

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第5452496号
(P5452496)

(45) 発行日 平成26年3月26日 (2014. 3. 26)

(24) 登録日 平成26年1月10日 (2014. 1. 10)

(51) Int. Cl.	F I
G 0 6 F 9/48 (2006. 01)	G 0 6 F 9/46 4 5 2 H
G 0 6 F 9/50 (2006. 01)	G 0 6 F 9/46 4 6 2 A

請求項の数 12 (全 20 頁)

(21) 出願番号	特願2010-528981 (P2010-528981)	(73) 特許権者	500046438
(86) (22) 出願日	平成20年10月7日 (2008. 10. 7)		マイクロソフト コーポレーション
(65) 公表番号	特表2011-501268 (P2011-501268A)		アメリカ合衆国 ワシントン州 9805
(43) 公表日	平成23年1月6日 (2011. 1. 6)		2-6399 レッドモンド ワン マイ
(86) 国際出願番号	PCT/US2008/079117		クロソフト ウェイ
(87) 国際公開番号	W02009/048892	(74) 代理人	100140109
(87) 国際公開日	平成21年4月16日 (2009. 4. 16)		弁理士 小野 新次郎
審査請求日	平成23年10月7日 (2011. 10. 7)	(74) 代理人	100089705
(31) 優先権主張番号	11/870, 981		弁理士 社本 一夫
(32) 優先日	平成19年10月11日 (2007. 10. 11)	(74) 代理人	100075270
(33) 優先権主張国	米国 (US)		弁理士 小林 泰
		(74) 代理人	100080137
			弁理士 千葉 昭男
		(74) 代理人	100096013
			弁理士 富田 博行

最終頁に続く

(54) 【発明の名称】 階層的予約資源スケジューリング・インフラストラクチャ

(57) 【特許請求の範囲】

【請求項 1】

コンピューティング環境においてプロセッサが以下の各行為を実施する、システム資源のスケジューリング方法であって、

第1の作業負荷に対して、作業負荷内に含まれる動作をスケジューリングするための第1のシステム資源スケジューリング方針を割り当てる行為であって、前記第1のシステム資源スケジューリング方針は、当該方針に基づく動作の実行に対して予め定められた時間間隔で配分されるべきシステム資源の割合を規定する比率基準方針からなる、行為と、

前記比率基準方針に従って前記第1の作業負荷のためにシステム資源を予約する行為と

、

第2の作業負荷に対して第2のシステム資源スケジューリング方針を割り当てる行為であって、前記第2のシステム資源スケジューリング方針は、前記比率基準方針とは異なる容量基準方針からなる、行為と、

前記容量基準方針に従って前記第2の作業負荷のためにシステム資源を予約する行為と

、

スケジューリングされずに残っているシステム資源の少なくとも一部を、システムのデフォルト・スケジューリング方針を用いる別の作業負荷のために確保する行為と、

を備え、

前記比率基準方針に基づく前記第1の作業負荷のための前記予約と前記容量基準方針に基づく前記第2の作業負荷のための前記予約が、同一の実行計画に含まれる、方法。

10

20

【請求項 2】

請求項 1 記載の方法において、前記作業負荷が他の作業負荷よりも階層的に下にあり、前記作業負荷のためのシステム資源を予約する行為は、前記作業負荷に対する方針と、前記作業負荷よりも階層的に上にある前記別の作業負荷に対する方針との双方によって指定されるように実行する、方法。

【請求項 3】

請求項 1 記載の方法において、容量によって資源の予約を指定するまたは期限によって資源の予約を指定する少なくとも 1 つの別の方針をスケジューリングする行為を含む、方法。

【請求項 4】

10

請求項 1 記載の方法において、前記システム資源は、プロセッサ、ネットワーク資源、メモリー資源、またはディスク資源の内少なくとも 1 つである、方法。

【請求項 5】

請求項 1 記載の方法において、前記方針によって指定されるように前記作業負荷のためにシステム資源を予約する行為は、

前記複数のシステム資源間で前記作業負荷の負荷均衡化を目指すようにシステム資源を予約する行為と、
を備えている、方法。

【請求項 6】

20

請求項 1 記載の方法において、前記方針によって指定されるように前記作業負荷のためにシステム資源を予約する行為は、

1 つのデバイスから別のデバイスへ作業負荷を移動することを目指すようにシステム資源を予約する行為と、
を備えている、方法。

【請求項 7】

請求項 1 記載の方法において、前記方針によって指定されるように前記作業負荷のためにシステム資源を予約する行為は、

上限クォータを強制することを目指すようにシステム資源を予約する行為と、
を備えている、方法。

【請求項 8】

30

請求項 1 記載の方法であって、更に、

前記作業負荷を実行したときに、前記システムの使用に関する情報を含むフィードバックを受ける行為と、

更に 1 つ又は複数の作業負荷を前記システム資源に配分するために、前記フィードバックに基づいて方針決定を行う行為と、
を備えている、方法。

【請求項 9】

請求項 8 記載の方法であって、更に、前記比率基準方針と前記容量基準方針にしたがって、前記第 1 及び第 2 の作業負荷を実行する行為を備えている、方法。

【請求項 10】

40

請求項 8 記載の方法において、前記システム資源は、プロセッサ、ネットワーク資源、メモリー資源、またはディスク資源の内の 1 つ又は複数を含む、方法。

【請求項 11】

請求項 8 記載の方法において、前記作業負荷をシステム資源に配分するために前記方針を用いる行為は、

前記方針とは別個に実装されたディスパッチャにおいて、該ディスパッチャがいずれの特定の方針とも独立して動作するように、前記システム資源によって実行されるべき作業負荷を示す前記方針からの情報を受け取る行為と、

前記ディスパッチャが、前記作業負荷を選択し、前記システム資源に前記作業負荷を実行させる行為と、

50

を備えている、方法。

【請求項12】

請求項8記載の方法において、前記フィードバックに基づいて方針決定を行う行為は、前記フィードバックに基づいて、新たな方針または同じ方針の内少なくとも1つを選択することを含む、方法。

【発明の詳細な説明】

【背景技術】

【0001】

背景および関連技術

コンピューターおよびコンピューティングシステムは、現代生活のほぼあらゆる面に影響を及ぼしてきている。コンピューターは、一般に、仕事、リクリエーション、健康管理、輸送、娯楽、家事管理などに関わっている。家庭用コンピューター、業務用ワークステーション、およびその他のシステムのような、汎用コンピューターを含む多くのコンピューターは、種々の異なる動作を実行する。動作は、作業負荷(workload)群に分類することができ、1つの作業負荷が、個々のタスクまたは目的を遂行するための1組の動作を定める。例えば、1つの作業負荷は、メディア・プレーヤ・アプリケーションを実施するために割り振ることができる。別の作業負荷は、ワード・プロセッサ・アプリケーションを実施するために割り振ることができる。更に別の作業負荷は、カレンダー管理、電子メール、またはその他管理アプリケーションを実施するために割り振ることができる。以前から暗に言われているように、多数の異なる作業負荷が1つのシステム上で一緒に動作している場合もあり得る。

10

20

【発明の概要】

【発明が解決しようとする課題】

【0002】

作業負荷群が1つのシステム上で一緒に動作することを可能にするには、システム資源の適性なスケジューリングを行い、異なる作業負荷群に割り当てなければならない。例えば、1つのシステム資源はプロセッサを含む。このプロセッサは、メディア・プレーヤ・アプリケーションのためにはディジタル・メディア・デコード処理を実行し、ワード・プロセッサ・アプリケーションのためにはフォントの助言(hinting)およびその他の表示機能を実行し、個人管理アプリケーションのためにはアルゴリズム計算を実行する能力を有することができる。しかしながら、1つのプロセッサでは、通例、いずれの所与の1時点においても1つのタスク、または限られた数のタスクしか実行することができない。このため、システム資源を種々の作業負荷間で共有できるように、スケジューリング・アルゴリズムが、プロセッサのような、システム資源のスケジューリングを行うとよい。

