(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7160442号
(P7160442)

(45)発行日　令和4年10月25日(2022.10.25)

(24)登録日　令和4年10月17日(2022.10.17)

(51)国際特許分類

		FI		
G06F 16/188 (2019.01)		G06F	16/188	
G06F 9/50 (2006.01)		G06F	9/50	120Z

請求項の数　9　（全22頁）

(21)出願番号　特願2019-571200(P2019-571200)
(86)(22)出願日　平成30年6月14日(2018.6.14)
(65)公表番号　特表2020-525909(P2020-525909A)
(43)公表日　令和2年8月27日(2020.8.27)
(86)国際出願番号　PCT/IB2018/054378
(87)国際公開番号　WO2019/003029
(87)国際公開日　平成31年1月3日(2019.1.3)
審査請求日　令和2年11月30日(2020.11.30)
(31)優先権主張番号　15/636,770
(32)優先日　平成29年6月29日(2017.6.29)
(33)優先権主張国・地域又は機関
米国(US)
(31)優先権主張番号　15/824,356
(32)優先日　平成29年11月28日(2017.11.28)
最終頁に続く

(73)特許権者　390009531
インターナショナル・ビジネス・マシーンズ・コーポレーション
INTERNATIONAL BUSINESS MACHINES CORPORATION
アメリカ合衆国10504 ニューヨーク州 アーモンク ニュー オーチャード ロード
New Orchard Road, Armonk, New York 10504, United States of America
(74)代理人　100112690
弁理士　太佐 種一
最終頁に続く

(54)【発明の名称】　読み取り／書き込み要求を管理するための方法、コンピュータ・プログラムおよびコンピュータ・システム

(57)【特許請求の範囲】

【請求項１】

読み取り／書き込み要求を管理するための方法であって、

テナント識別子のセット内の第１のテナント識別子に対応する第１のディレクトリを決定することであって、前記第１のディレクトリが第１の分散ファイル・システムを使用して構造化され、前記第１のテナント識別子が第１のテナントに対応する、前記決定することと、

接続サーバであるコネクタ・サービスを前記第１のテナントに対応する前記第１のディレクトリに割り当てることと、

前記コネクタ・サービスに対応する第２のディレクトリを決定することとであって、前記第２のディレクトリが、前記第１の分散ファイル・システムとは異なる第２の分散ファイル・システムを使用して構造化され、前記第２のディレクトリ上には、前記コネクタ・サービスとは別個に構成される第１のノードが含んでいる第１のファイルのセットがあり、前記第１のファイルのセットが前記第１のテナントに対応するものである、前記決定することと、

前記コネクタ・サービスおよび前記第１のノードを使用して読み取り／書き込み要求のセット内の第１の読み取り／書き込み要求を処理することであって、前記コネクタ・サービスが、前記第１のテナントから来る、前記第１の分散ファイル・システムへの前記第１の読み取り／書き込み要求を、前記第２の分散ファイル・システムに向けて方向付ける、前記処理することと、

前記第１の読み取り／書き込み要求に対して第１の結果を生成することと
を含み、少なくとも前記コネクタ・サービスおよび前記第１のノードを使用して前記第１の読み取り／書き込み要求を処理することが、コンピュータ・ハードウェア上で実行されるコンピュータ・ソフトウェアによって実行される、方法。
【請求項２】
前記コネクタ・サービスと前記第１のノードに含まれる複数のテナントのそれぞれとを対応付けた経路を生成することをさらに含む、請求項１に記載の方法。
【請求項３】
前記テナント識別子のセット内の第２のテナント識別子に対応する第３のディレクトリを決定することであって、前記第３のディレクトリが、前記第１の分散ファイル・システムを使用して構造化される、前記決定することと、
前記コネクタ・サービスを前記第２のテナント識別子に対応する前記第３のディレクトリに割り当てることと、
前記コネクタ・サービスおよび第２のノードを使用して第２の読み取り／書き込み要求を処理することであって、前記第２のディレクトリ上には、前記第２のノードが含む第２のファイルのセットがあり、前記第２のファイルのセットが第２のテナントに対応するものである、前記処理することと、
前記第２の読み取り／書き込み要求に対して第２の結果を生成することと
をさらに含んでいる、請求項１または２に記載の方法。
【請求項４】
前記第２のノードが前記第１のノードである、請求項３に記載の方法。
【請求項５】
前記第１の結果が、
新しいデータ・エントリと、
データのセットを含むメッセージと
から成る群から選択される、請求項１ないし４のいずれかに記載の方法。
【請求項６】
前記第２の分散ファイル・システムは、ＰＯＳＩＸ互換である、請求項１ないし５のいずれかに記載の方法。
【請求項７】
前記第１の分散ファイル・システムは、Ａｐａｃｈｅ　Ｈａｄｏｏｐ　Ｄｉｓｔｒｉｂｕｔｅｄ　Ｆｉｌｅ　Ｓｙｓｔｅｍ（「ＨＤＦＳ」）である、請求項１ないし６のいずれかに記載の方法。
【請求項８】
コンピュータ可読媒体に格納された、デジタル・コンピュータの内部メモリに読み込み可能なコンピュータ・プログラムであって、前記プログラムがコンピュータ上で実行された場合に請求項１ないし７のいずれかに記載の方法を実行するためのソフトウェア・コード部分を含んでいる、コンピュータ・プログラム。
【請求項９】
読み取り／書き込み要求を管理するためのコンピュータ・システムであって、
プロセッサ・セットと、
コンピュータ可読記憶媒体と
を備えており、
前記プロセッサ・セットが、前記コンピュータ可読記憶媒体に格納された命令を実行するように構造化されるか、配置されるか、接続されるか、またはプログラムされるか、あるいはその組み合わせが実行され、
前記命令が、
デバイスに、テナント識別子のセット内の第１のテナント識別子に対応する第１のディレクトリを決定させるように、前記デバイスによって実行可能なプログラム命令であって、前記第１のディレクトリが第１の分散ファイル・システムを使用して構造化され、前記

