

(19)日本国特許庁(JP)

(12)公表特許公報(A)

(11)公表番号
特表2025-506015
(P2025-506015A)

(43)公表日 令和7年3月5日(2025.3.5)

(51)国際特許分類

G 0 6 F 9/50 (2006.01)

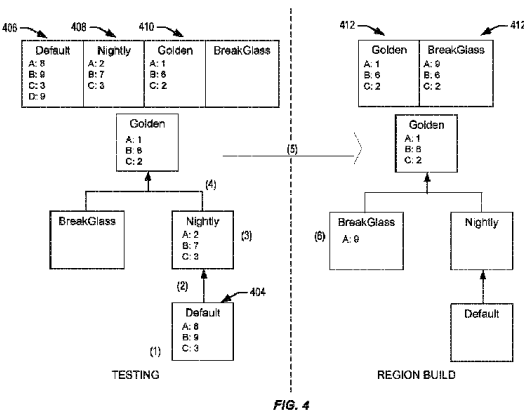
F I G 0 6 F 9/50 1 5 0 Z

審査請求 未請求 予備審査請求 未請求 (全61頁)			
(21)出願番号	特願2024-547098(P2024-547098)	(71)出願人	502303739
(86)(22)出願日	令和5年1月31日(2023.1.31)	(74)代理人	オラクル・インターナショナル・コーポレーション
(85)翻訳文提出日	令和6年8月16日(2024.8.16)		アメリカ合衆国 9 4 0 6 5 カリフォルニア州 レッドウッド ショアーズ, メール ストップ 5 オービー 7 オラクル パークウェイ 5 0 0
(86)国際出願番号	PCT/US2023/012035		110001195
(87)国際公開番号	WO2023/154198		弁理士法人深見特許事務所
(87)国際公開日	令和5年8月17日(2023.8.17)	(72)発明者	ミラー, エリック・ジョセフ
(31)優先権主張番号	63/308,003		アメリカ合衆国、9 4 0 6 5 カリフォルニア州、レッドウッド・シティ、オラクル・パークウェイ、5 0 0、オラクル・インターナショナル・コーポレーション内
(32)優先日	令和4年2月8日(2022.2.8)		
(33)優先権主張国・地域又は機関	米国(US)		
(31)優先権主張番号	63/312,814		
(32)優先日	令和4年2月22日(2022.2.22)		
(33)優先権主張国・地域又は機関	米国(US)		
(31)優先権主張番号	63/315,005		
最終頁に続く		最終頁に続く	

(54)【発明の名称】 クラウドリージョンにデータセンタを構築するための技術

(57)【要約】

自動化されたリージョン構築を実行するための技術が説明されている。クラウドコンピューティング環境のオーケストレーションサービス（例えば、マルチブロックオーケストレータ）は、1つまたは複数のデータセンタに対応するリージョン内でブートストラップされるサービスに対応する構成ファイルを取得してもよい。サービスの各々は、少なくとも1つのインフラストラクチャコンポーネントまたは対応するソフトウェアアーティファクトを含むリソースのそれぞれのセットと関連付けられてもよい。オーケストレーションサービスは、構成ファイルに少なくとも部分的に基づいてサービス間の依存を識別してもよい。サービスをブートストラップするためのオペレーションが実行される順序は、識別された依存に少なくとも部分的に基づいて決定されてもよい。オーケストレーションサービスは、決定された順序に従ってサービスをブートストラップするための対応するオペレーションを実行するようにプロビジョニングおよびデプロイメントマネージャにインクリメントに命令してもよい。



【特許請求の範囲】**【請求項 1】**

コンピュータが実行する方法であって、

クラウドコンピューティング環境のオーケストレーションサービスが、1つまたは複数のデータセンタに対応するリージョン内でブートストラップされる複数のサービスに対応する複数の構成ファイルを取得することを含み、前記複数のサービスの各々は、インフラストラクチャコンポーネントまたは対応するソフトウェアアーティファクトのうちの少なくとも1つを含むリソースのそれぞれのセットと関連付けられており、

前記方法は、

前記オーケストレーションサービスが、前記複数の構成ファイルに少なくとも部分的に基づいて前記複数のサービスの間の1つまたは複数の依存を識別することと、

前記オーケストレーションサービスが、識別された前記1つまたは複数の依存に少なくとも部分的に基づいて、前記複数のサービスをブートストラップするためのオペレーションが実行される順序を決定することと、

前記オーケストレーションサービスが、決定された前記順序に従って前記複数のサービスをブートストラップするための対応するオペレーションを実行するようにプロビジョニングおよびデプロイメントマネージャにインクリメントに命令することと

をさらに含む、コンピュータが実行する方法。

10

【請求項 2】

前記1つまたは複数の依存のレコードを維持することと、

20

第1のサービスが、利用可能な第2のサービスのケイパビリティへの対応する依存を有することを識別することと、

前記第2のサービスの前記ケイパビリティが利用不可能であることを識別することと、

前記ケイパビリティが利用可能になったときに通知されるリクエストを示すデータをケイパビリティサービスへ送信することと

をさらに含む、請求項1に記載のコンピュータが実行する方法。

【請求項 3】

前記第2のサービスの前記ケイパビリティが利用不可能であることを識別することは、前記第2のサービスの前記ケイパビリティが利用可能であることを示す通知が前記クラウドコンピューティング環境の前記ケイパビリティサービスからまだ受信されていないことを決定することを含む、請求項2に記載のコンピュータが実行する方法。

30

【請求項 4】

前記プロビジョニングおよびデプロイメントマネージャにインクリメントに命令することは、

前記ケイパビリティサービスから、前記ケイパビリティが利用可能であることを示す対応する通知を受信することと、

前記ケイパビリティが利用可能であるという前記通知を受信することに応答して、前記第2のサービスに対応するリソースを前記リージョン内でブートストラップするためのオペレーションを実行するように前記プロビジョニングおよびデプロイメントマネージャに命令することと

40

をさらに含む、請求項1～請求項3のいずれかに記載のコンピュータが実行する方法。

【請求項 5】

前記複数のサービスをブートストラップするための前記対応するオペレーションを実行するように前記プロビジョニングおよびデプロイメントマネージャにインクリメントに命令することにより、前記プロビジョニングおよびデプロイメントマネージャは、少なくとも1つのインフラストラクチャコンポーネントをプロビジョニングし、前記少なくとも1つのインフラストラクチャコンポーネントへ1つまたは複数のアーティファクトをデプロイする、請求項1～請求項4のいずれかに記載のコンピュータが実行する方法。

【請求項 6】

前記1つまたは複数の依存は、前記複数の構成ファイルの1つまたは複数のパースを実

50

行することに少なくとも部分的に基づいて識別される、請求項 1～請求項 5 のいずれかに記載のコンピュータが実行する方法。

【請求項 7】

前記オーケストレーションサービスが、前記 1 つまたは複数の依存および前記複数のサービスをブートストラップするためのオペレーションが実行される順序を識別することに少なくとも部分的に基づいて、1 つまたは複数の有向グラフを生成することをさらに含み、決定された前記順序に従って前記複数のサービスをブートストラップするための前記対応するオペレーションを実行するように前記プロビジョニングおよびデプロイメントマネージャにインクリメントに命令することは、前記 1 つまたは複数の有向グラフの 1 つまたは複数のトラバースに少なくとも部分的に基づく、請求項 1～請求項 6 のいずれかに記載のコンピュータが実行する方法。

10

【請求項 8】

クラウドコンピューティングシステムであって、

1 つまたは複数のプロセッサと、

前記 1 つまたは複数のプロセッサによって実行されると前記クラウドコンピューティングシステムのオーケストレーションサービスに、

1 つまたは複数のデータセンタに対応するリージョン内でブートストラップされる複数のサービスに対応する複数の構成ファイルを取得させ、前記複数のサービスの各々は、インフラストラクチャコンポーネントおよび対応するソフトウェアアーティファクトを含むリソースのそれぞれのセットと関連付けられており、

20

前記複数の構成ファイルに少なくとも部分的に基づいて前記複数のサービスの間の 1 つまたは複数の依存を識別させ、

識別された前記 1 つまたは複数の依存に少なくとも部分的に基づいて、前記複数のサービスをブートストラップするためのオペレーションが実行される順序を決定させ、

決定された前記順序に従って前記複数のサービスをブートストラップするための対応するオペレーションを実行するようにプロビジョニングおよびデプロイメントマネージャにインクリメントに命令させる

コンピュータ実行可能命令を記憶する 1 つまたは複数のメモリとを含む、クラウドコンピューティングシステム。

【請求項 9】

30

前記命令を実行することは、さらに、前記オーケストレーションサービスに、

第 1 のサービスが、利用可能な第 2 のサービスのケイパビリティへの対応する依存を有することを識別させ、

前記第 2 のサービスの前記ケイパビリティが利用不可能であることを識別させ、

前記ケイパビリティが利用可能になったときに通知されるリクエストを示すデータをケイパビリティサービスへ送信させる、請求項 8 に記載のクラウドコンピューティングシステム。

【請求項 10】

前記オーケストレーションサービスに、前記第 2 のサービスの前記ケイパビリティが利用不可能であることを識別させる前記命令を実行することは、さらに、前記オーケストレーションサービスに、前記第 2 のサービスの前記ケイパビリティが利用可能であることを示す通知が前記クラウドコンピューティングシステムの前記ケイパビリティサービスからまだ受信されていないことを決定させる、請求項 9 に記載のクラウドコンピューティングシステム。

40

【請求項 11】

前記プロビジョニングおよびデプロイメントマネージャにインクリメントに命令するために前記命令を実行することは、さらに、前記オーケストレーションサービスに、

前記ケイパビリティサービスから、前記ケイパビリティが利用可能であることを示す対応する通知を受信させ、

前記ケイパビリティが利用可能であるという前記通知を受信することに応答して、前記

50

第 2 のサービスに対応するリソースを前記リージョン内でブートストラップするためのオペレーションを実行するように前記プロビジョニングおよびデプロイメントマネージャに命令させる、請求項 8 ～請求項 10 のいずれかに記載のクラウドコンピューティングシステム。

【請求項 12】

前記複数のサービスをブートストラップするための前記対応するオペレーションを実行するように前記プロビジョニングおよびデプロイメントマネージャにインクリメントに命令するために前記命令を実行することは、前記プロビジョニングおよびデプロイメントマネージャに、少なくとも 1 つのインフラストラクチャコンポーネントをプロビジョニングさせ、前記少なくとも 1 つのインフラストラクチャコンポーネントへ 1 つまたは複数のアーティファクトをデプロイさせる、請求項 8 ～請求項 11 のいずれかに記載のクラウドコンピューティングシステム。

10

【請求項 13】

前記 1 つまたは複数の依存は、前記複数の構成ファイルの 1 つまたは複数のパースを実行することに少なくとも部分的に基づいて識別される、請求項 8 ～請求項 12 のいずれかに記載のクラウドコンピューティングシステム。

【請求項 14】

前記命令を実行することは、さらに、前記オーケストレーションサービスに、前記 1 つまたは複数の依存および前記複数のサービスをブートストラップするためのオペレーションが実行される順序を識別することに少なくとも部分的に基づいて、1 つまたは複数の有向グラフを生成させ、決定された前記順序に従って前記複数のサービスをブートストラップするための前記対応するオペレーションを実行するように前記プロビジョニングおよびデプロイメントマネージャにインクリメントに命令することは、前記 1 つまたは複数の有向グラフの 1 つまたは複数のトラバースに少なくとも部分的に基づく、請求項 8 ～請求項 13 のいずれか 1 項に記載のクラウドコンピューティングシステム。

20

【請求項 15】

クラウドコンピューティングシステムのオーケストレーションサービスに対応する 1 つまたは複数のプロセッサによって実行されると、前記オーケストレーションサービスに、1 つまたは複数のデータセンタに対応するリージョン内でブートストラップされる複数のサービスに対応する複数の構成ファイルを取得させ、前記複数のサービスの各々は、インフラストラクチャコンポーネントおよび対応するソフトウェアアーティファクトを含むリソースのそれぞれのセットと関連付けられており、

30

前記複数の構成ファイルに少なくとも部分的に基づいて前記複数のサービスの間の 1 つまたは複数の依存を識別させ、

識別された前記 1 つまたは複数の依存に少なくとも部分的に基づいて、前記複数のサービスをブートストラップするためのオペレーションが実行される順序を決定させ、

決定された前記順序に従って前記複数のサービスをブートストラップするための対応するオペレーションを実行するようにプロビジョニングおよびデプロイメントマネージャにインクリメントに命令させる

コンピュータ実行可能命令を記憶する非一時的コンピュータ可読記憶媒体。

40

【請求項 16】

前記命令を実行することは、さらに、前記オーケストレーションサービスに、

第 1 のサービスが、利用可能な第 2 のサービスのケイパビリティへの対応する依存を有することを識別させ、

前記第 2 のサービスの前記ケイパビリティが利用不可能であることを識別させ、

前記ケイパビリティが利用可能になったときに通知されるリクエストを示すデータをケイパビリティサービスへ送信させる、請求項 15 に記載の非一時的コンピュータ可読記憶媒体。

【請求項 17】

前記オーケストレーションサービスに、前記第 2 のサービスの前記ケイパビリティが利

50

用不可能であることを識別させる前記命令を実行することは、さらに、前記オーケストレーションサービスに、前記第2のサービスの前記ケイパビリティが利用可能であることを示す通知が前記クラウドコンピューティングシステムの前記ケイパビリティサービスからまだ受信されていないことを決定させる、請求項16に記載の非一時的コンピュータ可読記憶媒体。

【請求項18】

前記プロビジョニングおよびデプロイメントマネージャにインクリメントに命令するために前記命令を実行することは、さらに、前記オーケストレーションサービスに、

前記ケイパビリティサービスから、前記ケイパビリティが利用可能であることを示す対応する通知を受信させ、

前記ケイパビリティが利用可能であるという前記通知を受信することに応答して、前記第2のサービスに対応するリソースを前記リージョン内でブートストラップするためのオペレーションを実行するように前記プロビジョニングおよびデプロイメントマネージャに命令させる、請求項15～請求項17のいずれかに記載の非一時的コンピュータ可読記憶媒体。

【請求項19】

前記複数のサービスをブートストラップするための前記対応するオペレーションを実行するように前記プロビジョニングおよびデプロイメントマネージャにインクリメントに命令するために前記命令を実行することは、前記プロビジョニングおよびデプロイメントマネージャに、少なくとも1つのインフラストラクチャコンポーネントをプロビジョニングさせ、前記少なくとも1つのインフラストラクチャコンポーネントへ1つまたは複数のアーティファクトをデプロイさせる、請求項15～請求項18のいずれかに記載の非一時的コンピュータ可読記憶媒体。

【請求項20】

前記1つまたは複数の依存は、前記複数の構成ファイルの1つまたは複数のパースを実行することに少なくとも部分的に基づいて識別され、前記命令を実行することは、さらに、前記オーケストレーションサービスに、前記1つまたは複数の依存および前記複数のサービスをブートストラップするためのオペレーションが実行される順序を識別することに少なくとも部分的に基づいて、1つまたは複数の有向グラフを生成させ、決定された前記順序に従って前記複数のサービスをブートストラップするための前記対応するオペレーションを実行するように前記プロビジョニングおよびデプロイメントマネージャにインクリメントに命令することは、前記1つまたは複数の有向グラフの1つまたは複数のトラバースに少なくとも部分的に基づく、請求項15～請求項19のいずれかに記載の非一時的コンピュータ可読記憶媒体。

【発明の詳細な説明】

【技術分野】

【0001】

関連出願の相互参照

本PCT出願は、「Techniques for Bootstrapping a Region Build」という名称の、2022年2月8日に出願された米国仮特許出願第63/308,003号、「Techniques for Implementing Virtual Data Centers」という名称の、2022年2月22日に出願された米国仮特許出願第63/312,814号、「Techniques for Building Data Centers in Cloud Regions」という名称の、2022年2月28日に出願された米国仮特許出願第63/315,005号、および「Techniques for Building Data Centers in Cloud Regions」という名称の、2022年12月6日に出願された米国特許出願第18/076,238号の優先権を主張し、それらの開示は、全ての目的のためにそれらの全体が参照により本明細書に組み込まれる。

【0002】

発明の分野

本開示は、クラウドコンピューティング環境にデータセンタを構築するための技術に関

10

20

30

40

50

する。より具体的には、本開示は、自動化された形式においてデータセンタ内に様々なクラウドコンピューティングコンポーネント（例えば、サービス）に対応するリソースをプロビジョニングおよびデプロイするための技術を説明する。

【背景技術】

【0003】

背景

今日、クラウドインフラストラクチャサービスは、データセンタを構築するために（例えば、特定の地理的リージョンのデータセンタに様々なリソースをブートストラップするために）多くの個々のサービスを利用する。いくつかの例において、リージョンは、1つまたは複数のデータセンタが配置された（またはされる）局所的な地理的エリアに対応する論理的抽象である。データセンタを構築することは、（例えば、様々なサービスのために）インフラストラクチャリソースをプロビジョニングおよび構成し、これらのリソースにコードをデプロイすることを含んでもよい。データセンタを構築するためのオペレーションは、集合的に「リージョン構築」を実行すると呼ばれることがある。あらゆる適切な数のデータセンタが、リージョンに含まれてもよく、したがって、リージョン構築は、多数のデータセンタを構築するためのオペレーションを含んでもよい。リージョンを構築するための従来のツールは、かなりの手作業の労力を必要とする。加えて、1つのサービスのためのブートストラップオペレーションは、まだ利用可能ではない場合があるリージョンの他の機能および／またはサービスに依存する場合がある。サービスチームおよびリージョンの数が増えるほど、プロビジョニングおよびデプロイをオーケストレーションするために実行されるタスクが劇的に増加する。サービスをブートストラップするためのおよび／またはリージョンを構築するための手作業の労力に実質的に依存することは、非常に時間がかかり、リスクを招き、十分なスケールにならない場合がある。

【発明の概要】

【発明が解決しようとする課題】

【0004】

簡単な概要

本開示の実施形態は、自動化されたリージョン構築を実行することに関する（例えば、リージョン（例えば、1つまたは複数のデータセンタに関連した地理的位置）内のあらゆる適切な数のサービスのためのリソース（例えば、インフラストラクチャコンポーネント、アーティファクトなど）をブートストラップ（例えば、プロビジョニングおよび／またはデプロイ）すること）。ブートストラップオペレーションは、これらのオペレーション間の依存の自動的な検出に少なくとも部分的に基づいてオーケストレーションサービス（例えば、マルチブロックオーケストレータ）によってコーディネートおよびオーケストレーションされることができる。マルチブロックオーケストレータは、さらに、構成ファイルおよび／またはソフトウェアアーティファクトの様々なバージョンを維持してもよく、それによってリージョン構築が実行される特定のバージョンセットを知的にかつ自動的に識別することができる。

【0005】

少なくとも1つの実施形態は、コンピュータが実行する方法に関する。方法は、クラウドコンピューティング環境のマルチブロックオーケストレータが、1つまたは複数のデータセンタに対応するリージョン内でブートストラップされる複数のサービスに対応する複数の構成ファイルを取得することを含んでもよい。いくつかの実施形態において、複数のサービスのそれぞれは、インフラストラクチャコンポーネントおよび対応するソフトウェアアーティファクトを含むリソースのそれぞれのセットに関連付けられている。方法は、マルチブロックオーケストレータが、複数の構成ファイルに少なくとも部分的に基づいて複数のサービスの間の1つまたは複数の依存を識別することをさらに含んでもよい。方法は、マルチブロックオーケストレータが、識別された1つまたは複数の依存に少なくとも部分的に基づいて、複数のサービスをブートストラップするためのオペレーションが実行される順序を決定することをさらに含んでもよい。方法は、マルチブロックオーケストレー

ータが、決定された順序に従って複数のサービスをブートストラップするための対応するオペレーションを実行するようにブートストラップコントローラにインクリメントに命令することをさらに含んでもよい。

【0006】

別の実施形態は、別のコンピュータが実行する方法に関する。方法は、クラウドコンピューティング環境のマルチブロックオーケストレータが、複数のサービスに関連した複数の構成ファイルの構成ファイルのそれぞれのセットを識別する複数のバージョンセットを維持することを含んでもよい。方法は、マルチブロックオーケストレータが、複数の構成ファイルから構成ファイルの第1のセットを識別する第1のバージョンセットを決定することをさらに含んでもよい。方法は、マルチブロックオーケストレータが、第1のバージョンセットによって識別された構成ファイルの第1のセットを検証するための検証過程を実行することをさらに含んでもよい。方法は、マルチブロックオーケストレータが、ブロック構成ファイルの第2のセットを識別する第2のバージョンセットを生成することをさらに含んでもよい。いくつかの実施形態において、構成ファイルの第2のセットは、検証過程に合格した構成ファイルを識別することに少なくとも部分的に基づいて構成ファイルの第1のセットから識別されてもよい。方法は、マルチブロックオーケストレータが、第2のバージョンセットによって識別された構成ファイルの第2のセットを利用するリージョン構築を実行することをさらに含んでもよい。

10

【0007】

別の実施形態は、1つまたは複数のプロセッサと、1つまたは複数のプロセッサによって実行されると、コンピューティングデバイスに、本明細書に開示された方法を実行させる命令と、を含むコンピューティングデバイスに関する。

20

【0008】

さらに別の実施形態は、コンピューティングクラスタの1つまたは複数のプロセッサによって実行されると、コンピューティングクラスタに、本明細書に開示された方法を実行させるコンピュータ実行可能命令を記憶する非一時的コンピュータ可読媒体に関する。

【0009】

あらゆる特定の要素または行為の議論を容易に識別するために、参照番号における最も重要な1つまたは複数の数字は、その要素が最初に紹介される図面の番号を表す。

【図面の簡単な説明】

30

【0010】

【図1】少なくとも1つの実施形態による、クラウドインフラストラクチャオーケストレーションサービス(CIOS)102が、リージョンにおいてブートストラップサービスをダイナミックに提供するために動作し得る環境のブロック図である。

【図2】少なくとも1つの実施形態による、仮想ブートストラップ環境(ViBE)を構築するための環境および方法を示すためのブロック図である。

【図3】少なくとも1つの実施形態による、ViBEを利用するターゲットリージョンにサービスをブートストラップするための環境および方法を示すためのブロック図である。

【図4】少なくとも1つの実施形態による、どの構成ファイルバージョンおよび対応するアーティファクトが、ターゲットリージョンにサービスをブートストラップするために利用されるかを識別する多数のバージョンセットを維持するための例示的な方法を示すブロック図である。

40

【図5】少なくとも1つの実施形態による、リージョンに多数のサービスをプロビジョニングおよびデプロイするためのオペレーションを実行するための例示的なフローを示すブロック図である。

【図6】少なくとも1つの実施形態による、リージョン構築に関連付けられたサービスに関連する情報を表す例示的なインターフェースを示すブロック図である。

【図7】少なくとも1つの実施形態による、リージョン構築に関連付けられたイベントに関連する情報を表す例示的なインターフェースを示すブロック図である。

【図8】少なくとも1つの実施形態による、リージョン構築を実行するための(例えば、

50

ブートストラップオペレーションをインクリメントに実行することによってリージョンを修正するための) 例示的な方法を示す図である。

【図 9】少なくとも 1 つの実施形態による、リージョン構築を実行するためにバージョンセットを利用するための例示的な方法を示す図である。

【図 10】少なくとも 1 つの実施形態による、サービスシステムとしてのクラウドインフラストラクチャを実施するための 1 つのパターンを示すブロック図である。

【図 11】少なくとも 1 つの実施形態による、サービスシステムとしてのクラウドインフラストラクチャを実施するための別のパターンを示すブロック図である。

【図 12】少なくとも 1 つの実施形態による、サービスシステムとしてのクラウドインフラストラクチャを実施するための別のパターンを示すブロック図である。

【図 13】少なくとも 1 つの実施形態による、サービスシステムとしてのクラウドインフラストラクチャを実施するための別のパターンを示すブロック図である。

【図 14】少なくとも 1 つの実施形態による、例示的なコンピュータシステムを示すブロック図である。

【発明を実施するための形態】

【0011】

詳細な説明

以下の記述において、説明の目的で、ある実施形態の完全な理解を提供するために具体的な詳細が示される。しかしながら、これらの具体的な詳細なく様々な実施形態が実施され得ることが明らかになるであろう。図面および記述は、制限的であることは意図されていない。「例示的」という用語は、「例、実例または例示として働く」ことを意味するために本明細書において使用される。「例示的」なものとして本明細書に記述されたあらゆる実施形態または設計は、必ずしも他の実施形態または設計よりも好ましいまたは有利であると解釈されるべきではない。

【0012】

例示的な自動化されたデータセンタ構築（リージョン構築）インフラストラクチャ

クラウドサービスの採用は、近年、急速な増加を示している。今では、様々なタイプのクラウドサービスが、様々な異なるクラウドサービスプロバイダ（CSP）によって提供されている。クラウドサービスという用語は、一般的に、CSPによって提供されるシステムおよびインフラストラクチャ（クラウドインフラストラクチャ）を使用してオンデマンドで（例えば、サブスクリプションモデルを介して）ユーザまたはカスタマーにCSPによって利用可能にされるサービスまたは機能を指すために用いられる。典型的には、カスタマーにクラウドサービスを提供するために使用される、CSPのインフラストラクチャを構成するサーバおよびシステムは、カスタマーの自己の自社運用のサーバおよびシステムとは別のものである。したがって、カスタマーは、サービスのための別のハードウェアおよびソフトウェアリソースを購入する必要なく、CSPによって提供されるクラウドサービスを利用することができる。クラウドサービスは、カスタマーが、サービスまたは機能を提供するために使用されるインフラストラクチャを調達することに投資する必要なく、アプリケーションおよびコンピューティングリソースへの容易で、スケーラブルかつオンデマンドのアクセスを、サブスクライブするカスタマーに提供するように設計されている。サービスとしてのソフトウェア（SaaS）、サービスとしてのプラットフォーム（PaaS）、サービスとしてのインフラストラクチャ（IaaS）、およびその他のものなどの、クラウドサービスの様々な異なるタイプまたはモデルが提供され得る。カスタマーは、CSPによって提供される1つまたは複数のクラウドサービスをサブスクライブすることができる。カスタマーは、個人、組織、企業等のあらゆるエンティティであることができる。

【0013】

上記で示したように、CSPは、サブスクライブするカスタマーにクラウドサービスを提供するために使用されるインフラストラクチャおよびリソースを提供する責任を負う。CSPによって提供されるリソースは、ハードウェアリソースおよびソフトウェアリソ

10

20

30

40

50

スの両方を含むことができる。これらのリソースは、例えば、計算リソース（例えば、仮想マシン、コンテナ、アプリケーション、プロセッサ）、メモリリソース（例えば、データベース、データストア）、ネットワークリソース（例えば、ルータ、ホストマシン、ロードバランサ）、アイデンティティ、およびその他のリソースを含むことができる。ある実装形態では、クラウドサービスCSPのセットを提供するためにCSPによって提供されるリソースは、データセンタに組織化されている。データセンタは、クラウドサービスの特定のセットを提供するように構成されてもよい。CSPは、クラウドサービスの特定のセットを提供するために使用されるインフラストラクチャおよびリソースをデータセンタに備え付ける責任を負う。CSPは、1つまたは複数のデータセンタを構築してもよい。

