

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5890839号
(P5890839)

(45) 発行日 平成28年3月22日(2016.3.22)

(24) 登録日 平成28年2月26日(2016.2.26)

(51) Int.Cl.

F I

G 0 6 F 9/50 (2006.01)

G 0 6 F 9/46 4 6 2 A

G 0 6 F 11/34 (2006.01)

G 0 6 F 11/34 S

請求項の数 15 (全 29 頁)

(21) 出願番号 特願2013-534921 (P2013-534921)
 (86) (22) 出願日 平成23年9月23日(2011.9.23)
 (65) 公表番号 特表2013-543626 (P2013-543626A)
 (43) 公表日 平成25年12月5日(2013.12.5)
 (86) 国際出願番号 PCT/US2011/053030
 (87) 国際公開番号 W02012/054177
 (87) 国際公開日 平成24年4月26日(2012.4.26)
 審査請求日 平成26年9月18日(2014.9.18)
 (31) 優先権主張番号 12/907, 979
 (32) 優先日 平成22年10月19日(2010.10.19)
 (33) 優先権主張国 米国 (US)

(73) 特許権者 314015767
 マイクロソフト テクノロジー ライセン
 シング、エルエルシー
 アメリカ合衆国 ワシントン州 9805
 2 レッドモンド ワン マイクロソフト
 ウェイ
 (74) 代理人 100140109
 弁理士 小野 新次郎
 (74) 代理人 100075270
 弁理士 小林 泰
 (74) 代理人 100101373
 弁理士 竹内 茂雄
 (74) 代理人 100118902
 弁理士 山本 修

最終頁に続く

(54) 【発明の名称】 参照データサービスのための可用性管理

(57) 【特許請求の範囲】

【請求項 1】

コンテンツプロバイダによってホストされている少なくとも1つのデータベースに格納されたデータに対する複数のデータ要求を受信し管理する、ネットワークを介して前記コンテンツプロバイダと前記データの消費者との間に通信可能に接続された管理ユニットによって実施される方法であって、

前記複数のデータ要求への応答に関連した応答性能の特性評価を決定するステップと、

前記応答性能の特性評価に関連したリソース要求を前記コンテンツプロバイダへ送信するステップであって、前記リソース要求を含んだAPI(Application Programming Interface)コールを前記コンテンツプロバイダに送信するステップを含む、ステップと、

前記コンテンツプロバイダから、前記データを提供するための前記コンテンツプロバイダによって割り当てられた1つまたは複数の追加リソースに関する1つまたは複数の位置を特定する応答を受信するステップと、

を含む方法。

【請求項 2】

前記割り当てられた1つまたは複数の追加リソースは複製データベースを含む、請求項1に記載の方法。

【請求項 3】

前記コンテンツプロバイダからスケーラビリティインターフェースを介してスケーラビリティ入力を受信するステップであって、前記割り当てられた1つまたは複数の追加リソ

10

20

ースに関連した命令を前記コンテンツプロバイダから受信することを含むステップをさらに含む、請求項 1 に記載の方法。

【請求項 4】

コンテンツプロバイダによってホストされているコンテンツプロバイダデータを示すデータ使用状況を決定するステップであって、前記データ使用状況は前記コンテンツプロバイダデータに対する複数の要求に基づくものであり、前記複数の要求は、ネットワークを介して前記コンテンツプロバイダと前記コンテンツプロバイダデータの消費者との間に通信可能に接続された管理ユニットによって受信され管理される、ステップと、

前記データ使用状況の少なくとも 1 つの態様を前記コンテンツプロバイダに表示するスケラビリティインターフェースを介して、前記コンテンツプロバイダからの入力を受信するステップであって、前記コンテンツプロバイダからビジネス指標を受信するステップを含む、ステップと、

前記ビジネス指標をスケラビリティ指標に変換するステップと、

前記入力に応答して前記コンテンツプロバイダデータを提供するための 1 つまたは複数の追加のデータ可用性リソースを割り当てるステップであって、前記スケラビリティ指標に基づいて前記 1 つまたは複数の追加のデータ可用性リソースを割り当てるステップを含む、ステップと、

前記割り当てられた 1 つまたは複数の追加のデータ可用性リソースに関する 1 つまたは複数の位置の特定を前記管理ユニットに提供するステップと、

を含む方法。

【請求項 5】

前記ビジネス指標を受信するステップは、前記コンテンツプロバイダデータに関連した需要と、スケールされた量の前記コンテンツプロバイダデータをホストするコストとのうちの少なくとも 1 つに関連した指標を受信するステップを含み、

前記変換するステップは、前記ビジネス指標を、単位時間あたりのクエリ数または単位時間あたりのデータトランザクション数を処理する能力に関連した能力指標に変換するステップを含む、

請求項 4 に記載の方法。

【請求項 6】

前記入力を受信するステップは、所与のビジネス指標の乗数を受信するステップを含み、

前記変換するステップは、前記乗数に基づいて前記スケラビリティ指標を変換するステップを含む、

請求項 4 に記載の方法。

【請求項 7】

前記スケラビリティインターフェースを介して前記入力を受信するステップは、前記スケラビリティインターフェースの可変乗数ユーザインターフェース要素を介して、前記コンテンツプロバイダからの入力を受信するステップを含む、請求項 6 に記載の方法。

【請求項 8】

前記スケラビリティインターフェースを介して前記入力を受信するステップは、前記スケラビリティインターフェースを介して前記データ使用状況の少なくとも 1 つの態様のグラフィカル表現との対話から入力を受信するステップを含む、請求項 4 に記載の方法。

【請求項 9】

コンテンツプロバイダによってホストされている少なくとも 1 つのデータベースに格納されたデータに対する複数のデータ使用要求を受信し管理する、ネットワークを介して前記コンテンツプロバイダと前記データの消費者との間に通信可能に接続された管理ユニットによって実施され、情報のスケラブルな可用性を促進する方法であって、

前記データ使用要求への応答に関連した応答性能の特性評価に基づいて API (Application Programming Interface) コールを生成するステップと、

10

20

30

40

50

前記APIコールを前記コンテンツプロバイダに送信するステップと、

前記データをホストするためのリソースのスケールが前記応答性能の特性評価に基づいて実行されたか否かを示すとともに、さらに前記データを提供するための前記コンテンツプロバイダによって割り当てられた1つまたは複数の追加リソースに関する1つまたは複数の位置を示す前記APIコールへの応答を、前記コンテンツプロバイダから受信するステップと、

を含む方法。

【請求項10】

前記生成するステップは、複数の階層化された性能閾値に基づく前記応答性能の階層化された特性評価に従って前記APIコールを生成するステップを含む、請求項9に記載の方法。

10

【請求項11】

警告閾値のトリガを検出するステップをさらに含み、

前記APIコールを送信するステップは、前記応答性能の前記特性評価に関連した警告メッセージを送信するステップを含む、請求項10に記載の方法。

【請求項12】

強制スケール閾値のトリガを検出するステップをさらに含み、

前記APIコールを送信するステップは、前記応答性能の前記特性評価に関連した強制スケールメッセージを送信するステップを含む、請求項10に記載の方法。

【請求項13】

20

前記APIコールを送信するステップは、前記データの複製の現在の数、前記強制スケール閾値の前記トリガに関連した応答時間、または平均応答時間のうち少なくとも1つの表示のうち、少なくとも1つを送信するステップを含む、請求項12に記載の方法。

【請求項14】

前記応答は、前記データの複製の新たな数、またはスケール処理手順の状況のうち少なくとも1つをさらに示す、請求項12に記載の方法。

【請求項15】

ポリシー修正閾値のトリガを検出するステップをさらに含み、

前記データ使用要求を処理するステップは、前記データ使用要求に関連したキャッシュポリシーを修正するステップを含む、請求項10に記載の方法。

30

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、IaaS(Information as a Service)の提供に関連してデータの可用性を促進するためにリソースをスケールする参照データサービスのための可用性管理に関し、より詳細にはデータ使用要求への応答に関連した応答性能を監視し、応答性能の特性評価を確定し、特性評価に基づいてリソースのスケールを促進する参照データサービスのための可用性管理に関する。

【背景技術】

【0002】

40

幾つかの従来型のシステムに関わる背景として、伝統的に、コンピューティング装置は、情報、関連したアプリケーションおよびデータサービスを装置に対してローカルに格納していた。しかし、オンラインとクラウドサービスの発展とともに、情報はますます、装置の代わりに、サービスを実施しないかまたはサービスの一部または全部を実施する、ネットワークプロバイダに移されつつある。テラバイトのデータ(将来はペタバイト、エクサバイト等のデータになる可能性がある)を格納できるネットワークストレージファームの発展により、主装置と外部ストレージとを分離して、ローカルでのシナリオをクラウドで模倣する機会がもたらされた。

【0003】

しかしながら、どのクラウドサービスまたはネットワークストレージプロバイダでも、

50

関与する全ての関係者に対して追跡し監査できるように、発行者、開発者および消費者があらゆる種類のデータを容易に発行し、あらゆる種類のデータをアプリケーションに特化させ、あらゆる種類のデータを使用できるように、IaaSを任意のプラットフォーム上で効果的に提供することはできない。加えて、今日では、異なるコンテンツプロバイダおよび、典型的データを定義するための独占的な枠組みがあり、異なるコンテンツプロバイダは、クラウドへの自己の発行行動をお互いに調整しないので、このような無数の発行行動の結集力を利用する機会がほとんどない。

【0004】

ウェブサービスは典型的にある特定の顧客数を想定して構築されており、データサービスをホストして利用可能にするデータセンタ/情報基盤が、これらの目標を想定して構築され、調達されている。しかしながら、ウェブサービスがより人気になると、処理容量の増大により読み取るべき要求があまりに多くなり、その結果サービスが停止するおそれがある。即ち、要求の増加に適切に応答するためのリソースが不十分な場合がある。同様に、データ処理容量が減少する状況において、コンテンツプロバイダがリソースを過度に割り当てることがあり、その結果、コンテンツプロバイダが未使用のリソースへの料金を不必要に支払うこととなる。結果として、今日では、ウェブサービスでのデータの顕著な需要の増加または減少を取り扱うことは、共通解のない断片化された問題である。

【0005】

今日の装置およびデータサービスの上述した欠点は、単に、従来型のシステムの問題の一部の概要を提供しようとするものであり、包括的であることを意図するものではない。最先端技術に関する他の問題、および様々な非限定的な実施形態の一部の対応する利点は、下記の発明を実施するための形態を検討すればさらに明らかになる。

【発明の概要】

【0006】

ここでは、簡潔な概要を提供して、より詳細な説明および添付図面に従う例示的で非限定的な諸実施形態の様々な態様の基本的または一般的な理解を可能にするよう支援する。しかしながら、この概要は、広範囲な概要として意図したものでも包括的な概要として意図したものでもない。寧ろ、この概要の唯一の目的は、幾つかの例示的で非限定的な実施形態に関する幾つかの概念を、下記の様々な実施形態のより詳細な説明に対する前置きとして簡潔な形で提示することである。

【0007】

1つまたは複数の実施形態と対応する開示に従って、様々な非限定的な態様を、任意のプラットフォームからIaaSを提供することと関連して説明する。一態様においては、情報のスケーラブルな可用性を促進する方法を提供する。方法は、コンテンツプロバイダに関連したデータの消費に対する複数の要求を受信することを含む。本実施形態はさらに、複数の要求への応答に関連した応答性能を監視すること、ならびに、応答性能の特性評価を決定することを含むことができる。そして、データに割り当てられたリソースのスケールを、特性評価に基づいて促進する。

【0008】

別の態様において、情報のスケーラブルな可用性を促進するための方法を提供する。コンテンツプロバイダとサーバとの間の通信を確立し、データ使用状況を確定する。ここで、データ使用状況とは、サーバを経由して使用されたコンテンツプロバイダのデータを示すものである。方法はさらに、コンテンツプロバイダにスケーラビリティインターフェースを提供することを含むことができる。本実施形態では、スケーラビリティインターフェースは、データ使用状況の少なくとも一態様をコンテンツプロバイダに対して表示し、また、コンテンツプロバイダから入力を受信する。次に、コンテンツプロバイダに関連したデータ可用性リソースの割り当てを入力に応答して修正する。

