。又、前記説明により教示されるプロセスは、プログラムコードを実行せずにプロセス（又はその一部分）を遂行するように設計された電子回路により遂行されてもよい（マシンに代わって又はマシンとの組み合わせにおいて）。

10

【0099】

又、本発明は、ここに述べるオペレーションを遂行するための装置にも関する。この装置は、必要とされる目的で特別に構成されてもよいし、又はコンピュータに記憶されたコンピュータプログラムにより選択的にアクチベートされ又は再構成される汎用コンピュータでもよい。そのようなコンピュータプログラムは、これに限定されないが、フロッピー（登録商標）ディスク、光学ディスク、CD-ROM、及び磁気・光学ディスクを含む任意の形式のディスク、リードオンリメモリ（ROM）、RAM、EPROM、EEPROM、磁気又は光学カード、或いは電子的インストラクションを記憶するのに適したものでコンピュータシステムバスに各々結合された任意の形式の媒体のようなコンピュータ読み取り可能な記憶媒体に記憶される。

20

【0100】

マシン読み取り可能な媒体は、マシン（例えば、コンピュータ）により読み取り可能な形態で情報を記憶又は送信するためのメカニズムを含む。例えば、マシン読み取り可能な媒体は、リードオンリメモリ（ROM）、ランダムアクセスメモリ（RAM）、磁気ディスク記憶媒体、光学的記憶媒体、フラッシュメモリ装置、等を含む。

30

【0101】

製造物品は、プログラムコードを記憶するのに使用される。プログラムコードを記憶する製造物品は、これに限定されないが、1つ以上のメモリ（例えば、1つ以上のフラッシュメモリ、ランダムアクセスメモリ（スタティック、ダイナミック、その他）、光学的ディスク、CD-ROM、DVD-ROM、EPROM、EEPROM、磁気又は光学カード、或いは電子的インストラクションを記憶するのに適した他の形式のマシン読み取り可能な媒体として実施される。又、プログラムコードは、伝播媒体で実施されるデータ信号により（例えば、通信リンク（例えば、ネットワーク接続）を経て）リモートコンピュータ（例えば、サーバー）から要求側コンピュータ（例えば、クライアント）へダウンロードされてもよい。

40

【0102】

以上の詳細な説明は、コンピュータメモリ内のデータビットに対するオペレーションのアルゴリズム及び記号表示に関して表現された。これらのアルゴリズム記述及び表現は、データ処理分野の当業者により、彼等の仕事の実質を他の当業者に最も効率的に伝えるために使用される。アルゴリズムとは、ここでは、一般的に、望ましい結果を導く自己矛盾のないオペレーションシーケンスであると考えられる。オペレーションは、物理量の物理的操作を要求するものである。通常、必ずしもそうでないが、これらの量は、記憶、転送、合成、比較、その他、操作することのできる電氣的又は磁氣的信号の形態をとる。主と

50

して、慣用上、これらの信号は、ビット、値、エレメント、記号、キャラクタ、期間、数、等として参照するのが便利であると分かっている。

【0103】

しかしながら、これら及び同様の用語は、全て、適当な物理量に関連付けられ、そしてそれらの量に適用される便宜的表示に過ぎないことを銘記されたい。特に指示のない限り、以上の説明から明らかのように、この説明全体を通して、「受信」「発信」「処理」「送信」「昇格」「降格」「返送」「計算」「エンキュー」「移送」「起動」「転送」「デクリメント」「維持」、等の用語を使用する説明は、コンピュータシステムのレジスタ及びメモリ内で物理的（電子的）量として表されたデータを操作し、そしてコンピュータシステムのメモリ又はレジスタ或いは他のそのような情報記憶、伝達又は表示装置内で物理的 10
量的量として同様に表される他のデータへと変換するコンピュータシステム又は同様の電子的コンピューティング装置のアクション及び処理を参照するものである。

【0104】

ここに述べるアルゴリズム及びディスプレイは、特定のコンピュータ又は他の装置に固有に関係していない。種々の汎用システムを、ここに述べる技術に基づくプログラムと共に使用してもよいし、又はここに述べるオペレーションを遂行するように更に特殊な装置を構成するのが便利であると分かっている。種々のこれらシステムに要求される構造は、以上の説明から明らかとなろう。更に、本発明は、特定のプログラミング言語を参照して説明していない。上述した本発明の実施形態の教示を具現化するのに種々のプログラミング言語を使用できることが明らかであろう。 20