10

【0014】

CSPによって提供されるデータセンタは、異なるリージョンにおいてホストされてもよい。リージョンは、局所的な地理的エリアであり、リージョン名によって識別されてもよい。リージョンは、一般的に、互いに独立しており、国またはさらには大陸を横断するなど、大きな距離によって分離されることができる。リージョンは、レルムにグループ化される。CSPのためのリージョンの例は、US西部、US東部、オーストラリア東部、オーストラリア南東部などを含んでもよい。

【0015】

リージョンは、1つまたは複数のデータセンタを含むことができ、データセンタは、リージョンに対応するある地理的エリア内に配置されている。一例として、リージョンにおけるデータセンタは、そのリージョン内の都市に配置されてもよい。例えば、特定のCSPのために、US西部リージョンにおけるデータセンタは、カリフォルニア州サンホセに配置されてもよく、US東部リージョンにおけるデータセンタは、バージニア州アッシュバーンに配置されてもよく、オーストラリア東部リージョンにおけるデータセンタは、オーストラリアのシドニーに配置されてもよく、オーストラリア南東部リージョンにおけるデータセンタは、オーストラリアのメルボルンに配置されてもよい、などである。

20

【0016】

リージョン内のデータセンタは、1つまたは複数のアベイラビリティドメインに組織化されてもよく、アベイラビリティドメインは、高いアベイラビリティおよび災害復旧目的のために使用される。アベイラビリティドメインは、リージョン内の1つまたは複数のデータセンタを含むことができる。リージョン内のアベイラビリティドメインは、互いに隔離されており、フォールトトレラントであり、多数のアベイラビリティドメインにおけるデータセンタが同時に故障する可能性が非常になくなるように設計されている。例えば、リージョン内のアベイラビリティドメインは、リージョン内の1つのアベイラビリティドメインにおける故障が、同じリージョン内の他のアベイラビリティドメインにおけるデータセンタのアベイラビリティに影響する可能性が低くなるように構築されてもよい。

30

【0017】

カスタマーまたはサブスクライバが、CSPによって提供される1つまたは複数のサービスをサブスクライブするまたは1つまたは複数のサービスと契約すると、CSPは、カスタマーのためのテナンシを生成する。テナンシは、カスタマーのために生成されるアカウントのようなものである。ある実装形態において、カスタマーのためのテナンシは、1つのレルムに存在し、そのレルムに属する全てのリージョンにアクセスすることができる。したがって、カスタマーのユーザは、このテナンシの下でカスタマーによってサブスクライブされるサービスにアクセスすることができる。

40

【0018】

上記に示したように、CSPは、カスタマーにクラウドサービスを提供するためにデータセンタを構築またはデプロイする。CSPのカスタマーベースが成長するにつれて、CSPは、一般的に、カスタマーの増大する需要にサービスを提供しかつカスタマーにより良いサービスを提供するために、新たなリージョンに新たなデータセンタを構築するかまたは既存のデータセンタのキャパシティを増大する。好ましくは、データセンタは、その

50

データセンタによってサービスを受けるカスタマーのロケーションの地理的近傍に構築される。データセンタと、データセンタによってサービスを受けるカスタマーとの間の地理的近接は、リソースのより効率的な利用およびより迅速かつより信頼できるサービスがカスタマーに提供されることにつながる。したがって、C S P は、一般的に、データセンタによってサービスを受けるカスタマーに地理的に近接する地理的エリアにおける新たなリージョンに新たなデータセンタを構築する。例えば、ドイツにおける成長するカスタマーベースのために、C S P は、ドイツにおける新たなリージョンに1つまたは複数のデータセンタを構築してもよい。

【0019】

リージョンにおいてデータセンタ（または多数のデータセンタ）を構築することは、時には、リージョンを構築するとも呼ばれる。「リージョン構築」という用語は、リージョンにおいて1つまたは複数のデータセンタを構築することを指すために使用される。リージョンにおいてデータセンタを構築することは、そのデータセンタが提供するように構成されているサービスのセットを提供するために必要とされるまたは使用される新たなリソースのセットをプロビジョニングまたは生成することを必要とする。リージョン構築プロセスの最終結果は、リージョンにおけるデータセンタの生成であり、データセンタは、そのデータセンタのために意図されたサービスのセットを提供することができ、サービスのセットを提供するために使用されるリソースのセットを含む。

【0020】

リージョンにおいて新たなデータセンタを構築することは、様々なブートストラップアクティビティ間の広範囲の調整を必要とする非常に複雑なアクティビティである。ハイレベルにおいて、これは、データセンタによって提供されるサービスのセットを識別する、サービスのセットを提供するために必要とされる様々なリソースを識別する、識別されたリソースを生成、プロビジョニング、およびデプロイする、意図した形式で使用する、ことができるようにリソースを適切にワイヤリングする、などの様々なタスクの実行および調整を必要とする。これらのタスクのそれぞれは、調整される必要があるサブタスクをさらに有し、複雑性をさらに高める。この複雑性により、現在、リージョンにおけるデータセンタの構築は、注意深い手動調整を必要とするいくつかの手動で開始されるまたは手動でコントロールされるタスクを必要とする。その結果、新たなリージョンを構築する（すなわち、リージョンに1つまたは複数のデータセンタを構築する）タスクは、非常に時間がかかるものである。データセンタを構築するために、時間、例えば数カ月かかる可能性がある。加えて、プロセスは、非常にエラーを起こしやすく、時には、データセンタの所望の構成が達成される前に何回かの反復を必要とし、これは、データセンタを構築するためにかかる時間をさらに長くする。これらの制限および問題は、増大するカスタマーニーズに回答してタイムリーな形式でコンピューティングリソースを成長させるC S Pのケイパビリティを著しく制限する。

【0021】

技術的效果

本開示は、構築時間を短縮し、コンピューティングリソースの無駄を削減し、リージョンに1つまたは複数のデータセンタを構築することに関連するリスクを低減するための技術を記述する。過去にリージョンにおいてデータセンタを構築するために数週間および数カ月が必要とされた代わりに、本明細書に記載された技術は、従来のアプローチよりもエラーのリスクを減じながら、比較的大幅に短い時間でリージョンに新たなデータセンタを構築するために使用することができる。

【0022】

データセンタに対する所与の変更を実施するためのリソース（例えば、インフラストラクチャコンポーネントおよびデプロイされるソフトウェア）を識別する所定の構成ファイルに基づいてサービスを新たなデータセンタにブートストラップ（例えば、プロビジョニングおよびデプロイ）するように構成されたクラウドインフラストラクチャオーケストレーションサービス（C I O S）が本明細書に開示される。C I O Sは、リソース、実行タ

ーゲット、フェーズ、およびブロックの間の依存を識別するために構成ファイル（例えば、ブロック設定）をパースおよび分析することができる。C I O Sは、分析から特定のデータ構造を生成してもよく、これらのデータ構造を用いて、オペレーションをドライブし、サービスがリージョンにブートストラップされる順序を管理してもよい。C I O Sは、これらのデータ構造を利用して、いつC I O Sがサービスをブートストラップすることができるか、いつブートストラップがブロックされるかおよび／または前にブロックされたサービスに関連したブートストラップオペレーションをいつ再開することができるかを識別してもよい。有利には、C I O Sは、データ構造内の循環依存を識別し、タスク実行の前にこれらの循環依存を排除／解消するためにオペレーションを実行することができる。これらの技術を用いて、C I O Sは、それらのタスクが依存するリソースのアベイラビリティの前にタスクを実行するリスクを実質的に提言する。

10

【0023】

本明細書に開示された技術を利用して、C I O Sは、それらのタスクが依存する機能がリージョンにおいて利用可能になるまでタスクが開始されないことを保証しながらデータセンタに対する変更を実行するために並列処理を最適化し得る。これにより、C I O Sは、リージョン構築がより効率的に実行されることを可能にし、これは、データセンタを構築するために必要とされる時間および従来のアプローチに見られる無駄な計算リソース使用を大幅に減じる。

【0024】

マルチブロックオーケストレータ（M F O）が本明細書に開示される。M F Oは、テストおよび／またはリージョン構築（例えば、リージョンのデータセンタへのプロビジョニングおよびデプロイ）のために利用される構成ファイルの異なるセットを識別する多数のバージョンセットを利用するように構成されてもよい。特定のバージョンセットは、リージョンのデータセンタを構築するために用いられることができる。そのバージョンセットは、そこからリージョンにおけるブートストラップタスク（例えば、プロビジョニングおよびデプロイタスク）が決定され得る構成ファイルの特定のセットを識別することができる。M F Oは、構成ファイルの静的解析を行うことによって、サービス間の依存を識別し、リージョン構築の前に循環依存を識別および解消してもよい。M F Oは、それらのタスクが依存するリソースが利用可能になるまで、依存するブートストラップタスクが開始されないことを保証しながら、ブートストラップタスクを実行するようにプロビジョニングおよびデプロイメントマネジャ（例えば、図1～図3において検討されるC I O Sセントラル）にインクリメントに命令してもよい。リージョンの様々なケイパビリティが利用可能になるので、M F Oは、実行される後続のブートストラップタスクを識別および実施してもよく、構築プロセスを完了までインクリメントに駆動する。開示された技術は、それらのファイルがリージョン構築のために利用される前に単体および／または統合テストが構成ファイルとともに実行されることを可能にし、これは、リージョン構築が成功する可能性を高める。開示された技術を用いて、M F Oは、従来のシステムにおいて必要とされるエラーのリスクおよび時間を減じながら、自動化されたリージョン構築が実行されることを可能にする。

20

30

【0025】

定義

「リージョン」は、地理的位置に対応する論理的抽象である。リージョンは、あらゆる適切な数の1つまたは複数の実行ターゲットを含むことができる。いくつかの実施形態において、実行ターゲットは、データセンタに対応することができる。

40

【0026】

「実行ターゲット」は、リリースを実行するための変化の最小単位を意味する。「リリース」は、サービスに対する特定の変更をオーケストレーションする意図の表現を意味する（例えば、バージョン8をデプロイする、「内部DNSレコードを加える」など）。ほとんどのサービスにとって、実行ターゲットは、サービスの「インスタンス」を表す。1つのサービスは、1つまたは複数の実行ターゲットのそれぞれにブートストラップされる

50

ことができる。実行ターゲットは、デバイスのセット（例えば、データセンタ）と関連付けられてもよい。

【0027】

「ブートストラップ」は、1つのサービスに対応するあらゆる適切な数のリソース（例えば、インフラストラクチャコンポーネント、アーティファクトなど）のプロビジョニングおよびデプロイに関連付けられた集合的タスクを意味することが意図されている。

【0028】

「サービス」は、リソースのセットによって提供される機能を意味する。サービスのためのリソースのセットは、サービスの機能を提供するように構成されることができるクラウドプロバイダによってホストされる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（例えば、アプリケーション）のあらゆる適切な組合せを含む。サービスは、インターネットを通じてユーザに利用可能にすることができる。

【0029】

「アーティファクト」は、インフラストラクチャコンポーネントまたはKubernetes エンジンクラスタにデプロイされるコードを意味し、これは、ソフトウェア（例えば、アプリケーション）、インフラストラクチャコンポーネントのための構成情報（例えば、構成ファイル）などを含んでもよい。

【0030】

「フロック設定」は、1つのサービスに関連付けられた全てのリソース（例えば、インフラストラクチャコンポーネントおよびアーティファクト）のセットを記述する構成ファイル（または構成ファイルのセット）を意味する。フロック設定は、サービスのリソースの所望の状態に対応する1つまたは複数の態様を指定する宣言文を含んでもよい。

【0031】

「サービス状態」は、サービスと関連付けられた全てのリソース（例えば、インフラストラクチャリソース、アーティファクトなど）の現時点のスナップショットを意味する。サービス状態は、サービスリソースに関連付けられたプロビジョニングおよび／またはデプロイメントタスクに対応する状態を示す。

【0032】

IaaSプロビジョニング（または「プロビジョニング」）は、使用のためにコンピュータまたは仮想ホストを獲得すること、さらには必要なライブラリまたはサービスをそれらにインストールすることを意味する。「デバイスをプロビジョニングする」という表現は、それらの特定の使用のためのエンドユーザによって利用されることができる状態にデバイスを発展させることを意味する。プロビジョニングプロセスを受けたデバイスは、「プロビジョニングされたデバイス」と呼ばれてもよい。プロビジョニングされたデバイスを準備すること（ライブラリおよびデーモンをインストールすること）は、プロビジョニングの一部であり得る。この準備は、準備されたデバイスに新たなアプリケーションまたは新たなバージョンのアプリケーションをデプロイすることとは異なる。ほとんどのケースでは、デプロイメントはプロビジョニングを含まず、プロビジョニングは最初に行われる必要がある場合がある。準備されると、装置は、「インフラストラクチャコンポーネント」と呼ばれてもよい。

【0033】

IaaSデプロイメント（または「デプロイメント」）は、プロビジョニングされたインフラストラクチャコンポーネントに新たなアプリケーションまたは新たなバージョンの炙りケーションを提供するおよび／またはインストールするプロセスを意味する。インフラストラクチャコンポーネントがプロビジョニング（例えば、獲得、アサイン、準備など）されると、追加のソフトウェアがデプロイ（例えば、インフラストラクチャコンポーネントに提供およびインストール）されてもよい。インフラストラクチャコンポーネントは、プロビジョニングおよびデプロイメントが完了した後、「リソース」と呼ぶことができる。リソースの例は、仮想マシン、データベース、オブジェクトストレージ、ブロック記憶域、ロードバランサなどを含み得るが、それらに限定されない。

10

20

30

40

50

【0034】

「ケイパビリティ」は、サービスと関連付けられた機能のユニットを識別する。ユニットは、サービスによって提供される機能の一部または全部であることができる。例えば、リソースが許可／認証処理（例えば、リソースによって提供される機能のサブセット）のために利用可能であることを示すケイパビリティをパブリッシュすることができる。別の例として、サービスの完全な機能が利用可能であることを示すケイパビリティをパブリッシュすることができる。ケイパビリティは、リソースまたはサービスが依存する機能および／または使用のために利用可能なリソースまたはサービスの機能を識別するために使用することができる。

【0035】

「仮想ブートストラップ環境」（V i B E）は、既存のリージョン（例えば、「ホストリージョン」）のオーバーレイにおいてプロビジョニングされる仮想クラウドネットワークを意味する。プロビジョニングされると、V i B Eは、通信チャネル（例えば、I P s e c トンネルV P N）を使用して新たなリージョンに接続される。デプロイメントオーケストレータ、パブリックキーインフラストラクチャ（P K I）サービスなどの、ある必須のコアサービス（または「シード」サービス）をV i B Eにおいてプロビジョニングすることができる。これらのサービスは、ハードウェアをオンラインにもたらし、新たなリージョンへのトラストのチェーンを確立し、新たなリージョンにおいて残りのサービスをデプロイするために必要とされるケイパビリティを提供することができる。仮想ブートストラップ環境を利用することは、ホストリージョンのリソースを利用することによってブートストラップリソースの間の循環依存を防止することができる。サービスは、物理的リージョン（例えば、ターゲットリージョン）が利用可能になる前にV i B Eにおいてステージングおよびテストされることができる。

【0036】

「クラウドインフラストラクチャオーケストレーションサービス」（C I O S）は、リージョン構築の一部としてあらゆる適切な数のサービスのためのプロビジョニングおよびデプロイメントオペレーションを管理するように構成されたシステムを意味し得る。

【0037】

マルチブロックオーケストレータ（M F O）は、ターゲットリージョン（例えば、新たなリージョン）にサービスをプロビジョニングおよびデプロイするためにC I O Sのコンポーネント間でイベントを調整するコンピューティングコンポーネント（例えば、サービス）であり得る。M F Oは、リージョン構築の各々のサービスのための関連するイベントを追跡し、これらのイベントに応答して動作を行う。

【0038】

「ホストリージョン」は、仮想ブートストラップ環境（V i B E）をホストするリージョンを意味する。ホストリージョンは、V i B Eをブートストラップするために使用され得る。

【0039】

「ターゲットリージョン」は、構築中のリージョンを意味する。

「ケイパビリティをパブリッシュする」とは、特定のケイパビリティが利用可能である（または利用不可能である）ことの指示を提供する「パブリッシャーサブスクリバ」コンピューティング設計またはその他のものにおいて使用される場合の「パブリッシング」を意味する。ケイパビリティは、リソース／サービスの機能が利用可能であるという指示を提供するために「パブリッシュ」される（例えば、ケイパビリティサービスによって収集される、ケイパビリティサービスへ提供される、プッシュされる、プルされるなど）。いくつかの実施形態において、ケイパビリティは、イベント、通知、データ伝送、ファンクションコール、A P I呼び出しなどを介してパブリッシュ／送信され得る。特定のケイパビリティのアベイラビリティを示すイベント（またはその他の通知／データ伝送など）は、ケイパビリティサービスへブロードキャスト／アドレス（例えば、パブリッシュ）されることができる。

10

20

30

40

50

【0040】

「ケイパビリティサービス」は、異なるブロック間の依存をモデル化するように構成されたブロックであり得る。ケイパビリティサービスは、クラウドインフラストラクチャオーケストレーションサービス内に提供されてもよく、どのケイパビリティ、サービス、フィチャがリージョンにおいて利用可能になっているかを定義し得る。

【0041】

「リアルタイムリージョナルデータディストリビュータ」(RRDD)は、リージョンデータを管理するように構成されたサービスまたはシステムであり得る。このリージョンデータは、新たなリージョンのための実行ターゲットをダイナミックに生成するためにブロック設定へ注入されることができる。

【0042】

いくつかの例において、クラウドインフラストラクチャオーケストレーションサービス(CIOS)を実施するための技術が本明細書に記載される。このような技術は、上記で簡単に記述されているように、クラウド環境(例えば、リージョン)内にインフラストラクチャコンポーネントをブートストラップすること(例えば、インフラストラクチャコンポーネントにソフトウェアをプロビジョニングおよびデプロイすること)を管理するように構成されることができる。いくつかの例において、CIOSは、与えられたサービスのためのブートストラップタスク(プロビジョニングおよびデプロイメント)を管理するように構成され得るコンピューティングコンポーネント(例えば、CIOSセントラルおよびCIOSリージョナルであり、それらは両方とも後でさらに詳細に説明される)およびリージョン構築(例えば、多数のサービスに対応するブートストラップオペレーション)を開始/管理するように構成されたマルチブロックオーケストレータ(これも後でさらに詳細に説明される)を含むことができる。

【0043】

CIOSは、サービスチームからの最小限の手動ランタイム努力で(例えば、いくつかの例において、ハードウェアの初期承認および/または物理的輸送を超えて)リージョン構築およびワールドワイドのインフラストラクチャプロビジョニングおよびコードデプロイメントを可能にする。CIOSのハイレベルな責任は、リージョン構築を調整すること、ユーザに(例えば、リージョンの、リージョンを横断する、ワールドワイドの、など)CIOSによって管理されるリソースの現在の状態のビューを提供すること、およびリージョン内にリソースをブートストラップするためのブートストラップオペレーションを管理することを含むが、これらに限定されない。

【0044】

CIOSは、ビューリコンシリエーションを提供してもよく、そこでは、リソースの所望の状態(例えば、所望の構成)のビューが、リソースの現在の/実際の状態(例えば、現在の構成)によってリコンサイルされ得る。いくつかの例において、ビューリコンシリエーションは、どのリソースが実際に動作しているかおよびそれらの現在の構成および/または状態を識別するために状態データを取得することを含み得る。リコンシリエーションは、サービスレベルにおけるなど、様々なグラニュラリティにおいて実行されることができる。

【0045】

CIOSは、プラン生成を実行することができ、そこでは、リソースの所望の状態と現在の状態との差が識別される。プラン生成の部分は、リソースを現在の状態から所望の状態へもたらすために実行される必要があるオペレーションを識別することを含むことができる。いくつかの例において、CIOSは、生成されたプラントを承認のためにユーザに提示することができる。これらの例において、CIOSは、ユーザからのユーザ入力に基づいてプランを承認または拒絶されたものとしてマークすることができる。したがって、ユーザは、プランについて判断することにより少ない時間を費やすことができ、プランは、機械生成されるため、より正確である。プランは、人間の消費のためにはほとんど詳細すぎる。しかしながら、CIOSは、このデータを、非常に高度なユーザインターフェー

10

20

30

40

50

ス（UI）を介して提供することができる。

【0046】

いくつかの例において、CIOSは、承認されたプランを実行することによってチェンジマネジメントの実行を処理することができる。実行プランが生成および承認されると、エンジニアは、CIOSがロールバックを開始するまでもはやチェンジマネジメントに関与する必要がない場合がある。CIOSは、（例えば、CIOSが、実行しながらサービスヘルス低下を検出したとき）サービスを前の（例えば、リリース前の）状態に戻すプランを生成することによって前のサービスバージョンへのロールバックを処理することができる。

【0047】

CIOSは、アラームをモニタリングしかつ統合テストを実行することによってサービスヘルスを計測することができる。CIOSは、サービス低下のイベント時にチームが迅速にロールバック挙動を定義することを助けることができ、後でロールバック挙動を実行することができる。CIOSは、プランを生成および表示することができ、承認を追跡することができる。CIOSは、リージョン構築を横断してこれらのタスクを調整する1つのシステムにおいてプロビジョニングおよびデプロイメントの機能を組み合わせることができる。CIOSは、フロック（例えば、あらゆる適切な数のサービスに対応するフロック設定などのサービスリソース）、アーティファクト、リソース、および依存の発見もサポートする。CIOSは、1つまたは複数の構成ファイルの静的解析（例えば、コンテンツをパースおよび処理することを含む）によって全てのレベル（例えば、リソースレベル、実行ターゲットレベル、フェーズレベル、サービスレベルなど）において実行タスクの間の依存を発見することができる。これらの依存を用いて、CIOSは、これらの依存から様々なデータ構造を生成することができ、これらのデータ構造は、タスク実行（例えば、リージョンを横断するインフラストラクチャリソースのプロビジョニングおよびアーティファクトのデプロイメントに関するタスク）を推進するために使用されることができる。

【0048】

図1は、少なくとも1つの実施形態による、クラウドインフラストラクチャオーケストレーションサービス（CIOS）が、リージョンにおいてブートストラップサービスをダイナミックに提供するために動作し得る環境100のブロック図である。CIOS102は、以下のコンポーネント、すなわちリアルタイムリージョナルデータディストリビュータ（RRDD）104、マルチフロックオーケストレータ（MFO）106、CIOSセントラル108、CIOSリージョナル110、およびケイパビリティサービス112を含むことができるが、これらに限定されない。CIOSセントラル108およびCIOSリージョナル110の特定の機能は、その内容全体が全ての目的のためにその全体が組み込まれる、「Techniques for Deploying Infrastructure Resources with a Declarative Provisioning Tool」という名称の米国特許出願第17／016,754号により詳細に提供されている。いくつかの実施形態において、CIOS102のコンポーネントのあらゆる適切な組合せがサービスとして提供されてもよい。いくつかの実施形態において、CIOS102のある部分は、リージョン（例えば、ホストリージョン103によって表されるデータセンタ）にデプロイされてもよい。いくつかの実施形態において、CIOS102は、米国特許出願第17／016,754号にさらに詳細におよび以下で図2および図3に関して検討されるあらゆる適切な数のクラウドサービス（図1には示されていない）を含んでもよい。

【0049】

リアルタイムリージョナルデータディストリビュータ（RRDD）104は、レム、リージョン、実行ターゲット、およびアベイラビリティドメインを識別するリージョンデータを維持および提供するように構成されてもよい。いくつかのケースにおいて、リージョンデータは、あらゆる適切な形式（例えば、JSONフォーマット、データオブジェクト／コンテナ、XMLなど）であってもよい。RRDD104によって維持されるリージ

10

20

30

40

50

ョンデータは、対応する識別子によって個々に参照可能であることができるあらゆる適切な数のデータのサブセットを含んでもよい。例えば、「all_regions」という識別子は、全ての定義されたリージョンのためのメタデータを含むデータ構造（例えば、リスト、構造、オブジェクトなど）に関連付けられることができる。別の例として、「realm」などの識別子は、多数のレルムおよび各レルムに対応するリージョンのセットのためのメタデータを識別するデータ構造に関連付けられることができる。一般的に、リージョンデータは、識別子、DNS拡張子、状態（例えば、リージョンの状態）などの、1つまたは複数のレルム、リージョン、アベイラビリティドメイン（AD）、実行ターゲット（ET）などのあらゆる適切な属性を維持してもよい。RRDD104は、リージョン状態をリージョンデータの一部として管理するように構成されてもよい。リージョン状態は、リージョン内のブートストラップの状態を示すあらゆる適切な情報を含んでもよい。例えば、いくつかの例示的なリージョン状態は、「初期」、「構築」、「生産」、「休止」または「廃止予定」を含むことができる。「初期」状態は、まだブートストラップされていないリージョンを示してもよい。「構築」状態は、リージョン内の1つまたは複数のブロックのブートストラップが開始したことを示してもよい。「生産」状態は、ブートストラップが完了し、リージョンが検証のために準備されていることを示してもよい。「休止」状態は、CIOSセントラル108またはCIOSリージョナル110が、恐らく運用問題により、リージョナルスタックとの内部相互作用を休止していることを示してもよい。「廃止予定」状態は、リージョンが廃止予定であり、利用できなくなる可能性が高いおよび／または再びコンタクトされないことを示してもよい。

10

20

【0050】

CIOSセントラル108は、それによってユーザ（例えば、ユーザ109）がCIOS102と相互作用し得るあらゆる適切な数のユーザインターフェースを提供するように構成されている。例えば、ユーザは、CIOSセントラル108によって提供されるユーザインターフェースを介してリージョンデータに変更を加えることができる。CIOSセントラル108は、加えて、ユーザが、ブロック設定および／またはアーティファクトに加えられた変更を見る、プランを生成しかつ見る、プランを承認／拒絶する、プラン実行（例えば、CIOS102によって管理されるあらゆる適切な数のリソースのインフラストラクチャプロビジョニング、デプロイメント、リージョン構築、および／または所望の状態に対応する）におけるステータスを見ることを可能にする様々なインターフェースを提供してもよい。CIOSセントラル108は、あらゆる適切な数のCIOSリージョナル110インスタンスを管理するように構成された制御プレーンを実施してもよい。CIOSセントラル108は、リージョンデータを表すための1つまたは複数のユーザインターフェースを提供することができ、ユーザ109がリージョンデータを見るおよび／または変更することを可能にする。CIOSセントラル108は、あらゆる適切な数のインターフェースを介してRRDD104の機能呼び出すように構成されることができる。一般的に、CIOSセントラル108は、直接的または間接的に（例えば、RRDD104を介して）、リージョンデータを管理するように構成されてもよい。CIOSセントラル108は、ブロック設定内の変数としてリージョンデータを注入するためにブロック設定をコンパイルするように構成されてもよい。