【0009】

さらに別の態様では、情報のスケーラブルな可用性を促進する追加の方法を提供する。本実施形態は、データ使用要求への応答に関連した応答性能を監視することを含む。こ

10

20

30

40

50

で、データ使用要求とは、コンテンツプロバイダによってホストされるデータに対する要求である。方法はさらに、応答性能の特性評価に基づいてAPI(Application Programming Interface)コールを生成すること、および当該APIコールをコンテンツプロバイダに送信することを含む。次いで、APIレスポンスをコンテンツプロバイダから受信する。当該APIレスポンスは、データをホストするためにリソースのスケーリングが実行されたか否かを示すものであり、その結果、データ使用要求の処理がリソースのスケーリングに従って修正される。

【0010】

他の実施形態および様々な非限定的な例、シナリオ、および実装形態については下記により詳細に説明する。

【図面の簡単な説明】

【0011】

【図1】一実施形態にかかる、リソースのスケーリングを促進する例示的なシステムを示す図である。

【図2】一実施形態にかかる、リソースのスケーリングを促進する例示的なリソース管理ユニットを示すブロック図である。

【図3】一実施形態にかかる、リソースのスケーリングのための例示的で非限定的な実施形態を示すフローチャートである。

【図4】一実施形態にかかる、リソースのスケーリングを促進する例示的なインターフェースを示す図である。

【図5】一実施形態にかかる、リソースのスケーリングを促進する例示的なスケーラビリティインターフェースを示すブロック図である。

【図6】一実施形態にかかる、リソースのスケーリングを促進するインターフェースを提供するための例示的で非限定的な実施形態を示すフローチャートである。

【図7】一実施形態にかかる、リソースの外部的なスケーリングを促進する例示的なシステムを示す図である。

【図8】一実施形態にかかる、リソースのスケーリングを促進する例示的なAPI(Application Programming Interface)ユニットを示すブロック図である。

【図9】一実施形態における、リソースをスケーリングするための例示的で非限定的な実施形態を示すフローチャートである。

【図10】任意のプラットフォームから提供されるIaaSに対する非限定的な情報基盤のための例示的なシーケンスを示すフローチャートである。

【図11】任意のプラットフォームから提供されるIaaSに対する例示的で非限定的な情報基盤を示すブロック図である。

【図12】任意のプラットフォームから提供されるIaaSに対する情報基盤のための例示的で非限定的な一組の実装形態の具体的な詳細を示すブロック図である。

【図13】任意のプラットフォームから提供されるIaaSに対する例示的な情報基盤からの例示的なデータ使用を示す図である。

【図14】本明細書で説明する様々な実施形態を実装できる例示的で非限定的なネットワーク環境を表すブロック図である。

【図15】本明細書で説明する様々な実施形態の1つまたは複数の態様を実装できる例示的で非限定的なコンピューティングシステムまたは動作環境を表すブロック図である。

【発明を実施するための形態】

【0012】

概要

背景技術で論じたように、ウェブサービスは典型的に所定の顧客数を想定して構築されており、データサービスをホストして利用可能にするデータセンタ/情報基盤がこれらの目標を想定して構築され、提供されている。しかしながら、ウェブサービスがより人気になると、処理容量の増大により読み取るべき要求があまりに多くなり、その結果サービスが停止するおそれがあり、この場合、応答を処理するためのリソースが不十分であるか、

10

20

30

40

50

または、他の予測不可能な変動が生じる可能性がある。グローバルな情報市場が望まれる場合、このような市場のアーキテクチャにとっては、データが管理システムの内部でホストされているか外部でホストされているかに関わらず、データセンタを必要に応じてスケールアップまたはスケールダウンするための方法をコンテンツプロバイダに提供することにより、これらの制約を克服することが望ましいはずである。

【 0 0 1 3 】

本明細書で開示される様々な実施形態は、スケールアウトシステムおよび方法、ならびに、コンテンツプロバイダと通信してこのようなリソースのスケールを促進するために使用されるインターフェース、ならびにこのようなシステム、方法、およびインターフェースに付随する通信に関する。

10

【 0 0 1 4 】

リソーススケーリングシステム

図 1 は、任意のプラットフォームから提供される IaaS に関連したリソースのスケールを促進する例示的なシステムを図示する。図示したように、システム 100 は、リソース管理ユニット 120、消費者 130、およびコンテンツプロバイダ 140 を含むことができ、これらはネットワーク 110 を介して通信可能に接続される。リソース管理ユニット 120 は、コンテンツプロバイダ 140 に関連付けられたデータに対する利用者 130 からのデータ使用要求を管理するように構成されている。さらに、リソース管理ユニット 120 は、特定のコンテンツプロバイダに対応するデータ使用要求の量および / または使用要求の状況を監視することができ、次いでかかるデータに割り当てられたリソースを適切にスケールすることができる。例えば、コンテンツプロバイダのデータをリソース管理ユニット 120 によりホストする場合は、内部のスケール動作を実施することができ、スケーラブルデータベース 122 は適切に削除および / または複製される。しかしながら、使用されるデータがコンテンツプロバイダ 140 および / またはサードパーティによりホストされる場合は、リソース管理ユニット 120 はコンテンツプロバイダ 140 に応答時間の低下および / または急増を通知し、コンテンツプロバイダ 140 がそれに応じてデータベースを削除および / または複製、例えば、追加のデータベースインスタンスを生成できるようにすることができる。

20

【 0 0 1 5 】

次に図 2 を参照すると、一実施形態にかかる、情報のスケーラブルな可用性を促進する例示的なリソース管理ユニットのブロック図が提供されている。示されているように、リソース管理ユニット 200 は、プロセッサコンポーネント 210、メモリコンポーネント 220、通信コンポーネント 230、監視コンポーネント 240、評価コンポーネント 250、スケーリングコンポーネント 260、生成コンポーネント 270、および課金コンポーネント 280 を含むことができる。

30

【 0 0 1 6 】

一態様においては、プロセッサコンポーネント 210 は、複数の機能のうち何れかを実施することに関するコンピュータ読取可能命令を実行するように構成されている。プロセッサコンポーネント 210 は、リソース管理ユニット 200 から通信される情報を分析し、ならびに / あるいはメモリコンポーネント 220、通信コンポーネント 230、監視コンポーネント 240、評価コンポーネント 250、スケーリングコンポーネント 260、生成コンポーネント 270、および / または課金コンポーネント 280 によって利用可能な情報を生成するのに特化した、単一のプロセッサまたは複数のプロセッサでとすることができる。追加または代替として、プロセッサコンポーネント 210 を、リソース管理ユニット 200 の 1 つまたは複数のコンポーネントを制御するように構成することができる。

40

【 0 0 1 7 】

別の態様においては、メモリコンポーネント 220 は、プロセッサコンポーネント 210 に接続され、およびプロセッサコンポーネント 210 が実行するコンピュータ読取可能命令を格納するように構成される。メモリコンポーネント 220 は、通信コンポーネント

50

２３０、監視コンポーネント２４０、評価コンポーネント２５０、スケーリングコンポーネント２６０、生成コンポーネント２７０、および／または課金コンポーネント２８０のうち何れかによって生成されたデータを含め、複数の他の種類のデータのうち何れかを格納するように構成されることもある。メモリコンポーネント２２０は、ランダムアクセスメモリ、バッテリバックアップメモリ、ハードディスク、磁気テープ等を含め、幾つかの異なる態様で構成されることができる。圧縮自動バックアップ、例えばRAID(Redundant Array of Independent Drives)構成の利用といった様々な機能を、メモリコンポーネント２２０に実装することもできる。

【００１８】

さらに別の態様では、リソース管理ユニット２００は通信コンポーネント２３０を含み、通信コンポーネント２３０は、プロセッサコンポーネント２１０に接続され、リソース管理ユニット２００と外部エンティティとのインターフェースとなるように構成される。例えば、通信コンポーネント２３０は、特定のコンテンツプロバイダに関連したデータのデータ使用要求を受信するように構成されることができる。

【００１９】

図示したように、リソース管理ユニット２００はさらに、監視コンポーネント２４０および評価コンポーネント２５０を備えることができる。このような実施形態において、監視コンポーネント２４０はデータ消費要求への応答に関連した応答性能を監視するように構成され、一方、評価コンポーネント２５０は応答性能の特性評価を確定するように構成される。

【００２０】

リソース管理ユニット２００はまた、スケーリングコンポーネント２６０を含むこともでき、スケーリングコンポーネント２６０は、上記特性評価に基づいてデータに割り当てられたリソースのスケーリングを促進するように構成される。この目的のため、スケーリングコンポーネント２６０は、複数の方法のうち何れかでこのようなスケーリングを促進するように構成できることに留意されたい。例えば、スケーリングコンポーネント２６０を、リソース管理ユニット２００の外部で実施されるリソースのスケールを促進するように構成できることが想定されている。実際、コンテンツプロバイダ自身が自己のデータをホストできることが想定されているので、スケーリングコンポーネント２６０は、応答性能状況メッセージを生成するように構成されることができ、当該応答性能状況メッセージは、通信コンポーネント２３０を介してコンテンツプロバイダに送信される。特定の実施形態においては、生成コンポーネント２７０が含まれることがあり、生成コンポーネント２７０は、コンテンツプロバイダに対するAPI(Application Programming Interface)コールを生成するように構成されることができる。APIコールは、応答性能の少なくとも１つの態様に関連した要求、例えば、警告メッセージ、強制スケールメッセージ等を含む。次いで、要求に対するAPIレスポンスをコンテンツプロバイダから受信することができ、当該APIレスポンスは、例えば、コンテンツプロバイダによって割り当てられた追加のデータベースの位置など、データに関連したスケールされた位置のセットを特定する。

【００２１】

スケーリングコンポーネント２６０は、しかしながら、リソース管理ユニット２００内部で実施されるリソースのスケールを促進するように構成されることができる。このような実施形態では、スケーリングコンポーネント２６０が、複製のデータベースを削除および／または追加することでリソースをスケールするよう構成されるように、リソース管理ユニット２００は、コンテンツプロバイダのデータを一組の複製されたデータベース内でホストする。コンテンツプロバイダに割り当てられたリソースのスケールに関連した命令を、コンテンツプロバイダから受信することを促進するために、スケーラビリティインターフェースをコンテンツプロバイダに提供することもできる。

【００２２】

さらなる別の態様においては、リソース管理ユニット２００は課金コンポーネント２８０を含む。このような実施形態では、課金コンポーネント２８０は、コンテンツプロバイ

10

20

30

40

50

ダに割り当てられたリソースのスケーリングを収益化するように構成することができる。さらに、リソースをスケールアップまたはスケールダウンすることに関連したコストを自動的に定量化できることが想定されている。したがって、リソースのスケーリングに関連した実際のおよび／または仮定的なコストを、課金コンポーネント 280 を介して容易に利用可能とすることができる。

【0023】

図3は、一実施形態にかかる、リソースのスケーリングのための例示的で非限定的な実施形態を示すフローチャートである。300において、コンテンツプロバイダに関連したデータの使用に対する要求を受信する。次に、310において、要求への応答に関連した応答性能を監視し、次いで320において、応答性能の特性評価を行う。次に330において、特性評価に基づいて、データに割り当てられたリソースのスケーリングを促進する。

10

【0024】

図4は、一実施形態にかかる、リソースのスケールを促進する例示的なインターフェースを図示する。図示されるように、インターフェース400はディスプレイ410、制御パネル420、およびスケール調整430を含むことができる。この特定の実施形態では、インターフェース400をコンテンツプロバイダに提供して、その結果、コンテンツプロバイダが必要に応じてリソースをシームレスにスケールすることができると想定されている。この目的のため、ディスプレイ410を、現在、将来、または過去のいずれかに、データ使用要求を処理することに関連した様々な態様のうち何れかを表示するように構成することができる。例えば、図示されているように、時間に対する受信したデータ要求数のグラフィカル表現を表示することができる。次に、制御パネル420を使用して、例えば、異なるビジネス指標（例えば、データ要求数ではなく、時間に対するコスト）および／または異なるディスプレイタイプ（例えば、様々なグラフタイプ）を含め、異なるパラメータを切り替えることができる。

