

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6025856号
(P6025856)

(45) 発行日 平成28年11月16日(2016.11.16)

(24) 登録日 平成28年10月21日(2016.10.21)

(51) Int. Cl.		F I			
G06F	9/445	(2006.01)	G06F	9/06	610A
G06F	9/46	(2006.01)	G06F	9/46	350
G06F	9/50	(2006.01)	G06F	9/46	462Z

請求項の数 20 (全 18 頁)

(21) 出願番号	特願2014-543549 (P2014-543549)	(73) 特許権者	390009531
(86) (22) 出願日	平成24年11月20日(2012.11.20)		インターナショナル・ビジネス・マシーンズ・コーポレーション
(65) 公表番号	特表2015-504562 (P2015-504562A)		INTERNATIONAL BUSINESS MACHINES CORPORATION
(43) 公表日	平成27年2月12日(2015.2.12)		アメリカ合衆国10504 ニューヨーク州 アーモンク ニュー オーチャードロード
(86) 国際出願番号	PCT/US2012/066164		New Orchard Road, Armonk, New York 10504, United States of America
(87) 国際公開番号	W02013/078258	(74) 代理人	100108501
(87) 国際公開日	平成25年5月30日(2013.5.30)		弁理士 上野 剛史
審査請求日	平成26年12月24日(2014.12.24)		最終頁に続く
(31) 優先権主張番号	13/304,571		
(32) 優先日	平成23年11月25日(2011.11.25)		
(33) 優先権主張国	米国 (US)		

(54) 【発明の名称】 共有リソース・システム、リソースを割り振る方法、およびコンピュータ・プログラム製品 (共有リソースをプロビジョニングするための、コストを意識したテンプレート選択のためのシステム)

(57) 【特許請求の範囲】

【請求項1】

複数のクライアント・デバイスと、
前記複数のクライアント・デバイスに対してリソース・キャパシティを選択的に利用可能にする1つまたは複数のサービス・プロバイダ・コンピュータと、
前記1つまたは複数のサービス・プロバイダ・コンピュータから前記複数のクライアント・デバイスにリソース・キャパシティが提供される際に経由されるネットワークと、
全ての割り振り可能なリソース・キャパシティ構成についてのリソース・テンプレートを生成し、生成されたリソース・テンプレートのサブセットを選択し、前記サブセットから選択されたいくつかのリソース・テンプレートを維持するリソース・プロビジョニング・ユニットであって、各テンプレートが、割り振り可能なリソース・キャパシティ構成を指定し、各要求元クライアント・デバイスに、前記サブセットの前記リソース・テンプレートのうちの1つによって決定されるリソースが割り振られる、リソース・プロビジョニング・ユニットとを備える共有リソース・システム。

【請求項2】

前記リソース・プロビジョニング・ユニットが、
割り振り可能なリソースのコストおよび関連する属性を記憶する構成ストアと、
前に受け取られた、キャパシティを求める要求を記憶するサーバ要求ストアと、
テンプレート・リスト・オプションを記憶する入力パラメータ・ストアと、
前記前に受け取られた要求を最適なコストで満たすためのリソース・テンプレートのリ

ストを決定するテンプレート・ジェネレータであって、前記リストが、前記サブセットから選択された前記リソース・テンプレートをリストするものであるテンプレート・ジェネレータと、

生成されたテンプレート・リストを記憶するテンプレート・ストアとを備える、請求項 1 に記載の共有リソース・システム。

【請求項 3】

前記テンプレート・ジェネレータが各サービス・プロバイダ・コンピュータに対してテンプレート・リストを生成し、前記生成された各テンプレート・リストが前記テンプレート・ストアに記憶される、請求項 2 に記載の共有リソース・システム。

【請求項 4】

前記テンプレート・リスト・オプションが完全リスト・オプションおよび上位 k リスト・オプションを含む、請求項 2 に記載の共有リソース・システム。

【請求項 5】

前記上位 k リスト・オプションが k の複数の値を示し、前記テンプレート・ジェネレータが前記各サービス・プロバイダ・コンピュータに対して k の全ての値についてテンプレート・リストを生成し、前記生成されたテンプレート・リストのうちの 1 つが前記各サービス・プロバイダ・コンピュータに対して選択されて前記テンプレート・ストアに記憶される、請求項 4 に記載の共有リソース・システム。

【請求項 6】

選択された各テンプレート・リストが、前記各サービス・プロバイダ・コンピュータに対して生成された前記テンプレート・リストの最小コストを有する、請求項 5 に記載の共有リソース・システム。

【請求項 7】

前記各テンプレートが、コアの数と、メモリ空間と、ディスク空間とを含む割振り可能な構成を指定する、請求項 2 に記載の共有リソース・システム。

【請求項 8】

複数のコンピュータからの、リソース・キャパシティを求める要求の集まりを提供するステップと、

割振り可能なコンピュータ・リソースにコストを関連付けるステップと、

全ての割振り可能な構成についてのリソース・テンプレートを前記割振り可能なコンピュータ・リソースから生成するステップと、

各要求を 1 つまたは複数のテンプレートとマッチさせるステップと、

最小コストを有するテンプレートを各要求に対して選択するステップと、

マッチした要求の数に従って、選択された最小コスト・テンプレートをソートするステップと、および、

マッチした数の多い上位 k 個のテンプレートについて、マッチする要求のコストを決定するステップであって、前記 k 個の最も頻りにマッチしたテンプレートが上位 k リスト中でリストされる、前記決定するステップとを含む、リソースを割り振る方法。

【請求項 9】

前記各要求をマッチさせるステップが、

リソース・テンプレート・マッチのリストを生成するステップであって、前記リストが、少なくとも 1 つの要求にマッチする各テンプレートをリストする、前記生成するステップと、

前記リストされた各テンプレートにコストを関連付けるステップと、および、

コストが関連する全てのテンプレートから、最小コストを有するマッチするテンプレートを各要求に対して決定するステップとを含む、請求項 8 に記載のリソースを割り振る方法。

【請求項 10】

最小コストを有する前記マッチするテンプレートを前記各要求に対して選択するステップが、

10

20

30

40

50

各要求に対して最小リソース・マッチを有するテンプレートを見つけるステップであって、前記テンプレートが最小マッチ・テンプレートである、前記見つけるステップと、および、

前記最小マッチ・テンプレートのリソースのコストを決定するステップとを含む、請求項 9 に記載のリソースを割り振る方法。

【請求項 11】

上位 k リストを決定するステップが、

各テンプレートにマッチする要求の数に従って、前記ソートされたマッチするテンプレートをリストするステップと、

各 k 値についてコスト区分を決定するステップと、および、

前記コスト区分を上位 k テンプレート・テーブル中でリストするステップとを含む、請求項 10 に記載のリソースを割り振る方法。

【請求項 12】

前記コスト区分が、各上位 k リストについて、最小コストで満たされる要求、最小コストより高いコストで満たされる要求、および満たされない要求の数を含む、請求項 11 に記載のリソースを割り振る方法。

【請求項 13】

前記上位 k リスト中の、最低コストが関連するテンプレートが、クラウド環境におけるリソースのプロビジョニングに使用される、請求項 12 に記載のクラウド環境におけるリソースを割り振る方法。

【請求項 14】

クラウド環境リソースを割り振る方法であって、

全ての割り振り可能な構成についてのリソース・テンプレートを、割り振り可能なコンピュータ・リソースから生成するステップと、

サーバ要求データベースに含まれる各リソース・キャパシティ要求を、割り振り可能なクラウド・リソースを指定する 1 つまたは複数のテンプレートとマッチさせるステップと、