第１のテナント識別子が第１のテナントに対応する、前記プログラム命令と、
デバイスに、接続サーバであるコネクタ・サービスを前記第１のテナントに対応する前記第１のディレクトリに割り当てさせるように、前記デバイスによって実行可能なプログラム命令と、
デバイスに、前記コネクタ・サービスに対応する第２のディレクトリを決定させるように、前記デバイスによって実行可能なプログラム命令であって、前記第２のディレクトリが、前記第１の分散ファイル・システムとは異なる第２の分散ファイル・システムを使用して構造化され、前記第２のディレクトリ上には、前記コネクタ・サービスとは別個に構成される第１のノードが含んでいる第１のファイルのセットがあり、前記第１のファイルのセットが前記第１のテナントに対応するものである、前記プログラム命令と、
デバイスに、前記コネクタ・サービスおよび前記第１のノードを使用して読み取り／書き込み要求のセット内の第１の読み取り／書き込み要求を処理させるように、前記デバイスによって実行可能なプログラム命令であって、前記コネクタ・サービスが、前記第１のテナントから来る、前記第１の分散ファイル・システムへの前記第１の読み取り／書き込み要求を、前記第２の分散ファイル・システムに向けて方向付ける、前記プログラム命令と、
デバイスに、前記第１の読み取り／書き込み要求に対する第１の結果を生成させるように、前記デバイスによって実行可能なプログラム命令と
を含んでいる、コンピュータ・システム。

【発明の詳細な説明】

【技術分野】

【０００１】

本発明は、一般に、ストレージのアクセスおよび制御の分野に関し、特に、メモリの構成に関する。

【背景技術】

【０００２】

集中型システムにおいて、仮想化は、計算リソース、ストレージ容量、またはアプリケーションのモビリティ、あるいはその組み合わせの弾力性を実現する。集中型インフラストラクチャは、情報技術のコンポーネントをソフトウェア・パッケージにグループ化する。仮想化コンテナは、ソフトウェアを信頼できる方法でサーバにインストールするためのファイル・システムを含むソフトウェア・パッケージである。仮想化コンテナの例は、Ｄｏｃｋｅｒである。一部の仮想化コンテナは、ソフトウェア・ライブラリ・フレームワーク（software library frameworks）を含む。ソフトウェア・ライブラリ・フレームワークは、プログラミング・モデルを使用して、大規模なデータ・セットの分散処理を可能にする。そのようなソフトウェア・ライブラリ・フレームワークの一例は、Ｈａｄｏｏｐである。ポータブル・オペレーティング・システム・インターフェイスは、さまざまなオペレーティング・システム間の互換性を維持する。ポータブル・オペレーティング・システム・インターフェイスは、一連のアプリケーション・プログラミング・インターフェイスを定義する。ポータブル・オペレーティング・システム・インターフェイス規格の一例は、ＰＯＳＩＸである。

【０００３】

ビッグ・データ解析は、構造化データおよび非構造化データの両方を含むデータの指数関数的増加および可用性にもかかわらず、解析技術を可能にする。ビッグ・データ解析は、（ｉ）関係データベースに基づく超並列処理、および（ｉｉ）ソフトウェア・ライブラリ・フレームワークに基づく解析という２つの方向に発達した。

【０００４】

さまざまなユーザの複数のクラスタを管理することは、極めて困難である。１つの独立したネットワークＩＰアドレスを監視しながら、１つのテナントのクラスタ・インスタンスごとに１つのコネクタ・サービスを開始し、監視し、維持することができるが、この方法は、多量のシステム・リソースが必要になるため、拡張可能ではない。