【0105】

以上の説明は、本発明の規範的实施形態を単に記述したものである。当業者であれば、そのような説明、添付図面及び特許請求の範囲から、本発明の精神及び範囲から逸脱せずに種々の変更がなされ得ることが容易に明らかであろう。

【符号の説明】

【0106】

- 100：移動装置
- 102：オペレーティングシステム
- 104：ユーザアプリケーションプロセス
- 106A-N：デーモンプロセス
- 108：高重要度プロセス
- 110：低重要度プロセス
- 200：移動装置
- 202：オペレーティングシステム
- 204：ユーザアプリケーションプロセス
- 212：デーモンプロセス
- 206A N：デーモンプロセス
- 208：高重要度プロセス
- 210：低重要度プロセス
- 300：移動装置
- 302：オペレーティングシステム
- 304：ユーザアプリケーションプロセス
- 306A-N：デーモンプロセス
- 308：高重要度プロセス
- 310：低重要度プロセス
- 318：プロセス
- 320：カーネル
- 322：プロセス管理
- 326：起動サービス
- 902：ユーザアプリケーションプロセス

10

20

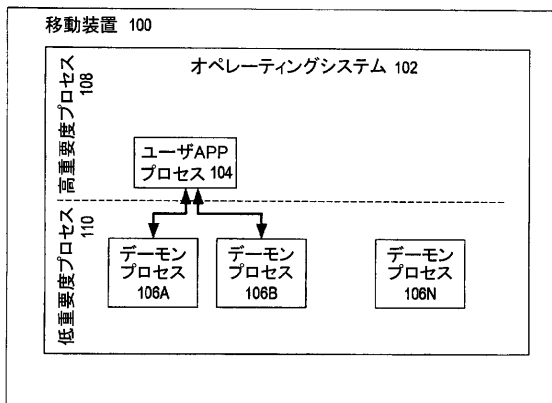
30

40

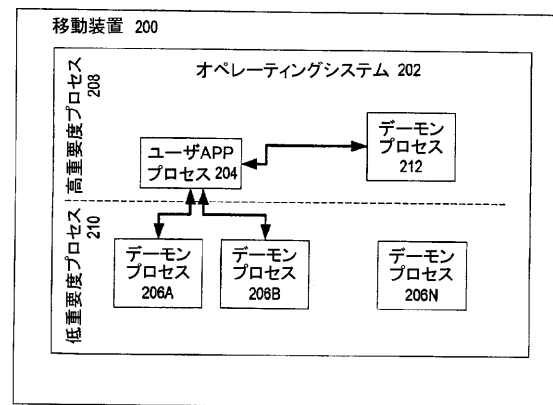
50

- 904 : 適応デーモンプロセス
- 906 : カーネル
- 908 : 重要度セット
- 910 : アサーションテーク
- 912 : ポートキュー
- 918 : ユーザAPP接続
- 926 : アサーションドロップ
- 930 : 返事オブジェクト破壊
- 932 : デーモン接続
- 936 : 返事オブジェクト
- 940 : メッセージサービス