少なくとも 1 つのマッチを有するマッチしたテンプレートを収集し、マッチを有さないテンプレートがあれば廃棄するステップと、

前記マッチしたテンプレートを、マッチした要求の数に従ってソートするステップと、および、

複数 (k 個) のマッチしたテンプレートについて、上位 k リストおよびソートされたテンプレートを使用して、各 k について満たされるマッチの要求の数を決定するステップとを含む方法。

【請求項 15】

前記各テンプレートが、コアの数と、メモリ空間と、ディスク空間とを含む割り振り可能な構成を指定する、前記クラウド環境リソースを割り振る方法であって、前記各要求をマッチさせるステップが、

マッチする各要求とテンプレートとをリストするためのリソース・マッチ・テーブルを生成するステップと、

マッチした各テンプレートに関連するコストをリストするための最小コスト・リストを生成するステップと、

前記サーバ要求データベースから要求を選択するステップと、

前記選択された要求にマッチするテンプレートを識別するステップと、

前記マッチするテンプレートによって指定される割り振り可能なリソースに関連するコストを決定するステップと、および、

前記マッチしたテンプレートおよび要求をリソース・マッチ・テーブルに追加し、前記関連するコストを前記最小コスト・リストに追加するステップと

を含む、請求項 14 に記載のクラウド環境リソースを割り振る方法。

【請求項 16】

前記各要求をマッチさせるステップが、

マッチさせていない要求が前記サーバ要求データベース中に残っているかどうか決定するステップと、ならびに、未選択の要求がなくなるまで、

戻って次の要求を選択し、前記次の要求に対してマッチするテンプレートおよび関連するコストを決定するステップとを含む、請求項 15 に記載のクラウド環境リソースを割り振る方法。

【請求項 17】

上位 k リストを決定するステップが、

前記マッチするテンプレートの前記テーブルを、各テンプレートの割振り可能なリソースにマッチする要求の数でソートするステップと、

第 1 の k 値を選択するステップと、

前記サーバ要求データベースから要求を選択するステップと、

前記 k 値についてコスト区分を決定するステップと、

全ての要求が選択されるまで、戻って次の要求を選択するステップと、および、全ての要求が選択されたとき、

前記コスト区分を上位 k テンプレート・テーブル中でリストするステップとを含む、請求項 16 に記載のクラウド環境リソースを割り振る方法。

【請求項 18】

前記コスト区分が、各上位 k リストについて、最小コストで満たされる要求、最小コストよりも高いコストで満たされる要求、および満たされない要求の数を含み、前記上位 k リスト中の、最低コストが関連するテンプレートが、リソースのプロビジョニングに使用される、請求項 17 に記載のクラウド環境リソースを割り振る方法。

【請求項 19】

コンピュータに、請求項 8 乃至 18 のいずれか 1 項に記載の方法の各ステップを実行させるためのコンピュータ・プログラム。

【請求項 20】

請求項 19 に記載のコンピュータ・プログラムを記録したコンピュータ読み取り可能な記録媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、共有リソースを割り振ることに関し、より詳細には、最小限のオーバープロビジョニングおよびアンダープロビジョニングでリソース要求を満たすためにテンプレートの最適セットを選択することに関する。

【背景技術】

【0002】

情報技術 (IT) の獲得および管理は、現代のどんな組織にとっても、主要な予算上の懸念である。さらに、ローカル IT ハードウェアが全キャパシティで使用されることはめったにない。したがって、IT インフラストラクチャのコストおよび無駄を削減するために、組織は、物理的なハードウェアを取得する代わりに、より少ないサーバ上でホストされるバーチャル・マシン (VM) 上に作業量をますます統合しつつある。リモート・サーバ・コンピュータが、仮想リソース (例えば処理力、メモリ、およびディスク空間) を、仮想サーバとしての各 VM に提供する。通常、各 VM 構成は、いくつかの仮想リソース・テンプレート (VRT または テンプレート) から選択される。各 VRT は、VM を定義するために割当て可能な、所定の仮想リソース能力を定義する。したがって、サーバ・コンピュータ (複数可) は、各 VM の意図される目的および予想されるニーズに最も近く構成された VRT (ソフトウェア・スタックおよびライセンス) を割り当てることによって、VM にキャパシティ (例えばディスク空間、処理リソース、およびメモリ) を割り振る。

【0003】

これらの VM を管理する際、VM の最適なキャパシティおよび最適な構成を決定すること、すなわち最適な VRT を選択することは困難であった。通常、サービス・プロバイダ

10

20

30

40

50

が、プロバイダ・システム最適化と、作業量予測と、VMリソース使用の継続的な監視から収集されたリソース使用履歴とに主に基づいて、各VMに対して、VRTを選択し、対応する物理リソースを割り振る。しかし、予測結果が適切であっても、割り振りが多すぎるかまたは少なすぎるせいでユーザ体験が損なわれる可能性がある。割り振りが多すぎると、通常なら他のユーザに利用可能だったかまたは追加のVMをサポートするのに利用可能だったであろうエネルギーおよびリソース、キャパシティが無駄になる。割り振りが少なすぎると、1つまたは複数のVMに割り振られるリソースが不十分になるので、これらのVM上のサービス品質(QoS)に影響を及ぼす(例えばビデオが停止したり音声聞き取れなくなったりする)。

【0004】

ユーザ要件は非常に変わりやすく、これによりプロバイダは、提供するテンプレートの定義および数を変動させることを余儀なくされる場合がある。リソース・プロバイダは、全ての要求を満たすように、提供するVRTの数を増やすこともできるが、数を増やすと、リソース・オーバープロビジョニングおよびテンプレート・スプローリングを引き起こす可能性がある。リソース・オーバープロビジョニングは、割り振り過多の場合と同様、提供されるキャパシティに必要であるよりも多くのリソースおよびエネルギーを消費する。この余分なリソースおよびエネルギーは、通常なら他のユーザにまたは追加のVM用に利用可能にできたはずのものである。テンプレート・スプローリングは、イメージ・スプロールとも呼ばれ、これは、あるユーザのニーズに適合するあるテンプレートまたはイメージが別のユーザのニーズに適合するようにいじられたときに生じ、提供されるVRTに別のテンプレートが追加される。最終的に、テンプレートの数は、管理不可能な数にまで拡大する。

【0005】

利用可能なリソースにアプリケーション・サーバ要求をマッチさせる手法として、いくつかの手法が試みられてきた。典型的なクラウド・コンピューティング手法は、選択のコストを無視して、ターゲット・ユーザ・アプリケーションの要件に最も近くマッチするVMテンプレートを選択するものであった。他のいくつかの従来手法は、主にVMテンプレートおよびネットワーク・テンプレートを指定し記憶して、VM、関連するネットワーク、および必要とされるソフトウェアのインストールを加速させる。さらに他の手法は、実行されるVMのプロファイルを作成し、ターゲット・マイグレーション・サーバを選択するものであった。別法として、他の技法は、仮想リソース・テンプレートを選択することに焦点を合わせるのではなく、アプリケーションをテンプレートにスケーリングすること、すなわち、VMをアプリケーションに適合させるのではなくアプリケーションをVMに適合させることに焦点を合わせる。最後に、最新の手法は、ユーザによって提供されるオープン仮想化フォーマット(OVF)インスタンスを、適切なクラウド提供品にマッチさせる。