【発明の概要】
【発明が解決しようとする課題】
【０００５】
したがって、当技術分野において、前述の問題に対処する必要がある。
【課題を解決するための手段】
【０００６】
第１の態様から見ると、本発明は、読み取り／書き込み要求を管理するための方法を提供し、この方法は、テナント識別子のセット内の第１のテナント識別子に対応する第１のディレクトリを決定することであって、第１のインターフェイス規格を使用して第１のディレクトリが構造化され、第１のテナント識別子が第１のディレクトリの第１のテナントに対応する、決定することと、コネクタ・サービスを第１のディレクトリおよび第１のテナント識別子に割り当てることと、コネクタ・サービスに対応する第２のディレクトリを決定することとであって、第２のインターフェイス規格を使用して第２のディレクトリが構造化され、第１のノードが第２のディレクトリ上に第１のファイルのセットを含んでおり、第１のファイルのセットが第１のテナントに対応する、決定することと、コネクタ・サービスおよび第１のノードを使用して読み取り／書き込み要求のセット内の第１の読み取り／書き込み要求を処理することであって、第１の読み取り／書き込み要求が第１のテナントから来る、処理することと、第１の読み取り／書き込み要求に対して第１の結果を生成することであって、コネクタ・サービスおよび第１のノードを使用して少なくとも第１の読み取り／書き込み要求を処理することが、コンピュータ・ハードウェア上で実行されるコンピュータ・ソフトウェアによって実行される、生成することとを含む。
【０００７】
さらに別の態様から見ると、本発明は読み取り／書き込み要求を管理するためのコンピュータ・システムを提供し、このシステムは、プロセッサ・セットと、コンピュータ可読記憶媒体とを備えており、このプロセッサ・セットは、コンピュータ可読記憶媒体に格納された命令を実行するように構造化されるか、配置されるか、接続されるか、またはプログラムされるか、あるいはその組み合わせが実行され、それらの命令は、デバイスに、テナント識別子のセット内の第１のテナント識別子に対応する第１のディレクトリを決定させるように、デバイスによって実行可能なプログラム命令であって、第１のインターフェイス規格を使用して第１のディレクトリが構造化され、第１のテナント識別子が第１のディレクトリの第１のテナントに対応する、プログラム命令と、デバイスに、コネクタ・サービスを第１のディレクトリおよび第１のテナント識別子に割り当てさせるように、デバイスによって実行可能なプログラム命令と、デバイスに、コネクタ・サービスに対応する第２のディレクトリを決定させるように、デバイスによって実行可能なプログラム命令であって、第２のインターフェイス規格を使用して第２のディレクトリが構造化され、第１のノードが第２のディレクトリ上に第１のファイルのセットを含んでおり、第１のファイルのセットが第１のテナントに対応する、プログラム命令と、デバイスに、コネクタ・サービスおよび第１のノードを使用して読み取り／書き込み要求のセット内の第１の読み取り／書き込み要求を処理させるように、デバイスによって実行可能なプログラム命令であって、第１の読み取り／書き込み要求が第１のテナントから来る、プログラム命令と、デバイスに、第１の読み取り／書き込み要求に対する第１の結果を生成させるように、デバイスによって実行可能なプログラム命令とを含む。
【０００８】
さらに別の態様から見ると、本発明は、読み取り／書き込み要求を管理するためのコンピュータ・プログラム製品を提供し、このコンピュータ・プログラム製品は、処理回路によって読み取り可能なコンピュータ可読記憶媒体を備えており、コンピュータ可読記憶媒体は、本発明のステップを実行するための方法を実行するためにこの処理回路によって実行される命令を格納している。
【０００９】
さらに別の態様から見ると、本発明は、コンピュータ可読媒体に格納された、デジタル

・コンピュータの内部メモリに読み込み可能なコンピュータ・プログラムを提供し、このコンピュータ・プログラムは、コンピュータ上で実行された場合に本発明のステップを実行するためのソフトウェア・コード部分を含んでいる。

【００１０】

本発明の態様によれば、方法、コンピュータ・プログラム製品、またはシステム、あるいはその組み合わせが存在し、これらは、（ｉ）テナント識別子のセット内の第１のテナント識別子に対応する第１のディレクトリを決定することであって、（ａ）第１のインターフェイス規格を使用して第１のディレクトリが構造化され、（ｂ）第１のテナント識別子が第１のディレクトリの第１のテナントに対応する、決定することと、（ｉｉ）コネクタ・サービスを第１のディレクトリおよび第１のテナント識別子に割り当てることと、（ｉｉｉ）コネクタ・サービスに対応する第２のディレクトリを決定することであって、（ａ）第２のインターフェイス規格を使用して第２のディレクトリが構造化され、（ｂ）第１のノードが第２のディレクトリ上に第１のファイルのセットを含んでおり、（ｃ）第１のファイルのセットが第１のテナントに対応する、決定することと、（ｉｖ）コネクタ・サービスおよび第１ノードを使用して読み取り／書き込み要求のセット内の第１の読み取り／書き込み要求を処理することであって、第１の読み取り／書き込み要求が第１のテナントから来る、処理することと、（ｖ）第１の読み取り／書き込み要求に対して第１の結果を生成することとを実行する動作を（必ずしもこの順序でなく）実行する。コネクタ・サービスおよび第１のノードを使用して少なくとも第１の読み取り／書き込み要求を処理することは、コンピュータ・ハードウェア上で実行されるコンピュータ・ソフトウェアによって実行される。