30

【0003】

通例、システム資源のスケジューリングを行うには、異なる作業負荷の本質が異なっても、全ての作業負荷に対して汎用アルゴリズムを用いる。言い換えると、所与のシステムに対して、システム全体の作業負荷に捕らわれない方針を用いて、システム資源のスケジューリングが実行される。

40

【課題を解決するための手段】

【0004】

本願において特許請求する主題は、先に述べたような環境における欠点をいずれも解消する実施形態、またはその環境だけで動作する実施形態に限定されるのではない。逆に、この背景は、本明細書において記載する実施形態の一部を実用化することができる技術分野の一例を例示するために提唱したに過ぎない。

【0005】

摘要

本明細書において記載する一実施形態は、システム資源のスケジューリング方法を含む。この方法は、システム資源スケジューリング方針を作業負荷に割り当てることを含む。

50

この方針は、作業負荷内部における作業負荷動作のスケジューリングを行うためにある。この方針は、作業負荷に基づいて指定され、方針が作業負荷に特定のようになるようにする。システム資源は、方針によって指定されるように、作業負荷のために予約される。

【0006】

別の実施形態は、システム資源を用いて作業負荷を実行する方法を含む。システム資源は、システム特定方針にしたがって、作業負荷のための予約において予約されており、予約は、作業負荷特定方針を適用するために作業負荷によって用いられる。この方法は、方針を選択することを含む。方針は、作業負荷内部における作業負荷動作のスケジューリングを行うためにある。方針は、作業負荷をシステム資源に配分するために用いられる。作業負荷を実行したときのシステムの使用に関する情報を含むフィードバックを受ける。更に作業負荷をシステム資源に配分するために、フィードバックに基づいて、方針決定を行う。

10

【0007】

更に別の実施形態では、システム資源上において作業負荷を実行する方法を実現する。この方法は、1つ又は複数の作業負荷に対する1つ又は複数のシステム資源スケジューリング方針にアクセスすることを含む。方針は、作業負荷内部における作業負荷動作のスケジューリングを行うためにあり、所与の方針が所与の作業負荷に特定のようになるように、作業負荷に基づいて指定される。方針によって指定されるようにシステム資源の予約を示す実行計画を定式化する。作業負荷は、実行計画に基づいて、システム資源に配分される。

【0008】

20

この摘要は、詳細な説明において以下で更に説明する概念から選択したものを、簡略化した形態で紹介するために設けられている。この摘要は、特許請求する主題の主要な特徴や必須の特徴を特定することを意図するのではなく、特許請求する主題の範囲を判断するために用いることも意図していない。

【0009】

本発明の更に別の特徴および利点は、以下に続く説明に明記されており、部分的にこの説明から自明であり、あるいは本発明の実施によって習得することもできる。本発明の特徴および利点は、添付した特許請求の範囲に特に指定した手段(instrument)および組み合わせによって実現し、獲得することができる。本発明の特徴は、以下の説明および添付した特許請求の範囲から更に一層明らかとなるであろうし、あるいは以下に明記するように、本発明の実施によって習得することができる。

30

【図面の簡単な説明】

【0010】

本発明の先に明記した利点および特徴ならびにその他の利点および特徴を得ることができる態様について説明するために、添付図面に例示する具体的な実施形態を参照して、先に端的に説明した本発明の一層特定の説明を行う。これらの図面は本発明の典型的な実施形態のみを図示するのであり、したがってその範囲を限定するように解釈すべきでないことを理解した上で、添付図面の使用により、実施形態について更に具体的にそして詳細に記載し説明する。

【図1】図1は、階層的作業負荷および方針構造を示す。

40

【図2】図2は、システム資源の予約を示す実行計画を示す。

【図3】図3は、資源管理システムおよびシステム資源を示す。

【図4】図4は、プロセッサ管理の一例を示す。

【図5】図5は、デバイス資源管理部を示す。

【図6】図6は、システム資源の予約方法を示す。

【図7】図7は、予約にしたがってシステム資源を管理する方法を示す。

【図8】図8は、一部の実施形態を実現することができる環境例を示す。

【発明を実施するための形態】

【0011】

本明細書において、実施形態の一部は、種々のコンピューター・ハードウェアを含む特

50

殊目的コンピューターまたは汎用コンピューターを備えることができる。これについては、以下で更に詳しく論ずることとする。また、実施形態の一部は種々の方法要素を含むこともできる。

【0012】

作業負荷に特定した方針にしたがって、作業負荷動作のためのシステム資源予約方針を摘要する実施形態を実現することができる。言い換えると、システム資源のスケジューリングを行う全ての作業負荷に全般的に適用可能な総合的汎用方針にしたがって資源予約を実行するのではなく、所与の作業負荷に特定して指定する方針に基づいて、システム資源をスケジューリングする。加えて、階層的に適用可能な方針にしたがって、作業負荷の予約を遂行することができる実施形態も実現することができる。図1は、数々の実施形態に適用することができる種々の特徴および態様を実現する一実施形態を示す原理の一例を示す。

10

【0013】

図1は、システム資源100を示す。システム資源は、例えば、処理資源、ネットワーク・アダプター資源、メモリー資源、ディスク資源等のようなハードウェアを含むことができる。システム資源は、作業負荷を実行することができる。作業負荷は、プログラムがシステム資源に向けて発生するサービス要求を含む。例えば、プロセッサに該当する作業負荷は、例えば、プロセッサの計算を実行する要求を含む。ネットワーク・アダプター資源に該当する作業負荷は、例えば、ネットワーク送信および受信動作、ネットワーク帯域の使用等を含む。メモリー資源に該当する作業負荷は、例えば、メモリー・リードおよびメモリー・ライトを含む。ディスク資源に該当する作業負荷は、例えば、ディスク・リードおよびディスク・ライトを含む。

20

【0014】

コンテキストに応じて、作業負荷は、ユーザーまたは他のプログラム活動の結果としてプログラムが発生する要求パターンを指すこともあり、異なるレベルの要求粒度(request granularity)を表すこともあり得る。例えば、電子商取引作業負荷は、多数のサーバーに跨る可能性があり、エンド・ユーザーまたは他の業務機能によって生成されるある種の要求資源パターンを伴う。

【0015】

作業負荷は、実行オブジェクトに関して定義することもできる。実行オブジェクトとは、資源を消費する作業負荷抽象化のインスタンスである。例えば、実行オブジェクトは、プロセッサおよびメモリーを消費するスレッド、NIC帯域幅を消費するソケット、ディスク帯域幅を消費するファイル記述子等である場合がある。

30

【0016】

作業負荷に対してシステム資源を予約することができる。図1に示す作業負荷の内2つは、メディア・プレーヤ作業負荷102およびワード・プロセッシング作業負荷104を含む。これらの作業負荷の各々は、それぞれ、メディア・プレーヤ・アプリケーションおよびワード・プロセッシング・アプリケーションを実施する際に用いられる動作を定める。更に、図1は、これら2つの作業負荷は、各々、これらには異なる方針106および108がそれぞれ関連付けられていることも示す。これらの方針は、作業負荷102および104を実行するようにスケジューリングするために、システム資源100をどのように予約すべきかを定めている。

40

【0017】

種々の方針を実現することができる。例えば、1つの方針は、比率基準予約方針である。比率予約は、所定間隔でシステム資源容量の割合の形態で予約を繰り返すことを含む。例えば、比率予約方針は、ある分量のプロセッサ・サイクルを予約すべきことを指定することができる。例えば、比率予約方針では、1,000,000回のプロセッサ・サイクル毎に、その内の2,000サイクルを、その方針が適用される作業負荷に割り当てるべきことを指定することができる。この種の予約は、反復作業負荷に適していることが多い。この方針の一例を、メディア・プレーヤ作業負荷102について示す。この場合、

50

方針106は10ms毎に1msをメディア・プレーヤ作業負荷102のために予約すべきことを指定している。

【0018】

別の方針は、容量に基づく予約に関する。容量予約では、デバイスの容量の百分率を指定し、この容量が利用可能でなければならない時間枠には制約がない。これらの種類の方針の方が柔軟にスケジューリングすることができる。何故なら、予約の保証は時間枠を有さないからである。これの一例を、ワード・プロセッサ作業負荷104について示す。この場合、方針108は、システム資源100の内10%をワード・プロセッサ作業負荷104のために予約すべきことを指定している。