【発明の概要】

【発明が解決しようとする課題】

【0006】

したがって、リソースを無駄にすることなく最小限のコストで十分なITリソースを割り振り、サーバQoSも維持することが必要とされている。より詳細には、最小限のコストでユーザ要件の大部分を効率的に満たす、クラウド・インフラストラクチャ中でVMをプロビジョニングするためのVMテンプレートのセットを選択することが必要とされている。

【課題を解決するための手段】

【0007】

本発明の一特徴は、クラウド・リソースのオーバープロビジョニングの最小化である。

【0008】

本発明の別の特徴は、クラウド環境におけるテンプレート・スプローリング/イメージ・スプロールの低減である。

10

20

30

40

50

【0009】

本発明のさらに別の特徴は、クライアントのニーズによりよく合致してそれによりクライアント・コストおよびリソースの無駄を削減するように、コストを意識したクラスタリングのために選択された、VMテンプレート/イメージのプールである。

【0010】

本発明は、コストを意識したクラスタリングにおけるテンプレートを編成するテンプレート・ジェネレータと、コストを意識したクラスタリングを使用してリソースを割り振る方法と、そのためのコンピュータ・プログラム製品とに関する。リソース・プロビジョニング・ユニットが、選択された数のリソース・テンプレートを生成し、選択し、維持する。各テンプレートは、割り振り可能なリソース・キャパシティ構成を指定する。要求元の各クライアント・デバイスには、選択されたリソース・テンプレートのうちの1つによって決定されるリソースが割り振られる。リソース・プロビジョニング・ユニットは、割り振り可能なリソースのコストおよび関連する属性を含む構成ストアと、前に受け取られた要求を含むサーバ要求ストアと、テンプレート・リスト・オプションを含む入力パラメータ・ストアとを備える。テンプレート・ジェネレータは、前に受け取られた要求を満たすための最適なテンプレート・リストを決定する。テンプレート・ストアは、生成されたテンプレート・リストを記憶する。結果として得られるテンプレート・リストは、非常に異なり変動するコスト（例えば、処理力、メモリ、およびディスク空間の間の、大きいコスト差）を個別に有する場合のある各リソースの、集合的なリソース・コストに基づく。

【0011】

前述のおよび他の目的、態様、および利点は、図面を参照しながら本発明の好ましい実施形態に関する以下の詳細な記述を読めば、よりよく理解されるであろう。

【図面の簡単な説明】

【0012】

【図1】本発明の一実施形態によるクラウド・コンピューティング・ノードを示す図である。

【図2】本発明の一実施形態によるクラウド・コンピューティング環境を示す図である。

【図3】本発明の一実施形態による抽象化モデル・レイヤを示す図である。

【図4】本発明の好ましい一実施形態による、テンプレートをリソース要求にマッチさせるための、コストを意識したクラスタリングのためのリソース・プロビジョニング・ユニットの例を示す図である。

【図5】好ましいテンプレート・ジェネレータによる上位k個のテンプレートの選択の例を示す図である。

【図6】最小コスト・マッチを見つける例を示す図である。

【図7】最小コスト・マッチをリストするために作成されるリソース・マッチ・テーブルの例を示す図である。

【図8】上位kリストを最小と最大で括ることによって最終的なテンプレート・リストを生成する例を示す図である。

【図9】747個のサーバ要求のセットに対して本発明を適用した例を示す図である。

【発明を実施するための形態】

【0013】

本開示はクラウド・コンピューティングに関する詳細な記述を含むが、本明細書に示す教示の実施はクラウド・コンピューティング環境に限定されないことを、あらかじめ理解されたい。そうではなく、本発明の実施形態は、現在知られているかまたは後に開発される、かつ以下にさらに示すような、他の任意のタイプのコンピューティング環境と共に実施することができる。

【0014】

クラウド・コンピューティングは、最小限の管理労力で、またはサービス・プロバイダとの最小限の対話で迅速にプロビジョニングでき解放できる構成可能なコンピューティング・リソース（例えばネットワーク、ネットワーク帯域幅、サーバ、処理、メモリ、スト

10

20

30

40

50

レージ、アプリケーション、バーチャル・マシン、およびサービス)の共有プールへの便利なオンデマンド・ネットワーク・アクセスを可能にするためのサービス送達モデルである。このクラウド・モデルは、少なくとも5つの特性、少なくとも3つのサービス・モデル、および少なくとも4つの展開モデルを有することができる。

【0015】

特性は以下のとおりである。

オンデマンド・セルフサービス：クラウド消費者は、サービスのプロバイダとの人的な対話を要することなく、必要に応じて自動的に、サーバ時間およびネットワーク・ストレージなどのコンピューティング能力を単方向にプロビジョニングすることができる。

【0016】

ブロード・ネットワーク・アクセス：能力はネットワーク全体にわたって利用可能であり、能力はまた、異種シン・クライアントまたはシック・クライアント・プラットフォーム(例えば携帯電話機、ラップトップ、およびPDA)による使用を推進する標準的なメカニズムを介してアクセスされる。

【0017】

リソース・プーリング：プロバイダのコンピューティング・リソースは、マルチテナント・モデルを使用する複数の消費者に役立つようにプールされ、種々の物理および仮想リソースが、需要に従って動的に割当ておよび再割当てされる。消費者は一般に、提供されるリソースの正確な場所に関する制御または知識を有さないが、より高レベルの抽象化(例えば国、州、またはデータセンタ)で場所を指定できる場合があるという点で、ある意味での場所独立性がある。

【0018】

迅速な弾性：能力を迅速かつ弾力的に、場合によっては自動的にプロビジョニングして素早くスケールアウトし、迅速に解放して素早くスケール・インすることができる。消費者にとっては、プロビジョニングに利用可能な能力は、しばしば無限に見え、いつでもどんな量でも購入することができる。

【0019】

測定されるサービス：クラウド・システムは、サービス・タイプ(例えばストレージ、処理、帯域幅、および有効なユーザ・アカウント)にとって適切な何らかの抽象化レベルで計測能力を利用することによって、リソース使用を自動的に制御および最適化する。リソース使用は、監視、制御、および報告することができ、利用されるサービスのプロバイダと消費者の両方に対するトランスペアレンシをもたらしすることができる。さらに、本発明は、リアルタイムの最適化されたリソース割振りのために、かつ最小限に抑えられた動作コストおよびエネルギー使用で、個別のリソース割振りおよび構成を進行中に調整するためのクライアント自己監視を可能にする。

【0020】

サービス・モデルは以下のとおりである。

ソフトウェア・アズ・ア・サービス(SaaS)：消費者に提供される能力は、クラウド・インフラストラクチャ上で実行されるプロバイダのアプリケーションを使用する能力である。アプリケーションへのアクセスは、ウェブ・ブラウザなどのシン・クライアント・インタフェースを介して様々なクライアント・デバイスから可能である(例えばウェブベースの電子メール)。消費者は、ネットワーク、サーバ、オペレーティング・システム、ストレージ、さらには個々のアプリケーション能力も含めた、基礎をなすクラウド・インフラストラクチャを管理することも制御することもしない(限られたユーザ特有アプリケーション構成設定については例外である可能性がある)。

【0021】

プラットフォーム・アズ・ア・サービス(PaaS)：消費者に提供される能力は、プロバイダによってサポートされるプログラミング言語およびツールを使用して作成された、消費者によって作成または取得されたアプリケーションを、クラウド・インフラストラクチャ上に展開する能力である。消費者は、ネットワーク、サーバ、オペレーティング・

10

20

30

40

50

システム、またはストレージを含めた、基礎をなすクラウド・インフラストラクチャを管理することも制御することもしないが、展開されたアプリケーションに対する制御、およびおそらくはアプリケーション・ホスティング環境構成に対する制御を有する。