20

【0025】

図示されたように、インターフェース400はスケールリング調整430も含むことができる。ここで、コンテンツプロバイダは、例えばコスト、要求数等のビジネス指標の観点からリソースをスケールすることを好み得るので、スケールリング調整430を含めることができ、この場合、コンテンツプロバイダは必要に応じて、単に乗数つまみ434をスケールリング調整430上でスライドさせる。例えば、ベースラインインジケータ432を提供することができ、この場合、乗数つまみ434を右にスライドするとビジネス指標乗数が増大、例えばリソースがスケールアップし、乗数つまみ434を左にスライドするとビジネス指標乗数が減少、例えばリソースがスケールダウンする。

30

【0026】

次に図5を参照すると、図5に係るブロック図は、様々な態様に従ってリソースのスケーリングを促進する例示的なスケラビリティインターフェースを図示する。図示されたように、スケラビリティインターフェース500は、プロセッサコンポーネント510、メモリコンポーネント520、ステータスコンポーネント530、ディスプレイコンポーネント540、入力コンポーネント550、修正コンポーネント560、および変換コンポーネント570を含むことができる。

40

【0027】

リソース管理ユニット200におけるプロセッサコンポーネント210と同様、プロセッサコンポーネント510は、複数の機能のうち何れかの実施に関するコンピュータ読取可能命令を実行するように構成されている。プロセッサコンポーネント510は、スケラビリティインターフェース500から通信される情報を分析し、ならびに／あるいはメモリコンポーネント520、ステータスコンポーネント530、ディスプレイコンポーネント540、入力コンポーネント550、修正コンポーネント560、および／または変換コンポーネント570によって利用可能な情報を生成するのに特化した単一のプロセッサまたは複数のプロセッサであることができる。追加または代替として、プロセッサコン

50

ポーネント 5 1 0 を、スケーラビリティインターフェース 5 0 0 の 1 つまたは複数のコンポーネントを制御するように構成することができる。

【 0 0 2 8 】

別の態様においては、メモリコンポーネント 5 2 0 は、プロセッサコンポーネント 5 1 0 に接続され、およびプロセッサコンポーネント 5 1 0 が実行するコンピュータ読取可能命令を格納するように構成される。メモリコンポーネント 5 2 0 は、ステータスコンポーネント 5 3 0、ディスプレイコンポーネント 5 4 0、入力コンポーネント 5 5 0、修正コンポーネント 5 6 0、および / または変換コンポーネント 5 7 0 のうち何れかにより生成されたデータを含め、複数の他の種類のデータのうち何れかを格納するように構成されることもできる。ここで、メモリコンポーネント 5 2 0 は、リソース管理ユニット 2 0 0 におけるメモリコンポーネント 2 2 0 と類似のものであることに留意されたい。したがって、メモリコンポーネント 2 2 0 の前述の機能 / 構成の何れもメモリコンポーネント 5 2 0 に適用可能であることは理解されよう。

【 0 0 2 9 】

図示されたように、スケーラビリティインターフェース 5 0 0 は、ステータスコンポーネント 5 3 0 およびディスプレイコンポーネント 5 4 0 も含み得る。このような実施形態では、ステータスコンポーネント 5 3 0 は使用されたコンテンツプロバイダのデータを示すデータ使用状況を確定するように構成され、一方、ディスプレイコンポーネント 5 4 0 はデータ使用状況の様々な態様のうち何れかをコンテンツプロバイダに対して表示するように構成される。ここで、このようなデータ使用状況の態様を複数の方法のうち何れかで表示できることが想定されている。例えば、ディスプレイコンポーネント 5 4 0 を、データ使用状況のグラフィカル表現、例えば、時間に対するデータ要求数を追跡する棒グラフを提供するように構成することができる。

【 0 0 3 0 】

スケーラビリティインターフェース 5 0 0 はさらに、入力コンポーネント 5 5 0 および修正コンポーネント 5 6 0 を含み得る。このような実施形態では、入力コンポーネント 5 5 0 はコンテンツプロバイダからの入力を受信するように構成され、一方、修正コンポーネント 5 6 0 は入力に回答してコンテンツプロバイダに関連したデータ可用性リソースの割り当てを修正するように構成される。このような修正を促進するために、スケーラビリティインターフェース 5 0 0 は変換コンポーネント 5 7 0 も含むことができ、当該変換コンポーネント 5 7 0 は、スケーラビリティ指標とビジネス指標との間での変換を実行するように構成される。さらに、特定のビジネス指標は対応するスケーラビリティ指標に変換可能であり、特定のスケーラビリティ指標は対応するビジネス指標に変換可能である。一態様においては、スケーラビリティ指標は、コンテンツプロバイダのデータ使用の処理に直接的に関連する複数の指標のうち何れか、例えば、時間単位あたりのクエリ数を処理する能力に関連付けることができ、一方、ビジネス指標は、複数のビジネス関連指標のうち何れか、例えば、コンテンツプロバイダのデータに関連した需要、スケールされた量のコンテンツプロバイダのデータをホストするコスト等に関連付けることができる。実際、コンテンツプロバイダはスケーラビリティ指標よりもビジネス指標の入力を好み得るので、このような指標の間を切り替えるための様々なツールを提供することができる。例えば、一態様においては、入力コンポーネント 5 5 0 を、コンテンツプロバイダの入力を所望のビジネス指標の乗数として受信するように構成することができ、変換コンポーネント 5 7 0 は、当該乗数に基づいて所望のビジネス指標に対応するスケーラビリティ指標に変換するように構成される。この目的のため、入力コンポーネント 5 5 0 を乗数つまみとして実装することができ、乗数は、当該乗数つまみを介してコンテンツプロバイダから確定することができる。

【 0 0 3 1 】

図 6 は、一実施形態にかかる、リソースをスケーリングするユーザインターフェースを提供するための例示的で非限定的な実施形態を示すフローチャートである。6 0 0 において、コンテンツプロバイダとサーバとの間の通信を確立する。次に 6 1 0 において、サー

バを介して使用されたコンテンツプロバイダのデータを示すデータ使用状況を確認する。データ使用状況は、サーバを介して使用されたコンテンツプロバイダのデータを示す。次に620において、スケーラビリティインターフェースをコンテンツプロバイダに提供するが、スケーラビリティインターフェースは、データ使用状況の諸態様を表示し、また、コンテンツプロバイダからの入力を受信する。次に630において、入力にตอบสนองして、コンテンツプロバイダに関連したデータ可用性リソースの割り当てを修正する。

【0032】

図7は、一実施形態にかかる、リソースの外部的なスケールを促進する例示的なシステムを示す。この特定の実施形態では、リソース管理ユニット710は、コンテンツプロバイダ720および/またはサードパーティによって、スケーラブルデータベース722内部でホストされるデータに対するデータ使用要求を管理するように構成される。さらに、リソース管理ユニット710は、コンテンツプロバイダのデータの使用に関連した応答性能をコンテンツプロバイダ720に通知するように構成されることができ、これにより、コンテンツプロバイダ720がそれに応じてスケーラブルデータベース722を削除および/または複製できるように。一態様においては、リソース管理ユニット710はAPIコールをコンテンツプロバイダ720に送信することができ、これらのAPIコールに対するAPIレスポンスが、続いてリソース管理ユニット720に提供される。例えば、特定の性能閾値を超過したことを示すAPIコールをコンテンツプロバイダ720に提供することができ、そしてこの場合、例えば、新たな複製されたデータベースの位置を提供するなど、スケーラブルデータベース722のスケールングが実施されたかどうか/どのように実行されたかを示すAPIレスポンスを、リソース管理ユニット710に提供することができる。

【0033】

次に図8を参照すると、ブロック図により、様々な態様にかかるリソースのスケールングを促進する例示的なAPI(Application Programming Interface)ユニットを図示されている。図示されるように、APIユニット800は、プロセッサコンポーネント810、メモリコンポーネント820、監視コンポーネント830、生成コンポーネント840、通信コンポーネント850、およびトリガコンポーネント860を含むことができる。

【0034】

それぞれリソース管理ユニット200およびスケーラビリティインターフェース500におけるプロセッサコンポーネント210および510と同様に、プロセッサコンポーネント810は、複数の機能のうち何れかを実施することに関連したコンピュータ読取可能命令を実行するように構成されている。プロセッサコンポーネント810は、APIユニット800から通信される情報を分析し、ならびに/あるいはメモリコンポーネント820、監視コンポーネント830、生成コンポーネント840、通信コンポーネント850、および/またはトリガコンポーネント860によって利用可能な情報を生成するのに特化した単一のプロセッサまたは複数のプロセッサであることができる。追加または代替として、プロセッサコンポーネント810を、APIユニット800の1つまたは複数のコンポーネントを制御するように構成することができる。

【0035】

別の態様においては、メモリコンポーネント820は、プロセッサコンポーネント810に接続され、およびプロセッサコンポーネント810が実行するコンピュータ読取可能命令を格納するように構成される。メモリコンポーネント820は、監視コンポーネント830、生成コンポーネント840、通信コンポーネント850、および/またはトリガコンポーネント860のうち何れかによって生成されたデータを含め、複数の他の種類のデータのうち何れかを格納するように構成されることもできる。ここで、メモリコンポーネント820は、それぞれリソース管理ユニット200およびスケーラビリティインターフェース500におけるメモリコンポーネント220および520と類似のものであることに留意されたい。したがって、メモリコンポーネント220および520の前述の機能/構成の何れもメモリコンポーネント820に適用可能であることは理解されよう。

【0036】

図示されるように、APIユニット800は監視コンポーネント830および生成コンポーネント840を備えることもできる。このような実施形態において、監視コンポーネント830は、コンテンツプロバイダによってホストされるデータに対するデータ使用要求への応答に関連した応答性能を監視するように構成され、一方、生成コンポーネント840は、応答性能の特性評価に基づいてAPIコールを生成するように構成される。

【0037】

別の態様において、APIユニット800は通信コンポーネント850を備え、当該通信コンポーネント850は、プロセッサコンポーネント810に接続され、APIユニット800と外部エンティティとのインターフェースとなるように構成される。例えば、通信コンポーネント850を、生成コンポーネント840によって生成されるAPIコールをコンテンツプロバイダに送信するように構成することができる。通信コンポーネント850を、コンテンツプロバイダからのAPIレスポンスを受信するように構成することもでき、この場合、このようなAPIレスポンスは、データをホストするためのリソースのスケーリングが実施されたかどうかを示す。このような実施形態では、続いて、データ使用要求の処理を、リソースのスケーリングに従って修正することができる。

【0038】

別の態様では、APIコールが、応答性能の階層化された特性評価に従って生成/送信されることが想定されている。このような実施形態においては、トリガコンポーネント860を含めることができ、当該トリガコンポーネント860は、いつ応答性能が特定の特性評価を示すかを複数の階層化された性能閾値のうち何れかに基づいて判定するように構成される。例えば、トリガコンポーネント860は、警告閾値のトリガを検出するように構成され、この場合、APIコールは、応答性能が所定の「警告」閾値、例えば25%の低下を超過したことを示す警告メッセージを含む。

【0039】

トリガコンポーネント860を、強制スケーリング閾値のトリガを検出するように構成することもでき、この場合、APIコールは、応答性能が所定の「強制スケーリング」閾値、例えば50%の低下を超過したことを示す強制スケーリングメッセージを含む。ここで、このような強制スケール閾値を促進するように、具体的な詳細事項をAPIコールおよび/またはAPIレスポンスに含めることができることに留意されたい。例えば、APIコールは、データの複製の現在の数の表示、強制スケーリング閾値のトリガに関連した応答時間、および/または平均応答時間を含むことができ、一方、APIレスポンスは、データの複製の新たな数の表示、データに関連した新たなデータソースの位置、ならびに/あるいは、コンテンツプロバイダおよび/またはもしくはコンテンツプロバイダに関連したサードパーティによって実施されるスケーリング処理手順の状況を含むことができる。

【0040】

より厳格な階層も想定されており、例えば50%の低下も階層として想定される。例えば、トリガコンポーネント860を、ポリシ修正閾値のトリガを検出するように構成することができる。このような実施形態では、ポリシ修正閾値のトリガに応答して、データ使用要求の処理に、データ使用要求に関連したキャッシュポリシの修正を含めることができる。