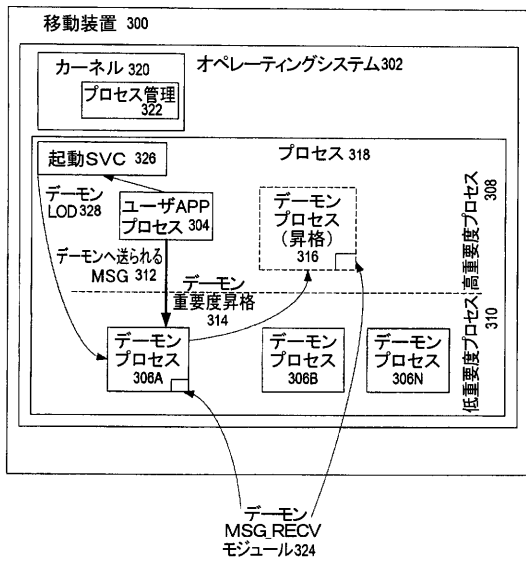
【 図 1 】



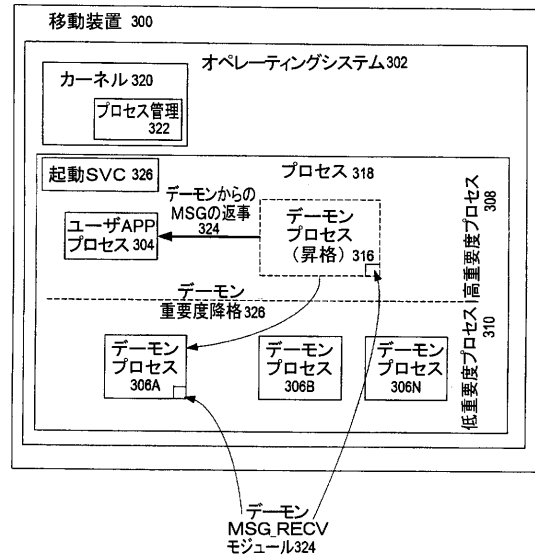
【 図 2 】



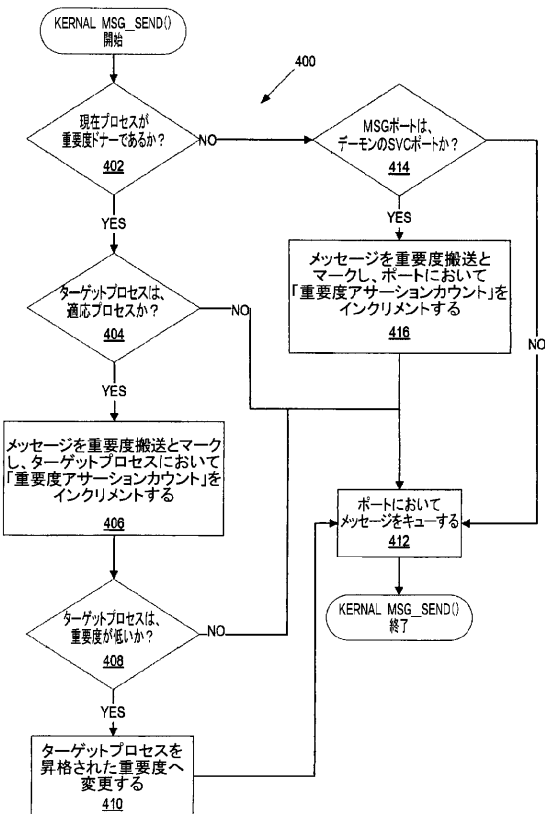
【図3A】



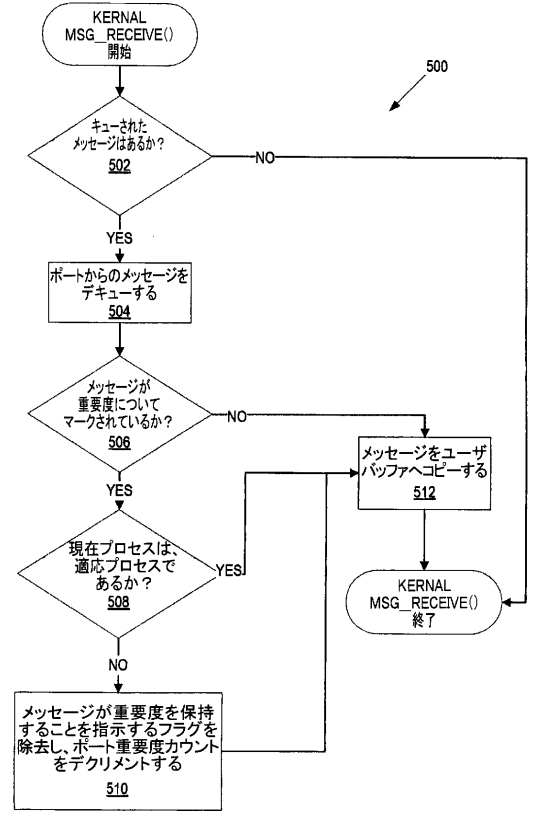
【図3B】



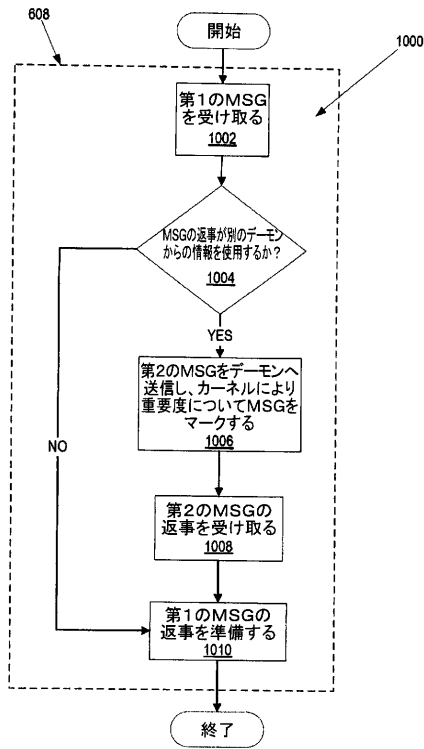
【図4】



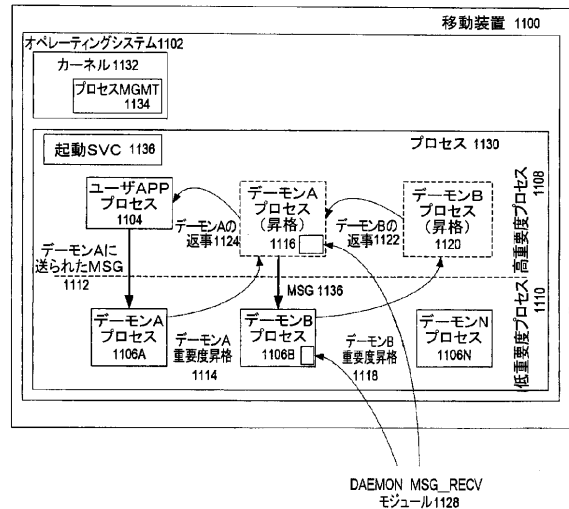
【図5】



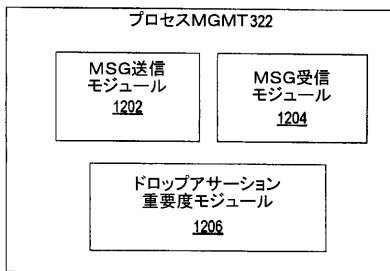
【 図 1 0 】



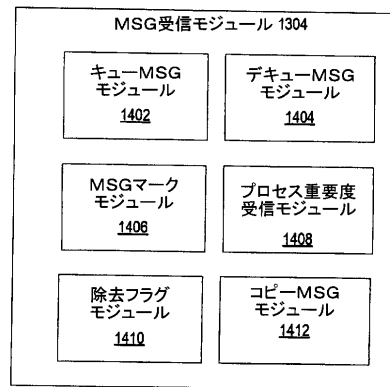
【 図 1 1 】



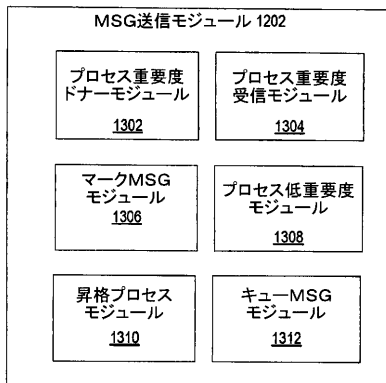
【 図 1 2 】