30

40

【0051】

CIOSリージョナル110の各インスタンスは、リージョンの1つのサービスに関連付けられたブートストラップタスクを実行するように構成されたモジュールに対応してもよい。CIOSリージョナル110は、CIOSセントラル108から所望の状態データを受信することができる。いくつかの実施形態において、所望の状態データは、サービスに関連付けられたリソースの所望の状態を（例えば、宣言文を介して）宣言するブロック設定を含んでもよい。CIOSセントラル108は、サービスに関連付けられたリソースの現在の状態のあらゆる適切な態様を示す現在の状態データを維持することができる。いくつかの実施形態において、CIOSリージョナル110は、所望の状態データと現在の状態データとの比較によって、1つまたは複数のリソースに対して変更が必要であること

50

を識別することができる。例えば、C I O Sリージョナル110は、1つまたは複数のインフラストラクチャコンポーネントがプロビジョニングされる必要がある、1つまたは複数のアーティファクトがデプロイされる、またはそれらのリソースの状態を所望の状態と一致させるためにサービスのリソースにあらゆる適切な変更が必要とされることを決定することができる。C I O Sリージョナル110がブートストラップオペレーションを実行するので、C I O Sリージョナル110は、それらが利用可能になるときのリソースの様々なケイパビリティを示すデータを発行してもよい。「ケイパビリティ」は、サービスと関連付けられた機能のユニットを識別する。ユニットは、サービスによって提供される機能の一部または全部であることができる。例えば、リソースが許可／認証プロセッシング（例えば、リソースによって提供される機能のサブセット）のために利用可能であることを示すケイパビリティが発行されることができる。別の例として、サービスの完全な機能が利用可能であることを示すケイパビリティが発行されることができる。ケイパビリティは、リソースまたはサービスが依存する機能および／または使用のために利用可能なリソースまたはサービスの機能を識別するために使用されることができる。

10

20

30

40

50

【0052】

ケイパビリティサービス112は、1) 様々なサービスのどのケイパビリティが現在利用可能であるか、2) あらゆるリソース／サービスが特定のケイパビリティにおいて待機しているかどうか、3) どの特定のリソースおよび／またはサービスが、与えられたケイパビリティにおいて待機しているか、または上記のあらゆる適切な組合せを示すケイパビリティデータを維持するように構成されている。ケイパビリティサービス112は、それによってケイパビリティサービスがリクエストされ得るインターフェースを提供してもよい。ケイパビリティサービス112は、ケイパビリティデータをM F O 1 0 6および／またはC I O Sリージョナル110（例えば、C I O Sリージョナル110の各インスタンス）に送信することを可能にする1つまたは複数のインターフェース（例えば、アプリケーションプログラミングインターフェース）を提供してもよい。いくつかの実施形態において、M F O 1 0 6および／またはC I O Sリージョナル110のあらゆる適切なコンポーネントまたはモジュールは、ケイパビリティサービス112からケイパビリティデータをリクエストするように構成されてもよい。

【0053】

いくつかの実施形態において、マルチブロックオーケストレータ（M F O）106は、リージョン構築努力を推進するように構成されてもよい。いくつかの実施形態において、M F O 1 0 6は、リージョン内の与えられたサービスをブートストラップするために（またはターゲットリージョンに変更のユニットを行うために）どのブロック／ブロック設定バージョンおよび／またはアーティファクトバージョンが利用されるかを記述する情報を管理することができる。いくつかの実施形態において、M F O 1 0 6は、リアルタイムリージョナルデータディストリビュータ104によって管理されるリージョンデータへの変更をモニタする（またはさもなければ通知される）ように構成されてもよい。いくつかの実施形態において、リージョンデータが変更されたことの指示を受信すると、リージョン構築がM F O 1 0 6によってトリガされてもよい。いくつかの実施形態において、M F O 1 0 6は、リージョン構築のために用いられる様々なブロック設定およびアーティファクトを収集してもよい。ブロック設定のいくつかまたは全ては、リージョンに依存しないように構成されてもよい。すなわち、ブロック設定は、どのリージョンにブロックがブートストラップされるかを明示的に識別しなくてもよい。いくつかの実施形態において、M F O 1 0 6は、データ注入プロセスをトリガしてもよく、このデータ注入プロセスを通じて、収集されたブロック設定が（例えば、C I O Sセントラル108によって）再コンパイルされる。再コンパイルの間、オペレーションは、リアルタイムリージョナルデータディストリビュータ104によって維持されるリージョンデータを設定ファイルへ注入させるために（例えば、C I O Sセントラル108によって）実行されてもよい。ブロック設定は、リージョンデータのハードコードされた識別を要求することなく変数／パラメータを通じてリージョンデータを参照することができる。ブロック設定は、リージョンデータが

ハードコードされる、したがって、変更することをより困難にするのではなく、このデータ注入を用いてランタイムにおいてダイナミックに修正されることができる。

【0054】

マルチフロックスオーケストレータ106は、静的フロックス分析を実行することができ、ここで、リソース、実行ターゲット、フェーズ、およびフロックスの間の依存を識別し、特に、除去される必要がある循環依存を識別するために、フロックス設定がパースされる。いくつかの実施形態において、MFO106は、識別された依存に基づいてあらゆる適切な数のデータ構造を生成することができる。これらのデータ構造（例えば、有向非巡回グラフ、連結リストなど）は、リージョン構築を実行するためのオペレーションを推進するためにクラウドインフラストラクチャオーケストレーションサービス102によって利用されてもよい。例えば、これらのデータ構造は、サービスがリージョン内でブートストラップされる順序を集散的に定義してもよい。このようなデータ構造の一例は、図3の構築依存グラフ338に関して以下でさらに検討される。循環依存（例えば、サービスAがサービスBを要求し、その逆でもある）が存在し、静的フロックス分析および／またはグラフを通じて識別されると、MFOは、これらの循環依存を修正するために、対応するフロックス設定に対して変更が要求されることを、あらゆる適切なサービスチームに通知するように構成されてもよい。MFO106は、サービスがリージョンにブートストラップされる順序を管理するために1つまたは複数のデータ構造を横断するように構成されることができる。MFO106は、あらゆる与えられた時間に、与えられたリージョン内で利用可能なケイパビリティを（例えば、ケイパビリティサービス112から取得されたデータを用いて）識別することができる。MFO106は、いつサービスをブートストラップすることができるか、いつブートストラップがブロックされるか、および／またはいつ前にブロックされたサービスに関連付けられたブートストラップオペレーションが再開することができるかを識別するために、このデータを用いることができる。この横断に基づいて、MFO106は、あらゆる適切な数のフロックス設定に対応するブートストラップオペレーションを実行するために命令がMFO106によってCIOSセントラル108へ送信される様々なリリースを実行することができる。いくつかの例において、MFO106は、グラフ内に見られる循環依存により1つまたは複数のフロックス設定が多数のリリースを要求し得ることを識別するように構成されてもよい。その結果、MFO106は、グラフにおいて識別された循環依存を破壊するために、与えられたフロックス設定のためのCIOSセントラル108へ多数の命令セットを送信してもよい。

【0055】

いくつかの実施形態において、ユーザは、新たなリージョン（例えば、ターゲットリージョン114）が構築されることをリクエストすることができる。これは、様々なサービスに対応するリソースをブートストラップすることを必要とする可能性がある。いくつかの実施形態において、ターゲットリージョン114は、リージョン構築リクエストが開始される時間に通信的に利用可能（および／またはセキュア）でなくてもよい。ターゲットリージョン114が、ブートストラップオペレーションを実行するために利用可能でありかつ構成されるような時間まで、ブートストラップを遅らせるのではなく、CIOS102は、仮想ブートストラップ環境116を用いてリージョン構築を開始してもよい。仮想ブートストラップ環境（ViBE）116は、ホストリージョン103（サービスのコアセットとともに前に構成されておりかつ通信的に利用可能かつセキュアである、既存のリージョン）によってホストされるオーバーレイネットワークであってもよい。MFO106は、ViBE116へリソースをブートストラップするために（一般的に「ViBEを構築する」と呼ばれる）ホストリージョン103のリソースを活用することができる。例えば、MFO106は、ホストリージョン（例えば、ホストリージョン103）内のCIOSリージョナル110のインスタンスにViBE116内のCIOSリージョナルの別のインスタンスをブートストラップさせる命令を、CIOSセントラル108を通じて提供することができる。ViBE内のCIOSリージョナルが処理のために利用可能になると、ターゲットリージョン114のためのサービスのブートストラップは、ViBE11

10

20

30

40

50

6 内で継続することができる。ターゲットリージョン 114 が、ブートストラップオペレーションを実行するために利用可能であるとき、V i B E 116 内の前にブートストラップされたサービスは、ターゲットリージョン 114 に移動させられてもよい。これらの技術を利用して、C I O S 102 は、あらゆる手動入力および／または構成が提供される必要性を劇的に減じることによって、リージョンが構築される速度を著しく高めることができる。

【0056】

図 2 は、少なくとも 1 つの実施形態による、仮想ブートストラップ環境 (V i B E) 202 (図 1 の V i B E 116 の一例) を構築するための環境 200 および方法を示すためのブロック図である。V i B E 202 は、既存のリージョン (例えば、ホストリージョン 204、図 1 のおよび実施形態におけるホストリージョン 103 の一例は、ホストリージョンサービスエンクレープである) のオーバーレイにおいてプロビジョニングされる仮想クラウドネットワークを表す。V i B E 202 は、ターゲットリージョンが利用可能となる前にサービスがターゲットリージョン (例えば、図 1 のターゲットリージョン 114 などの構築中のリージョン) のためにステージングされることができる環境を表す。

【0057】

新たなリージョン (例えば、図 1 のターゲットリージョン 114) をブートストラップするために、サービスのコアセットがブートストラップされてもよい。サービスのこれらのコアセットはホストリージョン 204 に存在してもよいが、まだ V i B E には (ターゲットリージョンにも) 存在しない。これらの必須のコアサービスは、デバイスをプロビジョニングし、新たなリージョンへのトラストのチェーンを確立し、リージョン内へ残りのサービス (例えば、負ロック) をデプロイするために必要とされる機能を提供する。V i B E 202 は、ホストリージョン 204 においてデプロイされるテナンシであってもよい。これは、仮想リージョンとして考えることができる。

【0058】

ターゲットリージョンが、ブートストラップオペレーションを提供するために利用可能になると、V i B E 202 をターゲットリージョンに接続することができ、これにより、V i B E におけるサービスは、ターゲットリージョンのサービスおよび／またはインフラストラクチャコンポーネントと相互作用することができる。これは、従前のシステムにおけるような自己完結型シードサービスの代わりに、生産レベルサービスのデプロイメントを可能にし、インターネット上でターゲットリージョンへの接続を必要とする。従来、シードサービスは、コンテナ収集の一部としてデプロイされ、リージョンを構築するために必要な依存をブートストラップするために使用されていた。既存のリージョンのインフラストラクチャ／ツーリングを用いて、リソースは、V i B E 202 内へブートストラップ (例えば、プロビジョニングおよびデプロイ) され、ターゲットリージョンが自給自足でき、直接通信されることができるまで、ハードウェアをプロビジョニングしかつサービスをデプロイするためにリージョン (例えば、ホストリージョン 204) のサービスエンクレープに接続されてもよい。V i B E 202 を利用することは、コアサービスの循環依存を破壊するためにホストリージョンのリソースを利用しながらインフラストラクチャをプロビジョニング／準備しかつソフトウェアをデプロイすることができるために必要とされる依存およびサービスをスタンドアップさせることを可能にする。

【0059】

マルチブロックオーケストレータ (M F O) 206 は、V i B E 202 を構築 (例えば、構成) するためにオペレーションを実行するように構成されてもよい。M F O 206 は、新たなリージョン (このケースでは、V i B E リージョン、V i B E 202) にブートストラップされる様々なリソースに対応する適用可能なブロック設定を取得することができる。例えば、M F O 206 は、ケイパビリティサービス 208 およびワーカー 210 をブートストラップする様子を識別するブロック設定 (例えば、「V i B E ブロック設定」) を取得してもよい。別の例として、M F O 206 は、V i B E 202 にドメイン名サービス (D N S) 212 をブートストラップすることに対応する別のブロック設定を取得し

10

20

30

40

50

てもよい。

【0060】

ステップ1において、MFO206は、CIOSセントラル214（例えば、それぞれ図1および図2のCIOSセントラル108およびCIOSセントラル214の一例）に命令してもよい。例えば、MFO206は、この時にはまだVIBE202に存在しないケイパビリティサービス208およびワーカー210のブートストラップを要求するためのリクエスト（例えば、VIBEフロック設定を含む）を送信してもよい。いくつかの実施形態において、CIOSセントラル214は、全てのフロック設定へのアクセスを有してもよい。したがって、いくつかの例において、MFO206は、ファイル自体ではなくVIBEフロック設定のための識別子を送信してもよく、CIOSセントラル214は、それをストレージから（例えば、図3のDB308またはフロックDB312から）独立して取得してもよい。

10

【0061】

ステップ2において、CIOSセントラル214は、CIOSリージョナル216への対応するリクエストによってVIBEフロック設定を提供してもよい。CIOSリージョナル216は、ステップ3において特定のインフラストラクチャプロビジョニングおよびデプロイメントオペレーションを識別および実行するためにVIBEフロック設定をパースしてもよい。

【0062】

いくつかの実施形態において、CIOSリージョナル216は、プロビジョニングおよびデプロイメントのための追加的な対応するサービスを利用してもよい。例えば、ステップ4において、CIOSリージョナル216は、CIOSリージョナルは、デプロイメントオーケストレータ218（例えば、ホストリージョン204の、コアサービスまたはその他の書き込み、構築およびデプロイアプリケーションソフトウェアの一例）に命令し、命令を実行させてもよく、この命令自体は、ケイパビリティサービス208およびワーカー210をVIBE202内でブートストラップさせる。

20

【0063】

ステップ5において、ケイパビリティは、（CIOSリージョナル216、ワーカー210を介してデプロイメントオーケストレータ218、またはその他から）ケイパビリティサービス208へ送信されてもよく、VIBEフロックに対応するリソースが利用可能であることを示す。ケイパビリティサービス208は、このデータを持続してもよい。いくつかの実施形態において、ケイパビリティサービス208は、VIBEとともに利用可能なケイパビリティのそれが維持するリストにこの情報を加える。例えば、ステップ5においてケイパビリティサービス208に提供されるケイパビリティは、ケイパビリティサービス208およびワーカー210が処理のために利用可能であることを示してもよい。

30

【0064】

ステップ6において、MFO206は、ケイパビリティが、ケイパビリティサービス208からデータ（ケイパビリティに対応する識別子）を受信または取得することに基づいてケイパビリティサービス208およびワーカー210が利用可能であることを示していることを識別してもよい。

40

【0065】

ステップ7において、ステップ6においてデータを受信／取得した結果、MFO206は、VIBE202にDNSサービス（例えば、DNS212）をブートストラップするようにCIOSセントラル214に命令してもよい。命令は、DNSサービスに対応する特定のフロック設定を識別してもよいまたは含んでもよい。

【0066】

ステップ8において、CIOSセントラル214は、VIBE202にDNS212をデプロイするようにCIOSリージョナル216に命令してもよい。いくつかの実施形態において、DNS212のためのDNSフロック設定は、CIOSセントラル214によって提供される。

50

【0067】

ステップ9において、今ではV i B E 2 0 2においてデプロイされているワーカー210は、C I O Sリージョナル216によって、D N S 2 1 2をデプロイするタスクに割り当てられてもよい。ワーカーは、D N S 2 1 2をデプロイするために実行される必要があるオペレーションのセットを識別するために（例えば、フロック設定（所望の状態）を、フロックに関連付けられた（現在は存在しない）リソースの現在の状態と比較することによって）、図3に関連して上記で記述された形式で宣言型インフラストラクチャプロビジョナを実行してもよい。

【0068】

ステップ10において、デプロイメントオーケストレータ218は、ステップ9において識別されたオペレーションに従ってD N S 2 1 2をデプロイするようにワーカー210に命令してもよい。示されているように、ワーカー210は、ステップ11においてV i B E 2 0 2にD N S 2 1 2をデプロイするためのオペレーションを実行することを進める。ステップ12において、ワーカー210は、V i B E 2 0 2においてD N S 2 1 2が利用可能であることをケイパビリティサービス208に通知する。M F O 2 0 6は、続いて、V i B Eフロック設定およびD N Sフロック設定に関連付けられたリソースが利用可能であることを識別してもよく、V i B Eにあらゆる適切な数の追加的なリソースをブートストラップするために進行してもよい。

【0069】

ステップ1～12が完了した後、V i B E 2 0 2を構築するためのプロセスが完了したと考えることができ、V i B E 2 0 2が構築されたと考えることができる。

【0070】

図3は、少なくとも1つの実施形態による、V i B Eを利用してターゲットリージョンにサービスをブートストラップするための環境300および方法を示すためのブロック図である。

【0071】

ステップ1において、ユーザ302は、リージョンデータを修正するために、C I O S セントラル304（それぞれ図1および図2のC I O S セントラル108およびC I O S セントラル214の一例）によって提供されるあらゆる適切なユーザインターフェースを利用してよい。例えば、ユーザ302は、そこに多数のサービスがブートストラップされる新たなリージョンを生成してもよい。

【0072】

ステップ2において、C I O S セントラル304は、R R D D 3 0 6（例えば、図1のR R D D 1 0 4の一例）に変更を送信するためのオペレーションを実行してもよい。ステップ3において、R R D D 3 0 6は、受信したリージョンデータを、データベース308、リージョン、A D、レルム、E Tなどのあらゆる適切な識別子、属性、状態などを含むリージョンデータを記憶するように構成されたデータストアに記憶してもよい。いくつかの実施形態において、アップデータ307は、リージョンデータをデータベース308またはあらゆる適切なデータストアに記憶するために利用されてもよく、そこからこのようなアップデートが（例えば、サービスチームに）アクセス可能であってもよい。いくつかの実施形態において、アップデータ307は、データベース308に対してなされたアップデートを（例えば、あらゆる適切な電子通知を介して）通知するように構成されてもよい。

【0073】

ステップ4において、M F O 3 1 0（それぞれ図1および図2のM F O 1 0 6および206の一例）は、リージョンデータにおける変更を検出してもよい。いくつかの実施形態において、M F O 3 1 0は、リージョンデータにおける変更のためのR R D D 3 0 6をポーリングするように構成されてもよい。いくつかの実施形態において、R R D D 3 0 6は、リージョン変更をM F O 3 1 0に発行またはさもなければ通知するように構成されてもよい。

10

20

30

40

50

【0074】

ステップ5において、リージョンデータにおける変更を検出することは、バージョンセット（例えば、「ゴールデンバージョンセット」識別子等の特定の識別子に関連付けられたバージョンセット）を取得するようにMFO310をトリガしてもよい。新たなリージョンにブートストラップされる各々のブロック（例えば、サービス）のための特定のバージョンおよびそのブロックに対応する各々のアーティファクトのための特定のバージョンを識別する。バージョンセットは、DB312から取得されてもよい。ブロックが発達および変化するにつれて、リージョン構築のために使用されるそれらの対応する設定およびアーティファクトのためのバージョンが変化し得る。これらの変更はブロックDB312において持続されてもよく、これにより、MFO310は、リージョン（例えば、VIBEリージョン、ターゲットリージョン／非VIBEリージョンなど）を構築するために使用するためにブロック設定およびアーティファクトのどのバージョンを使用するかを識別してもよい。ブロック設定（例えば、ブロック設定の全てのバージョン）および／またはアーティファクト（例えば、アーティファクトの全てのバージョン）は、DB308、DB312、またはCIOSセントラル304および／またはMFO310にアクセス可能なあらゆる適切なデータストアに記憶されてもよい。

10

【0075】

ステップ6において、MFO310は、現在のリージョンデータを備えるバージョンセットに関連付けられたブロック設定の各々を再コンパイルするようにCIOSセントラル304にリクエストしてもよい。いくつかの実施形態において、リクエストは、各々のブロック設定および／またはこれらのブロック設定に対応するアーティファクトのためのバージョンを示してもよい。

20

【0076】

ステップ7において、CIOSセントラル304は、（例えば、直接、またはリアルタイムリージョナルデータディストリビュータ306を介して）DB308から現在のリージョンデータを取得し、MFO310によってリクエストされたバージョンに従ってあらゆる適切なブロック設定およびアーティファクトを検索してもよい。

【0077】

ステップ8において、CIOSセントラル304は、現在のリージョンデータを備えるブロック設定を注入するために、ステップ7において取得されたリージョンデータを備えるブロック設定を再コンパイルしてもよい。CIOSセントラル304は、コンパイルされたブロック設定をMFO310に返してもよい。いくつかの実施形態において、CIOSセントラル304は、コンパイルが行われたことを単に示してもよい、またはMFO310は、RRDD306を介して、再コンパイルされたブロック設定にアクセスしてもよい。

30

【0078】

ステップ9において、MFO310は、再コンパイルされたブロック設定の静的解析を実行してもよい。静的解析の一部として、MFO310は、ブロック間の依存を識別するために、（例えば、宣言型インフラストラクチャプロビジョナ（例えば、Terraformなど）と関連付けられたライブラリを使用して）ブロック設定をパースしてもよい。識別された分析および依存から、MFO310は、構築依存グラフ338を生成することができる。構築依存グラフ338は、ブロックが新たなリージョンにブートストラップされる順序（および／またはブロック設定において示される変更が適用される順序）を識別する、非巡回有向グラフであってもよい。グラフにおける各々のノードは、特定のブロックのあらゆる適切な部分をブートストラップすることに対応してもよい。特定のブートストラップ順序は、依存に少なくとも部分的に基づいて識別されてもよい。いくつかの実施形態において、依存は、ノードの属性として表されてもよいおよび／またはノードを接続するグラフのエッジを介して示されてもよい。MFO310は、リージョン構築のオペレーションを推進するために、グラフ（例えば、開始ノードにおいて始まる）をトラバースしてもよい。

40

50

【0079】

いくつかの実施形態において、MFO310は、循環の存在（例えば、サービスAはサービスBに依存し、その逆でもある）を検出するための循環検出アルゴリズムを利用してよい。MFO310は、孤立したケイパビリティ依存を識別することができる。例えば、MFO310は、いかなる他のノードにも接続されていない構築依存グラフ338の孤立したノードを識別することができる。MFO310は、（例えば、ケイパビリティが早く発行されすぎたとき、および対応する機能が実際にはまだ利用可能ではないとき）、誤って発行されたケイパビリティを識別してもよい。MFO310は、同じケイパビリティを発行する1つまたは複数のインスタンスが存在することをグラフから検出することができる。いくつかの実施形態において、あらゆる適切な数のこれらのエラーが検出されてもよく、MFO310（またはCIOSセントラル304などの別の適切なコンポーネント）は、（例えば、電子通知、ユーザインターフェースなどを介して）この情報をユーザに通知またはさもなければ提示するように構成されてもよい。いくつかの実施形態において、MFO310は、循環依存を破壊するためにリソースを強制的に削除／再現させるように構成されてもよく、これらのリソースおよび／または対応するフロック設定のためのブートストラップオペレーションを実行するようにCIOSセントラル304に再び命令を提供してもよい。

10

【0080】

開始ノードはVIBEフロックをブートストラップすることに対応してもよく、第2のノードはDNSをブートストラップすることに対応してもよい。ステップ10～15は、（デプロイメントオーケストレータ317、図2のデプロイメントオーケストレータ218の一例を介して）VIBEフロックをVIBE316（例えば、それぞれ図1および図2のVIBE116および202の一例）にデプロイすることに対応する。すなわち、図3のステップ10～15は、概して、図2のステップ1～6に対応する。デプロイされるVIBEフロックに対応するケイパビリティが存在することが通知されると（例えば、図2のケイパビリティサービス208およびワーカー210に対応する、ケイパビリティサービス318およびワーカー320が利用可能であることを示す）、MFO310は、実行される次のオペレーションを識別するために構築依存グラフ338のトラバースを再開する。

20

【0081】

例えば、MFO310は、DNSフロックがデプロイされることを識別するために、構築依存グラフ338をトラバースし続けてもよい。ステップ16～21は、DNS322（図2のDNS212の一例）をデプロイするために実行されてもよい。これらのオペレーションは、概して、図2のステップ7～12に対応してもよい。

30

【0082】

ステップ21において、DNS322が利用可能であることを示すケイパビリティが記憶されてもよい。このケイパビリティを検出すると、MFO310は、構築依存グラフ338のトラバースを再開してもよい。このトラバースにおいて、MFO310は、CIOSリージョナル（例えば、CIOSリージョナル314の一例）のインスタンスのあらゆる適切な部分がVIBE316にデプロイされることを通知してもよい。いくつかの実施形態において、ステップ16～21は、CIOSリージョナル（VIBE）326（CIOSリージョナル314のインスタンス、図1のCIOSリージョナル110）およびワーカー328をVIBE316にデプロイすることに関して実質的に反復されてもよい。ケイパビリティは、CIOSリージョナル（VIBE）326が利用可能であるケイパビリティサービス318へ送信されてもよい。

40

【0083】

CIOSリージョナル（VIBE）326が利用可能であることを検出すると、MFO310は、構築依存グラフ338のトラバースを再開してもよい。このトラバースにおいて、MFO310は、デプロイメントオーケストレータ（例えば、デプロイメントオーケストレータ330、デプロイメントオーケストレータ317の一例）がVIBE316に

50

デプロイされることを識別してもよい。いくつかの実施形態において、ステップ 16～21 は、デプロイメントオーケストレータ 330 をデプロイすることに関して実質的に反復されてもよい。ケイパビリティを識別する情報がケイパビリティサービス 318 へ送信されてもよく、デプロイメントオーケストレータ 330 が利用可能であることを示す。

【0084】

デプロイメントオーケストレータ 330 がデプロイされると、V i B E 316 は、後続のリクエストを処理するために利用可能であると考えられてもよい。デプロイメントオーケストレータ 330 が利用可能であることを検出すると、M F O 310 は、ホストリージョンコンポーネント（ホストリージョン 322 のコンポーネント）を利用するのではなく、V i B E コンポーネントヘルディングされるように後続のブートストラップリクエストに命令してもよい。したがって、M F O 310 は、C I O S セントラル 304 を介して V i B E 316 へフロックデプロイメントを命令する各々のノードにおいて、構築依存グラフ 338 をトラバースし続けることができる。C I O S セントラル 304 は、フロック設定に従ってリソースをデプロイするように C I O S リージョナル（V i B E）326 にリクエストしてもよい。