【0041】

図9は、一実施形態にかかる、リソースをスケールするユーザインターフェースを提供するための例示的で非限定的な実施形態を示すフローチャートである。900において、データ使用要求への応答に関連した応答性能を監視する。この特定の実施形態について、データ使用要求は、コンテンツプロバイダによってホストされるデータに対する要求である。次いで910において、API(Application Programming Interface)コールを応答性能の特性評価に基づいて生成する。次に920において、APIコールをコンテンツプロバイダに送信する。次に930において、APIレスポンスをコンテンツプロバイダから受信するが、APIレスポンスは、データをホストするためにリソースのスケーリングが実施されたかどうかを示す。このような実施形態では、データ使用要求の処理をリソースのスケー

10

20

30

40

50

ルに従って修正する。

【 0 0 4 2 】

IaaSに関する補足的なコンテキスト

以下の説明には、上述の実施形態の1つまたは複数の理解をさらに支援するための可能な非限定的な情報基盤、アーキテクチャ、および/または関連したサービスに関する補足的なコンテキストが含まれる。この節で説明する任意の追加の機能のうち任意の1つまたは複数、任意のプラットフォームからのIaaSに対する、データセットの意味および対応するアクセスAPIの更新に関して上述した実施形態の任意の1つまたは複数に含めることができる。このような諸実施形態または諸機能の組合せが可能であるが、疑義を避けるため、本開示で説明した諸実施形態は、本明細書で説明した他のいずれかの実施形態に関する限定と考えられるべきではない。

10

【 0 0 4 3 】

幾つかの追加の背景として、今日では、情報をインターネットのようなネットワークを介して取得できるが、今日提供されるものはアクセスおよびフレームワークの点で独占的な所有物である傾向があり、従って、サードパーティプロバイダの参加という点で制限されている。例えば、現在、商用コンテンツのプロバイダが、このような商用コンテンツの価値の少なくとも一部を放棄しないで済む、自己のコンテンツデータを発行するための適切なビジネスモデルは存在しない。したがって、歴史的には、価値あるコンテンツの所有者はコンテンツを限定的で独占的な手段で公開する傾向にあった。あるいは、このようなコンテンツのストレージが、セキュアなクラウドストレージプロバイダにより提供される事例では、消費者が興味のある項目を結果として見つけるのに、何千ものテーブルをかき分ける場合には、ストレージ自体には殆ど価値がない。さらに、クラウドストレージプロバイダがデータを様々なプロバイダから収集しようとしても、このようなプロバイダはせいぜい、適度な、小規模の、または不完全なデータカタログを持っているに過ぎない。

20

【 0 0 4 4 】

当分野における潜在的な成長を阻害するものの多くは、取り扱い方法およびIP、例えば著作権に対する不信頼の結果である。手短に言えば、大企業はクラウドプロバイダに重要資産を委託しない。なぜならば、1回の情報漏洩でデータの価値がなくなる可能性があるからである。さらに、このようなこれまでのクラウドプロバイダは、自己の情報ニーズについてこのようなデータから価値を抽出したい情報労働者に対しては弱い立場にいたり、加えてこのようなシステムの独占的な性質のため、開発者はこれまでは情報機能を開発するためのツールを拡張することがほとんどできなかった。

30

【 0 0 4 5 】

したがって、1つまたは複数の実施形態について説明したように、あらゆるクラスの個人、即ち、発行者、開発者、情報労働者、および消費者に適合するIaaSに対する情報基盤を提供する。情報基盤は、情報探索、例えば、構造化されたプロブデータセットを発見、取得、使用する能力を可能にし、任意のスクリーンサイズなど任意のプラットフォームおよび任意のフォームファクタにおいて、任意のアプリケーションを強化することを可能にする。情報基盤により、さらに、仲介ビジネス、例えばパートナー主導型の世界規模なエコシステム(partner driven ecosystem and global reach)が、データおよび機能を開発者および情報労働者に提供することが可能となる。そして、情報基盤では、分析およびレポートが可能となり、例えばシングルクリック分析により秘密データを公開データで補完することができる。この点、情報基盤の様々な実装のオープンな性質のため、あらゆるアプリケーション開発者は、データの発行、処理、問合せ、および/または取り出しを促進するためのモバイルアプリケーション、クラウドアプリケーション、および/またはデスクトップアプリケーションを開発することができる。

40

【 0 0 4 6 】

アプリケーションを書くために、アカウント情報を登録することができ、例えばLiveIDでサインインし、アカウント鍵により、本明細書で1つまたは複数の実施形態について説

50

明したIaaSに対する情報基盤に対して可能となる、構造化されたリアルタイムなウェブサービスを介して、「プロブ」にアクセスするアカウント鍵を提供することができる。一態様では、開発者は、コーディングを開始する前にAPIを視覚的に調査することができる。例えば、サービスエクスポージャーモジュールまたは一組のモジュールは、開発者が視覚的にREST(REpresentational State Transfer)APIクエリを構築し、様々なパッケージに従ってコンテンツをプレビューするのに使用されることができる。様々なパッケージには、これらには限定されないが、例えば構造化データに対しては、XML(EXTensible Markup Language)、ATOM、(プロブおよびリアルタイムなコンテンツに対する)RAW、またはテーブルのビューがある。例えば、開発者は単に自己のアカウント鍵を提供して、コンテンツをプレビューすることを選択するだけである。

10

【0047】

従って、開発者は、モバイルアプリケーション、デスクトップアプリケーション、またはサービスアプリケーションを任意のプラットフォームで構築することができる。結果のRESTクエリを新たな開発者アプリケーションにコピーすることができ、そして、IaaSに対する情報基盤では、自動生成されたC#プロキシクラスを使用する能力により、ローカルオブジェクトモデルを作成する必要性またはXMLもしくはウェブサービスのコードの詳細を生成し理解する必要性を排除することができる。この点、別の非限定的な有益な態様では、開発者がプロキシクラスをダウンロードすることができ、それらを開発者のアプリケーションに含めて、サービスデータを出来るだけ少ない行数のコードで使うことができる。

20

【0048】

この点、IaaSに対する情報基盤の様々な実施形態では、開発者および情報労働者は、データを自分のアプリケーションおよび分析シナリオで即座に、定期的に、またはアプリケーション開発者によって指定された他の時間関数で、発見し、取得し、使用することができる。前述のように、開発者はアカウント鍵を取得し、これに続いて、コンテンツ要求および興味のあるコンテンツの申し込みについての1つまたは複数の実施形態に関して本明細書中で説明されたような、IaaSを可能とする、ネットワークサービスへのウェブサービスコールが行われる。

【0049】

つまり、一態様では、本明細書において1つまたは複数の実施形態について説明したIaaSのための情報基盤によって提供されるアカウント鍵は、開発者の秘密鍵である。この鍵により、開発者のアプリケーションと関連して使用されるコンテンツに対する課金およびレポートが可能となる。したがって、この開発者の秘密鍵は目的なしに共有されるものではなく、モバイルまたはデスクトップのソリューションで実行されるネイティブアプリケーションを開発し鍵をデプロイメントの一部として統合しようと計画する場合には、鍵の安全性確保において予防措置を取るべきである。例えば、暗号ストレージを使用して、確実に鍵が解読されないようにすることができる。

30

【0050】

コンテンツを使用するアプリケーションの追跡に使用されるアカウント鍵に加えて、ユーザIDは、開発者の個々のユーザを表すGUID(Globally Unique Identifier)である一意のユーザIDとすることができる。このフィールドにより、ユーザ単位ベースでプログラムの価格設定されるコンテンツに対して課金をすることができる。例えば、開発者がモバイルアプリケーションを開発し、個々のユーザがアプリケーションを使用している場合は、開発者は、要求が行われる度にユーザに代わって同一のGUIDを返す。しかしながら、開発者がウェブポータルを開発し、様々なユーザの代わりにウェブサービス要求を発行している場合には、ポータルを訪問するユーザ毎に新たなGUIDを割り当てるのは開発者に任されている。例えば、各登録ユーザにGUIDを割り当てることができ、または、各IPアドレス/ポートの組合せにユーザID、GUID等を割り当てることができる。

40

【0051】

REST APIは、本明細書において1つまたは複数の実施形態について説明したIaaSのため

50

の情報基盤からコンテンツを使用する、任意のプラットフォームで構築することができ、そして一実施形態では、プロキシクラスをC#でサービスに対して動的に生成することができる。これを達成することは、オブジェクトモデルをダウンロードし、それを現在の開発者のプロジェクトに追加し、アカウントおよびユーザ値、例えば一意のユーザ値を更新することに関する問題である。

【0052】

開発者が手動で要求を発行したい場合は、例を以下で提供するが、任意のサービスに対して、プレビュー機能を使用することができ、またはサービスエクスプローラから生成したURLを起動することもできる。サンプルのデータセットに対する例示的で非限定的なRESTクエリを以下に示す。一部または全部のコールは、セキュアなSSL(Secure Socket Layer)である得ることに留意されたい。

`https://api.sqlazureservices.com/UnService.svc/UNESCO(120)`

【0053】

"\$accountKey" および"\$uniqueUserId"要素も要求のヘッダに置かれ、これらは、例えば以下のように見える。

`$accountKey={developer account key}`

`$uniqueUserId={a GUID representing the user}`

【0054】

"\$accountKey"の値は、アカウントタブに存在する開発者のアカウント鍵を表し、"\$uniqueUserId"の値はサービスにアクセスするユーザのGUIDを表す。

【0055】

上述の認証アプローチの例は、他の認証アプローチも可能であるという点で非限定的であり、したがって、本明細書で説明した様々な実施形態で任意の代替的な認証アプローチを実装することができる。例えば、AccountKeyをパスワードとして用いる基本認証(RFC 1945)、トークンベースのアプローチを用いたACS(Access Control Server)、または他のトークンベースのアプローチを使用することもできる。

【0056】

したがって、本明細書における1つまたは複数の実施形態について説明したIaaSに対する情報基盤は、開発者および情報労働者が、任意のプラットフォームで貴重なデータサブスクリプションを容易に発見し、購入し、管理できるようにする新たなサービスまたはフレームワークである。情報基盤は、データ、画像、およびリアルタイムなウェブサービスを、有数の商用データプロバイダおよび権威ある公的なデータソース、ならびに非権威的な発行者から、共通のプロビジョニングおよび課金フレームワークのもとで統合された単一または複数の場所にまとめる、情報市場である。さらに、開発者および情報労働者はこの貴重なコンテンツを事実上任意のプラットフォーム、アプリケーションまたはビジネスワークフローで 사용할 ことができる。

【0057】

本明細書における1つまたは複数の実施形態について説明したIaaSに対する情報基盤を使用するための幾つかの例示的で非限定的なシナリオには、(a)消費者およびビジネスのシナリオに対する次世代の「キラーアプリケーション」に対する貴重なコンテンツを発見すること、(b)既存のアプリケーションまたはレポートを改善するための価値のあるデータを発見しライセンスを与えること、(c)異なるデータセットを独創的な方法でまとめて、ビジネスの性能およびプロセスについての新たな見識、例えば集約アルゴリズムを得ること、(d)プロブ、構造化データ、リアルタイムなウェブサービスに関する全てのコンテンツプロバイダにわたって即座かつ視覚的にAPIを開発 すること、ならびに(e)リッチなレポートおよび分析向けの、既存のワードプロセッサ、スプレッドシート、データベースクエリシステム等、既存のアプリケーションおよびデータベースシステムの内部でサードパーティのデータを使用すること、が含まれる。

【0058】

開発者に対する利点には、(a)試行的なサブスクリプションにより、データロイヤリテ

10

20

30

40

50

ィを支払うことなく、コンテンツを調査しアプリケーションを開発できること、(b)単純なトランザクションおよびサブスクリプションのモデルにより数百万ドルのデータセットに対する「従量課金制(pay as you go or grow)」のアクセスが可能となること、(c)データセットにわたる一貫したRESTベースのAPIにより任意のプラットフォームでの開発が促進されること、(d)APIの視覚的な構築および開発、結果のプレビュー、ならびに(e)自動的なC#プロキシクラスによりインスタントオブジェクトモデル(instant object model)が提供され、冗長なXMLおよびウェブサービスのコードを書く必要がなくなることが含まれる。

【0059】

情報労働者に対する利点には、(a)これには限定されないがExcelなどの既存のスプレッドシートソフトウェアにおけるデータで容易に作業するための、PowerPivotとの統合、(b)コンテンツを取得するための単純で予測可能なライセンスモデル、ならびに(c)SQL Server、SQL Azure Database、および他の既存の資産からのデータを使用する能力が含まれる。