【0022】

インフラストラクチャ・アズ・ア・サービス (IaaS) : 消費者に提供される能力は、処理、ストレージ、ネットワーク、および他の根本的なコンピューティング・リソースをプロビジョニングする能力であり (ハイパーバイザと呼ばれることもある)、消費者は任意のソフトウェアを展開および実行することができ、このソフトウェアには、オペレーティング・システムおよびアプリケーションを含めることができる。消費者は、基礎をなすクラウド・インフラストラクチャを管理することも制御することもしないが、オペレーティング・システム、ストレージ、展開されたアプリケーションに対する制御を有し、おそらくは、精選されたネットワーキング・コンポーネント (例えばホスト・ファイアウォール) の限られた制御を有する。

10

【0023】

展開モデルは以下のとおりである。

プライベート・クラウド : クラウド・インフラストラクチャは、組織に対してのみ運営される。このクラウド・インフラストラクチャは、この組織または第三者によって管理されてよく、敷地内または敷地外に存在してよい。

【0024】

コミュニティ・クラウド : クラウド・インフラストラクチャは、いくつかの組織によって共有され、共有の関心 (例えば任務、セキュリティ要件、ポリシー、およびコンプライアンスに関する考慮事項) を有する特定のコミュニティをサポートする。このクラウド・インフラストラクチャは、これらの組織または第三者によって管理されてよく、敷地内または敷地外に存在してよい。

20

【0025】

パブリック・クラウド : クラウド・インフラストラクチャは、一般大衆または大規模な業界団体が利用できるようにされ、クラウド・サービスを販売する組織によって所有される。

【0026】

ハイブリッド・クラウド : クラウド・インフラストラクチャは、2つ以上のクラウド (プライベート、コミュニティ、またはパブリック) の合成物である。これらのクラウドは、独自のエンティティのままだが、データおよびアプリケーションの可搬性を可能にする標準化された技術またはプロプラエタリ技術によって1つにまとめられる (例えば、クラウド間の負荷平衡のためのクラウド・バースティング)。

30

【0027】

クラウド・コンピューティング環境はサービス指向であり、その焦点は、ステートレス性、低カップリング、モジュラー性、およびセマンティックな相互運用性に合わせられる。クラウド・コンピューティングの中心には、相互接続されたノードのネットワークを構成するインフラストラクチャがある。

【0028】

次に図1を参照すると、クラウド・コンピューティング・ノードの例の概略図が示されている。クラウド・コンピューティング・ノード10は、適切なクラウド・コンピューティング・ノードの一例に過ぎず、本明細書に述べる本発明の実施形態の使用または機能の範囲に関してどんな限定を示唆するものともしない。ともかく、クラウド・コンピューティング・ノード10は、前述の機能のいずれかとの関連で実現されること、または前述の機能のいずれかを実施すること、あるいはその両方が可能である。

40

【0029】

クラウド・コンピューティング・ノード10中には、コンピュータ・システム/サーバ12があり、このコンピュータ・システム/サーバ12は、他の多くの汎用または専用コンピューティング・システム環境または構成で動作する。コンピュータ・システム/サー

50

バ 1 2 と共に使用するのに適する可能性のある周知のコンピューティング・システム、環境、または構成、あるいはこれらの全ての例としては、パーソナル・コンピュータ・システム、サーバ・コンピュータ・システム、シン・クライアント、シック・クライアント、ハンドヘルドまたはラップトップ・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ベースのシステム、セット・トップ・ボックス、プログラム可能な消費者電子機器、ネットワーク PC、ミニコンピュータ・システム、メインフレーム・コンピュータ・システム、および、以上のシステムまたはデバイスのいずれかを含む分散クラウド・コンピューティング環境などが挙げられるが、これらに限定されない。

【 0 0 3 0 】

コンピュータ・システム / サーバ 1 2 は、プログラム・モジュールなど、コンピュータ・システムによって実行されるコンピュータ・システム実行可能命令の一般的なコンテキストで記述することができる。一般に、プログラム・モジュールは、特定のタスクを実施するかまたは特定の抽象データ型を実装するルーチン、プログラム、オブジェクト、コンポーネント、ロジック、データ構造などを含む場合がある。コンピュータ・システム / サーバ 1 2 は、分散クラウド・コンピューティング環境で実践されてよく、その場合、タスクは、通信ネットワークを介してリンクされたりリモート処理デバイスによって実施される。分散クラウド・コンピューティング環境では、プログラム・モジュールは、メモリ記憶デバイスを含めたローカルとリモートの両方のコンピュータ・システム記憶媒体中に位置することができる。

【 0 0 3 1 】

図 1 に示すように、クラウド・コンピューティング・ノード 1 0 中のコンピュータ・システム / サーバ 1 2 は、汎用コンピューティング・デバイスの形で示されている。コンピュータ・システム / サーバ 1 2 のコンポーネントは、1 つまたは複数のプロセッサまたは処理ユニット 1 6 と、システム・メモリ 2 8 と、システム・メモリ 2 8 を含めた様々なシステム・コンポーネントをプロセッサ 1 6 に結合するバス 1 8 とを含んでよいが、これらに限定されない。

【 0 0 3 2 】

バス 1 8 は、様々なバス・アーキテクチャのいずれかを使用する、メモリ・バスまたはメモリ・コントローラ、周辺バス、アクセラレーテッド・グラフィックス・ポート、およびプロセッサ・バスまたはローカル・バスを含めた、いくつかのタイプのバス構造のうちのいずれか 1 つまたは複数を表す。限定ではなく例として、このようなアーキテクチャは、I S A (Industry Standard Architecture) バス、M C A (MicroChannel Architecture) バス、E I S A (EnhancedISA) バス、v e s a (Video Electronics Standards Association) ローカル・バス、および P C I (Peripheral Component Interconnects) バスを含む。

【 0 0 3 3 】

コンピュータ・システム / サーバ 1 2 は通常、様々なコンピュータ・システム可読媒体を含む。このような媒体は、コンピュータ・システム / サーバ 1 2 からアクセス可能な任意の利用可能な媒体とすることができ、揮発性と不揮発性媒体、取外し可能と非取外し可能媒体の両方を含む。

【 0 0 3 4 】

システム・メモリ 2 8 は、ランダム・アクセス・メモリ (R A M) 3 0 またはキャッシュ・メモリ 3 2 あるいはその両方など、揮発性メモリの形のコンピュータ・システム可読媒体を含んでよい。コンピュータ・システム / サーバ 1 2 はさらに、他の取外し可能 / 非取外し可能、揮発性 / 不揮発性コンピュータ・システム記憶媒体を備えてよい。例に過ぎないが、非取外し可能かつ不揮発性の磁気媒体 (図示せず。通常は「ハード・ドライブ」と呼ばれる) に対して読取りおよび書込みするためのストレージ・システム 3 4 が設けられてもよい。図示しないが、取外し可能かつ不揮発性の磁気ディスク (例えば「フロッピー (R) ・ディスク」) に対して読取りおよび書込みするための磁気ディスク・ドライブ、および、取外し可能かつ不揮発性の光学ディスク (C D - R O M、D V D - R O M、ま

10

20

30

40

50

たは他の光学媒体など)に対して読取りまたは書込みするための光学ディスク・ドライブが設けられてもよい。このような場合、各ドライブは、1つまたは複数のデータ媒体インタフェースによってバス18に接続されてよい。後でさらに図示および記述するように、メモリ28は、本発明の実施形態の機能を実施するように構成された1組(例えば少なくとも1つ)のプログラム・モジュールを有する少なくとも1つのプログラム製品を含んでよい。