【0085】

このプロセスの間のある時点において、ターゲットリージョン 334 が利用可能になってもよい。ターゲットリージョンが利用可能であることの指示は、（例えば、リージョンデータに対する更新として）ユーザ 302 によって提供されるターゲットリージョン 334 のためのリージョンデータから識別可能であってもよい。ターゲットリージョン 334 のアベイラビリティは、ターゲットリージョン 334 と外部ネットワーク（例えば、インターネット）との間のネットワーク接続を確立することに依存してもよい。ネットワーク接続は、公共ネットワーク（例えば、インターネット）上でサポートされてもよいが、1 つまたは複数の暗号化トンネル（例えば、トンネル 336 などの I P s e c トンネル）を V i B E 316 からターゲットリージョン 334 に提供するためにソフトウェアセキュリティツール（例えば、I P s e c）を使用してもよい。本明細書で使用される場合、「I P s e c」は、インターネットプロトコル（I P）を使用するネットワーク上でネットワークトラフィックを認証および暗号化するためのプロトコル群を意味し、プロトコル群（例えば、Openswan、Libreswan、strongSwan など）の 1 つまたは複数の利用可能な実装を含むことができる。ネットワークは、V i B E 316 をターゲットリージョン 334 のサービスエンクレーブに接続してもよい。

【0086】

I P s e c トンネルを確立する前に、ターゲットリージョン 334 への初期ネットワーク接続は、I P s e c がターゲットリージョン 334 におけるアセット（例えば、ベアメタルアセット）においてデプロイされ得るまで、ネットワーキングサービスのブートストラップを許容するように十分な接続（例えば、帯域外 V P N トンネル）上にあってもよい。ターゲットリージョン 334 のネットワークリソースをブートストラップするために、デプロイメントオーケストレータ 330 は、ターゲットリージョン 334 内のアセットにおいて I P s e c ゲートウェイをデプロイすることができる。次いで、デプロイメントオーケストレータ 330 は、V i B E 316 からの I P s e c トンネルを終了させるように構成されたターゲットリージョン 334 において V P N ホストをデプロイしてもよい。V i B E 316 におけるサービス（例えば、デプロイメントオーケストレータ 330、サービス A など）は、ターゲットリージョン 334 における V P N ホストとの I P s e c 接続を確立することができ、V i B E 316 からターゲットリージョン 334 へのブートストラップオペレーションが開始してもよい。

【0087】

いくつかの実施形態において、ブートストラップオペレーションは、V i B E 316 からデプロイされるときにコアサービスのインスタンスをホストすることをサポートするために、V i B E 316 におけるサービスがターゲットリージョン 334 においてリソースをプロビジョニングすることによって開始してもよい。例えば、ホストプロビジョニング

サービスは、VMのためのコンピューティングリソースを割り当てるためにターゲットリージョン334においてインフラストラクチャ（例えば、ベアメタルホスト）上にハイパーバイザをプロビジョニングしてもよい。ホストプロビジョニングサービスがターゲットリージョン334における物理的リソースの割り当てを完了すると、ホストプロビジョニングサービスは、ターゲットリージョン334における物理的リソースが割り当てられたことを示すケイパビリティを示す情報を発行してもよい。ケイパビリティは、CIOSリージョナル（V i B E）326を介して（例えば、ワーカー328によって）ケイパビリティサービス318へ発行されてもよい。

【0088】

ターゲットリージョン334のハードウェア割り当てが確立され、ケイパビリティサービス318へポストされると、CIOSリージョナル（V i B E）326は、V i B E 316からターゲットリージョン334へのコアサービスのインスタンスのデプロイメントをオーケストレーションすることができる。このデプロイメントは、V i B E 316を構築するための上記で説明されたプロセスと類似のものであってもよいが、ホストリージョン332サービスエンクレーブのコンポーネントの代わりに、V i B Eのコンポーネント（例えば、CIOSリージョナル（V i B E）326、ワーカー328、デプロイメントオーケストレータ330）を使用する。デプロイメントオペレーションは、概して、上記で説明されたステップ16～21に対応してもよい。

【0089】

サービスがV i B E 316からターゲットリージョン334へデプロイされるので、そのサービスと関連付けられたDNSレコードは、V i B E 316におけるサービスのインスタンスに対応してもよい。サービスと関連付けられたDNSレコードは、ターゲットリージョン334へのサービスのデプロイメントを完了するための後に更新されてもよい。言い換えれば、V i B E 316におけるサービスのインスタンスは、DNSレコードが更新されるまで、サービスヘトラフィック（例えば、リクエスト）を受信し続けてもよい。サービスは、部分的にターゲットリージョン334内へデプロイしてもよく、サービスが部分的にデプロイされるケイパビリティを示す情報を（例えば、ケイパビリティサービス318へ）発行してもよい。例えば、V i B E 316において動作するサービスは、対応するコンピュータインスタンス、ロードバランサ、および関連付けられたアプリケーションおよびその他のソフトウェアを備えるターゲットリージョン334内へデプロイされてもよいが、完全にデプロイされる前にデータベースデータがターゲットリージョン334へ移動するのを待つ必要があり得る。DNSレコード（例えば、DNS322によって管理される）は、依然として、V i B E 316におけるサービスと関連付けられていてもよい。サービスのためのデータ移動が完了すると、DNSレコードは、ターゲットリージョン334においてデプロイされたオペレーショナルサービスへのポイントに更新されてもよい。ターゲットリージョン334におけるデプロイされたサービスは、次いで、サービスのためのトラフィック（例えば、リクエスト）を受信してもよいのに対し、V i B E 316におけるサービスのインスタンスは、もはやサービスのためのトラフィックを受信しなくてもよい。

【0090】

バージョンセットとのマルチフロックオーケストレーション

図4は、少なくとも1つの実施形態による、多数のバージョンセットを維持するための例示的な方法400を示すブロック図である。方法400は、図1のクラウドインフラストラクチャオーケストレーションサービス102のあらゆる適切なコンポーネントによって実行されてもよい。いくつかの実施形態において、方法400は、図1のマルチフロックオーケストレータ106によって実行されてもよい。バージョンセットは、リージョン構築を実行するために用いられる全てのアーティファクトおよび構成ファイル（例えば、本明細書では略して「フロック設定」と呼ばれる、フロック構成ファイル）のスナップショットを指す。バージョンセットにおいて提供されるフロック設定および／またはアーティファクトは、そのアーティファクトまたはフロック設定の他のバージョンからアーティ

ファクトおよび／またはフロック設定の特定のバージョンを識別するバージョン識別子と関連付けられてもよい。例えば、フロック設定は、識別子「サービス1」と関連付けられてもよいが、多数のバージョン（例えば、バージョン1、バージョン2、バージョン3、1など）を有する。マルチフロックオーケストレータ（例えば、それぞれ図1、図2および図3のMFO106、206および310）は、潜在的に異なるフロック設定および／またはフロック設定のバージョン、または利用されるアーティファクトおよび／またはアーティファクトのバージョンに対応する多数のバージョンセットを管理するように構成されてもよい。バージョンセットは、リージョン構築を実行するために使用されることができ、その対し、テストおよび統合努力のために異なるバージョンセットを使用することができる。異なるバージョンセットを追跡することによって、マルチフロックオーケストレータは、リージョン構築のために、安定したバージョンセット（例えば、成功したリージョン構築を生成する可能性が高いバージョンセット）が使用されることを保証することができる。バージョンセットは、それによってバージョンセットが参照されることができる識別子（例えば、「us-ashburn-1」およびスコープ識別子と関連付けられてもよい。例示的なスコープ識別子は、「golden」、「nightly」、「break-glass」、および「default」を含むことができ、それらに限定されないが、その他の識別子を使用することができる。1つのスコープ識別子（例えば、golden）は、リージョン構築のために使用されるバージョンセットを示すことができる。別のスコープ識別子（例えば、nightly）は、テスト（例えば、毎夜実行される単位テスト）のために使用されるバージョンセットを示してもよい。別のスコープ識別子（例えば、break-glass）は、リージョン構築中に受けたエラーから回復するために使用することができるバージョンセットを示してもよい。別のスコープ識別子（例えば、default）は、リージョン構築における最終的な包含を期待してテストされるフロック設定／アーティファクトのバージョンセットを示してもよい。

10

20

30

40

50

【0091】

ステップ1において、各々のサービスチームは、フロック設定データ（例えば、フロック設定識別子、およびフロック設定の他のバージョンからフロック設定の特定のバージョンを一意的に識別する対応するバージョン識別子）を発行してもよい。いくつかの実施形態において、フロック設定データを発行することは、アプリケーションプログラミングインターフェース（API）コールまたは別の適切なファンクションコールを利用することによって実行されてもよい。このコールを実行することによって、フロック設定データ（例えば、フロック設定識別子およびバージョン識別子）が、defaultバージョンセット（例えば、defaultバージョンセット402）に加えられてもよい。いくつかの実施形態において、defaultバージョンセット402は、図3のDB312またはあらゆる適切なデータストアに記憶されてもよい。別の例として、フロック設定がdefaultバージョンセットに含まれるべきであることを示すフロック設定の特定のバージョンは、手動で（例えば、ラベル／タグを用いて）修正されてもよい。マルチフロックオーケストレータは、defaultバージョンセットと関連付けられるセットを識別するために全てのフロック設定をパースしてもよい。defaultバージョンセット404に示されているように、「A」～「D」は、それぞれのフロックと関連付けられたフロック設定識別子に対応するのに対し、defaultバージョンセット402において提供された「8」、「9」および「3」は、フロック設定のバージョンを指す。したがって、フロック設定A、バージョン8、フロック設定B、バージョン9、およびフロック設定C、バージョン2が、defaultバージョンセット404に加えられてもよい。defaultバージョンセットは、フロック設定「D」、バージョン9を予め含んでもよい。ステップ1の後に維持されるdefaultバージョンセットは、406に示されている。

【0092】

ステップ2において、あらゆる適切なオペレーションが、テストのためのnightlyバージョンセットを決定するために実行されてもよい。例えば、フロック設定識別子および対応するバージョン識別子のセットが、defaultバージョンセット（例えば、406に示さ

れている) から nightly バージョンセット (例えば、nightly バージョンセット 408) へコピーされてもよい。いくつかの実施形態において、nightly バージョンセットは、検証過程と関連付けられてもよい (例えば、所定のスケジュールに従って行われる nightly ユニットおよび／または統合テスト)。nightly バージョンセットは、あらゆる適切な検証過程に対応してもよく、毎夜実行される検証過程に必ずしも対応しなくてもよいことが認められるべきである。nightly バージョンセットは、フロック設定識別子、およびリージョン構築の各々のサービスのためのバージョンを識別してもよい。408 に示されているように、テストが開始されたとき、nightly バージョンセットは、フロック設定 A、バージョン 2、フロック設定「B」、バージョン 7、およびフロック設定「C」、バージョン 3 を含んでいてもよい。異なるバージョンが (406 に示されているように) 引き続き使用されてもよいが、テストが開始される時点の default バージョンセットは、nightly バージョンセットにコピーされるセットである。サービスは、nightly バージョンセットによって識別されたフロック設定 (およびこれらのフロック設定内で識別された対応するアーティファクトバージョン) を用いてテスト環境内でブートストラップされてもよく、リージョンのこのバージョンが検証されてもよい。

10

【0093】

ステップ 3 において、検証過程 (1 つまたは複数のユニットおよび／または統合テストを含む) が、テスト環境においてブートストラップされるサービスを検証するために実行されてもよい。いくつかの実施形態において、検証過程は、あらゆる適切な所定のオペレーションを含んでもよい。これらのオペレーションは、nightly バージョンセットによって識別されたサービスに対応する様々な機能をテストすることに向けられてもよい。マルチフロックオーケストレータ (MFO) は、検証過程の出力として提供される 1 つまたは複数のエラーを識別してもよい。これらのエラーは、1 つまたは複数のサービスに対応してもよい。いくつかの実施形態において、特定のサービスに対応するエラーが識別されると、MFO は、フロック設定のそのバージョンが、MFO によって維持されかつ後続のリージョン構築のために使用される golden バージョンセットに促進／コピーされることを制限するように構成されてもよい。

20

【0094】

ステップ 4 において、ステップ 3 において検証過程を完了した後、MFO は、golden バージョンセットを決定するように構成されてもよい。サービスが検証過程をうまく通過すると (例えば、エラーを発生しないことによってまたは少なくとも許容できるエラーを発生することによって)、MFO は、対応するフロック設定データ (例えば、設定識別子およびバージョン識別子) を golden バージョンセットにコピーしてもよい。サービスが検証過程に不合格になると (例えば、エラーを発生することによってまたは少なくとも許容できないエラーを発生することによって)、MFO は、検証過程を通過したことが分かっているフロック設定の最後の既知のバージョンを維持してもよい。例えば、フロック設定「A」、バージョン 1 が検証過程を通過すると、これは、golden バージョンセットへ促進されてもよい。その後、フロック設定「A」、バージョン 2 が、default バージョンセットへ発行され、後で、nightly バージョンセットへ促進されてもよい。しかしながら、フロック設定「A」、バージョン 2 が検証過程に不合格になると (例えば、1 つまたは複数の許容できないエラーを発生する)、フロック設定「A」、バージョン 2 は、golden バージョンセットへ促進されなくてもよい。代わりに、MFO は、golden バージョンセット内にフロック設定「A」、バージョン 1 (例えば、410 に示された golden バージョンセット) を維持してもよい。

30

40

【0095】

ステップ 5 において、リージョン構築 (例えば、新たなリージョンデータを識別すると MFO によって開始される、またはその他) を開始すると、golden バージョンセットは、リージョン構築のために利用されるために MFO によってコピーおよび／または取得されてもよい。このプロセスは、概して、上記で説明された図 5 のステップ 5 に対応してもよい。

50

【0096】

goldenバージョンセットは、リージョン構築のために利用されてもよい。いくつかの実施形態において、サービスチームは、フロック設定におけるエラーを識別してもよく、更新された（または異なる）バージョンのフロック設定を提供してもよい。ステップ6において、更新されたフロック設定は、Break Glassバージョンセットに加えられることができる。このBreak Glassバージョンセットは、Goldenバージョンセットのあらゆる適切なフロック設定のオーバーライドとして使用されてもよい。例えば、ステップ6において、フロック設定「A」、バージョン9が、Break Glassバージョンセットに加えてもよい。フロック設定「A」のバージョン9は、412に示されたgoldenバージョンセットのバージョン1とは異なってもよい。Break Glassバージョンセットにフロック設定「A」のバージョン9を加えることにより、MFOがトリガされてもよく、これにより、414に示されたBreak Glassバージョンセットのフロック構成のセットの静的解析を実行することによって構築依存グラフ314を再生し、ここで、フロック設定「A」、バージョン1が、フロック設定「A」、バージョン9と置き換えられる。いくつかの実施形態において、MFOが構築依存グラフ314を実行してもよいおよび／またはブートストラップオペレーションを管理するためのあらゆる適切なデータ構造（例えば、図3の宣言型インフラストラクチャプロビジョナ315によって生成および／または利用される1つまたは複数のデータ構造）が再生されてもよい。再生されたあらゆるデータ構造が、ノードが訪問される上記で説明されたのと類似の形式でトラバースされてもよい。ノードと関連付けられたケイパビリティ（例えば、ノードに対応するサービスが依存するケイパビリティ）が（図1のケイパビリティサービス112と協議することによって識別されるように）利用可能である場合、トラバースは、同じ評価が起こり得る次のノードへ進んでもよい。このプロセスは、ノードが依存するケイパビリティがそのために利用不可能であることが決定されるノードに達するまで、継続されてもよい。この時点で、MFOは、ケイパビリティが利用可能であることを確認するまで（例えば、ポーリングによっておよび／またはケイパビリティサービス112によって通知されることによって）、トラバースをパースしてもよい。例えば、MFOは、ノードが依存するケイパビリティが利用可能であるときに通知されるようにケイパビリティサービス112へリクエストを送信してもよく、ケイパビリティサービス112は、ケイパビリティが利用可能であることを示すデータを受信したときに対応する通知を提供するように構成されてもよい。代替的に、MFOは、ケイパビリティサービス112からケイパビリティのステータス（または全てのケイパビリティなどのケイパビリティのあらゆる適切な組合せ）をリクエストするために、あらゆる適切な所定のスケジュールにおいて1つまたは複数のリクエストを送信してもよい。いくつかの実施形態において、MFOは、ケイパビリティのステータス（例えば、ケイパビリティが利用可能または利用不可能である）を示すあらゆる適切な時間にケイパビリティサービス112からデータを受信してもよい。いくつかの実施形態において、ケイパビリティサービス112は、あらゆる適切な送信において、あらゆる適切な数のケイパビリティのステータスを示してもよい。

【0097】

図4に示された様々なバージョンセットを利用することによって、マルチフロックオーケストレータ（MFO）は、リージョン構築中にエラーの数および／またはエラーを発生する可能性を減じることができる。これは、新たなフロック設定（潜在的に新たなアーティファクトバージョンを含む）がdefaultバージョンの包含により指名されることを可能にし、defaultバージョンセットにおいて提供されたバージョンから生成されたnightlyバージョンセットにおける検証過程を実行し、合格したフロック設定バージョンのみを、それにより後続のリージョン構築が実行され得るgoldenバージョンセットへ促進することによって、達成される。

【0098】

図5は、少なくとも1つの実施形態による、リージョンヘルソースをブートストラップ（例えば、プロビジョニングおよびデプロイ）するためのオペレーションを実行するため

の例示的な方法 500 を示すブロック図である。方法 500 は、マルチブロックオーケストレータ 502（図 1 のマルチブロックオーケストレータ 106 の一例）によって実行されてもよい。ケイパビリティサービス 504 は、図 1 のケイパビリティサービス 112 の一例であってもよい。方法 500 のオペレーションを実行する前に、図 5 のステップ 1～9 が実行されてもよい。したがって、静的解析が（例えば、マルチブロックオーケストレータ 502 によって）実行されてもよい。「静的解析」という用語は、プログラムコード（例えば、ブロック設定などの構成ファイル）から、ブートストラップオペレーション間の依存のセットを識別するためのオペレーションを指すことが意図されている。リージョン構築のための静的解析を実行することは、構成ファイルのセット（例えば、図 4 に関して上記で説明された、golden バージョンセットに対応するブロック設定）を取得し、そこに定義されたリソース間の明確なおよび／または黙示的な依存を識別するためにこれらのファイルをパースすることを含んでもよい。いくつかの実施形態において、構成ファイルは、選択的なおよび／または所要の依存を識別してもよい。選択的な依存は、選択的に無視または実施され得る依存を識別してもよい。所要の依存は、例外なく依存が実施されることを示す。いくつかの実施形態において、マルチブロックオーケストレータ 502 は、これらの依存を維持および／または実施するための 1 つまたは複数のレコードを生成してもよい。例えば、マルチブロックオーケストレータ 502 は、識別された依存に基づいてグラフ（例えば、図 3 の構築依存グラフ 314、レコードの一例であるデータ構造など）を生成してもよい。グラフにおける各々のノードは、サービスのためのプロビジョニングおよびデプロイメントオペレーションに対応してもよい。いくつかの実施形態において、別々のグラフが、プロビジョニング対デプロイメントオペレーションのために維持されてもよい。これらのレコード（例えば、データ構造）を生成および／または利用するための例示的な技術は、その内容全体が全ての目的のために参照により組み込まれる、「Techniques for Managing Region Build Dependencies」という名称の米国特許仮出願第 63 / 315, 034 号に提供されている。以下の例は、有向グラフを利用するが、サービス／ブロック間の依存を維持するためのあらゆる適切なレコードが利用されてもよい。いくつかの実施形態において、方法 500 は、有向グラフの第 1 のノードが、依存を有さないブロックを示すことをマルチブロックオーケストレータ（MFO）502 が識別することに少なくとも部分的に基づいて開始してもよい。

【0099】

ブートストラップされるリソースのセットをカプセル化／識別するためにブロック設定が使用され得ることが認められるべきである。いくつかのケースにおいて、ブロック設定は、サービスの全てのリソースを識別してもよいが、必ずしも、サービスの全てのリソースが 1 つのブロック設定に含まれるケースである必要はない。

【0100】

ステップ 1 において、MFO 502 は、第 1 のブロック設定によるインフラストラクチャコンポーネントをプロビジョニングするための命令を実行してもよい。例えば、MFO 502 は、サービス 1 のためのインフラストラクチャコンポーネントをブートストラップするために CIOS セントラル（例えば、それぞれ図 1～図 3 の CIOS セントラル 108、214 および 304）に命令および／またはリクエストしてもよい。いくつかの実施形態において、命令／リクエストは、利用されるブロック設定を含んでもよいが、または、リクエストは、利用されるブロック設定に対応するバージョンを示してもよい。バージョンが識別されると、CIOS セントラル（ここには示されていない）は、リクエストされたバージョンに対応するブロック設定を（例えば、図 3 の DB 308 または 312 などから）取得してもよい。引き続き、CIOS セントラルは、図 2 および図 3 に関して上記で検討されたのと類似の形式で（例えば、リアルタイムリージョンデータがブロック設定に注入された後に）CIOS リージョナル（例えば、それぞれ図 1～図 3 の CIOS リージョナル 110、216 および／または 314 の一例）からサービス 1 のためのインフラストラクチャリソースのプロビジョニングを命令／リクエストすることへ進んでもよい。

【0101】

ステップ2において、M F O 5 0 2は、ステップ1においてプロビジョニングされたインフラストラクチャコンポーネントへ1つまたは複数のアーティファクトをデプロイするようにC I O S セントラルに命令／リクエストしてもよい。ステップ1におけるように、リクエストは、利用されるフロック設定（または利用されるフロック設定のためのバージョン識別子）および／またはデプロイされる各々のアーティファクトのためのバージョン識別子を含んでもよい。ステップ1およびステップ2におけるリクエストは各々、プロビジョニングリクエスト（例えば、インフラストラクチャコンポーネントをプロビジョニングするためのリクエスト）に対応するものとしてのリクエストまたはデプロイメントリクエスト（例えば、アプリケーション、構成ファイルまたは同様のものなどの様々なアーティファクトをデプロイするためのリクエスト）を識別する指示記号を有してもよい。いくつかの実施形態において、M F O 5 0 2は、サービスのためのインフラストラクチャコンポーネントをプロビジョニングするためのリクエストおよびこれらのインフラストラクチャコンポーネントに様々なアーティファクトをデプロイするためのデプロイメントリクエストを示すサービス1をブートストラップするための1つのリクエストを行ってもよい。したがって、いくつかの例において、ステップ1および2は、M F O 5 0 2からC I O S セントラルへの1つのリクエストによって実行されてもよい。他の実施形態において、ステップ1および2の各々（または、一緒に発行された場合、集合的にステップ1および2）は、グラフを通じたM F O 5 0 2の「パス」に対応してもよい。非限定的な例として、グラフをトラバースする第1のパスは、ステップ1のオペレーションを実行させ得るのに対し、グラフの第2のパスは、ステップ2のパスレーションを実行させ得る。

10

20

【0102】

ステップ3において、ケイパビリティ（ケイパビリティ1）は、ケイパビリティサービス504（例えば、それぞれ図1～図3のケイパビリティサービス112、208および／または308の一例）へ発行される。ケイパビリティを発行することは、ケイパビリティのための識別子（「ケイパビリティ識別子」と呼ばれる）によってアプリケーションプログラミングインターフェース（API）または別のファンクションコールをコールすることを含んでもよい。ケイパビリティ識別子は、あらゆる適切な長さの英数字値などのあらゆる適切な形式のものであってもよい。ケイパビリティ識別子は、特定の機能と関連付けられてもよい。いくつかの実施形態において、ケイパビリティは、1つまたは複数のアーティファクトがデプロイされているときに発行されてもよい。ケイパビリティは、サービスによって提供された機能の一部と関連付けられることができるか、またはケイパビリティは、サービスの全ての機能が利用可能であることを示すことができる。いくつかの実施形態において、多数のケイパビリティが、（例えば、サービス1のためにインフラストラクチャがプロビジョニングされるときおよび／またはアーティファクトがデプロイされるとき）ブートストラップサービス1の一部としてステップ3において発行されてもよい。ケイパビリティを発行する特定のコンピューティングコンポーネントは、図1のC I O S 102のあらゆる適切なコンポーネント（例えば、C I O S リージョナル110、または、V i B E が使用される場合、それぞれ図2および図3のC I O S リージョナル216または314、または図3のC I O S リージョナル（V i B E）326、デプロイメントオーケストレータ218、317、図2および図3のO S 330など）であってもよい。

30

40

【0103】

ステップ4において、M F O 5 0 2は、利用可能なケイパビリティの指示をポールまたはさもなければリクエストしてもよい。このリクエストは、所定の周期に従って、所定のスケジュールに従ってなど、あらゆる適切な時間に行われてもよい。いくつかの実施形態において、リクエストは、ケイパビリティサービス504に、発行／受信された全てのケイパビリティ識別子のリストを提供させてもよい。いくつかの実施形態において、リクエストは、特定のケイパビリティ識別子を含むことができ、提供される応答は、ケイパビリティ識別子が発行されたかどうか（例えば、ケイパビリティが発行されたかどうか、ケイパビリティが利用可能であるかどうか、など）を示すことができる。

【0104】

50

M F O 5 0 2 は、リージョン構築を促進するためにそれが使用しているグラフ（例えば、構築依存グラフ 3 1 4）を介して、他のリソースが、ステップ 4 において識別されたケイパビリティに依存しているかどうかを評価してもよい。例えば、グラフは、サービス 2 が、サービス 1 に関連付けられた機能に依存していることを示してもよい。この依存は、ケイパビリティとして表されてもよい。例えば、グラフは、サービス 1 が完全に（または部分的に）ブートストラップされている／動作可能であることを示すケイパビリティ（例えば、ケイパビリティ「1 2 3 4」）が発行されるまでブートストラップされることができないことを示すサービス 2 との関連を維持してもよい。所要のケイパビリティ（例えば、ケイパビリティ「1 2 3 4」）がステップ 4 を介して利用可能であることを識別すると、M F O 5 0 2 は、グラフのトラバースを継続してもよく、ステップ 5 へ進む。

10

【0 1 0 5】

いくつかの実施形態において、M F O 5 0 2 は、ケイパビリティが利用不可能であることを識別してもよい。いくつかの実施形態において、M F O 5 0 2 は、ケイパビリティサービス 5 0 4 へ、ケイパビリティが必要であるという指示を発行してもよい。M F O 5 0 2 は、サービス 2 がサービス 1（例えば、ケイパビリティ「1 2 3 4」）に対応する）に依存することを識別するためにグラフをトラバースしてもよい。いくつかの実施形態において、M F O 5 0 2 は、サービス 1 が利用可能であることを示すケイパビリティを待ちながらグラフのトラバースを休止してもよい。リージョンの様々なリソースが利用可能になるので、ケイパビリティは、ケイパビリティサービス 5 0 4 へ発行されてもよく、これらのケイパビリティが利用可能になったことを示す。いくつかの実施形態において、ケイパビリティサービス 5 0 4 が、特定のケイパビリティが利用可能になったことを識別すると、ケイパビリティサービス 5 0 4 は、M F O 5 0 2 に同じことを通知するために、アプリケーションプログラミングインターフェース、ファンクションコールなどのあらゆる適切な電子的方法を介してデータを送信してもよい。加えて、または代替的に、M F O 5 0 2 は、利用可能なケイパビリティ（例えば、1 つまたは複数のケイパビリティの特定のセット）の識別を周期的にポーリングまたはリクエストしてもよいおよび／またはケイパビリティサービス 5 0 4 は、利用可能なケイパビリティを周期的に発行してもよい。M F O 7 0 2 は、ケイパビリティサービス 5 0 4 が、ケイパビリティが利用可能であることを示すデータを提供するまで、ケイパビリティが利用不可能であることを見なしてもよく、グラフのさらなるトラバース（またはグラフの特定のパスのトラバース）を控えてもよい。M F O 7 0 2 が、ケイパビリティが利用可能になったことを識別すると（例えば、ケイパビリティサービス 5 0 4 によって提供されるデータに基づいて）、M F O 7 0 2 は、グラフのトラバースを継続してもよく、ここで追加のサービスがブートストラップされてもよいことを識別する。