【0060】

コンテンツパートナーに対する利益には、(a)プロブデータ、構造化データ、または動的なウェブサービスにとらわれない容易な発行およびオンボーディングのプロセス、(b)選択されたプラットフォームにおける、例えばVisual Studioおよび.NETまたは他のランタイムシステム開発を介した、開発を容易にするための開発者用ツールの構築、(c)グローバルな開発者および情報労働者のコミュニティに対する開発者のコンテンツの公開、(d)既存のコンテンツ使用アプリケーション内部でのコンテンツ発見および統合、ならびに(e)スケラブルなクラウドコンピューティングプラットフォームによる、全当事者に代わる記憶、提供、課金、およびレポート、が含まれる。

【0061】

本明細書における1つまたは複数の実施形態について説明したIaaSに対する情報基盤の非限定的な実装形態では、リッチウェブサービスの幾つかの機能には、(a)コンテンツカタログ全体にわたってサービスを使用するためのセキュアなRESTベースのモデル、(b)アクセスを簡略化するための、APIに組み込まれた動的ページネーション、(c)標準的なATOM 1.0フィードが大部分のサービスに利用可能であること、ならびに(d)全てのサービスにわたる一貫した課金、プロビジョニング、および利用レポート、が含まれる。

【0062】

本明細書における1つまたは複数の実施形態について説明したIaaSに対する情報基盤の非限定的な実装形態では、サービスエクスペローラのコンポーネントの幾つかの機能には、(a)開発を簡略化するために生成されたC#プロキシクラス、(b)タブ形式およびフィード、例えばサポートされる場合はATOM 1.0フィードとしてのデータのプレビュー、(c)コンパイルされたサービスコールが返す結果を理解するためのサービスの起動、(d)ドキュメンテーションの発見およびパラメータごとのサンプル値の発見、ならびに(e)開発を容易にするための、サービスコールに対する視覚的に構築されたURLのクリップボードへの即時コピー、が含まれる。

【0063】

本明細書における1つまたは複数の実施形態について説明したIaaSに対する情報基盤の非限定的な実装形態では、市場統合および発見ポータルコンポーネントの幾つかの機能には、(a)消費者および企業を含む領域にわたる新たなデータの発見、(b)サービスサブスクリプションおよび利用の制限の管理、(c)サービスにアクセスするためのアカウント鍵の管理、ならびに(d)日付およびアカウント鍵でグループ化した、アクセスされたサービス/データセットを含む詳細なアクセスレポートが含まれる。

【0064】

情報基盤は、コンテンツ所有者、アプリケーション開発者/ISV、情報労働者、および他の消費者/サブスクライバ(subscribers)に対して様々な価値提案を提供する。コンテンツ所有者にとって、データに課金しないデータプロバイダへの社会的な利益に加えて、

10

20

30

40

50

この情報基盤は、（データに課金するデータプロバイダにとって）データからの収益化を可能とするが、他の方法でもこのようなデータを発行することからの利益を引き出すことができる。情報基盤の開放性のため、コンテンツ所有者は、データのより高い可用性、およびISVおよび使用者／情報労働者へのデータの到達性を享受し、これらの全ては、特に過去と同様の独占モデルを開発するよりもより低いコストで利益が得られる。

【 0 0 6 5 】

アプリケーション開発者／ISVにとって、情報基盤は、一貫した形式でのデータへのアクセス、データアクセスに対する可変で低コストの価格モデル、予測可能な価格設定、そのデータを用いて形成された、使用者／IWがアプリケーションを利用する消費者／IWからの利益見込み、任意のプラットフォームを通じた消費者へのより広い到達性、ならびに単一の課金およびデータソース、を可能とする。

10

【 0 0 6 6 】

情報労働者／消費者にとって、情報基盤は、アプリケーションからの付加価値、現在の大規模オブジェクト(LOB)またはIWソフトウェアにおけるデータ使用、単一の課金およびデータソースに加えてデータアクセスに対する可変で低コストの価格モデル、を可能とする。

【 0 0 6 7 】

したがって、情報基盤は、現在の顧客および開発者の痛点を潜在的に大幅な収入上昇をもって解決し、モデル化、レポート、分析、およびトレンド分析のための即時データ(instant data)を伴ったビジネスインテリジェンスの機会をもたらし、データ所有者が情報基盤を介して自分のデータを格納するように促すことで任意のプラットフォームの採択およびそれへの固執性をもたらし、それにより、従来式のクラウド記憶環境と比較した場合の情報基盤の価値提案を差別化する。

20

【 0 0 6 8 】

図10のフローチャートに示すように、1000においては、コンテンツの所有者または発行者が情報基盤を介してデータを発行する様々な方法が記載されている。1010において、開発者が情報基盤を介してデータを使用するためのアプリケーションを開発できるようにする様々なツールが存在する。1020において、消費者または情報労働者はアプリケーションを使用し、または、直接データに対して問合わせてデータを使用することができる。最後に、1030において、情報基盤は、コンテンツチェーンにおける全当事者に代わって自動的な管理、監査、課金等を可能とする様々なリッチなツールを提供し、これらはトランザクションモデルにより可能となる。

30

【 0 0 6 9 】

この点、情報基盤の一部の主要な当事者には、データ所有者、アプリケーション開発者／ISV、および消費者／情報労働者が含まれる。一般に、データ所有者とは、データに課金したいか、または、他の理由により無償でデータを提供したいか、または、データに他の条件を強制したいエンティティである。次に、アプリケーション開発者／ISVとは、自分のアプリケーションを（例えば、広告を通じて、直接払い、間接払い等で）収益化したいか、または、自己に対する何らかの有益な理由から自分のアプリケーションを無償で提供したいエンティティである。情報労働者および使用者とは、生のデータを使用できる者、または、アプリケーション開発者が提供するアプリケーションを使用したいエンティティである。

40

【 0 0 7 0 】

図11は、本明細書で説明したIaaSを提供するエコシステムに参加できる様々な当事者を一般的に示すブロック図である。例えば、一組のネットワークアクセス可能な情報サービス1100が、データの機密性または他の特徴に応じて、様々な信頼されたまたは信頼されていないデータストア1110へのアクセスを提供する。図示されるように、したがって、エコシステムが任意の種類のデータ、プロブ、構造化データ、構造化されていないデータ等をサポートするので、データストア、即ち1112、1114、...、1116のタイプはあまり重要ではない。先に述べたように、システムは、データをエコシステ

50

ムに追加する発行者 1 1 2 0、データを使用するサブスクライバ 1 1 3 0、および自分達のアプリケーションでデータを使用することを支援するアプリケーション開発者またはプロバイダ 1 1 5 0を含む。アクセス情報ジェネレータ 1 1 7 0はまた、アカウント情報、鍵情報等の維持または実施を通じて、様々な当事者によるデータへのアクセスを管理することができる。この点、コンテンツ所有者 1 1 6 0は、コンテンツ所有者 1 1 6 0を発行者 1 1 2 0、サブスクライバ 1 1 3 0、および/またはアプリケーション開発者とすることもできるという点で、複数の役割のうちのいずれかに跨ることができる。一態様においては、全当事者に対する共通の情報基盤により、管理 1 1 6 5、監査 1 1 7 5、課金 1 1 7 5、ならびに情報基盤にわたって起こるデータランザクションに対する他の所望の付属サービスが可能となる。

10

【 0 0 7 1 】

この点、任意のプラットフォームからのIaaSを可能とするためのユーザフレンドリなデータプラットフォームに係る様々な実施形態は、データの消費者(IW、開発者、ISV)およびデータの消費者が、簡潔で費用対効果が良く便利な手法で処理することが可能な情報基盤である。情報基盤は貴重な(私的な)データおよびコミュニティの(公的な)データを手頃な方法で民主的にする(democratize)ことにより、IWが迅速に見識を見出すことができ、開発者が複数のデータソースを用いて創造的な手法で独創的なアプリケーションを構築することができ、開発者が任意のプラットフォームでの自分の作業を収益化できるようにする。例えば、情報基盤は、コンテンツに対する利用回数制料金ならびにサブスクリプション料金、コンテンツ料金(Pay for Content)(コンテンツ所有者が設定する「小売価格」)、支払データ手数料(Pay Data Fee)(「出荷および取扱手数料」)、およびBWをサポートし、さらに、論理ランザクション単位(レポート単位、API単位、ダウンロード単位等)の仲介料としてのデータ手数料をサポートする。

20

【 0 0 7 2 】

情報労働者(例えば、Office、SQL Server、Dynamicsのユーザ)に対して、情報基盤は、将来のEA統合ならびに予測可能な支出要件(ならびに、オンおよびオフを前提としたBIおよび「HPC」作業負荷をサポートするためのキャッシュ)を可能とするためのサブスクリプションをサポートする。したがって、代替手段には、コンテンツをユーザ毎、月毎に課金することが含まれ、これは、コンテンツパックまたはランザクション単位の価格設定を配信するためにバンドルしてもしなくともよく、例えばクラウドレポート/ビジネスインテリジェンスのオンデマンド価格設定は、利用単位の価格設定またはレポートギャラリーを介した垂直アプリケーションを可能とする一方で、大量のデータを移動させる必要性を排除することができる。

30

【 0 0 7 3 】

コンテンツプロバイダ(任意のデータタイプ、任意のクラウド)に対して、任意のプラットフォームを用いて、情報基盤は、任意の特定の所望のプラットフォームにおいて売上に刺激を与える貴重な提案となり、追加のコストなしに自動的にスケールでき、高レベルのSLAが得られる可能性がある。幾つかの非限定的な例では、データをセキュアにすることができ、データを以下の領域のデータと関連付けることができる。即ち、位置認識型サービスおよびデータ、商用および住宅の不動産、金融のデータおよびサービス等のデータである。非限定的なシナリオには、上位30社のNGO(Non-Governmental Organization)のデータセットにデータを提供することを含み得る。さらに、情報基盤が、「IaaS向けBing(Bing for Information as a Service)」、HPC等を通じてBIおよび可視化を示す機能を含むことができる。垂直アプリケーションの機会も同様に存在する。

40

【 0 0 7 4 】

1つの非限定的な実施形態では、データ仲介を従来式のブリックアンドモルタル(brick and mortar)戦略と類比させることができる。例えば、容量を棚の空間(例えば、構造化されたプロブデータと構造化されていないプロブデータとの混合)として表すことができ、商品原価(COGS)を面積(SA、プラットフォーム依存性、帯域幅)として表すことができ、コンテンツを商品として表すことができる(例えば、COGSをカバーするようにコンテン

50

ツプロバイダを最適化し、IWおよび開発者からの収益を最大化できる)。様々な実施形態では、オンボーディングプロセスを、データおよびサービスに関する品質基準を用いて、ならびにサービス内容合意書(SLA:Service Level Agreement)の適応により実装することができる。

【0075】

図12は、1つまたは複数の機能に従う上述のIaaSに対する情報基盤1210の例示的で非限定的な実装形態である。対話側には、情報労働者1200、開発者1202、および使用者1204があり、これらはSSL/RESTベースのAPI1206を介して情報基盤と通信することができる。負荷分散器1208を使用して、トラフィックの誘導を最適な方法で支援することができる。この点、入力ポータルウェブロール1220またはAPIウェブロール1222にルーティングされる。情報基盤1210からデータ側には、クラウドストレージフレームワーク1240のプロブデータセット1242もしくはプロブデータセット1255、または、関係データベースフレームワーク1250のデータセット1252もしくはデータセット1254にアクセスするための追加の負荷分散器1224または1226(例えば、WAまたはSA)がある。プロキシレイヤ1228を使用して、サードパーティクラウド1260のデータ1262またはデータ1264にアクセスすることができる。適用可能な場合は、コンテンツデータ抽象化レイヤ(DALs)1230を使用してコンテンツにアクセスすることができる。この点、データセットの複製または重複が様々なタイプのストレージに跨って存在することがあり、例えば、同一のデータがプロブデータおよび構造化データ、例えばSQLとして表されるかもしれない。

【0076】

データへの補足的なサービスとして、課金および発見サービス1270がオンライン課金1272(例えば、MOCp)または発見サービス1274(例えば、Pinpoint)を含むことができ、認証サービス1280が証明書管理1282(例えば、Live ID)またはコンテンツ認証1284、例えば、認証コンテンツサービス(authenticated content services, ACS)を含むことができる。アカウントサービス1290が、ログ取得/監査サービス1286またはアカウント管理1288サービスを含み得る。管理運用サービス1292は運用ダッシュボードサービス1294およびネットワーク運用サービス1296、例えばGomezを含み得る。