【0019】

注記すべきは、方針106および108はそれらの各アプリケーションに特定のであるということであり、方針は個々のアプリケーション毎に指定されることを意味する。個々のアプリケーション毎の指定は、各アプリケーションを特定の1つの方針に関連付けることによって遂行することができる。別の実施形態では、アプリケーションの種別を方針と関連付けることもできる。本明細書に開示する実施形態の範囲内では、他の集合体を実現することもできる。

【0020】

図1に示すように、各予約は、更に詳細予約(sub-reservation)に分割することができる。予約および詳細予約を用いると、予約およびデフォルト方針の樹木状階層(tree hierarchy)を作成することができる。この階層のリーフ・ノードは、予約方針を含む。例えば、図1は、メディア・プレーヤ作業負荷102の下に階層状にコーデック作業負荷110および表示作業負荷112があることを示す。これらの作業負荷と関連付けられているのは、それぞれ、方針114および116である。これらの方針114および116は、階層的に、メディア・プレーヤ作業負荷102に対する方針106の下にある。更に、図1は、他の階層的に配列した作業負荷および方針も示す。例えば、コーデック作業負荷118、120、および122は、階層的にコーデック作業負荷110の下にある。同様に、方針124、126、および128は、階層的に方針114の下にある。また、図1は、作業負荷130および132が階層的に作業負荷104の下にあること、そして方針134および136が階層的に方針108の下にあることも示す。

【0021】

図1は、この例では、方針が容量に基づく予約に関して予約を指定することができ、ワード・プロセッサ作業負荷104において示すように、資源の百分率を指定することを示す。ワード・プロセッサ作業負荷104では、全システム資源100の内10%が指定されている。図示のように、この全システム資源の内の10%の予約は、図1に示すような、階層的に下位にある作業負荷間において細分することができる。この場合、方針134は、全システム資源の内6%をUI作業負荷134のために予約すべきことを指定し、方針136は、全システム資源の内2%をフォント助言作業負荷132のために予約すべきことを指定している。更に、図1は、方針106が比率に基づく方針を指定しており、これによって、方針106は、10ms毎に1msをメディア・プレーヤ作業負荷102のために予約すべきことを指定していることも示す。

【0022】

実施形態によっては、2つの容量閾値パラメータ、即ち、ソフトおよびハードを用いて予約を行うこともできる。ソフト・パラメータは、ハード容量以上のシステム資源要求を指定する。ソフト値は、最適な性能を達成するために必要となる容量である。ハード値は、作業負荷を動作させるために必要となる最小限の予約値である。実施形態によっては、予約管理システムがソフト容量要求を満たそうとするが、ソフト容量要求を満たすことができない場合、予約管理システムは代わりにハード値を用いようとすることもある。予約管理システムは、動作のために予約した資源量を削減することによるというようにして、予約を減少させることができる。ハード容量値に見合う容量がデバイスの中にある場合、実施形態によっては、予約管理システムはアプリケーションを実行しないこともある。

【0023】

閾値に加えて、予約を予約緊急度と関連付けることもできる。予約緊急度とは、予約に関する優先度を決定する計量値である。予約緊急度は、システムが過大に割り当てられ、予約管理システムが、保留中の予約の内部分集合にしか資源を割り当てることができない場合に適用可能である。緊急度が高い方の予約が実行しようとする場合、予約管理システムは、緊急度が低い方の予約に、その予約を解除しなければならないことを通知する。この通知は、予約が解除されない場合、アプリケーションの終了に発展する。尚、予約緊急度は必ずしも先取りスケジューリング・メカニズムではなく、むしろ新たな予約が要求されたが資源が入手できないときに適用される割り当て優先度とすることもできる。

【0024】

実行オブジェクトにオブジェクト特定方針予約要件がない場合、デフォルトの方針を用いて、いずれの実行オブジェクトでもスケジューリングすることができる。図1は、方針138、140、および142を含む、多数のデフォルト方針を示す。予約管理システムは、比率予約で予約されていない全てのタイムスロットを、容量予約またはデフォルト予約のいずれかに割り当てる。全てのデバイスに対するデフォルト方針は、システム全体において同一とすることができる。これを行うのは、負荷均衡化動作を簡略化するためである。注記すべきは、デフォルト方針が、残っている容量であればいずれでも含むというような単純なことではない場合もあることである。例えば、比率容量に基づいて、方針108が10%の予約を指定し、方針106が10%の予約を指定している間、デフォルト・スケジューリング方針138は、他に何の予約もない場合、スケジューリングすることができる少なくとも80%のシステム資源を有する。メディア・プレーヤ作業負荷102またはワード・プロセッサ作業負荷104の一方または双方が、それぞれの予約量全部を必要とせず、システム資源予約の一部を、デフォルト・スケジューリング方針138が利用するために戻すと決定した場合には、デフォルト方針138に入手可能な資源は80%よりも多くなることもあり得る。

【0025】

デフォルト予約を、資源割り当ての備忘(reminder)を扱う方針と関連付けることもできる。ルート・ノードと同様に、各詳細予約は、このコンテキストにおいて動作し他に予約要件を有さない実行オブジェクトに対するデフォルト配置方針(default placing policy)を含むことができる。例えば、デフォルト方針140および142は、詳細予約デフォルト・スケジューリングのために用いられる。

【0026】

実行計画とは、資源管理システム・コンポーネントが予約およびデバイス容量に関する情報を捕獲ために用いる抽象概念(abstraction)である。具体的には、実行計画は、ディスパッチャ(dispatcher)によって作用する資源予約を表す下位計画である。実行計画の一例を図2に示す。実行計画200は、予約によって指定されたシステム資源のスケジューリングを示す。例示した実行計画200は、プロセッサのようなシステム資源についての、時間に基づく実行計画である。この例では、時間に基づく実行計画を例示するが、他のデバイスには、異なる実行計画を実施してもよいことは認められてしかるべきである。例えば、ネットワーク・デバイスに対する実行計画は、通信経路上で送られる一連のパケットで表すことができる。他の例には、メモリーのヒープのスライス、ディスクのブロック等が含まれる。ここで時間に基づく例に戻ると、実行計画は、一連のタイム・スライスであり、当該タイム・スライスを消費する役割を担う個々の方針によって管理される。予約タイム・スライスを所有する方針は、割り当て量(quantum)を用いて、更に細かい粒度にした間隔に予約の時間を区切り(time-slice)、それが管理する実行オブジェクト間で多重化することができる。スライスの粒度は、デバイスのコンテキストによって異なる。例えば、プロセッサは、タイマーの分解能、パケット・サイズに左右され、NICはパケット・サイズに左右され、メモリーはヒープ・サイズに左右され、ディスクはブロックに左右される等となる。

【0027】

実行計画 200 は、メディア・プレーヤ作業負荷 102 のための第 1 予約 202 と、ワード・プロセッサ作業負荷 104 のための第 2 予約 204 とを示す。この図示する例では、実行計画 200 は、特定の作業負荷のために予約されている資源の時間期間を例示する。この例では、予約 202 および 204 は周期的に繰り返すように示されているが、予約のスケジューリングを行うために用いられる方針によっては、他の割り当ても実施することができる。例えば、予約 202 は、10ms 毎に 1ms をメディア・プレーヤ作業負荷 102 のために予約する必要性のために、本質的に周期性が強くてよい。一方、予約 204 は、作業負荷をスケジューリングする方針が単にシステム資源の 10% を指定するだけであるため、もっと柔軟性があってもよい。

【0028】

10

実行計画は、様々な機能のために用いることができる。一例では、実行計画は、十分なデバイス容量が新たな予約に利用可能か否か査定するために用いることができる。例えば、実行計画 200 は、時間軸(time basis)上における利用可能なシステム資源の指示 206 を含む。予約要求を受けたとき、この指示 206 を参照して、予約要求に対応できるか否か判断することができる。