【0035】

1組(少なくとも1つ)のプログラム・モジュール42を有するプログラム/ユーティリティ40が、限定ではなく例としてメモリ28に記憶されてよく、また、オペレーティング・システム、1つまたは複数のアプリケーション・プログラム、他のプログラム・モジュール、およびプログラム・データも記憶されてよい。オペレーティング・システム、1つまたは複数のアプリケーション・プログラム、他のプログラム・モジュール、およびプログラム・データのそれぞれ、またはこれらの何らかの組合せは、ネットワーク環境の一実装形態を含んでよい。プログラム・モジュール42は一般に、本明細書に述べる本発明の実施形態の機能または方法あるいはその両方を実施する。

【0036】

コンピュータ・システム/サーバ12はまた、1つまたは複数の外部デバイス14と通信することができる。外部デバイス14は、キーボード、ポインティング・デバイス、表示装置24等;ユーザがコンピュータ・システム/サーバ12と対話できるようにする1つもしくは複数のデバイス;または、コンピュータ・システム/サーバ12が1つもしくは複数の他のコンピューティング・デバイスと通信できるようにする任意のデバイス(例えばネットワーク・カード、モデム等);あるいはこれらの全てなどである。このような通信は、入出力(I/O)インタフェース22を介して行うことができる。さらに、コンピュータ・システム/サーバ12は、ネットワーク・アダプタ20を介して、ローカル・エリア・ネットワーク(LAN)、一般的なワイド・エリア・ネットワーク(WAN)、または公衆ネットワーク(例えばインターネット)、あるいはこれらの全てなど、1つまたは複数のネットワークと通信することができる。図示のように、ネットワーク・アダプタ20が、バス18を介して、コンピュータ・システム/サーバ12の他のコンポーネントと通信する。図示していないが、他のハードウェア・コンポーネントまたはソフトウェア・コンポーネントあるいはその両方を、コンピュータ・システム/サーバ12と共に使用してもよいことを理解されたい。例としては、マイクロコード、デバイス・ドライバ、冗長処理ユニット、外部ディスク・ドライブ・アレイ、RAIDシステム、テープ・ドライブ、およびデータ・アーカイバル・ストレージ・システムなどが挙げられるが、これらに限定されない。

【0037】

次に図2を参照すると、例示的なクラウド・コンピューティング環境50が示されている。図示のように、クラウド・コンピューティング環境50は、1つまたは複数のクラウド・コンピューティング・ノード10を含み、クラウド消費者によって使用されるローカル・コンピューティング・デバイス(例えばパーソナル・デジタル・アシスタント(PDA)もしくはセルラー電話機54A、デスクトップ・コンピュータ54B、ラップトップ・コンピュータ54C、または自動車コンピュータ・システム54N、あるいはこれらの全て)が、このクラウド・コンピューティング・ノード10と通信することができる。ノード10は、相互と通信することができる。これらは、前述のようなプライベート、コミュニティ、パブリック、もしくはハイブリッド・クラウド、またはこれらの組合せなど、1つまたは複数のネットワーク中で、物理的または仮想的にグループ化することができる(図示せず)。これにより、クラウド・コンピューティング環境50は、クラウド消費者がローカル・コンピューティング・デバイス上にリソースを維持する必要のない、インフラストラクチャ、プラットフォーム、またはソフトウェア・アズ・サービス、あるいはこれらの全てを提供することができる。図2に示すコンピューティング・デバイス54A~Nのタイプは例示に過ぎないものとし、コンピューティング・ノード10およびクラウ

10

20

30

40

50

ド・コンピューティング環境50は、任意のタイプのネットワークまたはネットワーク・アドレス可能接続あるいはその両方を介して（例えばウェブ・ブラウザを使用して）、任意のタイプのコンピュータ化されたデバイスと通信できることを理解されたい。

【0038】

次に図3を参照すると、クラウド・コンピューティング環境50（図2）によって提供される1組の機能抽象化レイヤが示されている。図3に示すコンポーネント、レイヤ、および機能は例示に過ぎないものとし、本発明の実施形態はこれらに限定されないことを、あらかじめ理解されたい。図示のように、以下のレイヤおよび対応する機能が提供される。

【0039】

ハードウェアおよびソフトウェア・レイヤ60は、ハードウェアおよびソフトウェア・コンポーネントを含む。ハードウェア・コンポーネントの例としては、メインフレーム（一例ではIBM（R）zSeries（R）システム）、RISC（縮小命令セット・コンピュータ）アーキテクチャに基づくサーバ（一例ではIBM pSeries（R）システム）、IBM xSeries（R）システム、IBM BladeCenter（R）システム、ストレージ・デバイス、ネットワーク、およびネットワークング・コンポーネントが挙げられる。ソフトウェア・コンポーネントの例としては、ネットワーク・アプリケーション・サーバ・ソフトウェア（一例ではIBM WebSphere（R）アプリケーション・サーバ・ソフトウェア）、およびデータベース・ソフトウェア（一例ではIBM DB2（R）データベース・ソフトウェア）が挙げられる。（IBM zSeries、pSeries、xSeries、BladeCenter、WebSphere、およびDB2は、世界中の多くの管轄で登録されている国際的・ビジネス・マシンス・コーポレーションの商標である。）

【0040】

仮想化レイヤ62は、仮想エンティティの以下の例を提供できる抽象化レイヤを提供する。すなわち、仮想サーバ、仮想ストレージ、仮想ネットワーク（仮想プライベート・ネットワークを含む）、仮想アプリケーションおよびオペレーティング・システム、ならびに仮想クライアントを提供することができる。

【0041】

一例では、管理レイヤ64は、以下に述べる機能を提供することができる。リソース・プロビジョニング70は、クラウド・コンピューティング環境内でタスクを実施するのに利用されるコンピューティング・リソースおよび他のリソースの動的な調達を提供する。計測および価格設定は、クラウド・コンピューティング環境内でリソースが利用されるのに伴うコスト追跡、およびこれらのリソースの消費に対する課金または送り状送付を提供する。一例では、これらのリソースは、アプリケーション・ソフトウェア・ライセンスを含む場合がある。セキュリティは、クラウド消費者およびタスクに関する識別検証、ならびに、データおよび他のリソースのための保護を提供する。ユーザ・ポータルは、クラウド・コンピューティング環境へのアクセスを消費者およびシステム管理者に提供する。サービス・レベル管理は、必要とされるサービス・レベルが満たされるように、クラウド・コンピューティング・リソース割振りおよび管理を提供する。サービス・レベル・アグリーメント（SLA）プランニングおよび履行は、SLAに従って、将来必要とされることが予期されるクラウド・コンピューティング・リソースの事前構成および調達を提供する。

【0042】

ワークロード・レイヤ66は、クラウド・コンピューティング環境を利用できる対象となる機能の例を提供する。このレイヤから提供できるワークロードおよび機能の例としては、マッピングおよびナビゲーション、ソフトウェア開発およびライフサイクル管理、仮想教室教育の送達、データ分析処理、トランザクション処理、ならびにモバイル・デスクトップが挙げられる。