【0077】

図13は、IaaSを可能とするための一般的な情報基盤の1つまたは複数の実施形態に従う、データから当該データの消費者への例示的なエンドツーエンドの流れを示すブロック図である。例えば、IaaS1300は、商用データ1302および無償データ1304を含むことができ、これらは、営利開発者1310、動機が非営利である非営利開発者1312、および一般的に生産的な目的でデータを使用することに興味をもつ他の情報労働者1314といった様々な者が興味をもつ可能性がある。これらのエンティティは、探索サービス1320を使用して、どのアプリケーション1322、1324、、、1326が彼らにとって興味がありうるかを判定し、最終的にはデータをILA消費者1330およびDLA消費者1332へ同様に送信することができる。

【0078】

オープンデータプロトコル

ネットワーク対話および情報サービスを様々なコンピュータシステム構成およびプロトコルで実施できることは当業者には理解されよう。この点、本明細書で説明する1つまたは複数の実施形態で使用できるデータを問合わせて更新するための1つの非限定的な実装形態は、オープンデータプロトコル(OData)である。

【0079】

ODataは、データを問合わせて更新するためのウェブプロトコルである。ODataは、HTTP(HyperText Transfer Protocol)、AtomPub(Atom Publishing Protocol)、およびJSON(JavaScript Object Notation)のようなウェブ技術を適用して、様々なアプリケーション、サービス、およびストアからの情報へのアクセスを提供する。ある一般的な背景として、OD

ataは、AtomPubのクライアントおよびサーバを様々な製品で過去数年間実装した経験に基づいて、有機的に出現したものである。ODataを使用して、これらには限定されないが、リレーショナルデータベース、ファイルシステム、コンテンツ管理システム、および伝統的なウェブサイトを含め、様々なソースからの情報を公開し、情報にアクセスすることができる。ODataは、OSP(Open Specification Promise)のもとでリリースされ、エコシステムの発展を推進するために、誰もがOData実装と自由に相互運用できるようにするものである。ODataの機能を他のプロトコルに組み込んで、ネットワークデータを問合わせて更新するための同様な機能を有するハイブリッドなプロトコルを形成することもできる。

【0080】

ODataは、ウェブが動作する方法と一致する。一態様では、ODataは、例えばインターネットの動作方法と同様に、リソースを特定するための方法としてURI(Universal Resource Identifiers)を使用し、これらのリソースと対話するための均一なインターフェースを有するHTTP主体のプロトコルを使用する。1つの非限定的な態様では、ODataは、AtomPubにより普及したHTTP上の規約で構築されており、この規約は、データ、コンテンツ、および情報を独自に開発したシステムにわたり共有するプロセスを簡略化する。ODataは、交換すべき基本的なクエリおよびスキーマ情報をサポートするために諸実装形態が任意に実装できる、追加の規約を定義する。HTMLおよびJavaScript(登録商標)のクライアントとの統合を簡素化するために、ODataは、XMLベースのAtomPubフォーマットを補完するプロトコルの任意的なJSON表現を定義する。理解できるように、ODataプロトコルに関する追加の実装仕様または他の詳細についてはwww.odata.orgで見つけることができる。

【0081】

例示的なネットワーク環境および分散環境

当業者には、本明細書で説明した、任意のプラットフォームからのIaaSのための情報基盤向けの方法および装置の様々な実施形態および関連した諸実施形態を、コンピュータネットワークまたは分散コンピューティング環境の一部として展開することができ、任意の種類のデータストアに接続することができる、任意のコンピュータまたは他のクライアント装置もしくはサーバ装置と関連して実装することが、当業者には理解されよう。この点、本明細書で説明した様々な実施形態を、任意数のメモリまたはストレージユニットを有する任意のコンピュータシステムまたは環境で、および、任意数のストレージユニットにわたって生じる任意数のアプリケーションおよびプロセスで実装することができる。これには、リモートまたはローカルのストレージを有する、ネットワーク環境または分散コンピューティング環境において展開されたサーバコンピュータおよびクライアントコンピュータを有する環境が含まれるが、これには限定されない。

【0082】

図14は、例示的なネットワークコンピューティング環境または分散コンピューティング環境の非限定的な概略図を提供する。分散コンピューティング環境は、コンピューティングオブジェクトまたは装置1410、1412等およびコンピューティングオブジェクトまたは装置1420、1422、1424、1426、1428等を備え、これらのコンピューティングオブジェクトまたは装置は、アプリケーション1430、1432、1434、1436、1438によって表されるような、プログラム、メソッド、データストア、プログラム可能ロジック等を含むことができる。コンピューティングオブジェクトまたは装置1410、1412等およびコンピューティングオブジェクトまたは装置1420、1422、1424、1426、1428等が、PDA、オーディオ/ビデオ装置、携帯電話、MP3プレイヤー、ラップトップ等のような様々な装置を備えてもよいことは理解されよう。

【0083】

各コンピューティングオブジェクトまたは装置1410、1412等およびコンピューティングオブジェクトまたは装置1420、1422、1424、1426、1428等は、1つまたは複数の他のコンピューティングオブジェクトまたは装置1410、1412等ならびにおよびコンピューティングオブジェクトまたは装置1420、1422、1

4 2 4、1 4 2 6、1 4 2 8 等と、通信ネットワーク 1 4 4 0 により直接または間接に通信することができる。図 1 4 では単一の要素として図示したが、ネットワーク 1 4 4 0 が、サービスを図 1 4 のシステムに提供する他のコンピューティングオブジェクトおよびコンピューティング装置を備えてもよく、および/または、図示していないが複数の相互接続ネットワークを表してもよい。各コンピューティングオブジェクトまたは装置 1 4 1 0、1 4 1 2 等または 1 4 2 0、1 4 2 2、1 4 2 4、1 4 2 6、1 4 2 8 等が、様々な実施形態に従って提供される任意のプラットフォームからの IaaS に対する情報基盤との通信または情報基盤の実装に適した、API、または他のオブジェクト、ソフトウェア、ファームウェア、および/またはハードウェアを利用するアプリケーション 1 4 3 0、1 4 3 2、1 4 3 4、1 4 3 6、1 4 3 8 のようなアプリケーションを含むこともできる。

10

【0084】

分散コンピューティング環境をサポートする様々なシステム、コンポーネント、およびネットワーク構成が存在する。例えば、コンピューティングシステムを、有線または無線システムにより、ローカルネットワークまたは広域分散ネットワークにより、接続することができる。現在、多くのネットワークがインターネットに接続されており、インターネットは、広域分散コンピューティングに対する情報基盤を提供し多種多様なネットワークを包含するが、任意のネットワーク情報基盤を、様々な実施形態で説明した技法に伴う例示的な通信に使用することができる。

【0085】

したがって、クライアント/サーバ、ピアツーピア、またはハイブリッドアーキテクチャのような、多数のネットワークポロジおよびネットワーク情報基盤を利用することができる。クライアント/サーバのアーキテクチャ、特にネットワークシステムでは、クライアントは通常は、別のコンピュータ、例えばサーバにより提供される共有ネットワークリソースにアクセスするコンピュータである。図 1 4 の例では、非限定的な例として、コンピューティングオブジェクトまたは装置 1 4 2 0、1 4 2 2、1 4 2 4、1 4 2 6、1 4 2 8 等をクライアントと考えることができ、コンピューティングオブジェクトまたは装置 1 4 1 0、1 4 1 2 等をサーバと考えることができ、この場合、コンピューティングオブジェクトまたは装置 1 4 1 0、1 4 1 2 等が、コンピューティングオブジェクトまたは装置 1 4 2 0、1 4 2 2、1 4 2 4、1 4 2 6、1 4 2 8 等からデータを受信すること、データを記憶すること、データを処理すること、データをコンピューティングオブジェクトまたは装置 1 4 2 0、1 4 2 2、1 4 2 4、1 4 2 6、1 4 2 8 等に送信することのようなデータサービスを提供するが、状況に応じて任意のコンピュータをクライアント、サーバ、またはその双方と考えることができる。これらのコンピューティング装置のうち何れかが、1 つまたは複数の実施形態について本明細書で説明した任意のプラットフォームおよび関連する技術からの IaaS に対する情報基盤に関与するデータを処理し、またこれに関与するサービスもしくはタスクを要求してもよい。

20

30

【0086】

サーバは典型的に、インターネットまたは無線ネットワーク情報基盤のような、リモートまたはローカルのネットワーク上でアクセス可能なリモートコンピュータシステムである。クライアントプロセスが第 1 のコンピュータシステムでアクティブであり、サーバプロセスが第 2 のコンピュータシステムでアクティブであり、互いと通信媒体上で通信し、それにより分散機能を提供して、複数のクライアントがサーバの情報収集機能を利用できるようにしてもよい。ユーザプロファイリングに従って利用される任意のソフトウェアオブジェクトを、スタンドアロンで提供することができ、または、複数のコンピューティング装置もしくはオブジェクトにわたって分散させることができる。

40

【0087】

通信ネットワーク/パス 1 4 4 0 がインターネットであるネットワーク環境では、例えば、コンピューティングオブジェクトまたは装置 1 4 1 0、1 4 1 2 等は、コンピューティングオブジェクトまたは装置 1 4 2 0、1 4 2 2、1 4 2 4、1 4 2 6、1 4 2 8 等が、HTTP のような幾つかの公知なプロトコルのうち何れかを介して通信するウェブサーバで

50

あり得る。先に述べたように、コンピューティングオブジェクトまたは装置 1 4 1 0、1 4 1 2 等が、コンピューティングオブジェクトまたは装置 1 4 2 0、1 4 2 2、1 4 2 4、1 4 2 6、1 4 2 8 等の役割を果たしてもよく、その逆も成り立つが、これは分散コンピューティング環境の特徴であろう。

【 0 0 8 8 】

例示的なコンピューティング装置

先に述べたように、本明細書で説明した様々な実施形態は任意の装置に適用され、この場合、任意のプラットフォームからの IaaS に対する 1 つ以上の情報基盤を実装することが望ましいであろう。したがって、ハンドヘルドの、ポータブルの、および他のコンピューティング装置ならびにあらゆる種類のコンピューティングオブジェクトが本明細書で説明した様々な実施形態と関連して、即ち、任意のプラットフォームからの IaaS に対する情報基盤と関連して何らかの機能を装置が提供できる任意の場面で、使用されると想定されていることは理解されよう。したがって、図 1 5 において後述する下記の汎用目的のリモートコンピュータは、一例に過ぎず、本開示の諸実施形態を、ネットワーク/バスの相互運用性および対話を有する任意のクライアントで実装してもよい。

【 0 0 8 9 】

必要ではないが、諸実施形態のうち何れかを、装置またはオブジェクトに対するサービスの開発者による使用のためにオペレーティングシステムを介して部分的に実装することができ、および/または、動作可能コンポーネントと関連して動作するアプリケーションソフトウェアに含めることができる。ソフトウェアを、プログラムモジュールのような、1 つまたは複数のコンピュータ、例えばクライアントワークステーション、サーバ、または他の装置によって実行されているコンピュータ実行可能命令の一般的なコンテキストで説明してもよい。ネットワークの対話を様々なコンピュータシステム構成およびプロトコルで実行してもよいことは当業者には理解されよう。

【 0 0 9 0 】

図 1 5 は、1 つまたは複数の実施形態を実装できる適切なコンピューティングシステム環境の一例を示すが、上記で明らかにしたように、コンピューティングシステム環境 1 5 0 0 は適切なコンピューティング環境の一例に過ぎず、諸実施形態の何れかの使用または機能の範囲に関して何ら限定を示唆しようとするものではない。コンピューティング環境 1 5 0 0 は、例示的な動作環境 1 5 0 0 内に図示されるコンポーネントの任意の 1 つまたは組合せに関して、いかなる依存性または要件も有するように解釈されるべきではない。