【0029】

また、実行計画は、比率予約要求を満たすための間隔が得られるか否か査定するために用いることもできる。予約要件を満たすだけの十分な容量をデバイスが有していても、既存の比率予約と競合している場合、予約の頻度および継続期間を満たすのに適したスロットが得られない場合がある。

20

【0030】

また、実行計画は、予約管理部が新たな方針のコンテキストを選択するために効率的に概観する(walk through)ことができる一連の動作を作成するために用いることもできる。これについては、図 3 の説明と共に以下で更に詳しく論ずる。

【0031】

実行計画の計算は、新たな予約をデバイスに割り当てるときまたは予約構成(reservation configuration)が変更になったときに、広範な動作が生ずることが多い。一実施形態では、デバイス資源管理部によって計画を計算する。

【0032】

予約は、デバイスの種類に特定の容量計量値を用いる。この計量値は、資源およびオペレーティング・システムのコンフィギュレーションとは独立でなければならない。しかしながら、オペレーティング・システムはデバイスの容量に関する情報を提供することもできる。

30

【0033】

容量予約は、実行計画の一部として静的にスケジューリングすることができ、あるいは予約管理部によって割り当てられるタイム・スライスのように動的にスケジューリングすることもできる。静的な予約は、例えば、資源の動的な評価(evaluation)および査定(assessment)とは逆に、予め割り当てた分割部分を割り当てることを含むことができる。静的割り当ては、資源管理部の性能オーバーヘッドを低く抑える利点がある。動的割り当ては、スケジューリング階層の同じレベルのデフォルト方針において実行する負荷を扱う際に柔軟性を高めることができる。

40

【0034】

これより図 3 を参照すると、予約管理アーキテクチャ・システム 300 が示されている。先に説明したスケジューリング階層は、全てのデバイスについて従う共通のスケジューリング・パラダイムとすることができる。しかしながら、階層の深さおよび広さ、ならびに方針の複雑度は、デバイス毎に異なる。

【0035】

予約管理アーキテクチャ・システム 300 のコンポーネントは、2つのカテゴリー、即ち、記憶部および手順に編成されている。これらのコンポーネントは、方針、デバイス種類のいずれかに特定のであるか、あるいは汎用である。図 3 において、方針コンポーネン

50

ト群と一緒に纏められている。他の全ての手順は、デバイス種別に特定のである。記憶部は、方針状態記憶部 302 は例外であるが、システムのデバイス全てに共通である。以下の一連の動作は、典型的なスケジューリング・セッションにおいて実行され、新たな実行オブジェクトを予約管理システム 300 に投入することから開始する。

【0036】

1 に示すように、方針 304-1 にしたがって、新たな実行オブジェクトを予約管理システム 300 に投入する。配置アルゴリズム 306 が、この実行オブジェクトを、方針状態記憶部 302 に格納されているキューの 1 つに移動させる。方針状態記憶部 302 は、実行の優先順位または状態を表すことができるキューを含む、方針の内部状態を格納する。

10

【0037】

2 に示すように、配置アルゴリズム 306 は、方針配分アルゴリズム 308 をコールする。方針配分アルゴリズム 308 は、次の実行オブジェクトを実行のために選別する。

【0038】

3 において、デバイス・ディスパッチャ 310 をコールし、実行のために選択された実行オブジェクトに対してコンテキスト切替を行う。ディスパッチャ 310 は、方針 304-1 または方針 304-1 から 304-N までのいずれとも別個にかつ独立して実装されている。即ち、ディスパッチャ 310 は、適用する方針とは無関係に用いることができる。

【0039】

4 において、予約管理システム 300 のディスパッチャ 310 は、システム資源 312 に前述の実行オブジェクトを実行させる。注記すべきは、システム資源 312 は予約管理システム 300 とは別個であってもよいことである。デバイスのコンテキストに応じて、実行オブジェクトの実行を保留にするかまたは完了する。例えば、プロセッサの例では、プロセッサに対して割り当てられたタイム・スライスが終了するか、または実行オブジェクトは待機しておりしかも凍結されているか、または実行オブジェクトは自発的に放棄される。

20

【0040】

5 に示すように、方針状態遷移手順 314 を呼び出し、実行オブジェクト記憶部 316 および方針状態記憶部 302 において実行オブジェクトの状態を更新する。

30

【0041】

6 に示すように、時間アカウンティング手順(time accounting procedure) 318 が、資源コンテナ記憶部 320 を用いて、実行オブジェクトの使用統計を更新する。資源コンテナは、タスクを達成するために作業負荷によって用いられるシステム資源を論理的に收容する抽象概念である。例えば、資源コンテナは、ホストされるアプリケーションのコンポーネント全てについて定めることができる。資源コンテナは、アプリケーションによる資源の使用に関する統計情報を格納する。

【0042】

7 において、予約管理部 322 は、次の予約がどれであるか判断し、次の方針を実行するために然るべきスケジューラ・コンポーネントを呼び出す。これを達成するには、一実施形態では、図 2 に示す実行計画のような実行計画全体を調べる。図 3 に示す例では、この動作には 2 つの潜在的な成果がある。第 1 の成果は、図 2 に示したタイム・スライスの内の 1 つのようなスライス、またはパケット・スライス、ヒープ・スライス、ブロック・スライス等のその他のスライスが適宜スケジューリング階層の現在のレベルにおける現在の方針に割り当てられる。図 3 における 8 B に示すように、現在の方針の配分アルゴリズム 308 をコールする。第 2 の成果は、方針 304-2 のような異なる方針または 304-N までの他のいずれかの方針を用いての別の予約への切替を含み、ここで N は表される方針の数である。予約管理部 322 は次の予約（図では 8 A として示す）の実行計画に切り換えて、新たな計画を用いて同じ動作を実行する。

40

【0043】

50

総合実行オブジェクト記憶部 316 は、スケジューリング方針（例えば、304-1）からはアクセスできない場合もあるが、逆に方針によって現在管理されている実行オブジェクトのビューは見るることができる。潜在的な性能利得に加えて、これは、方針が、そのコンテキストにおいてスケジューリングされていない実行オブジェクトの状態を修正しようとしないうことを保証する。1つのデバイスから他のデバイスへの負荷均衡動作は、異なるデバイス上において実行している予約間において実行オブジェクトを移動させることによって遂行することができる。状態遷移手順 314 および配分手順 310 は、方針状態記憶部 302 と実行オブジェクト記憶部 316 との間における不一致を検出し、補正処置を講ずる。殆どの場合、補正処置は余分なスケジューリング動作を実行することを伴う。

【0044】

10

これより図 4 を参照すると、プロセッサ・スケジューラの潜在的实施態様が示されている。注記すべきは、他の実施態様、ならびにネットワーク資源、メモリー資源、ディスク資源等のような異なるシステム資源に対する別の実施態様も実現できることである。図 4 に提案するスケジューリング・フレームワークでは、共通インフラストラクチャによって調整される多数のスケジューリング方針によって、プロセッサのスケジューリングが行われる。インフラストラクチャによって提供されるプロセッサ・スケジューラのコンポーネント、および方針によって提供されるコンポーネントを図 4 に示す。プロセッサのコンテキストでは、以下の機能が実装される。タイマー・サポート、コンテキスト切換、および凍結通知。

【0045】

20

プロセッサ・スケジューラ・コンポーネントは、任意期間のタイマー割り込み（固定量ではなく）を定めることができてしかるべきである。タイマー割り込みのコンテキストは、予約、またはその予約に対応する方針からの予約の更なる細分のいずれかとすることができる。例えば、優先順位に基づく方針は、現在の予約のコンテキスト内において固定の割当量(quantum)を定めることができる。特定の時点において、多数のタイマーの期限が存在し、プロセッサ・スケジューラ・コンポーネントは、次の期限を指定し、コンテキストを設定し、その割り込みに対応するためにしかるべきスケジューラ・コンポーネントをコールすることによって、種々のタイマー割り込みを管理することができなければならない。タイマー間隔管理部 404 は、スケジューラ時間コンテキストのスタックを維持し、このスタックにおいて次に最も近いタイム・スライスを用いて、タイマー割り込みのスケジューリングを行う。タイマー・コンテキストは、多数の情報を含む。例えば、タイマー・コンテキストは、コンテキストの種類に関する情報を含む。これは、具体的に、スケジューリング方針によって定められた予約または実行オブジェクトタイム・スライスを指す。タイマー・コンテキストは、タイマー割り込みが発生する時間間隔に関する情報を含む。タイマー・コンテキストは、予約については現予約管理部 400 へのポインタ、またスケジューリング方針については状態遷移管理部 412 へのポインタを含む。タイマー・コンテキストは、予約について、現在の実行計画へのポインタを含む。