【００１１】

以下では、次の図に示された好ましい実施形態を単に例として参照し、本発明が説明される。

【図面の簡単な説明】

【００１２】

【図１】本発明に従うシステムの第１の実施形態を示すブロック図である。

【図２】第１の実施形態のシステムによって少なくとも一部が実行される第１の実施形態の方法を示すフローチャートである。

【図３】第１の実施形態のシステムの機械論理（例えば、ソフトウェア）部分を示すブロック図である。

【図４】本発明に従うシステムの第２の実施形態によって実行される第２の実施形態の方法を示すフローチャートである。

【図５】システムの第２の実施形態を示すブロック図である。

【図６】本発明に従うシステムの第３の実施形態によって生成されたルックアップ・テーブルである。

【図７】本発明に従うシステムの第４の実施形態によって実行される第３の実施形態の方法を示すフローチャートである。

【発明を実施するための形態】

【００１３】

ノード上のマルチテナント分散ファイル・システムの構成。さまざまなテナントおよびテナント・クラスタ（tenant clusters）は、分散ファイル・システムと相互関係があり、分散ファイル・システムは、コネクタ・サービスを介してさまざまなテナントと通信する。分散ファイル・システム全体は、物理ノード上に存在する。「発明を実施するための形態」のセクションは、（ｉ）ハードウェアおよびソフトウェア環境、（ｉｉ）実施形態例、（ｉｉｉ）追加のコメントまたは実施形態あるいはその両方、ならびに（ｉｖ）定義のサブセクションに分割されている。

【００１４】

Ｉ．ハードウェアおよびソフトウェア環境

本発明は、任意の可能な統合の技術的詳細レベルで、システム、方法、またはコンピュ

ータ・プログラム製品、あるいはその組み合わせであってよい。コンピュータ・プログラム製品は、プロセッサに本発明の態様を実行させるためのコンピュータ可読プログラム命令を含んでいるコンピュータ可読記憶媒体を含んでよい。

【００１５】

コンピュータ可読記憶媒体は、命令実行デバイスによって使用するための命令を保持および格納できる有形のデバイスにすることができる。コンピュータ可読記憶媒体は、例えば、電子ストレージ・デバイス、磁気ストレージ・デバイス、光ストレージ・デバイス、電磁ストレージ・デバイス、半導体ストレージ・デバイス、またはこれらの任意の適切な組み合わせであってよいが、これらに限定されない。コンピュータ可読記憶媒体のさらに具体的な例の非網羅的リストは、ポータブル・コンピュータ・ディスケット、ハード・ディスク、ランダム・アクセス・メモリ（ＲＡＭ：random access memory）、読み取り専用メモリ（ＲＯＭ：read-only memory）、消去可能プログラマブル読み取り専用メモリ（ＥＰＲＯＭ：erasable programmable read-only memoryまたはフラッシュ・メモリ）、スタティック・ランダム・アクセス・メモリ（ＳＲＡＭ：static random access memory）、ポータブル・コンパクト・ディスク読み取り専用メモリ（ＣＤ－ＲＯＭ：compact disc read-only memory）、デジタル多用途ディスク（ＤＶＤ：digital versatile disk）、メモリ・スティック、フロッピー（Ｒ）・ディスク、パンチカードまたは命令が記録されている溝の中の隆起構造などの機械的にエンコードされるデバイス、およびこれらの任意の適切な組み合わせを含む。本明細書において使用されるとき、コンピュータ可読記憶媒体は、それ自体が、電波またはその他の自由に伝搬する電磁波、導波管またはその他の送信媒体を伝搬する電磁波（例えば、光ファイバ・ケーブルを通過する光パルス）、あるいはワイヤを介して送信される電気信号などの一過性の信号であると解釈されるべきではない。

【００１６】

本明細書に記載されたコンピュータ可読プログラム命令は、コンピュータ可読記憶媒体から各コンピューティング・デバイス／処理デバイスへ、またはネットワーク（例えば、インターネット、ローカル・エリア・ネットワーク、広域ネットワーク、または無線ネットワーク、あるいはその組み合わせ）を介して外部コンピュータまたは外部ストレージ・デバイスへダウンロードされ得る。このネットワークは、銅伝送ケーブル、光伝送ファイバ、無線送信、ルータ、ファイアウォール、スイッチ、ゲートウェイ・コンピュータ、またはエッジ・サーバ、あるいはその組み合わせを備えてよい。各コンピューティング・デバイス／処理デバイス内のネットワーク・アダプタ・カードまたはネットワーク・インターフェイスは、コンピュータ可読プログラム命令をネットワークから受信し、それらのコンピュータ可読プログラム命令を各コンピューティング・デバイス／処理デバイス内のコンピュータ可読記憶媒体に格納するために転送する。