【 0 0 9 1 】

図 1 5 を参照すると、本明細書における 1 つまたは複数の実施形態を実装するための例示的なリモート装置は、ハンドヘルドコンピュータ 1 5 1 0 の形の汎用目的のコンピューティング装置を含めることができる。ハンドヘルドコンピュータ 1 5 1 0 のコンポーネントは、処理装置 1 5 2 0、システムメモリ 1 5 3 0、およびシステムメモリを含む様々なシステムコンポーネントを処理演算装置 1 5 2 0 に接続するシステムバス 1 5 2 1 を備えてもよいが、これらには限られない。

【 0 0 9 2 】

コンピュータ 1 5 1 0 は典型的に、様々なコンピュータ読取可能媒体を備え、コンピュータ 1 5 1 0 がアクセスできる任意の利用可能な媒体であり得る。システムメモリ 1 5 3 0 は、ROM(Read Only Memory)および/または RAM(Random Access Memory)のような揮発性および/または不揮発性のメモリの形のコンピュータ記憶媒体を備えてもよい。限定ではなく例として、メモリ 1 5 3 0 が、オペレーティングシステム、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を備えてもよい。

【 0 0 9 3 】

ユーザは、入力装置 1 5 4 0 を通じてコマンドおよび情報をコンピュータ 1 5 1 0 に入力することができる。モニタまたは他のタイプのディスプレイ装置も、出力インターフェース 1 5 5 0 のようなインターフェースを介してシステムバス 1 5 2 1 に接続される。モニタに加えて、コンピュータは、出力インターフェース 1 5 5 0 を介して接続される、ス

ピーカおよびプリンタのような他の周辺出力装置を備えてもよい。

【0094】

コンピュータ1510は、リモートコンピュータ1570のような1つまたは複数の他のリモートコンピュータへの論理接続を用いてネットワーク環境または分散環境で動作してもよい。リモートコンピュータ1570は、パーソナルコンピュータ、サーバ、ルータ、ネットワークPC、ピアデバイスもしくは他の共通ネットワークノード、または他の任意のリモート媒体使用装置または送信装置であってもよく、コンピュータ1510に関して上述した要素の一部または全部を備えてもよい。図15に示す論理接続には、LAN(Local Area Network)またはWAN(Wide Area Network)のようなネットワーク1571が含まれるが、他のネットワーク/バスを含むこともできる。このようなネットワーク環境は自宅、職場、企業規模のコンピュータネットワーク、イントラネット、およびインターネットにおいて一般的である。

10

【0095】

上述のように、例示的な諸実施形態を様々なコンピューティング装置、ネットワーク、および広告アーキテクチャと関連して説明したが、基礎となる概念を、クラウドまたはネットワークのサービスとの対話と関連してデータを発行し、データ向けのアプリケーションを構築し、データを使用するのが望ましい、任意のネットワークシステムおよび任意のコンピューティング装置またはシステムに適用してもよい。

【0096】

本明細書で説明した実施形態の1つまたは複数を実装する複数の方法、例えば、適切なAPI、ツールキット、ドライバコード、オペレーティングシステム、コントロール、スタンドアロンまたはダウンロード可能なソフトウェアオブジェクト等が存在する。これらにより、アプリケーションおよびサービスが、任意のプラットフォームからのIaaSに対する情報基盤を使用することができる。諸実施形態を、API(または他のソフトウェアオブジェクト)の観点から、ならびに説明した実施形態の1つまたは複数に従って任意のプラットフォームからのIaaSに対する情報基盤のプロビジョンを促進するソフトウェアオブジェクトまたはハードウェアオブジェクトの観点から検討してもよい。本明細書で説明した様々な実装形態および実施形態は、全体としてハードウェアである態様、一部がハードウェアで一部がソフトウェアである態様、ならびにソフトウェアである態様を有してもよい。

20

【0097】

「例示的」という用語は、本明細書では、例、事例、または実例としての役割を果たすことを意味するために使用される。疑義を避けるため、本明細書で開示した発明の技術的範囲はこのような例によっては限定されない。さらに、本明細書で「例示的」として説明した任意の態様または設計は、必ずしも他の態様または設計に対して好適または有利と考えるべきではなく、また、当業者に公知な等価の例示的な構造および技法を除外することを意味しない。さらに、「含まれる」、「有する」、「含む」という用語、および他の同様な単語が発明を実施するための形態または特許請求の範囲の何れかで使用される限りにおいては、疑義を避けるため、このような用語は、どの追加または他の要素も除外しないオープンな移行語としての「備える」という用語と同様に包括的であることを意図している。

30

40

【0098】

先に述べたように、本明細書で説明した様々な技法を、ハードウェアもしくはソフトウェアまたは、適切な場合は、その双方の組合せと関連して実装してもよい。本明細書で使用する際、「コンポーネント」、「システム」等という用語は同様に、コンピュータ関連のエンティティ、即ち、ハードウェア、ハードウェアおよびソフトウェアの組合せ、ソフトウェア、または実行中のソフトウェアの何れかを指すことが意図されている。例えば、コンポーネントは、プロセッサ上で実行されているプロセス、プロセッサ、オブジェクト、実行可能ファイル、実行スレッド、プログラム、および/またはコンピュータであってもよいがこれらに限られない。例として、コンピュータ上で実行されているアプリケーションおよびコンピュータの双方をコンポーネントとすることができる。1つまたは複数の

50

コンポーネントが、プロセスおよび／または実行スレッドの内部に存在してもよく、コンポーネントを１つのコンピュータのローカルに存在させてもよく、および／または、２つまたはそれ以上のコンピュータ間で分散させてもよい。

【００９９】

以上では、幾つかのコンポーネントの間の対話に関してシステムを説明した。このようなシステムおよびコンポーネントに、上述したものの様々な置換および組合せに従って、これらのコンポーネントまたは指定のサブコンポーネント、指定のコンポーネントもしくはサブコンポーネントの一部、および／または追加のコンポーネントを含めることができることは理解されよう。サブコンポーネントを、親コンポーネントに含める（階層的）のではなく他のコンポーネントに通信可能に接続されたコンポーネントとして実装することもできる。さらに、１つまたは複数のコンポーネントを、単一のコンポーネントに統合して集約的な機能を提供してもよく、または、管理レイヤのような幾つかの別個のサブコンポーネント、および任意の１つまたは複数の中間レイヤに分割してもよく、統合的な機能を提供するためにこのようなサブコンポーネントと通信可能に接続するために提供してもよいことに留意されたい。本明細書で説明した任意のコンポーネントが、本明細書では具体的に説明していないが当業者には一般に公知である１つまたは複数の他のコンポーネントと対話してもよい。

10

【０１００】

上述の例示的なシステムを考慮すると、開示された主題に従って実装できる方法を、様々な図面のフローチャートを参照して理解することができる。説明を簡単にするため、方法を一連のブロックで図示および説明しているが、一部のブロックを本明細書で図示および説明したのとは別の順序で、および／または、本明細書で図示および説明したのとは別のブロックと並行に実施してもよいので、特許請求される主題はブロックの順序によっては限定されないことを理解され、認識されたい。フローチャートを介して非逐次的なフローまたは分岐のフローを示す場合、同一または同様な結果を実現する様々な他の分岐、流れ、ブロックの順序を実装してもよいことは理解されよう。さらに、以降で説明する方法を実装するのに、必ずしも図示された全てのブロックが必要なわけではない。

20

【０１０１】

幾つかの実施形態では、クライアント側の視点を説明したが、疑義を避けるため、対応するサーバの視点も存在し、逆も成り立つことは理解されよう。同様に、方法を実施する場合、方法を１つまたは複数のコンポーネントを介して実施するように構成されたストレージおよび少なくとも１つのプロセッサを有する対応する装置を提供することができる。

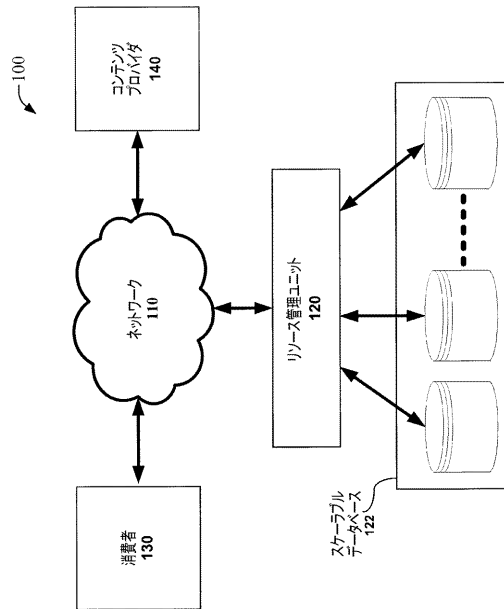
30

【０１０２】

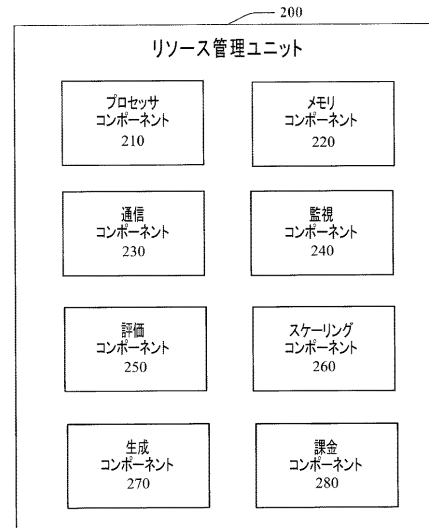
様々な実施形態を様々な図面の好適な諸実施形態と関連して説明したが、他の同様な諸実施形態を使用してもよく、または、同一の機能を実施するために様々な実施形態から逸脱することなく修正および追加を説明した実施形態に加えてもよいことは理解されよう。さらに、上述の諸実施形態の１つまたは複数の態様を複数の処理チップまたは装置においてまたはそれらに跨って実装してもよく、ストレージを同様に複数の装置に跨って実現してもよい。したがって、本発明はどの実施形態にも限定されず、添付された特許請求の範囲に従う範囲で解釈されるべきである。