20

30

【0 1 0 6】

ステップ 5、6 および 7 は、サービス 2 をブートストラップするという観点から、概してステップ 1、2 および 3 に対応してもよい。上記で説明されているように、ステップ 5 および 6 は、別々に実行されても、または 1 つのリクエストとして実行されてもよい。ステップ 7 において、ケイパビリティ 2 が発行されてもよく、サービス 2 の機能の一部または全てが利用可能であることを示す。上記で説明されているように、ケイパビリティサービス 5 0 4 は、このケイパビリティ情報を持続してもよい。

40

【0 1 0 7】

ステップ 8 において、M F O 5 0 2 は、サービス 2 に関連付けられたケイパビリティが利用可能であることを決定してもよい。上記で説明されているように M F O 5 0 2 は、（例えば、特にケイパビリティ識別子によって、または利用可能なケイパビリティのリストを取得することを介して）この情報をリクエストしてもよいまたはケイパビリティサービス 5 0 4 は、M F O 5 0 2 に、ケイパビリティのアベイラビリティを通知してもよい。いくつかの実施形態において、ケイパビリティサービス 5 0 2 は M F O 5 0 2 に特に通知してもよい、またはケイパビリティは、サービス 2 に関連付けられたケイパビリティが利用可能である 1 つまたは複数のコンピューティングコンポーネント（M F O 5 0 2 を含む）

50

へブロードキャストされてもよい。

【0108】

ステップ9、10、11は、サービス3をブートストラップするという観点から、ステップ1、2および3、および／または5、6および7に対応する。ステップ12は、ステップ4および8に対応してもよい。

【0109】

方法500のオペレーションは、あらゆる適切な回数実行されてもよい。例えば、ステップ1～4は、リージョン構築を推進するためにMFO502が使用しているグラフのトラバースに従ってあらゆる適切な回数（例えば、リージョンにおいてブートストラップされる各々のサービスのために）実行されてもよい。いくつかの実施形態において、図5のステップのあらゆる適切な部分が、図5に示されている順次方式ではなく並行方式で実行されてもよいことが認められるべきである。例えば、サービス1～3のリソースが依存を有さない場合、またはそれらが依存する全てのケイパビリティが利用可能である場合、それは、各々のサービスのために実行されるブートストラップオペレーションの少なくとも一部が実質的に並列方式で実行されることができ得る。方法500は、サービスおよび／またはリソースをVIBE（例えば、図5のVIBE524）へまたは（例えば、新たなリージョンが通信により利用可能である場合）VIBEを使用することなく新たなリージョンへ直接にブートストラップするために利用されてもよい。

【0110】

図6は、少なくとも1つの実施形態による、リージョン構築と関連付けられたサービスに関連した情報を表す例示的なユーザインターフェース600を示すブロック図である。いくつかの実施形態において、UI600は、図1のクラウドインフラストラクチャオーケストレーションサービス102のあらゆる適切なコンポーネント（例えば、それぞれ図1～図3のCIOSセントラル108、214および／または304）によってホストされてもよい。リージョン構築に対応するあらゆる適切なデータが、ユーザインターフェース（UI）600に表示されてもよい。いくつかの実施形態において、UI600は、特定のリージョン構築を選択するためのUIエレメント（例えば、UIエレメント602、ドロップダウンメニュー）を含んでもよい。UIエレメント602から特定のリージョン構築（例えば、リージョンX）を選択すると、そのリージョン構築に対応するデータが、UI600に表示されてもよい。

【0111】

UI600に表示されたデータは、最初、「サービス」タブ（例えば、サービスタブ604）に対応してもよい。いくつかの実施形態において、サービスタブ604は、選択されたリージョンにおいてブートストラップされるサービスに対応するあらゆる適切な情報を含んでもよい。非限定的な例として、サービスタブ604は、サービス識別子（コラム606）、バージョンセット識別子（コラム608）、変更タイプ（例えば、インフラストラクチャまたはアプリケーション／アーティファクト）およびステータス（例えば、成功／失敗）（コラム610）、消費されたケイパビリティ（コラム612）、生成されたケイパビリティ（コラム614）、および依存（コラム616）に対応するコラム606～616を含んでもよい。いくつかの実施形態において、コラム606は、選択されたリージョンにおいてブートストラップされる各々のサービスのための識別子を提供する。いくつかの実施形態において、サービス識別子（例えば、サービス識別子617）をクリックすることにより、UIは、エリア618内でUI600に提供されたデータが、選択されたサービス識別子と関連付けられたデータのみにフィルタリングされ得る。バージョンセット識別子619を選択することにより、UI600は、与えられたサービスと関連付けられたフロック設定および／またはアーティファクトのためのバージョン識別子と、バージョンセット識別子619とを（例えば、ポップアップウィンドウを介して）表示してもよい。

【0112】

コラム608は、サービスのための現在アクティブなバージョンセットを示すバージョ

10

20

30

40

50

ンセット識別子（例えば、バージョンセット識別子 619、「Golden」）を表示している。いくつかの実施形態において、ユーザが、同じサービスと関連付けられた 2 つのバージョンセットを見るおよび／または比較し得るように、現在のバージョンセット（例えば、Golden）を、サービスと関連付けられたあらゆるその他の選択されたバージョンセット識別子と比較するオプションが選択されてもよい。いくつかの実施形態において、アイコン（例えば、アイコン 628）が、各々のサービスのためのコラム 608 内に表示されてもよい。アイコンを選択することにより、バージョンセット情報を表示するためにポップアップまたはその他の適切なインターフェースが表示されてもよい。例えば、フロック設定（例えば、「Golden」バージョンセットにおいて使用される特定のバージョン）のための識別子が識別されてもよいおよび／または 1 つまたは複数のアーティファクト識別子および対応するバージョンが表示されてもよい。いくつかの実施形態において、アイコン 628 を選択することにより、あらゆる適切な数のバージョンセットに対応するバージョンセット情報が表示されてもよい。例えば、アイコン 628 を選択することにより、「Golden」バージョンセットと関連付けられた情報がその中で提供され得るポップアップまたはその他のユーザインターフェースが提供されてもよい。いくつかの実施形態において、サービスが、あらゆるその他のバージョンセット（例えば、Break Glassバージョンセット）に関連する場合、その他のバージョンセットに対応する情報が表示されてもよい。したがって、ユーザは、あらゆる適切なバージョンセットに対応する情報（例えば、フロック設定を識別するフロック識別子、フロック設定に対応するバージョン識別子、およびそのフロック設定によって参照されるあらゆる適切な数のアーティファクトに対応するデータを含む）へアクセスし得る。アーティファクトデータのインスタンスは、アーティファクトのための識別子（例えば、アプリケーション A）およびアーティファクトのバージョンに対応するバージョン識別子（例えば、バージョン 1）を含んでもよい。いくつかの実施形態において、バージョンセット情報は、サービスと関連付けられたリリースに対応するリリース識別子を含んでもよい。リリース

10

20

30

コラム 610 は、インフラストラクチャのプロビジョニングおよびアーティファクト（例えば、アプリケーション、構成ファイルなど）のデプロイメントに関連する情報を表示する。いくつかの実施形態において、識別子（例えば、インフラストラクチャおよび／またはアーティファクトに対応する特定のリリースを識別するリリース識別子）がコラム 610 に示されてもよい。ステータスは、リリースに対応するオペレーションの現在のステータスを示してもよい。

30

【0113】

コラム 612 は、どのケイパビリティが消費されたかを識別するインジケータを表示する。消費されたケイパビリティは、処理された対応するサービスが依存するケイパビリティである。言い換えれば、消費されたケイパビリティは、現在のサービスが依存する他のサービスに関連して発行されたケイパビリティを示す。

【0114】

コラム 614 は、どのケイパビリティをサービスが生成したかを識別するインジケータを表示する。生成されたケイパビリティは、与えられたサービスをブートストラップする経過において発行されたケイパビリティである。その他のサービスは、これらの生成されたケイパビリティのうちの 1 つまたは複数に依存してもよい。

40

【0115】

いくつかの実施形態において、ステータスを示すケイパビリティ識別子の隣にアイコンが現れることができる。例えば、アイコン 620 は、ケイパビリティがうまく消費／生成されたことを示してもよい。図 6 に示されているように、アイコン 620 は、ケイパビリティ 2 がうまく生成されたことを示している。同様に、アイコン 621 は、図 6 に表示されているように、ケイパビリティ（例えば、ケイパビリティ 5）がまだ生成されていないことを示している。アイコン 623 は、ケイパビリティが失敗した（例えば、生成されなかった、または消費されなかった）ことを示すために使用されることができる。アイコン 625 は、サービスが依存するケイパビリティが失敗したことを示すために使用されるこ

50

とができる。アイコン 6 2 7 は、まだ生成されていないケイパビリティへの依存を示すために使用されることができる。アイコン／ケイパビリティインジケータのいずれかを選択することにより、選択されたケイパビリティに対応するサービスを示すポップアップまたはその他のディスプレイが表示されてもよい。例えば、アイコン 6 2 7（またはアイコン 6 2 7 に対応するインジケータ）を選択することにより、そのケイパビリティ（この例では、サービス 2 および 3）に依存する全てのサービスを列挙するポップアップウィンドウが提供されてもよい。

【0116】

UI 6 0 0 は、エリア 6 3 0 を含んでもよい。エリア 6 3 0 は、リージョンと関連付けられたあらゆる適切な情報を表示するように構成されてもよい。例えば、エリア 6 3 0 は、リージョン構築のステータス（なし、構築、生成、休止、廃止予定など）、発行されたケイパビリティの数（例えば、3）、発行されることが予想される総ケイパビリティの数（例えば 1 2 3）を示してもよい。いくつかの実施形態において、エリア 6 3 0 は、ケイパビリティのステータスを示す進捗インジケータを表示してもよい。進捗インジケータの部分 6 3 2 は、うまく発行されたケイパビリティを示してもよい。進捗インジケータの部分 6 3 2 は、うまく発行されたケイパビリティを示してもよい。進捗インジケータの部分 6 3 4 は、まだ発行されていないケイパビリティを示してもよい。進捗インジケータの部分 6 3 6 は、失敗したケイパビリティを示してもよい。

10

【0117】

UI 6 0 0 は、エリア 6 3 8 を含んでもよい。エリア 6 3 8 は、エリア 6 1 8 に表示されたデータをフィルタリングするためのあらゆる適切な UI エlement を含んでもよい。ユーザは、識別子と関連付けられたフィルタのセットを定義してもよい。これらの所定のフィルタセットは、6 4 0 に示されているように、エリア 6 3 8 内に（例えば、識別子によって）表示されてもよい。いくつかの実施形態において、UI エlement 6 4 2 を選択することによってユーザ入力提供されてもよい。ユーザは、この Element を介してあらゆる適切なフィルタを入力してもよく、それによりエリア 6 1 8 に表示されたデータがフィルタリングされてもよい。

20

【0118】

イベントタブ 6 4 2 を選択することにより、ユーザが図 7 のユーザインターフェース（UI）7 0 0 へナビゲートされてもよい。

30

【0119】

図 7 は、少なくとも 1 つの実施形態による、リージョン構築と関連付けられたイベントに関連した情報を表示する例示的なユーザインターフェース 7 0 0 を示すブロック図である。UI 7 0 0 は、UI 6 0 0 を介した選択後のイベントタブのビューを表示してもよい。イベントタブ 7 0 2（図 6 のイベントタブ 6 4 2 に対応する）を選択することにより、データがエリア 7 0 4 に表示されてもよい。イベントは、ケイパビリティに対応してもよい。

【0120】

イベントタブ 7 0 2 は、イベント識別子（コラム 7 0 6）、イベントが受信された時間に対応する時間／タイムスタンプ（コラム 7 0 8）、ケイパビリティが発行されると仮定される順序を示すポジション（コラム 7 1 0）、イベントのタイプ（例えば、ケイパビリティ）を示すタイプ（コラム 7 1 2）、イベントが関連付けられた（例えば、サービスをブートストラップすることにより発行された）特定のサービス（フロック設定）を識別する名前（コラム 7 1 4）またはその他の識別子に対応するコラム 7 0 6～7 1 4 を含んでもよい。エリア 7 0 4 に表示されていないイベントと関連付けられたあらゆる適切な情報が、オプション 7 1 6 の選択に基づいて表示されてもよい。

40

【0121】

図 8 は、少なくとも 1 つの実施形態による、リージョン構築を実行するための（例えば、ブートストラップオペレーションをインクリメントに実行することによってリージョンを修正するための）例示的な方法 8 0 0 を示す。方法 8 0 0 は、図 1 のクラウドインフラ

50

ストラクチャオーケストレーションサービス 102（例えば、図 1 のマルチブロックオーケストレータ 106 などのオーケストレーションサービス）の 1 つまたは複数のコンポーネントによって実行されてもよい。コンピューティングデバイスの 1 つまたは複数のプロセッサによって実行されるとコンピューティングデバイスに方法 800 を実行させるコンピュータ可読命令を含むコンピュータ可読記憶媒体。方法 800 は、あらゆる適切な順序でまたは並行して実行されてもよい。方法 800 は、図 8 に示されているよりも多くのまたは少ないステップを含んでもよいことが認められるべきである。

【0122】

方法 800 は、802 において開始してもよく、ここで、1 つまたは複数のデータセンタに対応するリージョン内でブートストラップされる複数のサービスに対応する複数の構成ファイルが取得されてもよい。いくつかの実施形態において、複数のサービスの各々は、インフラストラクチャコンポーネントおよび対応するソフトウェアアーティファクトを含むリソースのそれぞれのセットと関連付けられてもよい。

【0123】

804 において、複数のサービスの間の 1 つまたは複数の依存が、複数の構成ファイルに少なくとも部分的に基づいて識別されてもよい。上記で説明されているように、静的解析が実行されてもよく、この静的解析は、1 つまたは複数の依存を識別するために構成ファイルをパースすることを含むことができる（他のサービスと関連付けられた変数および／または識別子の直接参照、他のサービスと関連付けられた変数および／または識別子の間接参照など）。

【0124】

806 において、複数のサービスをブートストラップするためのオペレーションが実行される順序が、1 つまたは複数の識別された依存に少なくとも部分的に基づいて決定されてもよい。上記で説明されているように、レコード、データ構造（例えば、図 5 の構築依存グラフ 314 などの有向グラフ）が、これらの依存の知識を維持するために生成されてもよい。

【0125】

808 において、プロビジョニングおよびデプロイメントマネジャ（例えば、図 1 の C I O S セントラル 108）が、決定された順序に従って複数のサービスをブートストラップするための対応するオペレーションを実行するようにインクリメントに命令されてもよい。いくつかの実施形態において、依存を管理するためにデータ構造が使用される場合など、データ構造は、リージョン構築に対応するあらゆる適切な数のサービスのブートストラップオペレーションをインクリメントに推進するために図 7 に関連して上記で説明されているのと類似の形式でトラバースされることができる。

【0126】

図 9 は、少なくとも 1 つの実施形態による、リージョン構築を実行する（例えば、リージョン内の 1 つまたは複数のデータセンタへリソースをブートストラップする）ためにバージョンセットを利用するための例示的な方法 900 を示す。方法 900 は、図 1 のクラウドインフラストラクチャオーケストレーションサービス 102 の 1 つまたは複数のコンポーネント（例えば、図 1 のマルチブロックオーケストレータ 106 などのオーケストレーションサービス）によって実行されてもよい。コンピューティングデバイスの 1 つまたは複数のプロセッサによって実行されるとコンピューティングデバイスに方法 900 を実行させるコンピュータ可読命令を含むコンピュータ可読記憶媒体。方法 900 は、あらゆる適切な順序でまたは並行して実行されてもよい。方法 900 は、図 900 に示されているよりも多いまたは少ないステップを含んでもよい。

【0127】

方法 900 は、902 において開始してもよく、ここで、複数のバージョンセット（例えば、図 6 のバージョンセット）が維持されてもよい。複数のサービスと関連付けられた複数の構成ファイルの構成ファイルのそれぞれのセットを識別する複数のバージョンセットの各々。

10

20

30

40

50

【0128】

904において、第1のバージョンセット（例えば、nightlyバージョンセット）が決定されてもよい。第1のバージョンセットは、複数の構成ファイルから構成ファイルの第1のセットを識別してもよい。

【0129】

906において、第1のバージョンセットによって識別された構成ファイルの第1のセットを検証するための検証過程が実行されてもよい。

【0130】

908において、構成ファイルの第2のセットを識別する第2のバージョンセット（例えば、goldenバージョンセット）が生成されてもよい。いくつかの実施形態において、検証過程をうまく通過したフロック構成ファイルを識別することに少なくとも部分的に基づいて構成ファイルの第1のセットから識別される構成ファイルの第2のセット。

【0131】

910において、リージョン構築は、第2のバージョンセットによって識別された構成ファイルの第2のセットを利用して実行されてもよい。

【0132】

例示的なクラウドサービスインフラストラクチャアーキテクチャ

上記のように、サービスとしてのインフラストラクチャ（IaaS）は、クラウドコンピューティングの1つの特定のタイプである。IaaSは、公共ネットワーク（例えば、インターネット）上で、仮想化されたコンピューティングリソースを提供するように構成されることができる。IaaSモデルにおいて、クラウドコンピューティングプロバイダは、インフラストラクチャコンポーネント（例えば、サーバ、記憶装置、ネットワークノード（例えば、ハードウェア）、デプロイメントソフトウェア、プラットフォーム仮想化（例えば、ハイパーバイザレイヤ）など）をホストすることができる。いくつかのケースにおいて、IaaSプロバイダは、これらのインフラストラクチャコンポーネントに伴うための様々なサービス（例えば、ビルディング、モニタリング、ロギング、ロードバランシングおよびクラスタリングなど）を供給してもよい。したがって、これらのサービスはポリシードリブンであり得るため、IaaSユーザは、アプリケーションアベイラビリティおよびパフォーマンスを維持するためにロードバランシングを推進するためのポリシーを実行することができてよい。

【0133】

いくつかの例において、IaaSカスタマーは、インターネットなどのワイドエリアネットワーク（WAN）を通じてリソースおよびサービスにアクセスしてもよく、アプリケーションスタックの残りのエレメントをインストールするためにクラウドプロバイダのサービスを使用することができる。例えば、ユーザは、IaaSプラットフォームにログインして、仮想マシン（VM）を生成し、各々のVMにオペレーティングシステム（OS）をインストールし、データベースなどのミドルウェアをデプロイし、ワークロードおよびバックアップのためのストレージバケットを生成し、さらには企業ソフトウェアをそのVMにインストールすることができる。次いで、カスタマーは、プロバイダのサービスを使用して、ネットワークトラフィックのバランスを取る、アプリケーション問題をトラブルシューティングする、パフォーマンスをモニタリングする、災害復旧を管理する、などを含む様々な機能を実行することができる。

【0134】

ほとんどのケースでは、クラウドコンピューティングモデルは、クラウドプロバイダの参加を必要とする。クラウドプロバイダは、IaaSを提供（オファー、レンタル、販売）することを専門とするサードパーティサービスであってもよいが、そうである必要はない。エンティティは、プライベートクラウドをデプロイすることを選択し、インフラストラクチャサービスの自己のプロバイダとなってもよい。

【0135】

いくつかの例において、IaaSデプロイメントは、準備されたアプリケーションサー

10

20

30

40

50

バなどに、新たなアプリケーションまたはアプリケーションの新たなバージョンを加えるプロセスである。これは、サーバを準備する（例えば、ライブラリ、デーモンなどをインストールする）プロセスも含んでもよい。これは、しばしば、ハイパーバイザレイヤ（例えば、サーバ、ストレージ、ネットワークハードウェア、および仮想化）の下で、クラウドプロバイダによって管理される。したがって、カスタマーは、（OS）、ミドルウェア、および／またはアプリケーションデプロイメント（例えば、セルフサービス仮想マシン（例えば、オンデマンドでスピンアップされることができるもの）において）をハンドリングするための責任があり得る。

【0136】

いくつかの例において、IaaSプロビジョニングは、使用するためのコンピュータまたは仮想ホストを獲得すること、さらには、それらに所要のライブラリまたはサービスをインストールすることを指す場合がある。ほとんどのケースでは、デプロイメントはプロビジョニングを含まず、プロビジョニングは最初に行われる必要はなくてもよい。

【0137】

いくつかのケースでは、IaaSプロビジョニングのための2つの異なる課題が存在する。第1に、あらゆるものが動作する前にインフラストラクチャの初期セットをプロビジョニングするという初期の課題がある。第2に、全てがプロビジョニングされると、既存のインフラストラクチャを発展させる（例えば、新たなサービスを追加する、サービスを変更する、サービスを除去するなど）という課題がある。いくつかのケースにおいて、これらの2つの課題は、インフラストラクチャの構成が宣言的に定義されることを可能にすることによって解決されてもよい。言い換えれば、インフラストラクチャ（例えば、どのコンポーネントが必要とされ、それらがどのように相互作用するか）は、1つまたは複数の構成ファイルによって定義されることができる。したがって、インフラストラクチャの全体的なトポロジー（例えば、どのリソースがどれに依存するか、それらが各々どのように協働するか）は、宣言的に記述されることができる。いくつかの例において、トポロジーが定義されると、構成ファイルに記述された異なるコンポーネントを生成および／または管理するワークフローが生成されることができる。

【0138】

いくつかの例において、インフラストラクチャは、多くの相互接続されたエレメントを有してもよい。例えば、コアネットワークとしても知られる、1つまたは複数の仮想プライベートクラウド（VPC）（例えば、構成可能および／または共有コンピューティングリソースの潜在的にオンデマンドのプール）が存在してもよい。いくつかの例において、ネットワークのインバウンドおよび／またはアウトバウンドトラフィックがどのようにセットアップされるかを定義するためにプロビジョニングされる1つまたは複数のインバウンド／アウトバウンドトラフィックグループおよび1つまたは複数の仮想マシン（VM）も存在してもよい。ロードバランサ、データベースなどのその他のインフラストラクチャエレメントもプロビジョニングされてもよい。より多くのインフラストラクチャエレメントが望まれるおよび／または追加されるので、インフラストラクチャは、インクリメントに発展し得る。

【0139】

いくつかの例において、様々な仮想コンピューティング環境を横断したインフラストラクチャコードのデプロイメントを可能にするために、連続的なデプロイメント技術が採用されてもよい。加えて、記述された技術は、これらの環境内でインフラストラクチャマネジメントを可能にすることができる。いくつかの例において、サービスチームは、1つまたは複数の、ただししばしば多くの、異なる生産環境へ（例えば、様々な異なる地理的ロケーションを横断して、時には全世界にわたって）デプロイされることが望まれるコードを書くことができる。しかしながら、いくつかの例において、コードがデプロイされるインフラストラクチャは、まずセットアップされなければならない。いくつかの例において、プロビジョニングは手動で行われることができる、リソースをプロビジョニングするためにプロビジョニングツールが利用されてもよい、および／またはインフラストラクチャ

10

20

30

40

50

がプロビジョニングされるとコードをデプロイするためにデプロイメントツールが利用されてもよい。

【0140】

図10は、少なくとも1つの実施形態による、IaaSアーキテクチャの例示的なパターンを示すブロック図1000である。サービスオペレータ1002は、仮想クラウドネットワーク(VCN)1006およびセキュアホストサブネット1008を含むことができるセキュアホストテナンシ1004に通信により結合されることができる。いくつかの例において、サービスオペレータ1002は、1つまたは複数のクライアントコンピューティングデバイスを使用してもよく、これは、Microsoft Windows Mobile (登録商標)などのソフトウェア、および／またはiOS、Windows Phone、Android、BlackBerry 8、Palm OSなどの様々なモバイルオペレーティングシステムを実行し、インターネット、Eメール、ショートメッセージサービス(SMS)、Blackberry (登録商標)、またはその他の通信プロトコルが有効化される、ポータブルハンドヘルドデバイス(例えば、iPhone (登録商標)、携帯電話、iPad (登録商標)、コンピューティングタブレット、パーソナルデジタルアシスタント(PDA))またはウェアラブルデバイス(例えば、Google Glass (登録商標)ヘッドマウントディスプレイ)であることができる。代替的に、クライアントコンピューティングデバイスは、例えば、Microsoft Windows (登録商標)、Apple Macintosh (登録商標)、および／またはLinux (登録商標)オペレーティングシステムの様々なバージョンを実行するパーソナルコンピュータおよび／またはラップトップコンピュータを含む汎用パーソナルコンピュータであることができる。クライアントコンピューティングデバイスは、例えば、Google Chrome OSなどの、限定することなく様々なGNU/Linuxオペレーティングシステムを含む、様々な市販のUNIX (登録商標)またはUNIXのようなオペレーティングシステムのいずれかを実行するワークステーションコンピュータであることができる。代替的に、または加えて、クライアントコンピューティングデバイスは、VCN1006および／またはインターネットへアクセスすることができるネットワーク上で通信することができる、シンクライアントコンピュータ、インターネット接続可能なゲーミングシステム(例えば、Kinect (登録商標)ジェスチャ入力デバイスを備えるまたは備えないMicrosoft Xboxゲーミングコンソール)、および／またはパーソナルメッセージングデバイスなどの、あらゆるその他の電子デバイスであってもよい。

10

20

30

【0141】

VCN1006は、SSH VCN1012に含まれたLPG1010を介してセキュアシェル(SSH)VCN1012に通信により結合されることができるローカルピアリングゲートウェイ(LPG)1010を含むことができる。SSH VCN1012は、SSHサブネット1014を含むことができ、SSH VCN1012は、コントロールプレーンVCN1016に含まれたLPG1010を介してコントロールプレーンVCN1016に通信により結合されることができる。また、SSH VCN1012は、LPG1010を介してデータプレーンVCN1018に通信により結合されることができる。コントロールプレーンVCN1016およびデータプレーンVCN1018は、IaaSプロバイダによって所有および／または運用されることができるサービステナンシ1019に含まれることができる。