【００１７】

本発明の動作を実行するためのコンピュータ可読プログラム命令は、アセンブラ命令、命令セット・アーキテクチャ（ＩＳＡ：instruction-set-architecture）命令、マシン命令、マシン依存命令、マイクロコード、ファームウェア命令、状態設定データ、集積回路のための構成データ、あるいは、Ｓｍａｌｌｔａｌｋ（Ｒ）、Ｃ＋＋などのオブジェクト指向プログラミング言語、および「Ｃ」プログラミング言語または同様のプログラミング言語などの手続き型プログラミング言語を含む１つまたは複数のプログラミング言語の任意の組み合わせで記述されたソース・コードまたはオブジェクト・コードであってよい。コンピュータ可読プログラム命令は、ユーザのコンピュータ上で全体的に実行すること、ユーザのコンピュータ上でスタンドアロン・ソフトウェア・パッケージとして部分的に実行すること、ユーザのコンピュータ上およびリモート・コンピュータ上でそれぞれ部分的に実行すること、あるいはリモート・コンピュータ上またはサーバ上で全体的に実行することができる。後者のシナリオでは、リモート・コンピュータを、ローカル・エリア・ネットワーク（ＬＡＮ：local area network）または広域ネットワーク（ＷＡＮ：wide areanetwork）を含む任意の種類のネットワークを介してユーザのコンピュータに接続する

ことができ、または接続を、（例えば、インターネット・サービス・プロバイダを使用してインターネットを介して）外部コンピュータに対して行うことができる。一部の実施形態では、本発明の態様を実行するために、例えばプログラマブル論理回路、フィールドプログラマブル・ゲート・アレイ（ＦＰＧＡ：field-programmable gate arrays）、またはプログラマブル・ロジック・アレイ（ＰＬＡ：programmable logic arrays）を含む電子回路は、コンピュータ可読プログラム命令の状態情報を利用することによって、電子回路をカスタマイズするためのコンピュータ可読プログラム命令を実行してよい。

【００１８】

本発明の態様は、本明細書において、本発明の実施形態に従って、方法、装置（システム）、およびコンピュータ・プログラム製品のフローチャート図またはブロック図あるいはその両方を参照して説明される。フローチャート図またはブロック図あるいはその両方の各ブロック、ならびにフローチャート図またはブロック図あるいはその両方に含まれるブロックの組み合わせが、コンピュータ可読プログラム命令によって実装され得るということが理解されるであろう。

【００１９】

これらのコンピュータ可読プログラム命令は、コンピュータまたはその他のプログラム可能なデータ処理装置のプロセッサを介して実行される命令が、フローチャートまたはブロック図あるいはその両方の１つまたは複数のブロックに指定される機能／動作を実施する手段を作り出すべく、汎用コンピュータ、専用コンピュータ、または他のプログラム可能なデータ処理装置のプロセッサに提供されてマシンを作り出すものであってよい。これらのコンピュータ可読プログラム命令は、命令が格納されたコンピュータ可読記憶媒体がフローチャートまたはブロック図あるいはその両方の１つまたは複数のブロックに指定される機能／動作の態様を実施する命令を含んでいる製品を備えるように、コンピュータ可読記憶媒体に格納され、コンピュータ、プログラム可能なデータ処理装置、または他のデバイス、あるいはその組み合わせに特定の方式で機能するように指示できるものであってもよい。

【００２０】

コンピュータ可読プログラム命令は、コンピュータ上、その他のプログラム可能な装置上、またはその他のデバイス上で実行される命令が、フローチャートまたはブロック図あるいはその両方の１つまたは複数のブロックに指定される機能／動作を実施するように、コンピュータ実装プロセスを生成すべく、コンピュータ、その他のプログラム可能なデータ処理装置、またはその他のデバイスに読み込まれて、コンピュータ上、その他のプログラム可能な装置上、またはその他のデバイス上で一連の動作可能なステップを実行させるものであってもよい。

【００２１】

図内のフローチャートおよびブロック図は、本発明のさまざまな実施形態に従って、システム、方法、およびコンピュータ・プログラム製品の可能な実装のアーキテクチャ、機能、および動作を示す。これに関連して、フローチャートまたはブロック図内の各ブロックは、規定された論理機能を実装するための１つまたは複数の実行可能な命令を備える、命令のモジュール、セグメント、または部分を表してよい。一部の代替の実装では、ブロックに示された機能は、図に示された順序とは異なる順序で発生してよい。例えば、連続して示された２つのブロックは、実際には、含まれている機能に応じて、実質的に同時に実行されるか、または場合によっては逆の順序で実行されてよい。ブロック図またはフローチャート図あるいはその両方の各ブロック、ならびにブロック図またはフローチャート図あるいはその両方に含まれるブロックの組み合わせは、規定された機能または動作を実行するか、または専用ハードウェアとコンピュータ命令の組み合わせを実行する専用ハードウェアベースのシステムによって実装され得るということにも注意する。