40

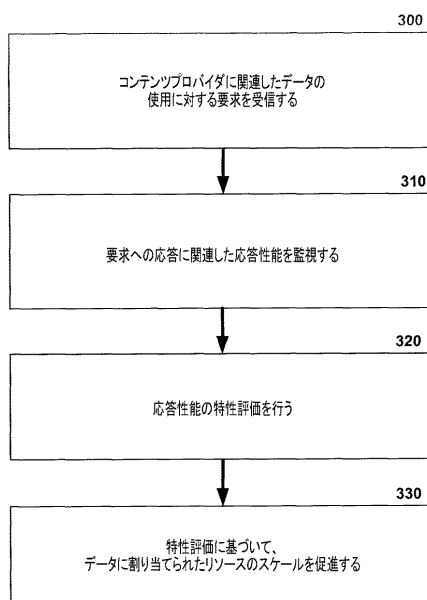
【図 1】



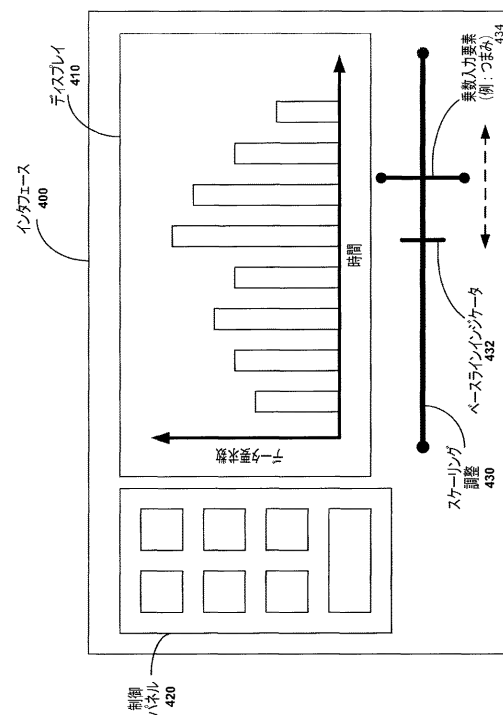
【図 2】



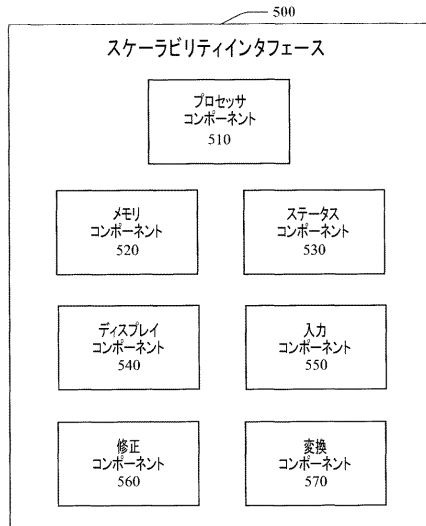
【図 3】



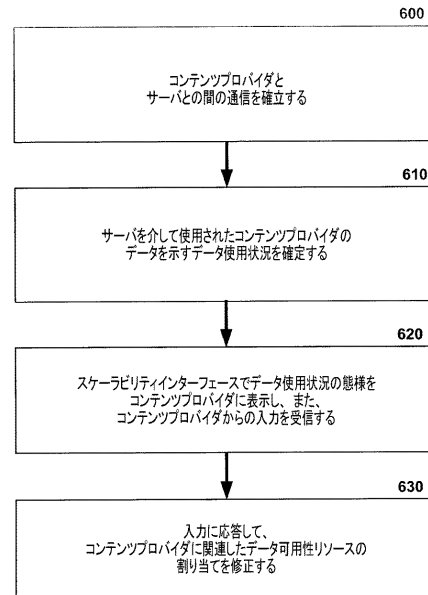
【図 4】



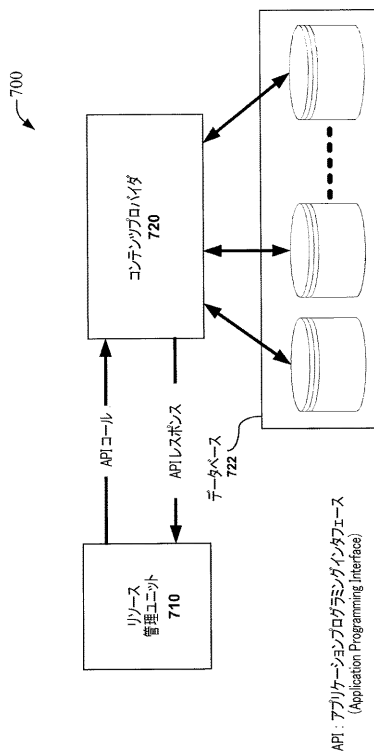
【図 5】



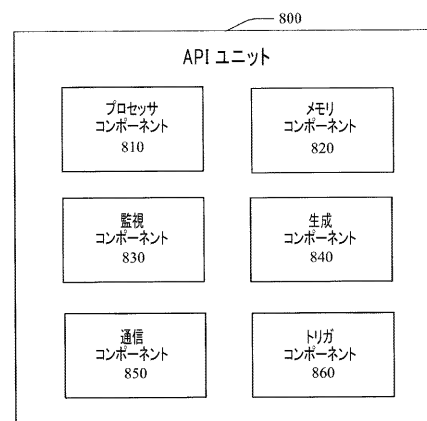
【図 6】



【図 7】

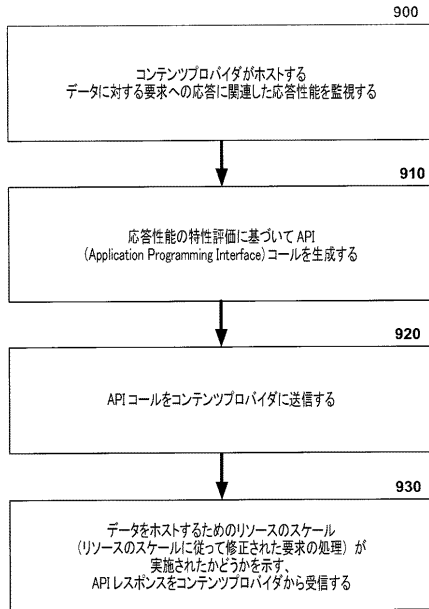


【図 8】

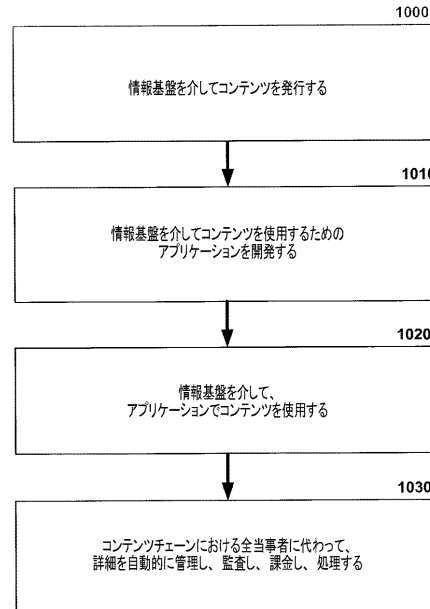


API : アプリケーションプログラミングインターフェース
(Application Programming Interface)

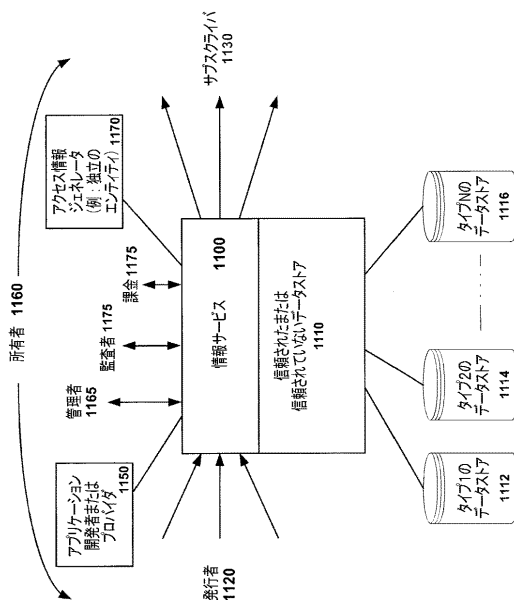
【図 9】



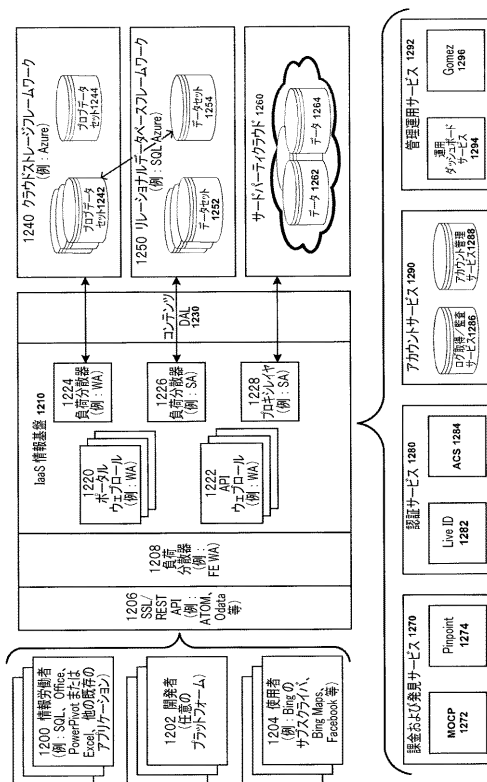
【図 10】



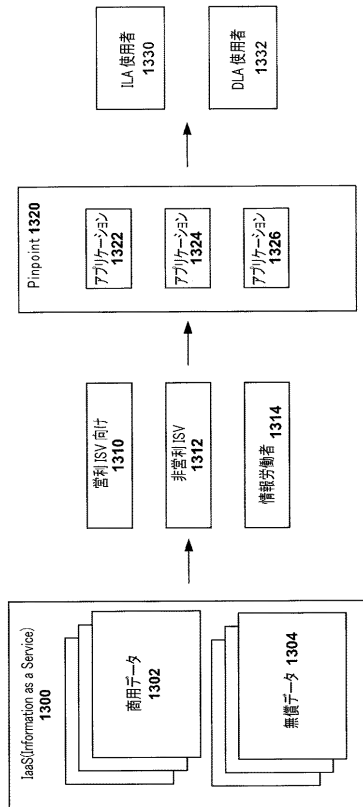
【図 11】



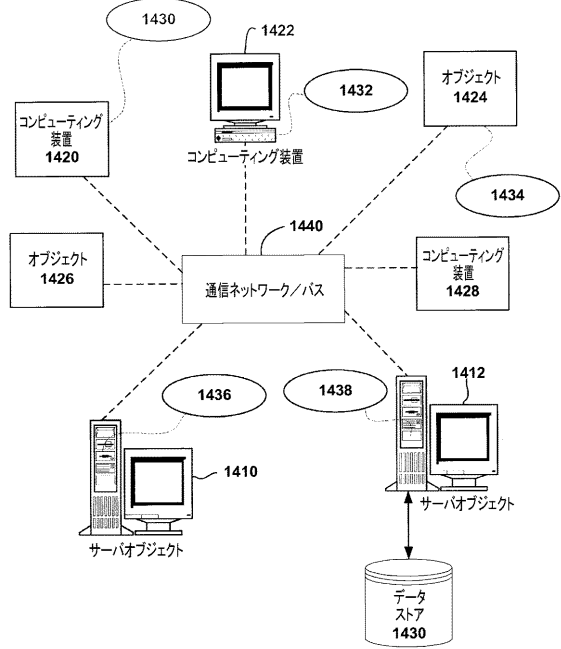
【図 12】



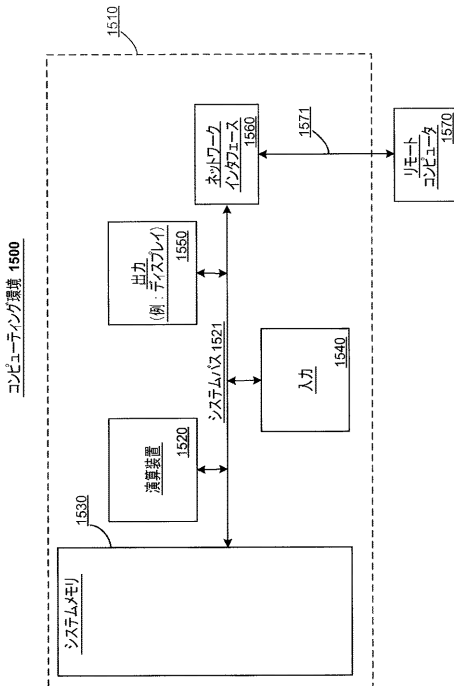
【図 13】



【図 14】



【図 15】



フロントページの続き

- (74)代理人 100153028
弁理士 上田 忠
- (74)代理人 100120112
弁理士 中西 基晴
- (74)代理人 100196508
弁理士 松尾 淳一
- (74)代理人 100147991
弁理士 鳥居 健一
- (74)代理人 100119781
弁理士 中村 彰吾
- (74)代理人 100162846
弁理士 大牧 綾子
- (74)代理人 100173565
弁理士 末松 亮太
- (74)代理人 100138759
弁理士 大房 直樹
- (72)発明者 ルネ ジャコ ボウ
アメリカ合衆国 98052 ワシントン州 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 エルシーエー - インターナショナル パテンツ内
- (72)発明者 ルーカス ゲオゼス
アメリカ合衆国 98052 ワシントン州 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 エルシーエー - インターナショナル パテンツ内
- (72)発明者 ムー コースラヴィ
アメリカ合衆国 98052 ワシントン州 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 エルシーエー - インターナショナル パテンツ内
- (72)発明者 クリスチャン リアンズパーガー
アメリカ合衆国 98052 ワシントン州 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 エルシーエー - インターナショナル パテンツ内
- (72)発明者 ロジャー ソーレン モール
アメリカ合衆国 98052 ワシントン州 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 エルシーエー - インターナショナル パテンツ内

審査官 篠塚 隆

- (56)参考文献 国際公開第2010/109952(WO, A1)
特開2007-200347(JP, A)

- (58)調査した分野(Int.Cl., DB名)

G06F3/01
3/048 - 3/0482
3/0485
3/0487 - 3/0489
9/46
9/48
9/50 - 9/52
9/54
11/28 - 11/34
12/00

19 / 00
G 0 6 Q 1 0 / 0 0 - 1 0 / 1 0
3 0 / 0 0 - 3 0 / 0 8
5 0 / 0 0 - 5 0 / 2 0
5 0 / 2 6 - 9 9 / 0 0
H 0 4 L 1 2 / 0 0 - 1 2 / 2 6
1 2 / 5 0 - 1 2 / 9 5 5