40

【0142】

コントロールプレーンVCN1016は、ペリメータネットワーク(例えば、企業イントラネットと外部ネットワークとの間の企業ネットワークの部分)として機能するコントロールプレーン非武装地帯(DMZ)ティア1020を含むことができる。DMZベースサーバは、制限された責任を有してもよく、違反を閉じ込めておくことを助けてもよい。加えて、DMZティア1020は、1つまたは複数のロードバランサ(LB)サブネット1022、アプリサブネット1026を含むことができるコントロールプレーンアプリティア1024、データベース(DB)サブネット1030(例えば、フロントエンドDBサブネットおよび／またはバックエンドDBサブネット)を含むことができるコントロー

50

ルプレーンデータティア1028を含むことができる。コントロールプレーンDMZティア1020に含まれたLBサブネット1022は、コントロールプレーンアプリティア1024に含まれたアプリサブネット1026およびコントロールプレーンVCN1016に含まれることができるインターネットゲートウェイ1034に通信により結合されることができ、アプリサブネット1026は、コントロールプレーンデータティア1028に含まれたDBサブネット1030、サービスゲートウェイ1036、およびネットワークアドレストランスレーション(NAT)ゲートウェイ1038に通信により結合されることができる。コントロールプレーンVCN1016は、サービスゲートウェイ1036およびNATゲートウェイ1038を含むことができる。

【0143】

コントロールプレーンVCN1016は、アプリサブネット1026を含むことができるデータプレーンミラーアプリティア1040を含むことができる。データプレーンミラーアプリティア1040に含まれたアプリサブネット1026は、コンピュータインスタンス1044を実行することができる仮想ネットワークインターフェースコントローラ(VNIC)1042を含むことができる。コンピュータインスタンス1044は、データプレーンミラーアプリティア1040のアプリサブネット1026を、データプレーンアプリティア1046に含まれることができるアプリサブネット1026に通信により結合することができる。

【0144】

データプレーンVCN1018は、データプレーンアプリティア1046、データプレーンDMZティア1048、およびデータプレーンデータティア1050を含むことができる。データプレーンDMZティア1048は、データプレーンアプリティア1046のアプリサブネット1026およびデータプレーンVCN1018のインターネットゲートウェイ1034に通信により結合されることができるLBサブネット1022を含むことができる。アプリサブネット1026は、データプレーンVCN1018のサービスゲートウェイ1036およびデータプレーンVCN1018のNATゲートウェイ1038に通信により結合されることができる。データプレーンデータティア1050は、データプレーンアプリティア1046のアプリサブネット1026に通信により結合されることができるDBサブネット1030も含むことができる。

【0145】

コントロールプレーンVCN1016およびデータプレーンVCN1018のインターネットゲートウェイ1034は、公共インターネット1054に通信により結合されることができるメタデータマネジメントサービス1052に通信により結合されることができる。公共インターネット1054は、コントロールプレーンVCN1016およびデータプレーンVCN1018のNATゲートウェイ1038に通信により結合されることができる。コントロールプレーンVCN1016およびデータプレーンVCN1018のサービスゲートウェイ1036は、クラウドサービス1056に通信により結合されることができる。

【0146】

いくつかの例において、コントロールプレーンVCN1016およびデータプレーンVCN1018のサービスゲートウェイ1036は、公共インターネット1054を通過することなくクラウドサービス1056へアプリケーションプログラミングインターフェース(API)コールを行うことができる。サービスゲートウェイ1036からクラウドサービス1056へのAPIコールは、一方向であることができる。サービスゲートウェイ1036はクラウドサービス1056へAPIコールを行うことができ、クラウドサービス1056は、リクエストされたデータをサービスゲートウェイ1036へ送信することができる。しかしながら、クラウドサービス1056は、サービスゲートウェイ1036へのAPIコールを開始しなくてもよい。

【0147】

いくつかの例において、セキュアホストテナンシ1004は、さもなければ隔離され得

10

20

30

40

50

るサービステナンス１０１９に直接接続されることができる。セキュアホストサブネット１００８は、さもなければ隔離されるシステム上の双方向通信を可能にし得るＬＰＧ１０１０を通じてＳＳＨサブネット１０１４と通信することができる。セキュアホストサブネット１００８をＳＳＨサブネット１０１４に接続することは、セキュアホストサブネット１００８に、サービステナンス１０１９内の他のエンティティへのアクセスを与えてもよい。

【０１４８】

コントロールプレーンＶＣＮ１０１６は、サービステナンス１０１９のユーザが、所望のリソースをセットアップまたはさもなければプロビジョニングすることを可能にし得る。コントロールプレーンＶＣＮ１０１６においてプロビジョニングされる所望のリソースは、データプレーンＶＣＮ１０１８においてデプロイまたはさもなければ使用されてもよい。いくつかの例において、コントロールプレーンＶＣＮ１０１６は、データプレーンＶＣＮ１０１８から隔離されることができ、コントロールプレーンＶＣＮ１０１６のデータプレーンミラーアプリティア１０４０は、データプレーンミラーアプリティア１０４０およびデータプレーンアプリティア１０４６に含まれることができるＶＮＩＣ１０４２を介して、データプレーンＶＣＮ１０１８のデータプレーンアプリティア１０４６と通信することができる。

10

【０１４９】

いくつかの例において、システムのユーザ、またはカスタマーは、メタデータマネジメントサービス１０５２へリクエストを通信することができる公共インターネット１０５４を通じて、リクエストを行う、例えば、オペレーションを作成、読み出し、更新、または削除（ＣＲＵＤ）することができる。メタデータマネジメントサービス１０５２は、インターネットゲートウェイ１０３４を通じてコントロールプレーンＶＣＮ１０１６へリクエストを通信することができる。リクエストは、コントロールプレーンＤＭＺティア１０２０に含まれたＬＢサブネット１０２２によって受信されることができる。ＬＢサブネット１０２２は、リクエストが有効であることを決定してもよく、この決定に応答して、ＬＢサブネット１０２２は、コントロールプレーンアプリティア１０２４に含まれたアプリサブネット１０２６へリクエストを送信することができる。リクエストが検証され、公共インターネット１０５４へのコールを要求すると、公共インターネット１０５４へのコールがＮＡＴゲートウェイ１０３８へ送信されてもよく、ＮＡＴゲートウェイ１０３８は公共インターネット１０５４へのコールを行うことができる。リクエストによって記憶されることが望まれ得るメタデータは、ＤＢサブネット１０３０に記憶されることができる。

20

30

【０１５０】

いくつかの例において、データプレーンミラーアプリティア１０４０は、コントロールプレーンＶＣＮ１０１６とデータプレーンＶＣＮ１０１８との間の直接通信を容易にすることができる。例えば、構成に対する変更、更新、またはその他の適切な修正が、データプレーンＶＣＮ１０１８に含まれたリソースに適用されることが望まれる場合がある。ＶＮＩＣ１０４２を介して、コントロールプレーンＶＣＮ１０１６は、データプレーンＶＣＮ１０１８に含まれたリソースと直接通信することができ、これにより、それらに対して構成への変更、更新、またはその他の適切な修正を実行することができる。

40

いくつかの実施形態において、コントロールプレーンＶＣＮ１０１６およびデータプレーンＶＣＮ１０１８は、サービステナンス１０１９に含まれることができる。このケースでは、システムのユーザ、またはカスタマーは、コントロールプレーンＶＣＮ１０１６またはデータプレーンＶＣＮ１０１８のいずれも所有または操作しない場合がある。その代わりに、ＩaaSプロバイダが、コントロールプレーンＶＣＮ１０１６およびデータプレーンＶＣＮ１０１８を所有または操作してもよく、それら両方は、サービステナンス１０１９に含まれてもよい。この実施形態は、ユーザまたはカスタマーが他のユーザまたは他のカスタマーのリソースと相互作用することを防止し得るネットワークの隔離を可能にすることができる。また、この実施形態は、記憶のために、所望のレベルの脅威防止を有さない場合がある公共インターネット１０５４に依存する必要なく、システムのユーザまたは

50

カスタマーが、プライベートでデータベースを記憶することを可能にしてもよい。

【0151】

その他の実施形態において、コントロールプレーンVCN1016に含まれたLBサブネット1022は、サービスゲートウェイ1036から信号を受信するように構成されることができる。この実施形態において、コントロールプレーンVCN1016およびデータプレーンVCN1018は、公共インターネット1054をコールすることなくIaaSプロバイダのカスタマーによってコールされるように構成されてもよい。IaaSプロバイダのカスタマーは、この実施形態を望む場合がある。なぜならば、カスタマーが使用するデータベースが、IaaSプロバイダによって制御されてもよく、公共インターネット1054から隔離されてもよいサービステナンシ1019に記憶されてもよいからである。

10

【0152】

図11は、少なくとも1つの実施形態による、IaaSアーキテクチャの別の例示的なパターンを示すブロック図1100である。サービスオペレータ1102（例えば、図10のサービスオペレータ1002）は、仮想クラウドネットワーク（VCN）1106（例えば、図10のVCN1006）およびセキュアホストサブネット1108（例えば、図10のセキュアホストサブネット1008）を含むことができるセキュアホストテナンシ1104（例えば、図10のセキュアホストテナンシ1004）に通信により結合されることができる。VCN1106は、SSH VCN1112に含まれたLPG1010を介してセキュアシェル（SSH）VCN1112（例えば、図10のSSH VCN1012）に通信により結合されることができるローカルピアリングゲートウェイ（LPG）1110（例えば、図10のLPG1010）を含むことができる。SSH VCN1112は、SSHサブネット1114（例えば、図10のSSHサブネット1014）を含むことができ、SSH VCN1112は、コントロールプレーンVCN1116に含まれたLPG1110を介してコントロールプレーンVCN1116（例えば、図10のコントロールプレーンVCN1016）に通信により結合されることができる。コントロールプレーンVCN1116は、サービステナンシ1119（例えば、図10のサービステナンシ1019）に含まれることができ、データプレーンVCN1118（例えば、図10のデータプレーンVCN1018）は、システムの、ユーザまたはカスタマーによって所有または操作され得るカスタマーテナンシ1121に含まれることができる。

20

30

【0153】

コントロールプレーンVCN1116は、LBサブネット1122（例えば、図10のLBサブネット1022）を含むことができるコントロールプレーンDMZティア1120（例えば、図10のコントロールプレーンDMZティア1020）、アプリサブネット1126（例えば、図10のアプリサブネット1026）を含むことができるコントロールプレーンアプリティア1124（例えば、図10のコントロールプレーンアプリティア1024）、およびデータベース（DB）サブネット1130（例えば、図10のDBサブネット1030に類似のもの）を含むことができるコントロールプレーンデータティア1128（例えば、図10のコントロールプレーンデータティア1028）を含むことができる。コントロールプレーンDMZティア1120に含まれたLBサブネット1122は、コントロールプレーンアプリティア1124に含まれたアプリサブネット1126およびコントロールプレーンVCN1116に含まれることができるインターネットゲートウェイ1134（例えば、図10のインターネットゲートウェイ1034）に通信により結合されることができ、アプリサブネット1126は、コントロールプレーンデータティア1128に含まれたDBサブネット1130、サービスゲートウェイ1136（例えば、図10のサービスゲートウェイ1036）、およびネットワークアドレ스트ランスレーション（NAT）ゲートウェイ1138（例えば、図10のNATゲートウェイ1038）に通信により結合されることができる。コントロールプレーンVCN1116は、サービスゲートウェイ1136およびNATゲートウェイ1138を含むことができる。

40

【0154】

50

コントロールプレーンVCN1116は、アプリアブネット1126を含むことができるデータプレーンミラーアプリア1140（例えば、図10のデータプレーンミラーアプリア1040）を含むことができる。データプレーンミラーアプリア1140に含まれたアプリアブネット1126は、コンピュータインスタンス1144（例えば、図10のコンピュータインスタンス1044に類似のもの）を実行することができる仮想ネットワークインターフェースコントローラ（VNIC）1142（例えば、1042のVNIC）を含むことができる。コンピュータインスタンス1144は、データプレーンミラーアプリア1140に含まれたVNIC1142およびデータプレーンアプリア1146に含まれたVNIC1142を介して、データプレーンミラーアプリア1140のアプリアブネット1126と、データプレーンアプリア1146（例えば、図10のデータプレーンアプリア1046）に含まれることができるサブアブネット1126との間の通信を容易にすることができる。

10

【0155】

コントロールプレーンVCN1116に含まれたインターネットゲートウェイ1134は、公共インターネット1154（例えば、図10の公共インターネット1054）に通信により結合されることができるメタデータマネジメントサービス1152（例えば、図10のメタデータマネジメントサービス1052）に通信により結合されることができる。公共インターネット1154は、コントロールプレーンVCN1116に含まれたNATゲートウェイ1138に通信により結合されることができる。コントロールプレーンVCN1116に含まれたサービスゲートウェイ1136は、クラウドサービス1156（例えば、図10のクラウドサービス1056）に通信により結合されることができる。

20

【0156】

いくつかの例において、データプレーンVCN1118は、カスタマーテナンシ1121に含まれることができる。このケースでは、IaaSプロバイダは、各々のカスタマーのためのコントロールプレーンVCN1116を提供してもよく、IaaSプロバイダは、各々のカスタマーのために、サービステナンシ1119に含まれた固有のコンピュータインスタンス1144をセットアップしてもよい。各々のコンピュータインスタンス1144は、サービステナンシ1119に含まれたコントロールプレーンVCN1116と、カスタマーテナンシ1121に含まれたデータプレーンVCN1118との間の通信を可能にし得る。コンピュータインスタンス1144により、サービステナンシ1119に含まれたコントロールプレーンVCN1116においてプロビジョニングされるリソースが、カスタマーテナンシ1121に含まれたデータプレーンVCN1118においてデプロイまたはさもなければ使用されることができる。

30

【0157】

その他の例において、IaaSプロバイダのカスタマーは、カスタマーテナンシ1121に存在するデータベースを有してもよい。この例において、コントロールプレーンVCN1116は、アプリアブネット1126を含むことができるデータプレーンミラーアプリア1140を含むことができる。データプレーンミラーアプリア1140は、データプレーンVCN1118に存在することができるが、データプレーンミラーアプリア1140は、データプレーンVCN1118に存在しなくてもよい。すなわち、データプレーンミラーアプリア1140は、カスタマーテナンシ1121へのアクセスを有してもよいが、データプレーンミラーアプリア1140は、データプレーンVCN1118に存在しなくてもよいまたはIaaSプロバイダのカスタマーによって所有または操作されなくてもよい。データプレーンミラーアプリア1140は、データプレーンVCN1118に対してコールを行うように構成されてもよいが、コントロールプレーンVCN1116に含まれたあらゆるエンティティに対するコールを行うように構成されなくてもよい。カスタマーは、コントロールプレーンVCN1116においてプロビジョニングされるデータプレーンVCN1118におけるリソースをデプロイまたはさもなければ使用することを望む場合があるが、データプレーンミラーアプリア1140は、カスタマーの、所望のデプロイメントまたはリソースのその他の使用を容易にすること

40

50

ができる。

【0158】

いくつかの実施形態において、IaaSプロバイダのカスタマーは、データプレーンVCN1118にフィルタを適用することができる。この実施形態において、カスタマーは、どのデータプレーンVCN1118がアクセスすることができるかを決定することができる。カスタマーは、データプレーンVCN1118から公共インターネット1154へのアクセスを制限してもよい。IaaSプロバイダは、フィルタを適用することができなくてもよいまたはさもなければあらゆる外部のネットワークまたはデータベースへのデータプレーンVCN1118のアクセスを制御することができなくてもよい。カスタマーテナンシ1121に含まれたデータプレーンVCN1118にカスタマーによってフィルタおよび制御を適用することは、データプレーンVCN1118を他のカスタマーおよび公共インターネット1154から隔離することに役立つことができる。

10

【0159】

いくつかの実施形態において、クラウドサービス1156は、コントロールプレーンVCN1116またはデータプレーンVCN1118における、公共インターネット1154に存在しない場合があるサービスにアクセスするために、サービスゲートウェイ1136によってコールされることができる。クラウドサービス1156と、コントロールプレーンVCN1116またはデータプレーンVCN1118との間の接続は、ライブまたは連続的でなくてもよい。クラウドサービス1156は、IaaSプロバイダによって所有または操作される異なるネットワークに存在してもよい。クラウドサービス1156は、サービスゲートウェイ1136からのコールを受信するように構成されてもよく、公共インターネット1154からのコールを受信しないように構成されてもよい。いくつかのクラウドサービス1156は、他のクラウドサービス1156から隔離されてもよく、コントロールプレーンVCN1116は、コントロールプレーンVCN1116と同じリージョンにはない場合があるクラウドサービス1156から隔離されてもよい。例えば、コントロールプレーンVCN1116は、「リージョン1」に配置されてもよく、クラウドサービス「デプロイメント10」は、リージョン1および「リージョン2」に配置されてもよい。リージョン1に配置されたコントロールプレーンVCN1116に含まれたサービスゲートウェイ1136によってデプロイメント10に対するコールが行われると、コールは、リージョン1におけるデプロイメント10へ送信されてもよい。この例において、コントロールプレーンVCN1116、またはリージョン1におけるデプロイメント10は、リージョン2におけるデプロイメント10に通信により結合されなくてもよいまたはさもなければリージョン2におけるデプロイメント10と通信しなくてもよい。

20

30

【0160】

図12は、少なくとも1つの実施形態による、IaaSアーキテクチャの別の例示的なパターンを示すブロック図1200である。サービスオペレータ1202（例えば、図10のサービスオペレータ1002）は、仮想クラウドネットワーク（VCN）1206（例えば、図10のVCN1006）およびセキュアホストサブネット1208（例えば、図10のセキュアホストサブネット1008）を含むことができるセキュアホストテナンシ1204（例えば、図10のセキュアホストテナンシ1004）に通信により結合されることができる。VCN1206は、SSH VCN1212に含まれたLPG1210を介してSSH VCN1212（例えば、図10のSSH VCN1012）に通信により結合されることができるLPG1210（例えば、図10のLPG1010）を含むことができる。SSH VCN1212は、SSHサブネット1214（例えば、図10のSSHサブネット1014）を含むことができ、SSH VCN1212は、コントロールプレーンVCN1216に含まれたLPG1210を介してコントロールプレーンVCN1216（例えば、図10のコントロールプレーンVCN1016）に、およびデータプレーンVCN1218に含まれたLPG1210を介してデータプレーンVCN1218（例えば、図10のデータプレーン1018）に通信により結合されることができる。コントロールプレーンVCN1216およびデータプレーンVCN1218は、サービ

40

50

ステナンス 1 2 1 9（例えば、図 10 のサービステナンス 1 0 1 9）に含まれることができる。

【0161】

コントロールプレーン VCN 1 2 1 6 は、ロードバランサ (LB) サブネット 1 2 2 2（例えば、図 10 の LB サブネット 1 0 2 2）を含むことができるコントロールプレーン DMZ ティア 1 2 2 0（例えば、図 10 のコントロールプレーン DMZ ティア 1 0 2 0）、アプリサブネット 1 2 2 6（例えば、図 10 のアプリサブネット 1 0 2 6 と類似のもの）を含むことができるコントロールプレーンアプリティア 1 2 2 4（例えば、図 10 のコントロールプレーンアプリティア 1 0 2 4）、および DB サブネット 1 2 3 0 を含むことができるコントロールプレーンデータティア 1 2 2 8（例えば、図 10 のコントロールプレーンデータティア 1 0 2 8）を含むことができる。コントロールプレーン DMZ ティア 1 2 2 0 に含まれた LB サブネット 1 2 2 2 は、コントロールプレーンアプリティア 1 2 2 4 に含まれたアプリサブネット 1 2 2 6、およびコントロールプレーン VCN 1 2 1 6 に含まれることができるインターネットゲートウェイ 1 2 3 4（例えば、図 10 のインターネットゲートウェイ 1 0 3 4）に通信により結合されることができ、アプリサブネット 1 2 2 6 は、コントロールプレーンデータティア 1 2 2 8 に含まれた DB サブネット 1 2 3 0、サービスゲートウェイ 1 2 3 6（例えば、図 10 のサービスゲートウェイ）およびネットワークアドレストラংশション (NAT) ゲートウェイ 1 2 3 8（例えば、図 10 の NAT ゲートウェイ 1 0 3 8）に通信により結合されることができる。コントロールプレーン VCN 1 2 1 6 は、サービスゲートウェイ 1 2 3 6 および NAT ゲートウェイ 1 2 3 8 を含むことができる。

10

20

【0162】

データプレーン VCN 1 2 1 8 は、データプレーンアプリティア 1 2 4 6（例えば、図 10 のデータプレーンアプリティア 1 0 4 6）、データプレーン DMZ ティア 1 2 4 8（例えば、図 10 のデータプレーン DMZ ティア 1 0 4 8）、およびデータプレーンデータティア 1 2 5 0（例えば、図 10 のデータプレーンデータティア 1 0 5 0）を含むことができる。データプレーン DMZ ティア 1 2 4 8 は、LB サブネット 1 2 2 2 を含むことができ、この LB サブネット 1 2 2 2 は、データプレーンアプリティア 1 2 4 6 の信頼できるアプリサブネット 1 2 6 0 および信頼できないアプリサブネット 1 2 6 2、ならびにデータプレーン VCN 1 2 1 8 に含まれたインターネットゲートウェイ 1 2 3 4 に通信により結合されることができる。信頼できるアプリサブネット 1 2 6 0 は、データプレーン VCN 1 2 1 8 に含まれたサービスゲートウェイ 1 2 3 6、データプレーン VCN 1 2 1 8 に含まれた NAT ゲートウェイ 1 2 3 8、およびデータプレーンデータティア 1 2 5 0 に含まれた DB サブネット 1 2 3 0 に通信により結合されることができる。信頼できないアプリサブネット 1 2 6 2 は、データプレーン VCN 1 2 1 8 に含まれたサービスゲートウェイ 1 2 3 6 およびデータプレーンデータティア 1 2 5 0 に含まれた DB サブネット 1 2 3 0 に通信により結合されることができる。データプレーンデータティア 1 2 5 0 は、データプレーン VCN 1 2 1 8 に含まれたサービスゲートウェイ 1 2 3 6 に通信により結合されることができる。信頼できないアプリサブネット 1 2 6 2 を含むことができる。

30

【0163】

信頼できないアプリサブネット 1 2 6 2 は、テナント仮想マシン (VM) 1 2 6 6 (1) ~ (N) に通信により結合されることができる 1 つまたは複数のプライマリ VNIC 1 2 6 4 (1) ~ (N) を含むことができる。各々のテナント VM 1 2 6 6 (1) ~ (N) は、それぞれのカスタマーテナンス 1 2 7 0 (1) ~ (N) に含まれることができるそれぞれのコンテナエグレス VCN 1 2 6 8 (1) ~ (N) に含まれることができるそれぞれのアプリサブネット 1 2 6 7 (1) ~ (N) に通信により結合されることができる。それぞれのセカンダリ VNIC 1 2 7 2 (1) ~ (N) は、データプレーン VCN 1 2 1 8 に含まれた信頼できないアプリサブネット 1 2 6 2 と、コンテナエグレス VCN 1 2 6 8 (1) ~ (N) に含まれたアプリサブネットとの間の通信を容易にすることができる。各々のコンテナエグレス VCN 1 2 6 8 (1) ~ (N) は、公共インターネット 1 2 5 4 (例

40

50

えば、図 10 の公共インターネット 1054) に通信により結合されることができる NAT ゲートウェイ 1238 を含むことができる。

【0164】

コントロールプレーン VCN 1216 およびデータプレーン VCN 1218 に含まれたインターネットゲートウェイ 1234 は、公共インターネット 1254 に通信により結合されることができるメタデータマネジメントサービス 1252 (例えば、図 10 のメタデータマネジメントシステム 1052) に通信により結合されることができる。公共インターネット 1254 は、コントロールプレーン VCN 1216 およびデータプレーン VCN 1218 に含まれた NAT ゲートウェイ 1238 に通信により結合されることができる。コントロールプレーン VCN 1216 およびデータプレーン VCN 1218 に含まれたサービスゲートウェイ 1236 は、クラウドサービス 1256 に通信により結合されることができる。

10

【0165】

いくつかの実施形態において、データプレーン VCN 1218 は、カスタマーテナンシ 1270 と統合されることができる。この統合は、コードを実行するときにサポートを望む場合があるケースなどのいくつかのケースでは、IaaS プロバイダのカスタマーにとって有用であるまたは望ましい可能性がある。カスタマーは、破壊的であり得る実行するためのコードを提供する場合がある、または他のカスタマーリソースと通信する場合がある、またはさもなければ望ましくない効果を生じる場合がある。これに応答して、IaaS プロバイダは、カスタマーによって IaaS プロバイダに与えられたコードを実行すべきかどうかを決定してもよい。

20

【0166】

いくつかの例において、IaaS プロバイダのカスタマーは、IaaS プロバイダに一時的なネットワークアクセスを許可してもよく、データプレーンアプリティア 1246 にアタッチされる機能をリクエストしてもよい。機能を実行するためのコードは、VM 1266 (1) ~ (N) において実行されてもよく、コードは、データプレーン VCN 1218 における他のどこでも実行されないように構成されてもよい。各々の VM 1266 (1) ~ (N) は、1 つのカスタマーテナンシ 1270 に接続されてもよい。VM 1266 (1) ~ (N) に含まれたそれぞれのコンテナ 1271 (1) ~ (N) は、コードを実行するように構成されてもよい。このケースでは、デュアルアイソレーション (例えば、コードを実行するコンテナ 1271 (1) ~ (N) であって、コンテナ 1271 (1) ~ (N) は、少なくとも、信頼できないアプリサブネット 1262 に含まれた VM 1266 (1) ~ (N) に含まれてもよい) が存在することができ、これは、誤ったまたはさもなければ望ましくないコードが IaaS プロバイダのネットワークを損傷することまたは異なるカスタマーのネットワークを損傷することを防止するのに役立ち得る。コンテナ 1271 (1) ~ (N) は、カスタマーテナンシ 1270 に通信により結合されてもよく、カスタマーテナンシ 1270 からデータを送信または受信するように構成されてもよい。コンテナ 1271 (1) ~ (N) は、データプレーン VCN 1218 におけるいかなる他のエンティティからもデータを送信または受信するように構成されなくてもよい。コードの実行が完了すると、IaaS プロバイダは、コンテナ 1271 (1) ~ (N) をキルまたはさもなければ廃棄してもよい。

30

40

【0167】

いくつかの実施形態において、信頼できるアプリサブネット 1260 は、IaaS プロバイダによって所有または操作されてもよいコードを実行してもよい。この実施形態において、信頼できるアプリサブネット 1260 は、DB サブネット 1230 に通信により結合されてもよく、DB サブネット 1230 において CRUD オペレーションを実行するように構成されてもよい。信頼できないアプリサブネット 1262 は、DB サブネット 1230 に通信により結合されてもよいが、この実施形態において、信頼できないアプリサブネットは、DB サブネット 1230 において読み出しオペレーションを実行するように構成されてもよい。各々のカスタマーの VM 1266 (1) ~ (N) に含まれることができ