30

【0046】

タイマー割り込みディスパッチャ 408 は、タイマー割り込みによって誘起され、そして先取の種類およびタイマー・コンテキストに応じて、スケジューリング機能のスケジューリング・エントリ・ポイントをコールする。ある実行オブジェクトに対するタイム・スライスが終了した場合、または実行オブジェクトが凍結された場合、現在の状態遷移管理部がコールされ、最終的に、予約コンテキストの中で次の実行オブジェクトをスケジューリングする。予約に対するタイム・スライスが終了した場合、次の予約および方針を選択するために、現在の実行計画コンテキストによって、予約管理部をコールする。

40

【0047】

図 4 は、プロセッサ・スケジューラ・コンポーネントの典型的な制御フローを示す。1Aにおける新たな予約の場合において示すように、予約管理部 400 は新たなタイマー・コンテキスト・オブジェクトを作成する。このタイマー・コンテキスト・オブジェクトは、予約時間間隔、それ自体のコールバック・エントリ・ポイント、および現実行計画に

50

対する参照を含む。1 Bにおける実行オブジェクトのスケジューリングの場合、ディスパッチャ402は、実行オブジェクト時間間隔、および状態遷移管理部コールバック機能へのポインタを有するコンテキストを作成する。2に示すように、時間間隔管理部404は、要求のコンテキストをタイマー・コンテキスト・スタック406に押し込む。3において、時間間隔管理部404は、最も近いタイム・スライスを発見し、タイマー割り込みディスパッチャ408にコンテキストを設定し、タイマー410をプログラムする。4において、タイマー410からのタイマー割り込みが発生し、タイマー割り込みディスパッチャ408を呼び出す。5において、タイマー割り込みディスパッチャ408はそのコンテキストを調べ、予約が終了した場合には予約管理部400コールバック機能をコールし、あるいは実行オブジェクトタイム・スライスが終了した場合には、状態遷移管理部412をコールする。6において、状態遷移管理部412をコールした後、実行オブジェクト・スケジューリング制御フローを実行し、本プロセスにおける別の繰り返しのためにディスパッチャ402をコールする。

10

【0048】

これまでは、1つのデバイスのスケジューリング・インフラストラクチャの設計を説明の中心とした。しかしながら、実施形態は、デバイス資源管理部によって多数のデバイスを管理する機能を含むこともできる。これは、多数の共有プロセッサを用いるマルチコア・デバイスや、多数のオペレーティング・システムを用いるハイパーバイザー技術(hypervisor technologies)が近年普及していることを考えると、特に有用であると言える。

【0049】

20

一実施形態では、デバイス資源管理部は、同じ種類のデバイス群に跨るタスクを実行する役割を担う。予約のデバイスへの割り当て、負荷均衡化、および負荷軽減というような動作は、通例、デバイス資源管理部によって実行する動作である。実施形態によっては、これを遂行するにあたって、異なるデバイスに合わせて実行計画を修正する場合もあり、更に予約を実行計画間で移動させることを含む場合もある。デバイス資源管理部は、デバイス・スケジューラのコンポーネントと比較すると、呼び出される頻度は相対的に低い。したがって、相対的に費用がかかる動作を実行することができる。

【0050】

実施形態によっては、デバイス資源管理部によって実行する動作は、4つのカテゴリーに分けることができ、これらのカテゴリーについてここで論ずる。最初のカテゴリーは、予約のデバイスへの割り当て、およびデバイス・スケジューラのための実行計画の作成である。予約割り当てが行われるのは、アプリケーションが新たな予約を要求したとき、または予約構成設定(configuration)が行われたときである。デバイス資源管理部は、最初に、デバイスの利用可能な容量を調べ、予約をデバイスに割り当てる。容量に加えて、ある種の作業負荷の実行を妨げる可能性もある、デバイスの電力状態、および性能というような、他にも潜在的な考慮事項がある。デバイス資源管理部は、予約緊急方針を適用する役割を担う。これは、予約のために利用可能な資源がない場合に該当する。新たな予約の予約緊急度が、既存の予約(群)と比較され、デバイス資源管理部は、緊急度が低い方の予約のアプリケーション(群)に、その予約を取り消すように通知するか、あるいは一定の時間枠内に遂行しない場合これらを終了させる。クォータ(quotas)は、特殊な方針である。クォータとは、作業負荷の資源使用を制限することを目的とするシステム強制方針である。クォータには、上限(cap)および累増(accrual)という2つの格別の種類が含まれる。上限クォータは、閾値として作用して、資源の利用を一定の限度に抑える。例えば、あるアプリケーションには、プロセッサ容量の10%の上限がある場合がある。累増クォータは、更に長い時間期間における資源の総使用に対する制限である。例えば、ホストされているウェブ・サイトは1回の請求期間では5GBよりも多いネットワーク帯域幅を使用してはいけないことを、1つの累増クォータが指定することができる。累増クォータにおいて用いられる同じ通知を、予約先取りの場合に適用することができる。資源不足および低い関連緊急度のために実行されない予約要求は、整列させて、資源が解放されたときに割り当てることができる。

30

40

50

【0051】

予約の割り当てが決定した後、デバイス資源管理部は、デバイスに合わせて実行計画を計算し直さなければならない。実施形態によっては、デバイス・スケジューリング階層のルート実行計画(root execution plan)の再計算だけで済む場合もある。また、デバイス資源管理部は、実行計画計算サービスを、デバイス・スケジューリング階層のルート・レベル以外において一次予約を細分することが必要なスケジューラに提供することができる。

【0052】

また、デバイス資源管理部は、同じ予約が多数のデバイスにおいて同じ開始時刻に行われなければならない集団スケジューリング(gang scheduling)をサポートすることもできなければならない。この特徴は、特に、同期化を必要とする場合もあるスレッド群の同時実行を必要とする場合もある同時ランタイム(concurrency run-times)には有用である。異なるデバイス上で全てのスレッドを同時に実行することにより、同期化が行われたときにこれらの全てが実行するので、調整コストは最小限に抑えられる。

【0053】

また、デバイス資源管理部は、デバイス・スケジューリング階層のルート・ノードに対するデフォルト・スケジューリング方針において実行する実行オブジェクトの負荷を均衡化する役割も担う。この動作は、異なるデバイスの方針状態記憶部間で実行オブジェクトを移動させることによって、実行計画間で実行オブジェクトを移動させることを伴う。これを遂行するには、この動作に参与するデバイスの実行オブジェクト・ビューを変更する。負荷均衡化についての判断には、レイテンシの考慮というような、オペレーティング・システムにおける発見法を必要とすることもあり得る。

【0054】

デバイス資源管理部は、システム資源を監視し、上限クォータ閾値を適用する。これは、デバイス資源管理部と方針ディスパッチャとの協力を必要とする動作である。デバイス資源管理部は、方針に提示された実行オブジェクト・ビューから実行オブジェクトを除外することによって、既定期間実行オブジェクトを保留にする。

【0055】

本例では、デバイス資源管理部は、オペレーティング・システム・サービスを用いて、デバイスを数え上げて、デバイス・コンフィギュレーションを検査し、容量および可用性を判断する。デバイス資源管理部が動作するためにオペレーティング・システムによって用いられるサービスは、ここではシステム資源管理部と呼ばれるコンポーネントに編入される。デバイス資源管理部は、デバイス・スケジューラおよび負荷均衡化動作の開始および終了に関して特殊な動作を必要とするハードウェア障害、ホット・スワップ等のために、システム資源管理部のイベント通知システムを利用する。