【００２２】

ここで、本発明に従うソフトウェアまたは方法あるいはその両方の可能なハードウェアおよびソフトウェア環境の実施形態が、各図を参照して詳細に説明される。図１は、マル

チテナント構成サブシステム１０２、ユーザ・サブシステム１０４、仮想コンテナ・サブシステム１０６、仮想コンテナ・サブシステム１０８、コネクタ・サービス１１２、および通信ネットワーク１１４を含んでいるネットワーク・コンピュータ・システム１００のさまざまな部分を示す機能ブロック図である。マルチテナント構成サブシステム１０２は、マルチテナント構成コンピュータ２００、ディスプレイ・デバイス２１２、および外部デバイス２１４を含んでいる。マルチテナント構成コンピュータ２００は、通信ユニット２０２、プロセッサ・セット２０４、入出力（Ｉ／Ｏ：input/output）インターフェイス・セット２０６、メモリ・デバイス２０８、および永続的ストレージ・デバイス２１０を含んでいる。メモリ・デバイス２０８は、ランダム・アクセス・メモリ（ＲＡＭ）デバイス２１６およびキャッシュ・メモリ・デバイス２１８を含んでいる。永続的ストレージ２１０は、マルチテナント構成プログラム３００を含んでいる。仮想コンテナ・サブシステム１０８は、ソフトウェア・ライブラリ・フレームワーク１１０を含んでいる。

【００２３】

マルチテナント構成サブシステム１０２は、多くの点で、本発明におけるさまざまなコンピュータ・サブシステムを代表している。したがって以下では、マルチテナント構成サブシステム１０２の複数の部分が説明される。

【００２４】

マルチテナント構成サブシステム１０２は、ラップトップ・コンピュータ、タブレット・コンピュータ、ネットブック・コンピュータ、パーソナル・コンピュータ（ＰＣ：personal computer）、デスクトップ・コンピュータ、パーソナル・デジタル・アシスタント（ＰＤＡ：personal digital assistant）、スマートフォン、または通信ネットワーク１１４を介してクライアント・サブシステムと通信できる任意のプログラム可能な電子デバイスであってよい。マルチテナント構成プログラム３００は、以下で、この「発明を実施するための形態」セクションの「実施形態例」サブセクションにおいて詳細に説明される、特定のソフトウェア機能を作成し、管理し、制御するために使用される機械可読の命令またはデータあるいはその両方の集合である。

【００２５】

マルチテナント構成サブシステム１０２は、通信ネットワーク１１４を介して他のコンピュータ・サブシステムと通信することができる。例えば、通信ネットワーク１１４は、ローカル・エリア・ネットワーク（ＬＡＮ）、インターネットなどの広域ネットワーク（ＷＡＮ）、またはこれらの組み合わせであることができ、有線接続、無線接続、または光ファイバ接続を含むことができる。一般に、通信ネットワーク１１４は、マルチテナント構成サブシステム１０２とクライアント・サブシステムの間の通信をサポートする接続およびプロトコルの任意の組み合わせであることができる。

【００２６】

マルチテナント構成サブシステム１０２は、多くの両方向の矢印を含むブロック図として示されている。これらの両方向の矢印（別の参照番号はない）は、マルチテナント構成サブシステム１０２のさまざまなコンポーネント間の通信を提供する通信ファブリックを表す。通信ファブリックは、プロセッサ（マイクロプロセッサ、通信プロセッサ、またはネットワーク・プロセッサ、あるいはその組み合わせなど）、システム・メモリ、周辺機器、およびシステム内の任意のその他のハードウェア・コンポーネントの間で、データまたは制御情報あるいはその両方を渡すために設計された、任意のアーキテクチャを使用して実装され得る。例えば、通信ファブリックは、少なくとも一部において、１つまたは複数のバスを使用して実装され得る。

【００２７】

メモリ・デバイス２０８および永続的ストレージ・デバイス２１０は、コンピュータ可読記憶媒体である。一般に、メモリ・デバイス２０８は、任意の適切な揮発性または不揮発性のコンピュータ可読記憶媒体を含むことができる。現在または近い将来あるいはその両方において、（ｉ）外部デバイス２１４が、マルチテナント構成サブシステム１０２の一部または全部のメモリを提供できてよいということ、または（ｉｉ）マルチテナント構