50

かつカスタマーからコードを実行してもよいコンテナ 1 2 7 1 (1) ~ (N) は、DB サブネット 1 2 3 0 に通信により結合されなくてもよい。

【0 1 6 8】

その他の実施形態において、コントロールプレーン VCN 1 2 1 6 およびデータプレーン VCN 1 2 1 8 は、直接に通信により結合されなくてもよい。この実施形態において、コントロールプレーン VCN 1 2 1 6 とデータプレーン VCN 1 2 1 8 との間には直接通信が存在しなくてもよい。しかしながら、通信は、少なくとも 1 つの方法を通じて間接的に行われることができる。LPG 1 2 1 0 が IaaS プロバイダによって確立されてもよく、この LPG 1 2 1 0 は、コントロールプレーン VCN 1 2 1 6 とデータプレーン VCN 1 2 1 8 との間の通信を容易にすることができる。別の例において、コントロールプレーン VCN 1 2 1 6 またはデータプレーン VCN 1 2 1 8 は、サービスゲートウェイ 1 2 3 6 を介してクラウドサービス 1 2 5 6 へコールを行うことができる。例えば、コントロールプレーン VCN 1 2 1 6 からクラウドサービス 1 2 5 6 へのコールは、データプレーン VCN 1 2 1 8 と通信することができるサービスのためのリクエストを含むことができる。

10

【0 1 6 9】

図 1 3 は、少なくとも 1 つの実施形態による、IaaS アーキテクチャの別の例示的なパターンを示すブロック図 1 3 0 0 である。サービスオペレータ 1 3 0 2 (例えば、図 1 0 のサービスオペレータ 1 0 0 2) は、仮想クラウドネットワーク (VCN) 1 3 0 6 (例えば、図 1 0 の VCN 1 0 0 6) およびセキュアホストサブネット 1 3 0 8 (例えば、図 1 0 のセキュアホストサブネット 1 0 0 8) を含むことができるセキュアホストテナンシ 1 3 0 4 (例えば、図 1 0 のセキュアホストテナンシ 1 0 0 4) に通信により結合されることができる。VCN 1 3 0 6 は、SSH VCN 1 3 1 2 に含まれた LPG 1 3 1 0 を介して SSH VCN 1 3 1 2 (例えば、図 1 0 の SSH VCN 1 0 1 2) に通信により結合されることができる LPG 1 3 1 0 (例えば、図 1 0 の LPG 1 0 1 0) を含むことができる。SSH VCN 1 3 1 2 は、SSH サブネット 1 3 1 4 (例えば、図 1 0 の SSH サブネット 1 0 1 4) を含むことができ、SSH VCN 1 3 1 2 は、コントロールプレーン VCN 1 3 1 6 に含まれた LPG 1 3 1 0 を介してコントロールプレーン VCN 1 3 1 6 (例えば、図 1 0 のコントロールプレーン VCN 1 0 1 6) に、およびデータプレーン VCN 1 3 1 8 に含まれた LPG 1 3 1 0 を介してデータプレーン VCN 1 3 1 8 (例えば、図 1 0 のデータプレーン VCN 1 0 1 8) に通信により結合されることができる。コントロールプレーン VCN 1 3 1 6 およびデータプレーン VCN 1 3 1 8 は、サービステナンシ 1 3 1 9 (例えば、図 1 0 のサービステナンシ 1 0 1 9) に含まれることができる。

20

30

【0 1 7 0】

コントロールプレーン VCN 1 3 1 6 は、LB サブネット 1 3 2 2 (例えば、図 1 0 の LB サブネット 1 0 2 2) を含むことができるコントロールプレーン DMZ ティア 1 3 2 0 (例えば、図 1 0 のコントロールプレーン DMZ ティア 1 0 2 0)、アプリサブネット 1 3 2 6 (例えば、図 1 0 のアプリサブネット 1 0 2 6) を含むことができるコントロールプレーンアプリティア 1 3 2 4 (例えば、図 1 0 のコントロールプレーンアプリティア 1 0 2 4)、および DB サブネット 1 3 3 0 (例えば、図 1 2 の DB サブネット 1 2 3 0) を含むことができるコントロールプレーンデータティア 1 3 2 8 (例えば、図 1 0 のコントロールプレーンデータティア 1 0 2 8) を含むことができる。コントロールプレーン DMZ ティア 1 3 2 0 に含まれた LB サブネット 1 3 2 2 は、コントロールプレーンアプリティア 1 3 2 4 に含まれたアプリサブネット 1 3 2 6 およびコントロールプレーン VCN 1 3 1 6 に含まれることができるインターネットゲートウェイ 1 3 3 4 (例えば、図 1 0 のインターネットゲートウェイ 1 0 3 4) に通信により結合されることができ、アプリサブネット 1 3 2 6 は、コントロールプレーンデータティア 1 3 2 8 に含まれた DB サブネット 1 3 3 0、サービスゲートウェイ 1 3 3 6 (例えば、図 1 0 のサービスゲートウェイ) およびネットワークアドレ스트ランスレーション (NAT) ゲートウェイ 1 3 3 8 (

40

50

例えば、図 10 の NAT ゲートウェイ 1038) に通信により結合されることができる。コントロールプレーン VCN 1316 は、サービスゲートウェイ 1336 および NAT ゲートウェイ 1338 を含むことができる。

【0171】

データプレーン VCN 1318 は、データプレーン アプリティア 1346 (例えば、図 10 のデータプレーン アプリティア 1046)、データプレーン DMZ ティア 1348 (例えば、図 10 のデータプレーン DMZ ティア 1048)、およびデータプレーン データティア 1350 (例えば、図 10 のデータプレーン データティア 1050) を含むことができる。データプレーン DMZ ティア 1348 は、データプレーン アプリティア 1346 の信頼できる アプリサブネット 1360 (例えば、図 12 の信頼できる アプリサブネット 1260) および信頼できない アプリサブネット 1362 (例えば、図 12 の信頼できない アプリサブネット 1262)、ならびにデータプレーン VCN 1318 に含まれたインターネットゲートウェイ 1334 に通信により結合されることができる LB サブネット 1322 を含むことができる。信頼できる アプリサブネット 1360 は、データプレーン VCN 1318 に含まれたサービスゲートウェイ 1336、データプレーン VCN 1318 に含まれた NAT ゲートウェイ 1338、およびデータプレーン データティア 1350 に含まれた DB サブネット 1330 に通信により結合されることができる。信頼できない アプリサブネット 1362 は、データプレーン VCN 1318 に含まれたサービスゲートウェイ 1336 およびデータプレーン データティア 1350 に含まれた DB サブネット 1330 に通信により結合されることができる。データプレーン データティア 1350 は、データプレーン VCN 1318 に含まれたサービスゲートウェイ 1336 に通信により結合されることができる DB サブネット 1330 を含むことができる。

10

20

【0172】

信頼できない アプリサブネット 1362 は、信頼できない アプリサブネット 1362 内に存在するテナント仮想マシン (VM) 1366 (1) ~ (N) に通信により結合されることができるプライマリ VNIC 1364 (1) ~ (N) を含むことができる。各々のテナント VM 1366 (1) ~ (N) は、それぞれのコンテナ 1367 (1) ~ (N) においてコードを実行することができ、コンテナエグレス VCN 1368 に含まれることができるデータプレーン アプリティア 1346 に含まれることができる アプリサブネット 1326 に通信により結合されることができる。それぞれのセカンダリ VNIC 1372 (1) ~ (N) は、データプレーン VCN 1318 に含まれた信頼できない アプリサブネット 1362 と、コンテナエグレス VCN 1368 に含まれた アプリサブネット との間の通信を容易にすることができる。コンテナエグレス VCN は、公共インターネット 1354 (例えば、図 10 の公共インターネット 1054) に通信により結合されることができる NAT ゲートウェイ 1338 を含むことができる。

30

【0173】

コントロールプレーン VCN 1316 およびデータプレーン VCN 1318 に含まれたインターネットゲートウェイ 1334 は、公共インターネット 1354 に通信により結合されることができるメタデータマネジメントサービス 1352 (例えば、図 10 のメタデータマネジメントサービス 1052) に通信により結合されることができる。公共インターネット 1354 は、コントロールプレーン VCN 1316 およびデータプレーン VCN 1318 に含まれた NAT ゲートウェイ 1338 に通信により結合されることができる。コントロールプレーン VCN 1316 およびデータプレーン VCN 1318 に含まれたサービスゲートウェイ 1336 は、クラウドサービス 1356 に通信により結合されることができる。

40

【0174】

いくつかの例において、図 13 のブロック図 1300 のアーキテクチャによって示されたパターンは、図 12 のブロック図 1200 のアーキテクチャによって示されたパターンに対する例外であると考えられてもよく、IaaS プロバイダがカスタマー (例えば、切断されたリージョン) と直接通信することができない場合に、IaaS プロバイダのカス

50

タマーにとって望ましい場合がある。各々のカスタマーのためにVM1366(1)～(N)に含まれたそれぞれのコンテナ1367(1)～(N)は、カスタマーによってリアルタイムでアクセスされることができる。コンテナ1367(1)～(N)は、コンテナエグレスVCN1368に含まれることができるデータプレーンアプリティア1346のアプリサブネット1326に含まれたそれぞれのセカンダリVNIC1372(1)～(N)へコールを行うように構成されてもよい。セカンダリVNIC1372(1)～(N)は、公共インターネット1354へコールを送信してもよいNATゲートウェイ1338へコールを送信することができる。この例において、カスタマーによってリアルタイムでアクセスされることができるコンテナ1367(1)～(N)は、コントロールプレーンVCN1316から隔離されることができ、データプレーンVCN1318に含まれた他のエンティティから隔離されることができる。コンテナ1367(1)～(N)は、他のカスタマーからのリソースから隔離されてもよい。

10

【0175】

その他の例において、カスタマーは、クラウドサービス1356をコールするためにコンテナ1367(1)～(N)を使用することができる。この例において、カスタマーは、クラウドサービス1356からサービスをリクエストするコードをコンテナ1367(1)～(N)において実行してもよい。コンテナ1367(1)～(N)は、このリクエストをセカンダリVNIC1372(1)～(N)へ送信することができ、セカンダリVNICは、リクエストをNATゲートウェイへ送信することができ、NATゲートウェイは、リクエストを公共インターネット1354へ送信することができる。公共インターネット1354は、インターネットゲートウェイ1334を介して、コントロールプレーンVCN1316に含まれたLBサブネット1322へリクエストを送信することができる。リクエストが有効であるという決定に回答して、LBサブネットは、リクエストをアプリサブネット1326へ送信することができ、アプリサブネット1326は、サービスゲートウェイ1336を介してリクエストをクラウドサービス1356へ送信することができる。

20

【0176】

図示されたIaaSアーキテクチャ1000、1100、1200、1300は、示されたもの以外のコンポーネントを有してもよいことが認められるべきである。さらに、図示された実施形態は、開示の実施形態を組み込んでもよいクラウドインフラストラクチャシステムのいくつかの例でしかない。いくつかの他の実施形態において、IaaSシステムは、図示されたよりも多いまたは少ないコンポーネントを有してもよい、2つ以上のコンポーネントを組み合わせてもよい、またはコンポーネントの異なる構成または配置を有してもよい。

30

【0177】

ある実施形態において、本明細書において説明されたIaaSシステムは、セルフサービス、サブスクリプションベース、弾力的にスケーラブル、信頼できる、非常に利用可能な、およびセキュアな形式でカスタマーに供給される、アプリケーション、ミドルウェア、およびデータベースサービスオフリングのスイートを含んでもよい。このようなIaaSシステムの例は、本譲受人によって提供されるOracle Cloud Infrastructure(OCI)である。

40

【0178】

図14は、様々な実施形態が実施され得る、例示的なコンピュータシステム1400を示す。システム1400は、上記で説明されたコンピュータシステムのいずれかを実施するために使用されてもよい。図示されているように、コンピュータシステム1400は、バスサブシステム1402を介して多数の周辺サブシステムと通信する処理ユニット1404を含む。これらの周辺サブシステムは、処理加速ユニット1406、I/Oサブシステム1408、記憶サブシステム1418、および通信サブシステム1424を含んでもよい。記憶サブシステム1418は、有形コンピュータ可読記憶媒体1422およびシステムメモリ1410を含む。

50

【0179】

バスサブシステム1402は、コンピュータシステム1400の様々なコンポーネントおよびサブシステムに意図したように互いに通信させるための機構を提供する。バスサブシステム1402は1つのバスとして概略的に示されているが、バスサブシステムの代替的な実施形態は、多数のバスを利用してもよい。バスサブシステム1402は、様々なバスアーキテクチャのうち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、およびローカルバスを含むいくつかのタイプのバス構造のうちのいずれかであってもよい。例えば、このようなアーキテクチャは、業界標準アーキテクチャ（ISA）バス、マイクロチャネルアーキテクチャ（MCA）バス、エンハンスドISA（EISA）バス、ビデオエレクトロニクススタンダードアソシエーション（VESA）ローカルバス、および周辺コンポーネントインターコネクト（PCI）バスを含んでもよく、これらは、IEEE P 1386.1規格に従って製造されたメザニンバスとして実施されることができる。

10

【0180】

1つまたは複数の集積回路（例えば、従来のマイクロプロセッサまたはマイクロコントローラ）として実施されることができる処理ユニット1404は、コンピュータシステム1400のオペレーションを制御する。1つまたは複数のプロセッサは、処理ユニット1404に含まれてもよい。これらのプロセッサは、シングルコアまたはマルチコアプロセッサを含んでもよい。ある実施形態において、処理ユニット1404は、各々の処理ユニットに含まれたシングルコアまたはマルチコアプロセッサを備える1つまたは複数の独立した処理ユニット1432および／または1434として実施されてもよい。その他の実施形態において、処理ユニット1404は、2つのデュアルコアプロセッサを1つのチップに統合することによって形成されるクアッドコア処理ユニットとして実施されてもよい。

20

【0181】

様々な実施形態において、処理ユニット1404は、プログラムコードに応答して様々なプログラムを実行することができ、多数の同時に実行するプログラムまたは処理を維持することができる。あらゆる与えられた時間において、実行されるプログラムコードのいくつかまたは全ては、プロセッサ1404および／または記憶サブシステム1418に存在することができる。適切なプログラミングにより、プロセッサ1404は、上記で説明された様々な機能を提供することができる。コンピュータシステム1400は、加えて、デジタル信号プロセッサ（DSP）、専用プロセッサ、および／または同様のものを含むことができる処理加速ユニット1406を含んでもよい。

30

【0182】

I/Oサブシステム1408は、ユーザインターフェース入力デバイスおよびユーザインターフェース出力デバイスを含んでもよい。ユーザインターフェース入力デバイスは、キーボード、マウスまたはトラックボールなどのポインティングデバイス、ディスプレイに組み込まれたタッチパッドまたはタッチスクリーン、スクロールホイール、クリックホイール、ダイヤル、ボタン、スイッチ、キーパッド、音声命令認識システムを備えるオーディオ入力デバイス、マイクロフォン、およびその他のタイプの入力デバイスを含んでもよい。ユーザインターフェース入力デバイスは、例えば、ジェスチャおよび音声指示を使用するナチュラルユーザインターフェースを通じて、ユーザが、Microsoft Xbox（登録商標）360ゲームコントローラなどの入力デバイスを制御しかつこれと相互作用することを可能にする、Microsoft Kinect（登録商標）モーションセンサなどのモーション感知および／またはジェスチャ認識デバイスを含んでもよい。ユーザインターフェース入力デバイスは、ユーザから目の動き（例えば、写真を撮影するときおよび／またはメニューを選ぶときの「まばたき」）を検出し、アイジェスチャを入力デバイス（例えば、Google Glass（登録商標））への入力として変換する、Google Glass（登録商標）まばたき検出器などのアイジェスチャ認識デバイスも含んでもよい。加えて、ユーザインターフェース入力デバイスは、音声命令を通じて、ユーザが音声認識システム（例えば、S

40

50

iri（登録商標）ナビゲータ）と相互作用することを可能にする音声認識センシングデバイスを含んでもよい。

【0183】

ユーザインターフェース入力デバイスは、これらに限定されることなく、3次元（3D）マウス、ジョイスティックまたはポインティングスティック、ゲームパッドおよびグラフィックタブレット、およびオーディオ／ビジュアルデバイス、例えば、スピーカ、デジタルカメラ、デジタルカムコーダ、ポータブルメディアプレーヤ、ウェブカム、イメージスキャナ、指紋スキャナ、バーコードリーダ3Dスキャナ、3Dプリンタ、レーザ距離計、視線検出デバイスも含んでもよい。加えて、ユーザインターフェース入力デバイスは、例えば、医用画像入力デバイス、例えば、コンピュータ断層撮影、磁気共鳴イメージング、陽電子放出断層撮影、医用超音波検査デバイスを含んでもよい。ユーザインターフェース入力デバイスは、例えば、オーディオ入力デバイス、例えば、MIDIキーボード、デジタル楽器なども含んでもよい。

10

【0184】

ユーザインターフェース出力デバイスは、ディスプレイサブシステム、インジケータライト、または非視覚的ディスプレイ、例えば、オーディオ出力デバイスなどを含んでもよい。ディスプレイサブシステムは、陰極線管（CRT）、フラットパネルデバイス、例えば、液晶ディスプレイ（LCD）またはプラズマディスプレイを使用するフラットパネルデバイス、プロジェクションデバイス、タッチスクリーンなどであってもよい。一般的に、「出力デバイス」という用語の使用は、コンピュータシステム1400からユーザまたは他のコンピュータへ情報を出力するための全ての可能なタイプのデバイスおよび機構を含むことが意図されている。例えば、ユーザインターフェース出力デバイスは、これらに限定されることなく、テキスト、グラフィックおよびオーディオ／ビデオ情報を視覚的に伝達する様々なディスプレイデバイス、例えば、モニタ、プリンタ、スピーカ、ヘッドフォン、自動車ナビゲーションシステム、プロッタ、音声出力デバイス、およびモデムを含んでもよい。

20

【0185】

コンピュータシステム1400は、本開示において説明された実施形態の機能を提供するソフトウェアおよびデータ構造を記憶するための有形非一時的コンピュータ可読記憶媒体を提供する記憶サブシステム1418を含んでもよい。ソフトウェアは、処理ユニット1404の1つまたは複数のコアまたはプロセッサによって実行されると上記で説明された機能を提供する、プログラム、コードモジュール、命令、スクリプトなどを含むことができる。記憶サブシステム1418は、本開示に従って使用されるデータを記憶するためのリポジトリも提供してもよい。

30

【0186】

図14における例に示されているように、記憶サブシステム1418は、システムメモリ1410、コンピュータ可読記憶媒体1422、およびコンピュータ可読記憶媒体リーダ1420を含む様々なコンポーネントを含むことができる。システムメモリ1410は、処理ユニット1404によってロード可能および実行可能なプログラム命令を記憶してもよい。システムメモリ1410は、命令の実行中に使用されるデータおよび／またはプログラム命令の実行中に生成されるデータを記憶してもよい。様々な異なる種類のプログラムが、クライアントアプリケーション、ウェブブラウザ、ミッドティアアプリケーション、リレーショナルデータベースマネジメントシステム（RDBMS）、仮想マシン、コンテナなどを含むがこれらに限定されないシステムメモリ1410へロードされてもよい。

40

【0187】

システムメモリ1410は、オペレーティングシステム1416を記憶してもよい。オペレーティングシステム1416の例は、Microsoft Window（登録商標）、Apple Macintosh（登録商標）、および／またはLinuxオペレーティングシステムの様々なバージョン、様々な市販のUNIX（登録商標）またはUNIXのようなオペレーティングシス

50

テム（制限されることなく様々なGNU/Linuxオペレーティングシステム、Google Chrome（登録商標）OSなどを含む）および／またはモバイルオペレーティングシステム、例えば、iOS、Windows（登録商標）Phone、Android（登録商標）OS、Black Berry（登録商標）OS、およびPalm（登録商標）OSオペレーティングシステムを含んでもよい。コンピュータシステム1400が1つまたは複数の仮想マシンを実行するある実装形態において、ゲストオペレーティングシステム（GOS）を伴う仮想マシンが、システムメモリ1410にロードされ、処理ユニット1404の1つまたは複数のプロセッサまたはコアによって実行されてもよい。

【0188】

システムメモリ1410は、コンピュータシステム1400のタイプに依存する異なる構成で提供されることができる。例えば、システムメモリ1410は、揮発性メモリ（例えば、ランダムアクセスメモリ（RAM））および／または不揮発性メモリ（例えば、読み出し専用メモリ（ROM）、フラッシュメモリなど）であってもよい。スタティックランダムアクセスメモリ（SRAM）、ダイナミックランダムアクセスメモリ（DRAM）およびその他を含む、異なるタイプのRAM構成が提供されてもよい。いくつかの実装形態において、システムメモリ1410は、スタートアップ中など、コンピュータシステム1400内のエレメント間で情報を伝達するのに役立つ基本ルーチンを含むベーシック入力／出力システム（BIOS）を含んでもよい。

【0189】

コンピュータ可読記憶媒体1422は、コンピュータシステム1400の処理ユニット1404によって実行可能な命令を含む、コンピュータシステム1400によって自要するためのコンピュータ可読情報を一時的におよび／またはより永久に含み、記憶するための、リモート、ローカル、固定、ならびに／またはリムーバブルな記憶デバイスおよび記憶媒体を表してもよい。

【0190】

コンピュータ可読記憶媒体1422は、情報の記憶および／または送信のためのあらゆる方法または技術において実施される、揮発性および不揮発性、リムーバブルおよび非リムーバブル媒体などの、ただしこれらに限定されない、記憶媒体および通信媒体を含む、当該技術分野において知られるまたは使用されるあらゆる適切な媒体を含むことができる。これは、有形コンピュータ可読記憶媒体、例えば、RAM、ROM、電子的に消去可能なプログラマブルROM（EEPROM）、フラッシュメモリまたはその他のメモリ技術、CD-ROM、デジタルバーサタイルディスク（DVD）、またはその他の光ストレージ、磁気カセット、磁気テープ、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、またはその他の有形コンピュータ可読媒体を含むことができる。

【0191】

例えば、コンピュータ可読記憶媒体1422は、非リムーバブル、不揮発性磁気媒体から読み出すまたはそれに書き込むハードディスクドライブ、リムーバブル、不揮発性磁気ディスクから読み出すまたはそれに書き込む磁気ディスクドライブ、CD-ROM、DVD、およびBlu-Ray（登録商標）ディスクなどのリムーバブル、不揮発性光ディスクまたはその他の光媒体から読み出すまたはそれに書き込む光ディスクドライブを含んでもよい。コンピュータ可読記憶媒体1422は、これらに限定されないが、Zip（登録商標）ドライブ、フラッシュメモリカード、ユニバーサルシリアルバス（USB）フラッシュドライブ、セキュアデジタル（SD）カード、DVDディスク、デジタルビデオテープなどを含んでもよい。コンピュータ可読記憶媒体1422は、フラッシュメモリベースSSD、企業フラッシュドライブ、ソリッドステートROMなどの不揮発性メモリに基づくソリッドステートドライブ（SSD）、ソリッドステートRAM、ダイナミックRAM、スタティックRAM、DRAMベースSSD、磁気抵抗RAM（MRAM）SSADなどの揮発性メモリに基づくSSD、DRAMおよびフラッシュメモリベースSSDの組合せを使用するハイブリッドSSDも含んでもよい。ディスクドライブおよびそれらの関連するコンピュータ可読媒体は、コンピュータシステム1400のためのコンピュータ可読命令、

10

20

30

40

50

データ構造、プログラムモジュール、およびその他のデータの不揮発性ストレージを提供してもよい。

【0192】

処理ユニット1404の1つまたは複数のプロセッサまたはコアによって実行可能な機械可読命令は、非一時的コンピュータ可読記憶媒体に記憶されてもよい。非一時的コンピュータ可読記憶媒体は、揮発性メモリ記憶装置および／または不揮発性記憶装置を含む物理的に有形のメモリまたは記憶装置を含むことができる。非一時的コンピュータ可読記憶媒体の例は、磁気記憶媒体（例えば、ディスクまたはテープ）、光記憶媒体（例えば、DVD、CD）、様々なタイプのRAM、ROM、またはフラッシュメモリ、ハードドライブ、フロッピードライブ、取り外し可能なメモリドライブ（例えば、USBドライブ）、またはその他のタイプの記憶装置を含む。

10

【0193】

通信サブシステム1424は、他のコンピュータシステムおよびネットワークへのインターフェースを提供する。通信サブシステム1424は、他のシステムからデータを受信し、コンピュータシステム1400から他のシステムへデータを送信するためのインターフェースとして働く。例えば、通信サブシステム1424は、コンピュータシステム1400がインターネットを介して1つまたは複数のデバイスに接続されることを可能にし得る。いくつかの実施形態において、通信サブシステム1424は、（例えば、携帯電話技術、3G、4GまたはEDGE（エンハンスドデータレートフォアグローバルエボリューション））などの、アドバンストデータネットワーク技術、WiFi（IEEE802.11ファミリー規格、またはその他のモバイル通信技術、またはそれらのあらゆる組合せ）、全地球測位システム（GPS）レシーバコンポーネント、および／またはその他のコンポーネントを使用して）無線音声および／またはデータネットワークにアクセスするための無線周波数（RF）トランシーバコンポーネントを含むことができる。いくつかの実施形態において、通信サブシステム1424は、無線インターフェースに加えてまたはその代わりに、有線ネットワーク接続（例えば、イーサネット）を提供することができる。

20

【0194】

いくつかの実施形態において、通信サブシステム1424は、コンピュータシステム1400を使用してもよい1つまたは複数のユーザの代わりに、構造化されたおよび／または構造化されていないデータフィード1426、イベントストリーム1428、イベントアップデート1430などの形式で入力通信を受信してもよい。

30

【0195】

例えば、通信サブシステム1424は、ソーシャルネットワークおよび／またはその他の通信サービスのユーザからリアルタイムでデータフィード1426、例えばTwitter（登録商標）フィード、Facebook（登録商標）アップデート、リッチサイトサマリ（RSS）フィードなどのウェブフィード、および／または1つまたは複数のサードパーティ情報ソースからのリアルタイムアップデートなどを受信するように構成されてもよい。

【0196】

加えて、通信サブシステム1424は、連続的であってもよいまたは本質的に明確な終わりがなく無制限であってもよい、リアルタイムイベントのイベントストリーム1428および／またはイベントアップデート1430を含んでもよい、連続的なデータストリームの形式のデータを受信するように構成されてもよい。連続的なデータを生成するアプリケーションの例は、例えば、センサデータアプリケーション、ファイナンシャルティッカー、ネットワークパフォーマンス測定ツール（例えば、ネットワークモニタリングおよびトラフィック管理アプリケーション）、クリックストリーム分析ツール、自動車トラフィックモニタリングなどを含んでもよい。

40

【0197】

通信サブシステム1424は、コンピュータシステム1400に結合された1つまたは複数のストリーミングデータソースコンピュータと通信してもよい1つまたは複数のデータベースへ、構造化されたおよび／または構造化されていないデータフィード1426、