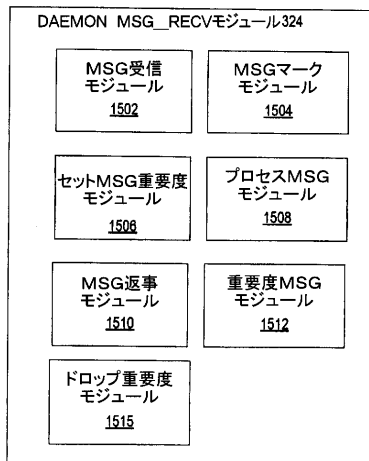
【 図 1 4 】



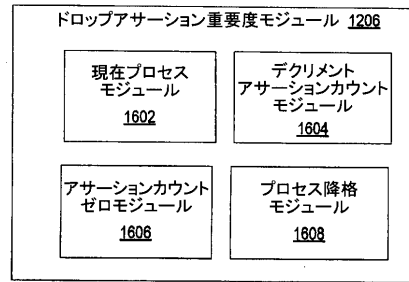
【 図 1 3 】



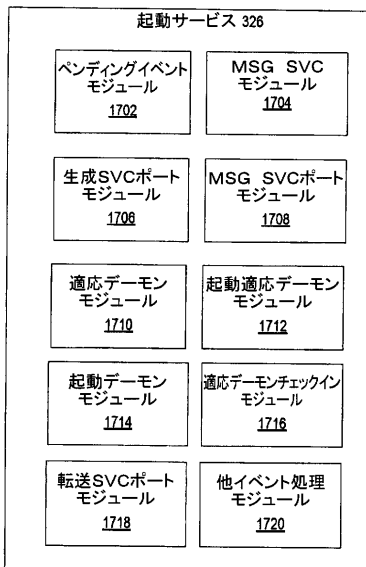
【 図 1 5 】



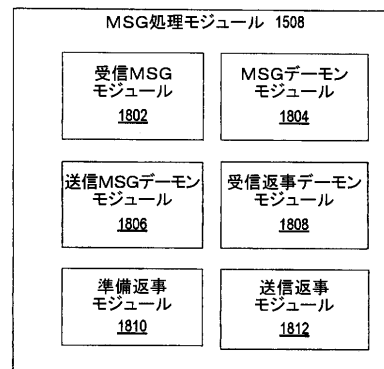
【 図 1 6 】



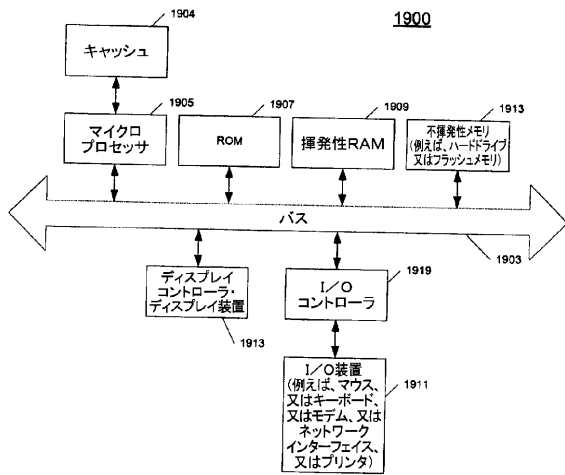
【 図 1 7 】



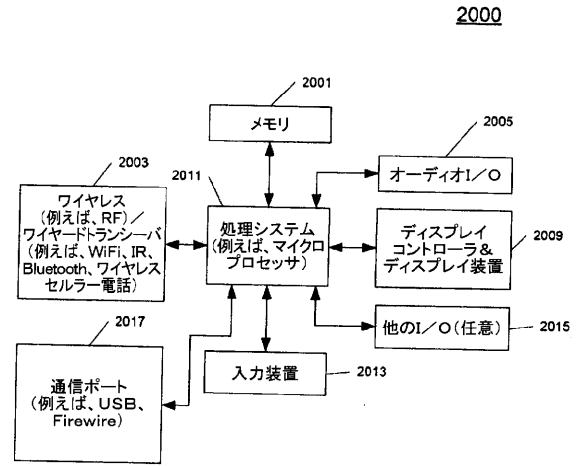
【 図 1 8 】



【 図 1 9 】



【 図 2 0 】



フロントページの続き

- (72)発明者 アナンサクリシュナ ラメシュ
アメリカ合衆国 95014 カリフォルニア州 クパチーノ インフィニット ループ 1 エ
ムエス 3 - シーオーエス
- (72)発明者 ヨーセン リン
アメリカ合衆国 95014 カリフォルニア州 クパチーノ インフィニット ループ 1 エ
ムエス 302 - 3オーエス3
- (72)発明者 ダミアン パスカル ソレッソ
アメリカ合衆国 95014 カリフォルニア州 クパチーノ インフィニット ループ 1 エ
ムエス 3 - シーオーエス
- (72)発明者 ジェイムス マイケル マギー
アメリカ合衆国 95014 カリフォルニア州 クパチーノ インフィニット ループ 1 エ
ムエス 111 - エイチオーエム
- (72)発明者 ケヴィン ジェイムス ヴァン ヴェクテン
アメリカ合衆国 95014 カリフォルニア州 クパチーノ インフィニット ループ 1 エ
ムエス 3 - シーオーエス
- (72)発明者 ウメシュ スレシュ ヴァイシャンパヤン
アメリカ合衆国 95014 カリフォルニア州 クパチーノ インフィニット ループ 1 エ
ムエス 3 - シーオーエス

【外国語明細書】

2013257873000001.pdf