成サブシステム１０２の外部にあるデバイスが、マルチテナント構成サブシステム１０２にメモリを提供できてよいということ、あるいはその両方のことに、さらに注意する。

【００２８】

マルチテナント構成プログラム３００は、プロセッサ・セット２０４の１つまたは複数のプロセッサによる、通常はメモリ・デバイス２０８を介したアクセスまたは実行あるいはその両方のために、永続的ストレージ・デバイス２１０に格納される。永続的ストレージ・デバイス２１０は、（ｉ）少なくとも送信中の信号より永続的であり、（ｉｉ）プログラム（ソフト・ロジックまたはデータあるいはその両方を含む）を有形の媒体（磁気または光領域など）に格納し、（ｉｉｉ）永久記憶装置より大幅に永続性が低い。代替としてデータ格納は、永続的ストレージ・デバイス２１０によって提供される格納のタイプより永続的または永久的あるいはその両方であってよい。

【００２９】

マルチテナント構成プログラム３００は、実質的データ（すなわち、データベースに格納されるデータのタイプ）または機械可読の実行可能な命令あるいはその両方を含んでよい。この特定の実施形態（すなわち、図１）では、永続的ストレージ・デバイス２１０は、磁気ハード・ディスク・ドライブを含んでいる。可能性のある変形をいくつか挙げると、永続的ストレージ・デバイス２１０は、固体ハード・ドライブ、半導体ストレージ・デバイス、読み取り専用メモリ（ＲＯＭ）、消去可能プログラマブル読み取り専用メモリ（ＥＰＲＯＭ）、フラッシュ・メモリ、あるいはプログラム命令またはデジタル情報を格納できる任意のその他のコンピュータ可読記憶媒体を含んでよい。

【００３０】

永続的ストレージ・デバイス２１０によって使用される媒体は、取り外し可能であってもよい。例えば、取り外し可能ハード・ドライブを、永続的ストレージ・デバイス２１０に使用できる。その他の例としては、永続的ストレージ・デバイス２１０の一部でもある別のコンピュータ可読記憶媒体に転送するためのドライブに挿入される、光ディスクおよび磁気ディスク、サム・ドライブ、ならびにスマート・カードが挙げられる。

【００３１】

これらの例において、通信ユニット２０２は、マルチテナント構成サブシステム１０２の外部にある他のデータ処理システムまたはデバイスとの通信を提供する。これらの例において、通信ユニット２０２は、１つまたは複数のネットワーク・インターフェイス・カードを含む。通信ユニット２０２は、物理的通信リンクまたは無線通信リンクのいずれか、あるいはその両方を使用して通信を提供できる。本明細書において説明される任意のソフトウェア・モジュールは、通信ユニット（通信ユニット２０２など）を介して永続的ストレージ・デバイス（永続的ストレージ・デバイス２１０など）にダウンロードされてよい。

【００３２】

Ｉ／Ｏインターフェイス２０６は、データ通信でマルチテナント構成コンピュータ２００にローカルに接続されることがある他のデバイスとのデータの入力および出力を可能にする。例えば、Ｉ／Ｏインターフェイス２０６は、外部デバイス２１４との接続を提供する。外部デバイス２１４は通常、キーボード、キーパッド、タッチ・スクリーン、またはその他の適切な入力デバイス、あるいはその組み合わせなどのデバイスを含む。外部デバイス２１４は、例えばサム・ドライブ、ポータブル光ディスクまたはポータブル磁気ディスク、およびメモリ・カードなどの、ポータブル・コンピュータ可読記憶媒体を含むこともできる。本発明の実施形態（例えば、マルチテナント構成プログラム３００）を実践するために使用されるソフトウェアおよびデータは、そのようなポータブル・コンピュータ可読記憶媒体に格納され得る。それらの実施形態では、関連するソフトウェアが、Ｉ／Ｏインターフェイス・セット２０６を介して、全体的または部分的に永続的ストレージ・デバイス２１０に読み込まれてよい（または、読み込まれなくてよい）。Ｉ／Ｏインターフェイス・セット２０６は、データ通信でディスプレイ・デバイス２１２にも接続される。

【００３３】

ディスプレイ・デバイス２１２は、データをユーザに表示するためのメカニズムを提供し、例えば、コンピュータのモニタまたはスマートフォンの表示画面であってよい。

【００３４】

本明細書に記載されたプログラムは、アプリケーションに基づいて識別され、本発明の特定の実施形態において、そのアプリケーションに関して実装される。ただし、本明細書における特定のプログラムの名前は単に便宜上使用されていると理解されるべきであり、したがって、本発明は、そのような名前によって識別されたか、または暗示されたか、あるいはその両方によって示された特定のアプリケーションのみで使用するように制限されるべきではない。