【0043】

図4に、本発明の好ましい一実施形態による、テンプレートをリソース要求にマッチさせるための、コストを意識したクラスタリングのためのリソース・プロビジョニング・ユニット70の例を示す。特に、コストを意識したこの好ましいクラスタリング技法は、非常に異なり変動するコスト（例えば、処理力、メモリ、およびディスク空間の間の、大きいコスト差）を個別に有する場合のあるリソースの、集会的なコストを考慮する。本質的には、各サーバ要求に対する最小コスト・マッチを決定した後、所与のテンプレートの満たされる要求の数に従って、テンプレートのリストを精緻化する。本発明は、クラウド・コンピューティングに適用され、VMテンプレートの数と、満たされるサーバ要件のレベルと、要件を満たすための非常に異なる可能性のあるコストとの間のトレードオフをプロバイダが分析できるようにする。

10

【0044】

好ましいリソース・プロビジョニング・ユニット70は、構成ストレージ72、サーバ要求ストレージ74、入力パラメータ・ストレージ76、テンプレート・ストレージ78、およびテンプレート・ジェネレータ80を備える。構成ストレージ72、サーバ要求ストレージ74、入力パラメータ・ストレージ76、テンプレート・ストレージ78は、個別のストレージ・ユニットであってもよく、または単一のストレージに、例えば図1のストレージ34に記憶されてもよい。好ましいテンプレート・ジェネレータ80は、サーバ要求のセットに対するコンピューティング要件を満たすのに必要なテンプレートの数を選択および決定する、コストを意識した好ましいクラスタリング技法を実施する。

【0045】

20

構成ストレージ72は、リソース・コストおよび関連する属性を含む。例えば、仮想2.1GHz中央処理装置(CPU)、1テラバイト(1TB)のディスク・ストレージ、および2GBのRAMを含むサーバ・テンプレートのエントリが、1時間当たり0.08ドルのコストを指定する場合がある。好ましくは、サーバ要求ストレージ74は、CPU（またはCPU均等物、例えばデュアル・コア64ビットi86）、メモリ、およびディスク空間などのリソース・タイプによって定義される、前に受け取られたサーバ要求を含むサーバ要求データベースを含む。入力パラメータ・ストレージ76は、テンプレート・ジェネレータ80が全てのテンプレートをリストするように「最良テンプレート」オプションが選択されるかどうかの指示を記憶する。または、上位kテンプレート・オプションが選択されるときにのみ使用される、値kおよびテンプレート・パーセンテージを記憶する。好ましいテンプレート・ジェネレータ80は、利用可能な組合せの数から、テンプレートの初期セットを生成する。次いで、テンプレート・ジェネレータ80は、コストを意識した好ましいクラスタリング技法を実施して、最小コスト・マッチでサーバ要求要件を満たすのに必要なテンプレートの数を決定する。各サーバ要求に対して最小コスト・マッチを計算した後、テンプレート・ジェネレータ80は、所与のテンプレートの満たされる要件の数に従ってテンプレートのリストを精緻化し、精緻化されたリストをテンプレート・ストレージ78に記憶する。

30

【0046】

好ましくは、サーバ要求は、サーバ要求ストレージ74中の要求データベースに記憶され、各要求は、リソース・タイプ、例えばCPU、メモリ、およびディスクによって定義される。好ましいテンプレート・ジェネレータ80は、例えば入力パラメータ・ストレージ76中で定義される「最良テンプレート」オプションに基づいて、全てのサーバ要求需要を満たすためのテンプレートの完全なリストを生成する。別法として、好ましいテンプレート・ジェネレータ80は、例えばやはり入力パラメータ・ストレージ76中で定義される「上位kテンプレート」オプションを使用して、最も多いサーバ要求の要件を満たす上位k個のテンプレートを生成する。

40

【0047】

テンプレート・ジェネレータ80は、全ての要求サーバ需要を満たすためのテンプレートの完全なリストから、各サーバ要件に対して最小コスト・マッチを計算する。これらの最小コスト・マッチから、テンプレート・ジェネレータ80は、例えば満たされるパーセ

50

ンテージまたは最低の全体コストによって、最も多いサーバ要求の要件を満たす上位 k リストについて k (k_1, k_2, \dots, k_n) を選択する。結果として得られる、最も頻繁に要求されるテンプレートのセットは、テンプレートのサブセットであり、総コストを示し、最終的なより小さいテンプレート・セットとして選択される。

【0048】

図5に、好ましいテンプレート・ジェネレータ80による上位 k 個のテンプレートの選択の例80Aを示す。最初に、構成72が、要求ストレージ74中の各サーバ要件と比較されて、各要求に対する最小コスト・マッチ86が決定される(82)。次に、セット中の要件のうちの少なくとも1つについての最小限の要件を満たすことのできないテンプレートがあれば廃棄される(88)。残りのテンプレートは、マッチする要求の数に従ってソートされ(90)、ソートされたテンプレートは記憶される(92)。上位 k リスト94およびソートされたテンプレート92を使用して、各 k についてマッチが決定される(96)。各 k についての満たされる要求の数、およびコストは、例えば表示に向けて、出力に渡される(98)。

10

【0049】

したがってまず、例えばCPU、メモリ、およびディスク要件74を、対応するリストされたターゲット構成72中の最も小さい値を有するテンプレートとマッチさせることによって、各要求に対する最小サーバ要件マッチが決定される(82)。単純な例として、CPUキャパシティ・ターゲット・リストにより、1, 2, 4, 8, 12, 16, . . . , \max_{CPU} 個のコアを選択することができ、 \max_{CPU} は、利用可能な最大数である。同様に、選択可能なメモリとディスク空間は両方とも、それぞれ、1, 2, 4, 8, 12, 16, 32, . . . , \max_{mem} ギガバイト(GB)、および100, 200, 300, 400, 500, 600, . . . , \max_{disk} GBとすることができる。この例ではCPU、メモリ、およびディスク・キャパシティが使用されているが、通常、好ましい一解決法では、追加のリソース、例えばネットワーク帯域幅などが考慮される。

20

【0050】

各サーバ要件に対する最小コスト・マッチ86は、リストされたリソース(例えばCPU、メモリ、およびディスク)に基づいて各サーバの要件を満たすためのコストの下限をもたらす。さらに、好ましいテンプレート・ジェネレータ80は、最小コスト86を使用して、全ての要求を最小コストで満たすことができるかどうかを判定する。本明細書では、最小コスト・テンプレートとの各マッチを「最小コスト・マッチ」と呼ぶ。

30

【0051】

本発明の好ましい一実施形態により、図6に、最小コスト・マッチを見つけること(82)の例を示し、図7に、最小コスト・マッチをリストするために作成されるリソース・マッチ・テーブル86の例を示す。最小コスト・マッチを見つけること(82)は、最小構成コスト・リスト824およびリソース・マッチ・テーブル86を作成することで開始する(822)。リソース・マッチ・テーブル86は、この例では、コアの数860、要求されるメモリ862、および要求されるディスク空間864のフィールドを含む。要求74が、全ての要求が選択される(828)まで、要求された順に選択される(826)。未選択の要求がある限り、現在選択されている要求がターゲット・リソース・リスト832と比較されて(830)、要求要件にマッチする最小リソースを含むテンプレートが見つけられる。リソース・マッチはリソース・マッチ・テーブル86に入力され、リソース・コストのリスト834を使用してマッチのコストが決定される(836)。マッチするテンプレートのコストは最小構成コスト・リスト824に入力され、次の要求が選択される(826)。全ての要求が選択されると(828)、後続のフィルタリングおよび分析に向けて、マッチした要求86および関連するコスト824が返される(838)。

40

【0052】

各テンプレートは、CPU、メモリ、およびディスク空間の、割振り可能な組合せを定義する。したがって、マッチした各要求(図5の86)に対して関連するコスト(図6の824)が決定された後、使用されないテンプレート(例えば複製)が廃棄される(88