【0056】

図5は、管理システム500のコンポーネントを示す。この例では、デバイス資源管理部510は4つの注目すべき動作を実行する。第1の動作は、実行計画の計算を含む。1に示すような、新たな予約に対して、和合性(affinity)計算部502は予約を実行するのに適したデバイスを選択する。予約和合性計算部502は、実行計画計算部504をコールして、そのデバイスに合わせた新たな実行計画を求め、次いでこの実行計画を、選択したデバイスの予約管理部506に受け渡す。予約構成の変更または既存の予約の細分の場合、和合性計算を飛び越す。

【0057】

第2の動作は、ハードウェア変更に関係する。2に示すように、ソフトウェア資源管理部508は、デバイス資源管理部510に、予約および実行オブジェクト軽減手順512を通じて、変更が発生したことを通知する。デバイス資源管理部510は、次いで、ハードウェアの変更に応じて、デバイスに現在割り当てられている予約および実行オブジェクトを軽減する。例えば、デバイスが低電力モードに移行しようとしている場合、実行オブジェクトおよび予約を他のデバイスに割り当てし直すことができる。実行計画計算部50

4をコールして、影響を受けるデバイスの実行計画を計算し直す。

【0058】

第3の動作は、負荷均衡化に関係する。3に示すように、実行オブジェクト負荷均衡部514は、関与するデバイスの実行オブジェクト・ビューを変更することによって、ルート・デバイス・スケジューリング階層においてデフォルトの方針で実行している実行オブジェクトを割り当てし直す。

【0059】

第4の動作は、上限クォータ強制に関係する。4に示すように、上限クォータ・エンジン516は、実行オブジェクトがその閾値を超過したか否か判断する。違反が検出された場合、実行オブジェクト記憶部518において実行オブジェクトの状態を修正する。方針の実行オブジェクト・ビューからこの実行オブジェクトを除外することによって、所定の時間量だけこの実行オブジェクトを保留にする。上限クォータ・エンジン516は、方針ビューにおいてこの実行オブジェクトを確立し直す。実行オブジェクトが現在実行中である場合、上限クォータ・エンジン516はこの実行オブジェクトにフラグを立てて、方針時間アカウントコンポーネントによって、ビューの変更を行う。

【0060】

これより図6に移ると、方法600が示されている。方法600は、システム資源のスケジューリングを行うための行為を含むことができる。この方法は、作業負荷のためにシステム資源スケジューリング方針にアクセスすることを含む（行為602）。この方針は、作業負荷の動作をスケジューリングするためにあり、方針が作業負荷に特定のとなるように、作業負荷に基づいて指定される。例えば、図1における多くの例の中に示すように、方針106は作業負荷102に特定のとなっている。一実施形態では、作業負荷はシステムの方針を用いて、当該作業負荷を実行するために用いられる作業負荷特定方針に基づいて、この作業負荷のための予約のスケジューリングを行うことができる。

【0061】

更に、方法600は、方針によって指定された作業負荷のためにシステム資源を予約する行為（行為604）を含む。これの一例を実行計画200に示す。ここでは、予約202および204が、作業負荷特定方針のために実施される。

【0062】

更に、方法600は、システムのデフォルト・スケジューリング方針を用いて、他の作業負荷のために、スケジューリングされていない残りのシステム資源の少なくとも一部を保存することも含むことができる。図2は、206において、システム・デフォルト・スケジューリング方針を用いた予約を示す。

【0063】

方法600の実施形態によっては、この作業負荷が階層的に別の作業負荷の下になる場合がある。例えば、図1は、例の中でもとりわけ、作業負荷102よりも階層的に下になっている作業負荷110および112を示す。一実施形態では、作業負荷のためにシステム資源を予約すること（行為604）は、この作業負荷に対する方針、およびこの作業負荷よりも階層的に高い作業負荷に対する方針双方によって指定されて実行する。一例をあげると、作業負荷110のための予約は、方針114および方針106双方に基づいてスケジューリングすることができる。

【0064】

方針は、多数の異なるパラメータに関して指定することができる。例えば、方針は、比率による資源の予約、容量による資源の予約を指定すること、または期限による資源の予約を指定することができる。

【0065】

一実施形態では、方針によって指定されたように作業負荷のためにシステム資源を予約する際（行為604）、多数のシステム資源について実行計画を参照することが含まれ、多数のシステム資源の中からのシステム資源の各々は、同じデバイス種別を含む。例えば、システムが多数の異なるプロセッサを含むこともあり得る。実行計画に基づいて、複

数のシステム資源の中で作業負荷の負荷均衡化を目指すように、システム資源の予約を行う。代替実施形態では、システム資源の予約は、1つのデバイスから他のデバイスへの作業負荷を軽減することを目指して行われる。例えば、システムからデバイスを除外しようとする場合、または容量が少なくなる低電力状態にデバイスを移行させる場合、またはその他の理由で、このようなデバイスから利用可能な容量がある他のデバイスに、作業負荷を移行させることが望ましいこともある。別の代替実施形態では、上限クォータを強制することを目指して、システム資源の予約を実行する。

【0066】

これより図7を参照すると、他の方法700の実施形態が示されている。方法700は、例えば、コンピューティング環境において実用化することができる。この方法は、システム資源を用いて作業負荷を実行する行為を含む。システム資源は、システム特定方針にしたがって、作業負荷のために既に予約されている。これらの方針は、作業負荷の動作をスケジューリングするためである。この方法は、方針を選択することを含み、この方針はある作業負荷に特定のであり（行為702）、この方針を用いて作業負荷をシステム資源に配分し、方針にしたがって作業負荷を実行し（行為704）、作業負荷を実行した場合システムの使用に関する情報を含むフィードバックを受け取り（行為706）、このフィードバックに基づいて、更に作業負荷をシステム資源に配分するために、方針決定を行う（行為708）ことを含む。これの一例を図3に示す。図3は、方針304-1から方針304-Nまでをどのようにしてディスパッチャ310と共に用いて、システム資源312によって作業負荷を実行させるかを示す。

【0067】

方法700において、方針決定を行う（行為708）のは、実行計画に基づくことができる。実行計画は、作業負荷のためのシステム資源の予約を定める。例えば、システム資源312上において作業負荷を実行した後、実行計画200のような実行計画を参照して、予約202および204の内の1つのような予約と比較して、作業負荷をシステム資源312上で実行した時間量に基づいて、方針変更を行うべきか否か判断することができる。

【0068】

本明細書において記載した実施形態の一部は、以前に実現されたスケジューリング・システムに対して1つ又は複数の利点を提供することができる。例えば、実施形態の中には、特殊化に配慮したものがある。即ち、システム資源のスケジューリングは、作業負荷の要件を満たすように、カスタム化が可能であるとよい。1つのスケジューリング方針が全ての作業負荷の要件を満たすことができない場合もある。本明細書における実施形態の中には、デフォルトの方針を用いるか、またはアプリケーションに合わせて特定の設計した新たなスケジューリング方針を定める選択肢を、作業負荷が有する場合もある。

【0069】

実施形態の中には、拡張性に配慮したものもある。本明細書に記載されている実施形態を用いると、スケジューリング方針は、作業負荷要件を取り込むように拡張可能にすることもできる。この属性は、所望の特殊化の実現に配慮する。デフォルトのシステムが供給する方針に加えて、資源管理インフラストラクチャは、単に既存の方針から選択するだけでなく、作業負荷がこれらの方針を指定できるように、挿入可能な(pluggable)方針アーキテクチャを提供することができる。

【0070】

実施形態の中には、一貫性に配慮したものもある。同じ資源管理インフラストラクチャを異なる資源に用いることができる。スケジューリング・アルゴリズムは、通例、デバイス種別の要件を満たすように特殊化されている。プロセッサ、ネットワーク、およびディスク・スケジューラは、異なるアルゴリズムを用いることもでき、オペレーティング・システムの異なる要素に実装されることもできる。しかしながら、実施形態の中には、全てのスケジューラが、コンポーネントを特徴化するために同じモデルを用い、更に同じ統計およびクォータ・インフラストラクチャを用いることができる場合もある。