50

イベントストリーム 1428、イベントアップデート 1430 などを出力するように構成されてもよい。

【0198】

コンピュータシステム 1400 は、ハンドヘルドポータブルデバイス（例えば、iPhone（登録商標）携帯電話、iPad（登録商標）コンピューティングタブレット、PDA）、ウェアラブルデバイス（例えば、Google Glass（登録商標）頭部装着型ディスプレイ）、PC、ワークステーション、メインフレーム、キオスク、サーバラック、またはあらゆるその他のデータ処理システムを含む、様々なタイプのうちの 1 つであることができる。

【0199】

コンピュータおよびネットワークの絶え間なく変化する性質により、図示されたコンピュータシステム 1400 の説明は、特定の例としてのみ意図されている。図示されたシステムよりも多いまたは少ないコンポーネントを有する多くのその他の構成が可能である。例えば、カスタマイズされたハードウェアが使用されてもよいおよび／または特定のエレメントがハードウェア、ファームウェア、ソフトウェア（タブレットを含む）、または組合せにおいて実施されてもよい。さらに、ネットワーク入力／出力デバイスなどの他のコンピューティングデバイスへの接続が採用されてもよい。本明細書に提供された開示および教示に基づいて、当業者は、様々な実施形態を実施するための他のやり方および／または方法を認識するであろう。

【0200】

特定の実施形態が説明されているが、様々な修正、変更、代替的な構成、および均等物も開示の範囲に含まれる。実施形態は、ある特定のデータ処理環境内のオペレーションに制限されず、複数のデータ処理環境内で自由に操作される。実施形態は、特定の連続のトランザクションおよびステップを用いて説明されているが、本開示の範囲は、説明された連続のトランザクションおよびステップに限定されないことが、当業者に明らかになるべきである。上述の実施形態の様々な特徴および態様は、個々にまたは組み合わせて使用されてもよい。

【0201】

さらに、ハードウェアおよびソフトウェアの特定の組合せを用いて実施形態が説明されているが、ハードウェアおよびソフトウェアの他の組合せも本開示の範囲内であることが認識されるべきである。実施形態は、ハードウェアにおいてのみ、ソフトウェアにおいてのみ、またはそれらの組合せを用いて、実施されてもよい。本明細書に説明された様々なプロセスは、同じプロセッサまたはあらゆる組合せにおける異なるプロセッサにおいて実施されることができる。したがって、コンポーネントまたはモジュールが、あるオペレーションを実行するように構成されていると説明されている場合、このような構成は、例えば、オペレーションを実行するために電子回路を設計することによって、オペレーションを実行するためにプログラマブル電子回路（マイクロプロセッサなど）をプログラムすることによって、またはそれらのあらゆる組合せで達成されることができる。プロセッサは、プロセス間通信のための従来技術を含むがこれに限定されない様々な技術を用いて通信することができ、プロセスの異なる対は異なる技術を用いてもよい、またはプロセスの同じ対は異なる時点で異なる技術を用いてもよい。

【0202】

実施形態は、プロセッサによって実行されるとプロセッサに開示において説明された方法のうちのいずれかを実行させるコンピュータプログラム／命令を含むコンピュータプログラム製品を使用することによって実施されてもよい。

【0203】

したがって、明細書および図面は、制限的な意味ではなく例示的な意味で見なされるべきである。しかしながら、特許請求の範囲に示されているより広い趣旨および範囲から逸脱することなくそれらに対して追加、削減、削除、およびその他の修正および変更が加えられ得ることが明らかになるであろう。したがって、特定の開示実施形態が説明されてい

10

20

30

40

50

るが、これらは限定的であることは意図されていない。様々な修正および均等物が、以下の請求項の範囲に含まれる。

【0204】

開示された実施形態を説明する文脈において（特に、以下の請求項の文脈において）「a」、「an」および「the」という用語ならびに類似の指示対象の使用は、本明細書に別段の指示がない限りまたは文脈によって明らかに矛盾しない限り、単数形および複数形の両方をカバーすると解釈されるべきである。「含む（comprising）」、「有する（having）」、「含む（including）」および「含む（containing）」という用語は、別段の注意がない限り無制限の用語（すなわち、「含むが、限定されない」を意味する）であると解釈されるべきである。「接続される」という用語は、介在する何かが存在するとしても、部分的にまたは全体的に内部に含まれる、取り付けられる、または接合されると解釈されるべきである。本明細書における値の範囲の列挙は、単に、本明細書において別段の指示がない限り、その範囲内に含まれる各々の別個の値にそれぞれ言及する簡潔な方法として働くことが意図されており、各々の別個の値は、本明細書において個々に列挙されたものとして明細書に組み込まれる。本明細書において説明された全ての方法は、本明細書において別段の指示がない限りまたは文脈によって明らかに矛盾しない限り、あらゆる適切な順序で実行されることができる。本明細書において提供されたあらゆるおよび全ての例、または例示的な言語（例えば、「など」）の使用は、単に実施形態をより明らかにすることが意図されており、そうでないことが請求されない限り開示の範囲に対する制限を加えない。明細書における言語は、あらゆる請求されていない要素を、開示の実行にとって必須であるものとして示すものとして解釈されるべきではない。

【0205】

「X、YまたはZのうちの少なくとも」という表現などの離接的言語は、特にそうでないことが述べられていない限り、アイテム、用語などが、X、YまたはZのいずれか、またはそれらのあらゆる組合せ（例えば、X、Yおよび／またはZ）であってもよいことを表すために一般的に使用されるものとして文脈の中で理解されることが意図されている。したがって、このような離接的言語は、概して、ある実施形態がXのうちの少なくとも1つ、Yのうちの少なくとも1つ、またはZのうちの少なくとも1つが各々存在することを要求することを暗示することは意図されておらず、また暗示すべきではない。

【0206】

開示を実行するために知られた最良の形態を含む、本開示の好ましい実施形態が本明細書に説明されている。これらの好ましい実施形態の変化態様は、前記説明を読むことにより、当業者に明らかになり得る。当業者は、このような変化態様を必要に応じて採用することができるべきであり、開示は、本明細書において具体的に説明されたもの以外の形式で実行されてもよい。したがって、本開示は、適用法によって許可された、それに添付された請求項において列挙された主題の全ての修正および均等物を含む。さらに、その全ての可能な変化態様における上述の要素のあらゆる組合せは、本明細書において別段の指示がない限り、開示によって包含される。

【0207】

本明細書において引用された、出版物、特許出願、および特許を含む全ての参照物は、各々の参照物が、参照によって組み込まれることが個々におよび具体的に指示されておりかつその全体が本明細書に示されたものとして、参照により同じ範囲に本明細書に組み込まれる。

【0208】

前記明細書において、開示の態様は、その特定の実施形態に関連して説明されているが、当業者は、開示がそれに限定されないことを認識するであろう。上述の開示の様々な特徴および態様は、個々にまたは組み合わせて使用されてもよい。さらに、実施形態は、明細書のより広い趣旨および範囲から逸脱することなく、本明細書に説明されたものを超えてあらゆる数の環境および用途において利用されることができる。したがって、明細書お

【図 3】

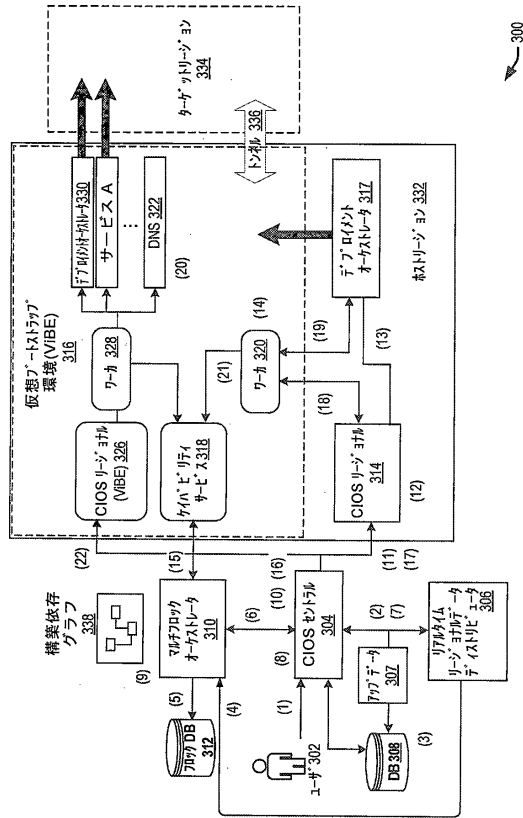


FIG. 3

【図 4】

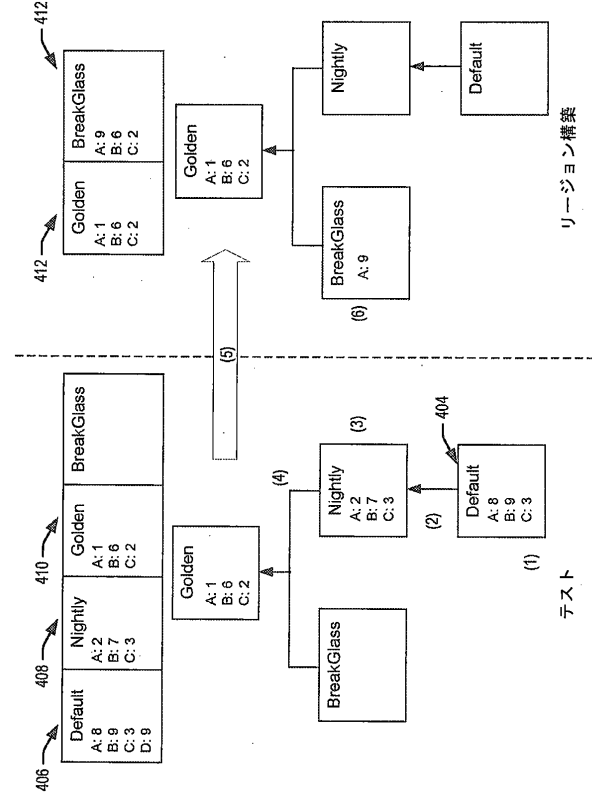


FIG. 4

【図 5】

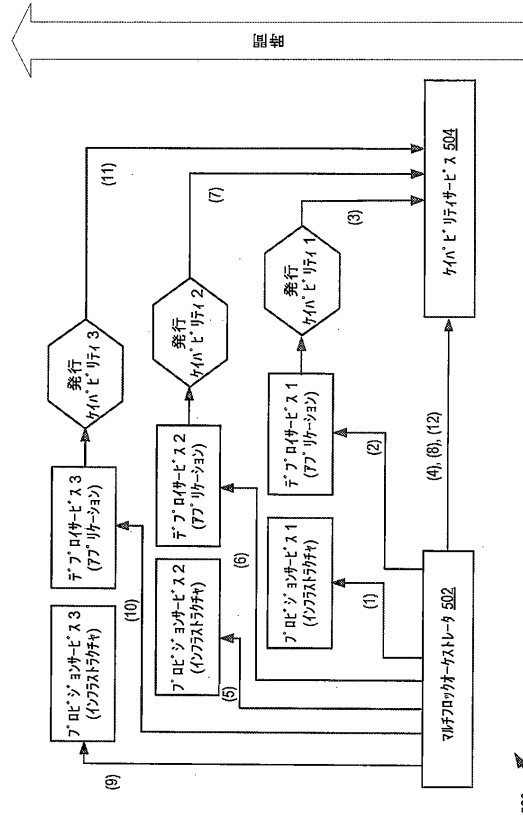


FIG. 5

【図 6】

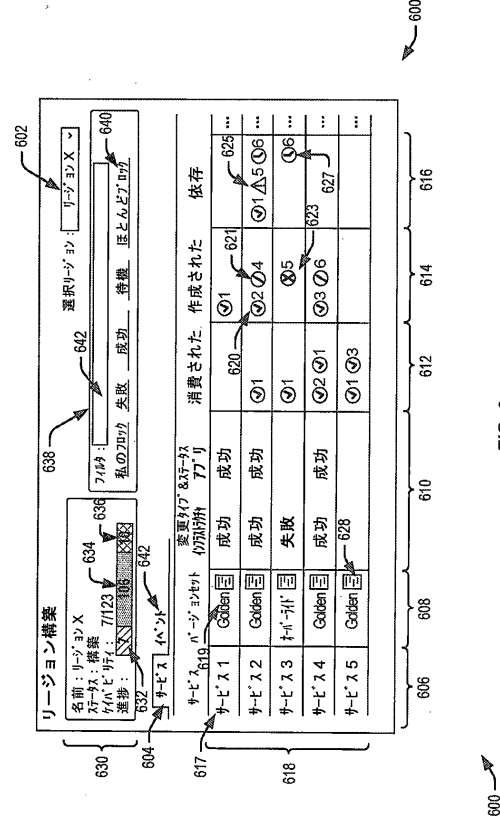


FIG. 6

【図 7】

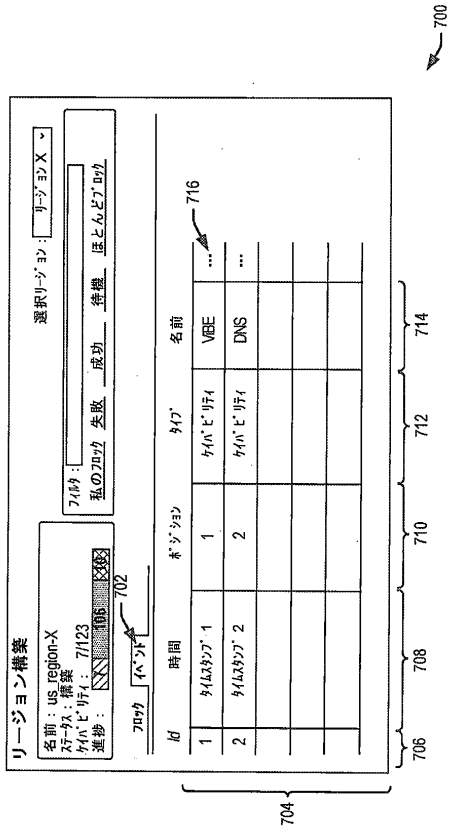


FIG. 7

【図 8】

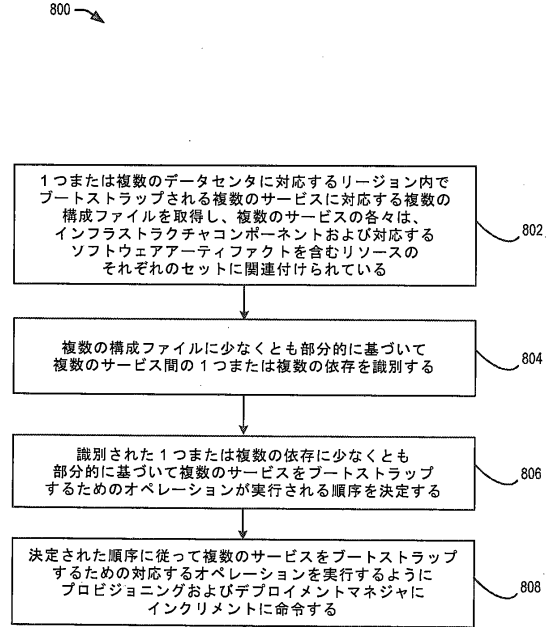


FIG. 8

【図 9】

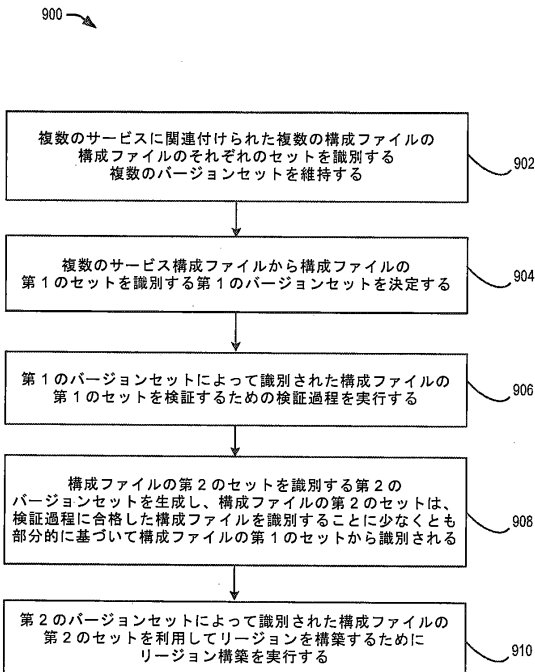


FIG. 9

【図 10】

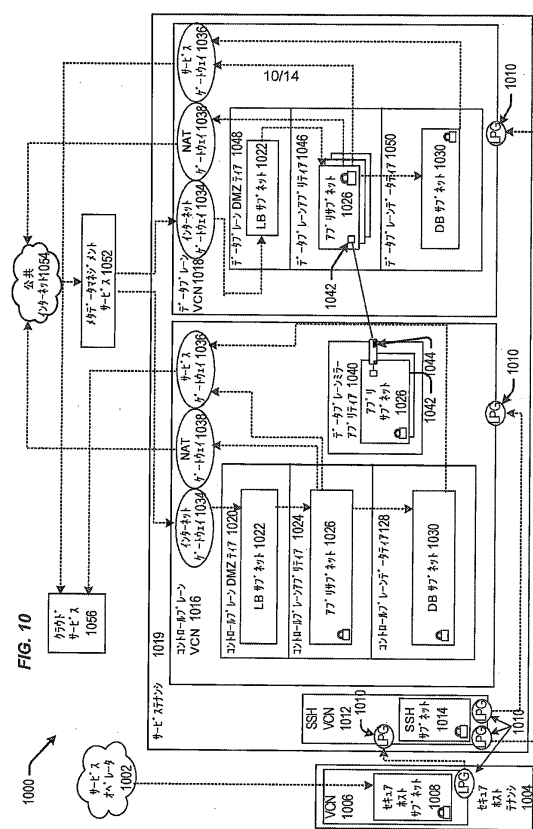
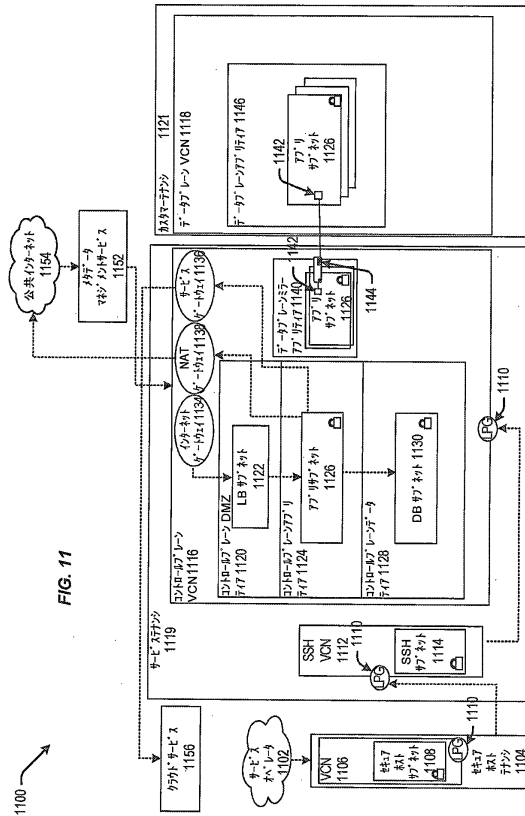
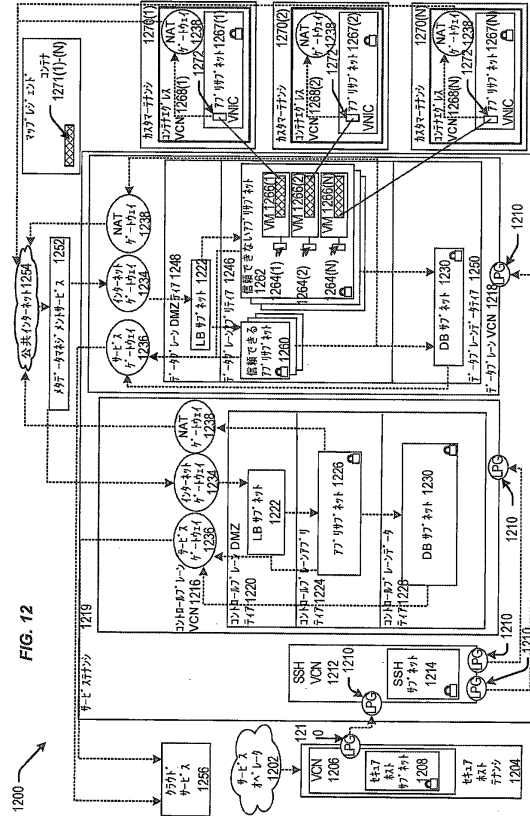


FIG. 10

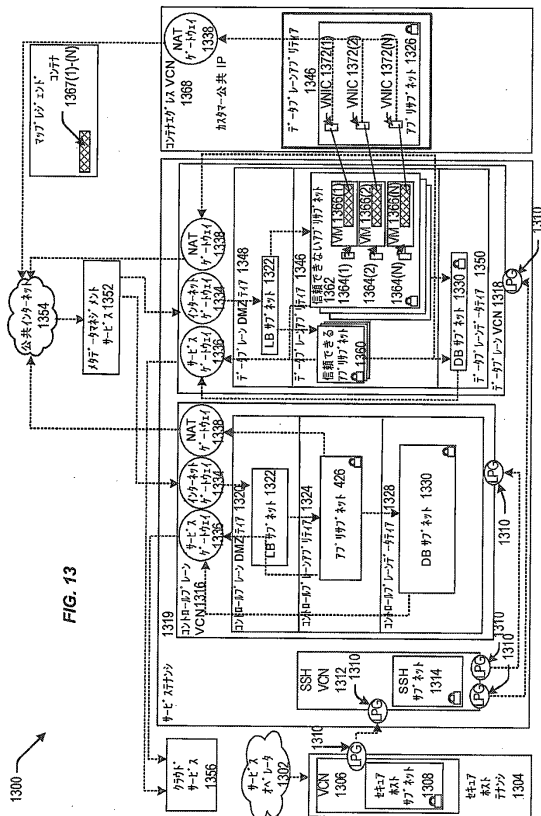
【図 1 1】



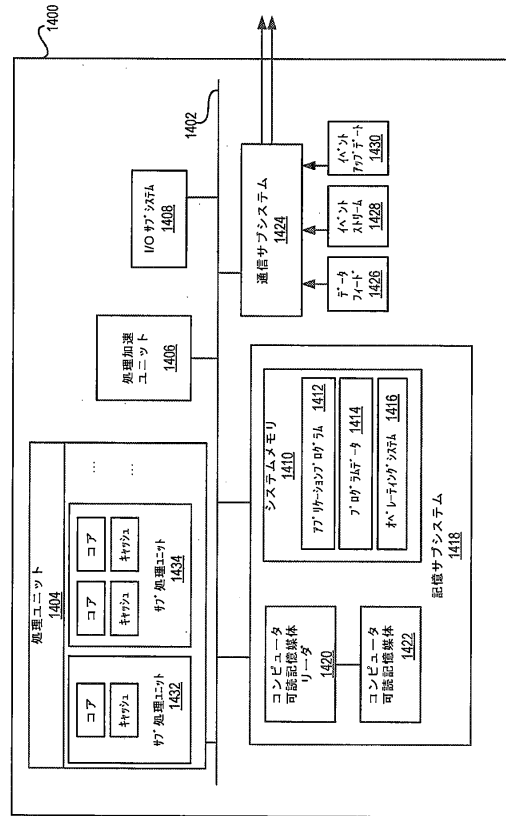
【圖 12】



【図 13】



【図 14】



【国際調査報告】

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2023/012035

A. CLASSIFICATION OF SUBJECT MATTER INV. G06F9/50 G06F8/60 ADD.		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED		
Minimum documentation searched (classification system followed by classification symbols) G06F		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) EPO-Internal, WPI Data		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2021/224076 A1 (DOCKTER CALEB [US] ET AL) 22 July 2021 (2021-07-22) the whole document -----	1-20
A	US 2017/228227 A1 (WINTERFELDT DAVID [US] ET AL) 10 August 2017 (2017-08-10) paragraphs [0028], [0086] - paragraph [0093]; figures 8B, 8C -----	1-20
☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.		
* Special categories of cited documents : "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier application or patent but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family		
Date of the actual completion of the international search 28 April 2023		Date of mailing of the international search report 09/05/2023
Name and mailing address of the ISA/ European Patent Office, P.B. 5618 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016		Authorized officer Renault, Sophie

Form PCT/ISA/210 (second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT				International application No	
Information on patent family members				PCT/US2023/012035	
Patent document cited in search report		Publication date	Patent family member(s)		Publication date
US 2021224076 A1		22-07-2021	NONE		
US 2017228227 A1		10-08-2017	US 2013232480 A1		05-09-2013
			US 2016019096 A1		21-01-2016
			US 2017228227 A1		10-08-2017

10

20

30

40

50

フロントページの続き

(32)優先日 令和4年2月28日(2022.2.28)

(33)優先権主張国・地域又は機関
米国(US)

(31)優先権主張番号 18/076,238

(32)優先日 令和4年12月6日(2022.12.6)

(33)優先権主張国・地域又は機関
米国(US)

(81)指定国・地域 AP(BW,CV,GH,GM,KE,LR,LS,MW,MZ,NA,RW,SD,SL,ST,SZ,TZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,RU,TJ,TM),EP(AL,AT,BE,BG,CH,CY,CZ,DE,DK,EE,ES,FI,FR,GB,GR,HR,HU,IE,IS,IT,LT,LU,LV,MC,ME,MK,MT,NL,NO,PL,PT,RO,RS,SE,SI,SK,SM,TR),OA(BF,BJ,CF,CG,CI,CM,GA,GN,GQ,GW,KM,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AO,AT,AU,AZ,BA,BB,BG,BH,BN,BR,BW,BY,BZ,CA,CH,CL,CN,CO,CR,CU,CV,CZ,DE,DJ,DK,DM,DO,DZ,EC,EE,EG,ES,FI,GB,GD,GE,GH,GM,GT,HN,HR,HU,ID,IL,IN,IQ,IR,IS,IT,JM,JO,JP,KE,KG,KH,KN,KP,KR,KW,KZ,LA,LC,LK,LR,LS,LU,LY,MA,MD,MG,MK,MN,MW,MX,MY,MZ,NA,NG,NI,NO,NZ,OM,PA,PE,PG,PH,PL,PT,QA,RO,RS,RU,RW,SA,SC,SD,SE,SG,SK,SL,ST,SV,SY,TH,TJ,TM,TN,TR,TT,TZ,UA,UG,US,UZ,VC,VN,WS,ZA,ZM,ZW

(特許庁注：以下のものは登録商標)

1. WINDOWS PHONE

2. iOS

3. イーサネット

4. PALM OS

(72)発明者 ドックター， ケイレブ

アメリカ合衆国、94065 カリフォルニア州、レッドウッド・シティー、オラクル・パークウェイ、500、オラクル・インターナショナル・コーポレーション内