50

)。その結果、残りの一意のテンプレートのセットは、少なくとも1つの要求を満たし、さらに、あらゆる要求は、少なくとも1つのテンプレートによって満たされ得る。また、通常はいくつかの単一テンプレートを使用して複数の要求が満たされる場合があり、どんな要求も満たさない他のテンプレートは通常は使用されない場合がある。しかし、最小コスト・マッチ86から選択された一意のテンプレートのセットは、全ての要求を満たすためのテンプレートを含む(88)。

【0053】

次に、選択されたテンプレート・セットは、選択された基準に従って、例えば、昇順のストレージ、昇順のメモリ、CPUコアによって、または、好ましくは、各テンプレートによって満たされる降順の要求数によって、ソートされる(90)。したがって、少なくとも1つのマッチを有するテンプレートのリストを収集して(88)リストをソートすると(90)、好ましいテンプレート・ジェネレータ80は、要求のリストをフィルタにかけて、様々な数の最も頻繁なまたは上位の要求を選択(96)する際の助けとする。これにより、n個のテンプレート・リスト、上位k個のテンプレートのn個のリスト、または上位kリストがもたらされる。好ましいテンプレート・ジェネレータ80は、上位kリストを使用して、各サーバに対する最小リソース・マッチを識別しリストする。

10

【0054】

したがって、図8に示すように、好ましいテンプレート・ジェネレータ(図4の80)は、上位kリストに対する要求を最小と最大で括ることによって、最終的なテンプレート・リスト78を生成する(96)。特に、好ましいテンプレート・ジェネレータ80は、満たすことのできない要求、または、最小コストで、および最小コストよりも高いコストで満たすことのできる要求の数を、リストごとに決定する。

20

【0055】

最初に、好ましいテンプレート・ジェネレータ80は、上位テンプレートが満たす要求の数(例えば5、10、15、20、25)について、上位k要求テーブル98を作成する(960)。kの第1の値(例えば5、10、15、20、25)が、上位kリスト94から選択される(962)。第1の要求が、要求ストレージ74から取り出される(964)。次いで、選択された要求が、現在の上位kリスト中のテンプレートと比較されて(966)、最小(コスト)であろうとより大きい(より高い)(コスト)であろうとテンプレートのいずれかが要求を満たすかどうか、または上位k個のテンプレートのうちのどれによっても要求を満たすことができないかが判定される。これを繰り返し(968)、全ての要求が比較されるまで(968)次の要求を選択して(964)比較する(966)。次に、これらのテンプレートによって満たされる要求および満たされない要求の数を示す現在の値kについて、上位k要求テーブル98中にエントリが作成される(970)。リストされる上位kの値94がまだある(972)場合は、962に戻り、次の値が選択され、上位k個のテンプレートによって満たされる要求および満たされない要求の数が決定される(962~970)。そうでない場合は、kの全ての値が選択されたとき(972)、上位k要求テーブル98が、最終的なテンプレート・リストとして返される(78)。好ましいテンプレート・ジェネレータ80は、最良の上位kリスト(例えば、最小kリスト・コストによって示される収穫逓減ポイント)を決定するためのトレードオフ分析において、この最終的なテンプレート・リスト78を使用する。

30

40

【0056】

図9に、k=5、10、15、20、25、および89(すなわち全てのテンプレート)の場合に、747個のサーバ要求のセットに対して本発明を適用した例を示す。この例は、上位k個のテンプレートによって最小コストでマッチしたパーセンテージ(下)、最大コストでマッチしたパーセンテージ(真ん中)、および満たされないパーセンテージ(上)を含む。

【産業上の利用可能性】

【0057】

したがって、本発明を適用することにより、高価なリソース・キャパシティ(例えば処

50

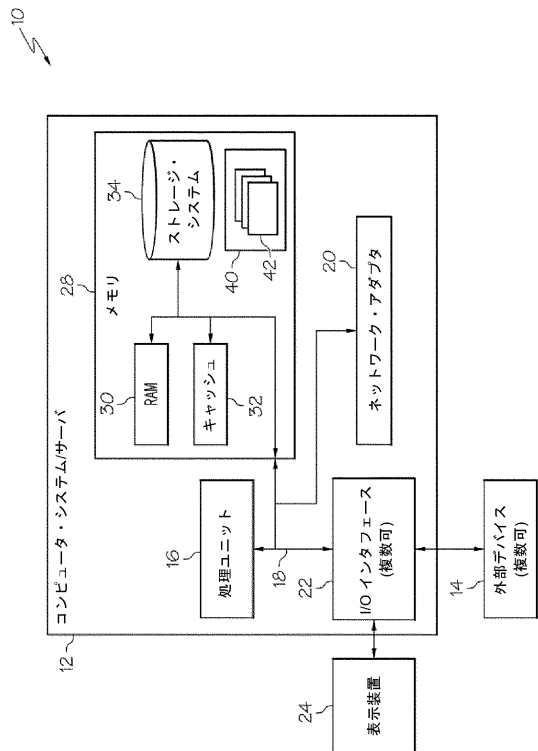
理力、メモリ、およびディスク空間)のオーバプロビジョニングとアンダープロビジョニングの両方が低減される。さらに、割振り可能なリソースは、非常に異なり変動するコストを個別に有する場合があるので、コストを意識した好ましいクラスタリング技法は、各テンプレート中のリソースの集合的なコストに基づいてテンプレートを選択する。このように、集合的なコストに従ってテンプレートを生成および編成することにより、最適なコストでサーバ要件を満たす最低コストのテンプレートがいくつか選択される。さらに、生成されたテンプレート・リストは、維持されるテンプレートの数と、サーバごとに満たされる要求の数と、これらの要求を満たすコストとの間のトレードオフを分析するのを容易にする。

【0058】

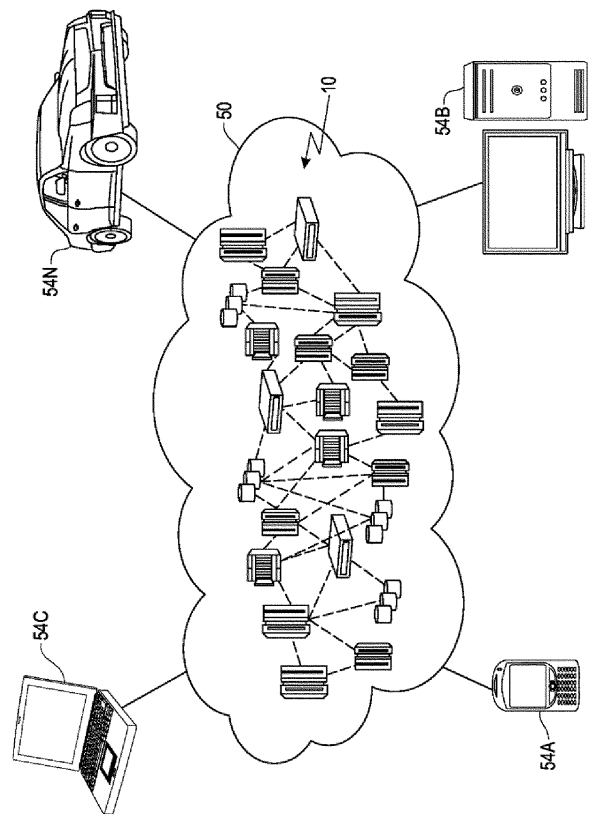
10

本発明を好ましい実施形態の点から述べたが、添付の請求項の主旨および範囲内で変更を加えて本発明を実践できることは、当業者なら認識するであろう。このような全ての変形および修正も、添付の特許請求の範囲内に含まれるものとする。したがって、例および図面は、限定ではなく例示と見なすべきである。