【0071】

実施形態の中には、周期性に配慮したものもある。作業負荷の部分集合の応答性は、システムの負荷およびスケジューリング方針とは独立しているとよい。オペレーティング・システムは、レイテンシに敏感なアプリケーションに対して、システム資源の既定部分を保証できるとよい。

【0072】

実施形態の中には、適応性に配慮したものもある。スケジューリング方針は、システムの動的挙動を取り込むように修正することができる。挿入可能な方針スケジューリングモデルによって、上位のシステム・コンポーネントおよびアプリケーションが、それらのシステム性能を調和させるように方針を調節することが可能になる。

10

【0073】

また、実施形態は、コンピューター実行可能命令またはデータ構造を搬送するまたは格納したコンピューター読み取り可能媒体を含むこともできる。このようなコンピューター読み取り可能媒体は、汎用コンピューターまたは特殊目的コンピューターによってアクセスすることができる入手可能な媒体であればいずれでも可能である。一例として、そして限定ではなく、このようなコンピューター読み取り可能媒体は、RAM、ROM、EEPROM、CD-ROMまたはその他の光ディスク・ストレージ、磁気ディスク・ストレージまたはその他の磁気記憶デバイス、あるいは所望のプログラム・コード手段をコンピューター実行可能命令またはデータ構造の形態で搬送または格納するために用いることができし、汎用コンピューターまたは特殊目的コンピューターがアクセス可能なその他のいずれの媒体も含むことができる。ネットワークまたはその他の通信接続（有線、ワイヤレス、あるいは有線またはワイヤレスの組み合わせのいずれか）を通じて情報をコンピュータに転送または提供する場合、コンピューターはこの接続をコンピューター読み取り可能媒体と見なすことは適正である。つまり、このような接続はいずれも、コンピューター読み取り可能媒体と呼ぶことは適正である。以上の組み合わせも、コンピューター読み取り可能媒体の範囲に含まれてしかるべきである。

20

【0074】

コンピューター実行可能命令は、例えば、命令およびデータを備えており、汎用コンピューター、特殊目的コンピューター、または特殊目的処理装置に、一定の機能または機能の一群を実行させる。本主題は、構造的特徴および／または方法論的行為に特定した文言で記載したが、添付する特許請求の範囲において定義する発明は、本明細書において記載した具体的な特徴や行為のいずれにも必ずしも限定されないことは言うまでもない。むしろ、これら具体的な特徴および行為は、特許請求の範囲を実現する形態例として開示したに過ぎない。

30

【0075】

図8および以下の論述は、本発明を実現可能とするのに適したコンピューティング環境の端的な総合的な説明を行うことを意図している。必須ではないが、本発明の説明は、ネットワーク環境においてコンピューターが実行するプログラム・モジュールのような、コンピューター実行可能命令という一般的なコンテキストで行う。一般に、プログラム・モジュールは、ルーチン、プログラム、オブジェクト、コンポーネント、データ構造等を含み、特定のタスクを実行するか、または特定の抽象的データ・タイプを実現する。コンピューター実行可能命令、付随するデータ構造、およびプログラム・モジュールは、本明細書に開示した方法のステップを実行するプログラム・コード手段の例を表す。このような実行可能命令または付随するデータ構造の特定のシーケンスは、このようなステップにおいて記載した機能を実施するための対応する行為の例を表す。

40

【0076】

本発明は、多くの種類のコンピューター・システム構成を有するネットワーク・コンピューティング環境においても実施できることは当業者には認められよう。コンピューター・システム構成には、パーソナル・コンピューター、ハンドヘルド・デバイス、マルチプロセッサ・システム、マイクロプロセッサ主体またはプログラマブル消費者用デバイ

50

ス、ネットワークPC、ミニコンピュータ、メインフレーム・コンピュータ等が含まれる。また、本発明は分散型システムにおいても実施することができ、この場合、ローカルおよびリモート・処理装置が、（有線接続データー・リンク、ワイヤレス・データー・リンク、または有線接続およびワイヤレス・データー・リンクの組み合わせにいずれかによって）通信ネットワークを通じてリンクされており、双方がタスクを実行する。分散型・環境では、プログラム・モジュールは、ローカルおよびリモート双方のメモリー記憶デバイスに配置することもできる。

【0077】

図8を参照すると、本発明を実施するシステム例は、コンピューター820の形態とした汎用コンピューターを含む。コンピューター820は、図示のように多数のプロセッサを含んでもよい演算装置821、システム・メモリー822、およびシステム・バス823を含むことができる。システム・バス823は、システム・メモリー822を含む種々のシステム構成要素を演算装置821に結合する。システム・バス823は、メモリー・バスまたはメモリー・コントローラ、周辺バス、および種々のバス構造のいずれかを用いるローカル・バスを含む数種類のバス構造のいずれでもよい。システム・メモリーは、リード・オンリ・メモリー（ROM）824およびランダム・アクセス・メモリー（RAM）825を含む。基本入／出力システム826（BIOS）は、起動中のように、コンピューター820内のエレメント間におけるデーター転送を補助する基本的なルーチンを含み、ROM824内に格納することができる。

【0078】

また、コンピューター820は、磁気ハード・ディスク839からの読み取りおよびこれへの書き込みを行なう磁気ハード・ディスク・ドライブ827、リムーバブル磁気ディスク829からの読み取りおよびこれへの書き込みを行なう磁気ディスク・ドライブ828、ならびにCD ROMまたはその他の光媒体のようなリムーバブル光ディスク831からの読み取りおよびこれへの書き込みを行なう光ディスク・ドライブ830を含むこともできる。磁気ハード・ディスク・ドライブ827、磁気ディスク・ドライブ828、および光ディスク・ドライブ830は、それぞれ、ハード・ディスク・ドライブ・インターフェース832、磁気ディスク・ドライブ・インターフェース833、および光ドライブ・インターフェース834によって、システム・バス823に接続されている。これらのドライブおよびそれに付随するコンピューター読み取り可能媒体は、コンピューター820のために、コンピューター実行可能命令、データー構造、プログラム・モジュール、およびその他のデーターの不揮発性ストレージを備えている。本明細書に記載する環境例では、磁気ハード・ディスク839、リムーバブル磁気ディスク829、およびリムーバブル光ディスク831を採用したが、データーを格納するための他の種類のコンピューター読み取り可能媒体も用いることができ、磁気カセット、フラッシュ・メモリー・カード、デジタル・バーサタイル・ディスク、ベルヌーイ・カートリッジ、RAM、ROM等が含まれる。

【0079】

プログラム・コード手段は、1つ又は複数のプログラム・モジュールを備えており、磁気ハード・ディスク839、リムーバブル磁気ディスク829、リムーバブル光ディスク831、ROM824、またはRAM825上に格納することができ、オペレーティング・システム835、1つ又は複数のアプリケーション・プログラム836、その他のプログラム・モジュール837、およびプログラム・データー838を含む。ユーザーは、コマンドおよび情報をコンピューター820に、キーボード840、ポインティング・デバイス842、またはマイクロフォン、ジョイスティック、ゲーム・パッド、衛星ディッシュ、スキャナ等のようなその他の入力デバイス（図示せず）によって入力することができる。これらおよびその他の入力デバイスは、多くの場合、システム・バスに結合されているシリアル・ポート・インターフェース846を介して、演算装置821に接続されている。あるいは、入力デバイスは、パラレル・ポート、ゲーム・ポート、またはユニバーサル・シリアル・バス（USB）のようなその他のインターフェースによって接続すること

も可能である。モニター 847 またはその他の形式の表示装置も、ビデオ・アダプター 848 のようなインターフェースを介して、システム・バス 823 に接続されている。モニターに加えて、パーソナル・コンピュータは、スピーカーおよびプリンター等のような、その他の周辺出力装置（図示せず）も含むことができる。