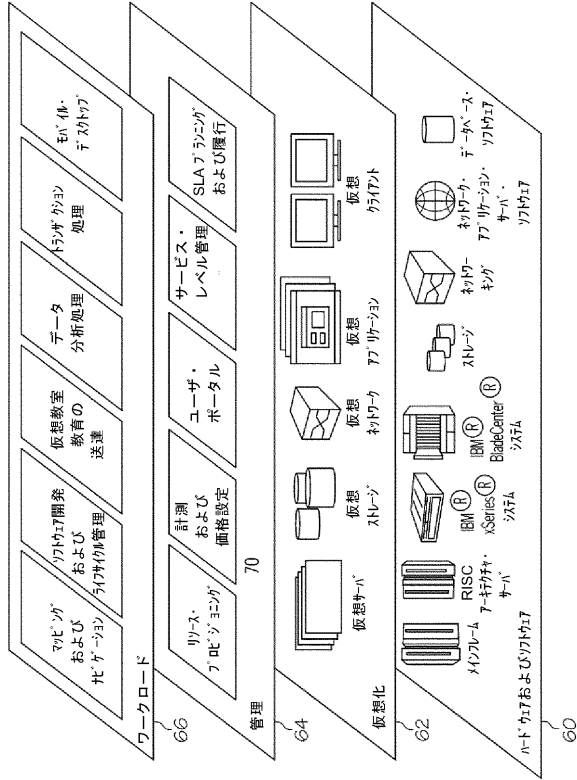
【図1】



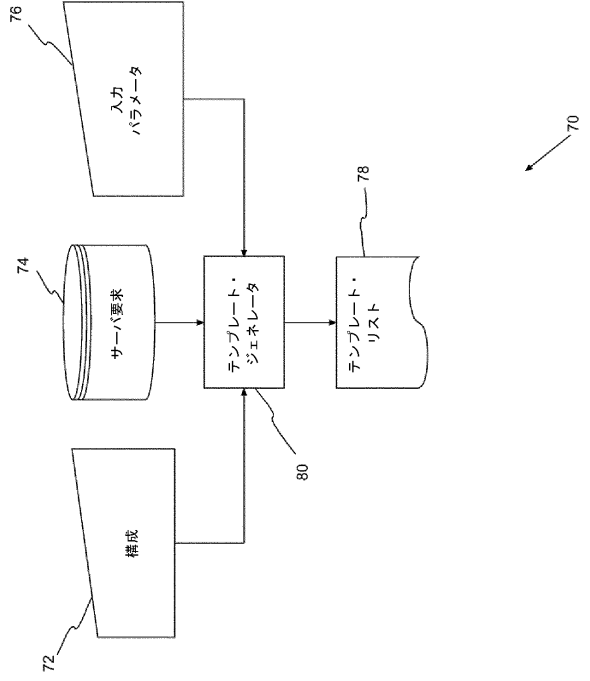
【図2】



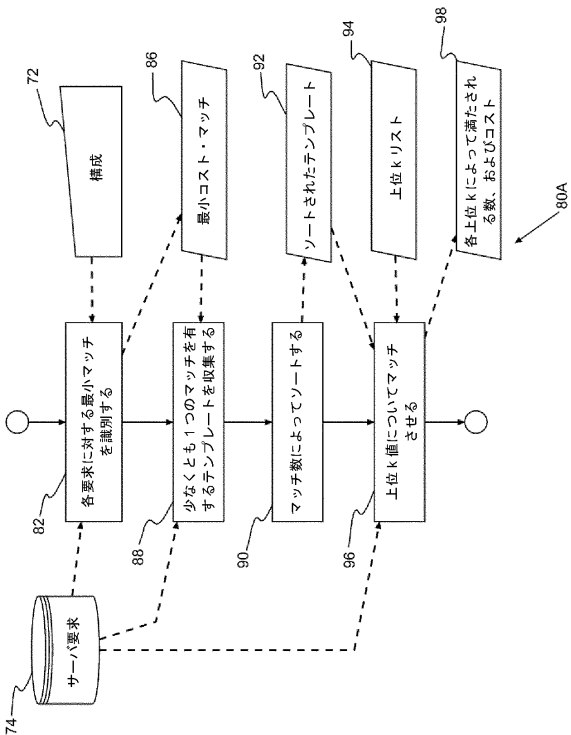
【図3】



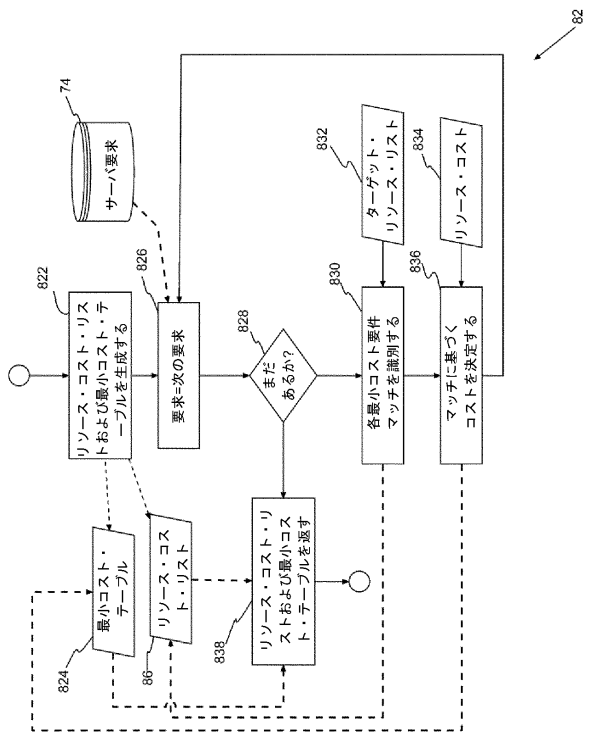
【図4】



【図5】



【図6】



フロントページの続き

- (74)代理人 100112690
弁理士 太佐 種一
- (72)発明者 アスンソン、マルコス、ディアス、デ
ブラジル サンパウロ ルア ットイア 1157
- (72)発明者 ネット、マルコ、アウレリオ、ステルマー
ブラジル サンパウロ ルア ットイア 1157
- (72)発明者 ピーターソン、ブライアン、エル
アメリカ合衆国10589-3216 ニューヨーク州 ソマーズ ルート100 294
- (72)発明者 レンガナラヤナ、ラクシュミナラヤナン
アメリカ合衆国10532 ニューヨーク州ホーソーン スカイライン・ドライブ19
- (72)発明者 ロフラーノ、ジョン、ジェイ
アメリカ合衆国10532 ニューヨーク州ホーソーン スカイライン・ドライブ19
- (72)発明者 ヤング、クリストファー、シー
アメリカ合衆国10532 ニューヨーク州ホーソーン スカイライン・ドライブ19

審査官 多胡 滋

- (56)参考文献 特開2007-133654(JP,A)
特開2008-310610(JP,A)
特開2011-192184(JP,A)

(58)調査した分野(Int.Cl., DB名)

G06F 9/445
G06F 9/46
G06F 9/50

- (54)【発明の名称】共有リソース・システム、リソースを割り振る方法、およびコンピュータ・プログラム製品(共有リソースをプロビジョニングするための、コストを意識したテンプレート選択のためのシステム、方法、およびプログラム製品)