【0080】

コンピューター 820 は、リモート・コンピューター 849 a および 849 b のような 1 つ又は複数のリモート・コンピューターへの論理接続を用いて、ネットワーク環境において動作することも可能である。リモート・コンピューター 849 a および 849 b は、各々、別のパーソナル・コンピューター、サーバー、ルーター、ネットワーク PC、ピア・デバイス、またはその他の共通ネットワーク・ノードとすることができ、通例、コンピューター 820 に関して先に説明したエレメントの多くまたは全てを含むが、図 8 には、メモリー記憶デバイス 850 a および 850 b、ならびにそれらに付随するアプリケーション・プログラム 36 a および 36 b のみを示す。図 8 に示す論理接続は、ローカル・エリア・ネットワーク（LAN）851 およびワイド・エリア・ネットワーク（WAN）852 を含む。ここでは、これらを一例として示するのであって、限定ではない。このようなネットワーク環境は、事務所、企業規模のコンピューター・ネットワーク、イントラネットおよびインターネットにおいては、珍しくはない。

10

【0081】

LAN ネットワーク環境で用いる場合、例えば、コンピューター 820 は、ネットワーク・インターフェースまたはアダプター 853 を介してローカル・ネットワーク 851 に接続する。WAN ネットワーク環境で用いる場合、コンピューター 820 は、モデム 854、ワイヤレス・リンク、またはインターネットのようなワイド・エリア・ネットワーク 852 上で通信を確立するその他の手段を含むことができる。モデム 854 は、内部または外部でも可能であり、シリアル・ポート・インターフェース 846 を介してシステム・バス 823 に接続することができる。ネットワーク環境では、コンピューター 820 に関して図示したプログラム・モジュール、またはその一部をリモート・メモリー記憶装置に格納することができる。尚、図示のネットワーク接続は一例であり、ワイド・エリア・ネットワーク 852 を通じて通信を確立する他の手段も使用可能であることは認められよう。

20

【0082】

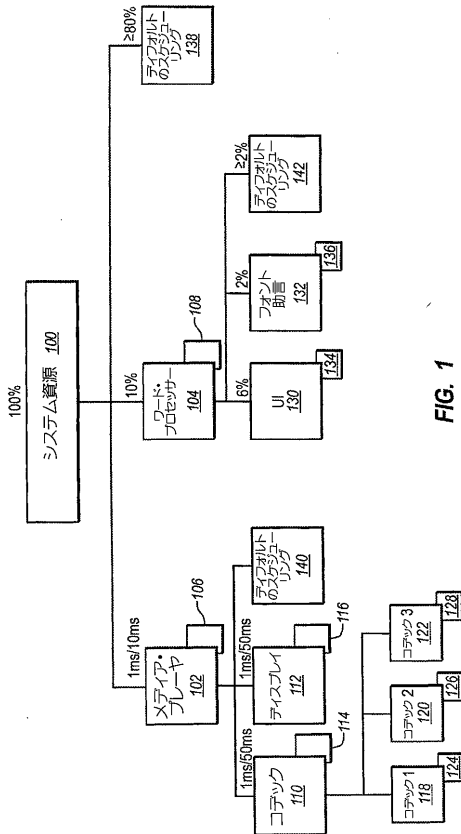
実施形態は、以上で論じた資源に合わせて作業負荷を処理する機能を含むことができる。この処理は、本明細書において既に説明したような、作業負荷特定方針を用いて遂行することができる。

30

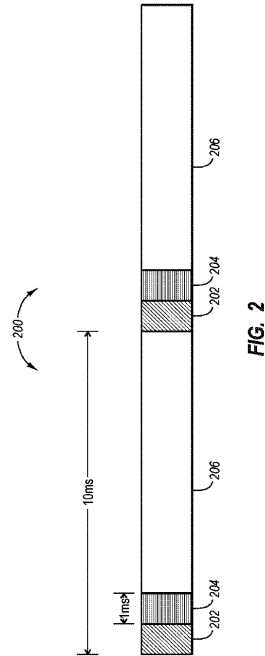
【0083】

本発明は、その主旨または本質的な特徴から逸脱することなく、他の具体的な形態においても具体化することができる。記載した実施形態は、あらゆる観点において、例示に過ぎず、限定としては見なさないものとする。したがって、本発明の範囲は、前述の説明ではなく、添付した特許請求の範囲によって示されることとする。特許請求の範囲の均等の意味および範囲に該当する全ての変更は、その範囲に含まれるものとする。

【図 1】



【図 2】



【図 3】

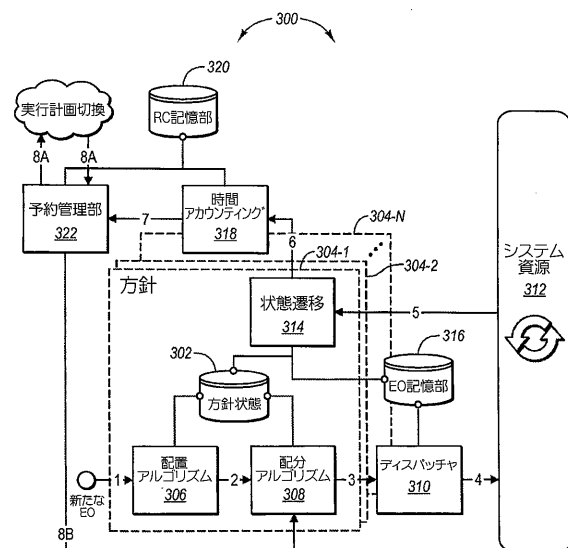


FIG. 3

【図 4】

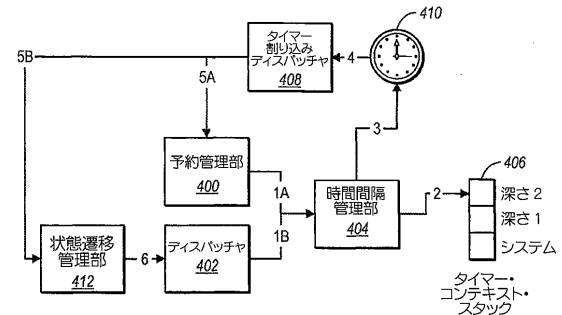


FIG. 4

【図 5】

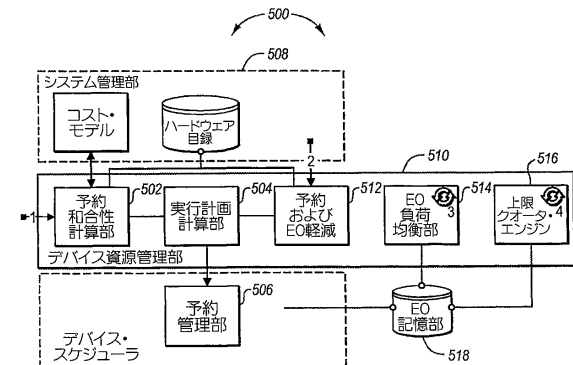


FIG. 5

フロントページの続き

(74)代理人 100153028

弁理士 上田 忠

(72)発明者 パパエフスタシオウ, エフスタシオス

アメリカ合衆国ワシントン州98052, レッドモンド, ワン・マイクロソフト・ウェイ, マイクロソフト コーポレーション, インターナショナル・パテント

(72)発明者 トロウブリッジ, シーン・イー

アメリカ合衆国ワシントン州98052, レッドモンド, ワン・マイクロソフト・ウェイ, マイクロソフト コーポレーション, インターナショナル・パテント

(72)発明者 トリブル, エリック・ディーン

アメリカ合衆国ワシントン州98052, レッドモンド, ワン・マイクロソフト・ウェイ, マイクロソフト コーポレーション, インターナショナル・パテント

(72)発明者 オクス, スタニスラヴ・エイ

アメリカ合衆国ワシントン州98052, レッドモンド, ワン・マイクロソフト・ウェイ, マイクロソフト コーポレーション, インターナショナル・パテント

審査官 井上 宏一

(56)参考文献 特表2001-515242 (JP, A)

特開2005-209203 (JP, A)

特表2005-513587 (JP, A)

(58)調査した分野(Int.Cl., DB名)

G06F 9/46 - 9/54