【００３５】

本発明のさまざまな実施形態の説明は、例示の目的で提示されているが、網羅的であることは意図されておらず、開示された実施形態に制限されない。説明された実施形態の範囲を逸脱することなく多くの変更および変形が可能であることは、当業者にとって明らかである。本明細書で使用された用語は、実施形態の原理、実際の適用、または市場で見られる技術を超える技術的改良を最も適切に説明するため、または他の当業者が本明細書で開示された実施形態を理解できるようにするために選択されている。

【００３６】

ＩＩ．実施形態例

図２は、本発明に従う方法を表すフローチャート２５０を示している。図３は、フローチャート２５０の方法の動作の少なくとも一部を実行するマルチテナント構成プログラム３００を示している。以降では、図２（方法の動作のブロックに関する）および図３（ソフトウェア・ブロックに関する）を詳細に参照して、この方法および関連するソフトウェアについて説明する。

【００３７】

動作Ｓ２５５で処理が開始し、要求受信モジュール（「ｍｏｄ」）３０２が要求のセットを受信する。本発明の一部の実施形態では、要求受信ｍｏｄ３０２が、要求元のセットから要求のセットを受信する。要求元の例としては、ソフトウェア・ライブラリ・フレームワーク、仮想コンテナ、またはユーザ、あるいはその組み合わせが挙げられるが、これらに限定されない。一部の実施形態では、要求のセットは、入出力（「Ｉ／Ｏ」）要求のセットである。さらに別の実施形態では、要求のセットは、読み取り／書き込み要求のセットである。これらの実施形態の一部では、要求のセットは、Ｉ／Ｏ読み取り／書き込み要求のセットである。仮想コンテナの例は、Ｄｏｃｋｅｒである。ソフトウェア・ライブラリ・フレームワークの例は、Ｈａｄｏｏｐである。さらに別の実施形態では、要求受信ｍｏｄ３０２が、要求元の動的インスタンス化のセットから要求のセットを受信する。

【００３８】

一部の実施形態では、要求元は第１の分散ファイル・システムである。これらの実施形態の一部では、第１の分散ファイル・システムはＰＯＳＩＸ互換である。さらに別の実施形態では、第１の分散ファイル・システムは、第１のインターフェイス規格を使用して構造化される。一部の実施形態では、要求のセットは第２の分散ファイル・システムに関連している。これらの実施形態の一部では、第２の分散ファイル・システムはＰＯＳＩＸ互換ではない。さらに別の実施形態では、第２の分散ファイル・システムは、第２のインターフェイス規格を使用して構造化される。代替として、一部の実施形態では、（ｉ）第１の分散ファイル・システムがＰＯＳＩＸ互換であり、（ｉｉ）第２の分散ファイル・システムがＰＯＳＩＸ互換ではない。さらに別の代替の実施形態では、第１の分散ファイル・システムも第２の分散ファイル・システムもＰＯＳＩＸ互換ではなく、異なるインターフェイス規格を使用して第１の分散ファイル・システムおよび第２の分散ファイル・システムが構造化される。

【００３９】

処理が動作Ｓ２６０に進み、ディレクトリ決定ｍｏｄ３０４が、要求元のセットに対応するディレクトリのセットを決定する。本発明の一部の実施形態では、ディレクトリ決定

ｍｏｄ３０４が、要求元のセットに対応するディレクトリのセットを決定する。ディレクトリは、コンピュータ・ファイルのセットを整理するための構造である。ディレクトリは、パス、フォルダ、またはドロワーと呼ばれることもあり、あるいはその組み合わせで呼ばれることもある。ディレクトリは、（ｉ）親フォルダ／子フォルダ／ファイル．拡張子、または（ｉｉ）親フォルダ＞子フォルダ＞ファイル、あるいはその両方を含む、さまざまな形態で表され得る。これらの実施形態の一部では、ディレクトリ決定ｍｏｄ３０４が、テナント識別子のセットに対応するディレクトリのセットを決定する。他の実施形態では、ディレクトリ決定ｍｏｄ３０４が、ディレクトリを要求元のセットに割り当てることによって、テナント識別子のセットに対応するディレクトリのセットを決定する。さらに別の実施形態では、ディレクトリ決定ｍｏｄ３０４が、サブディレクトリを要求元のセットに割り当てることによって、テナント識別子のセットに対応するディレクトリのセットを決定する。一部の実施形態では、要求元のセット内の第１の要求元が、第１のディレクトリに対応する。他の実施形態では、要求元のセットが、第１のディレクトリを共有する。一部の実施形態では、ディレクトリ決定ｍｏｄ３０４が、要求受信ｍｏｄ３０２が動作Ｓ２５５で受信した要求のセットの送信元である、要求元のセットに対応するディレクトリのセットを決定する。

【００４０】

処理が動作Ｓ２６５に進み、テナント識別子決定ｍｏｄ３０６が、要求のセットに対応するテナント識別子のセットを決定する。本発明の一部の実施形態では、テナント識別子決定ｍｏｄ３０６が、要求のセットに対応するテナント識別子のセットを決定する。一部の実施形態では、テナント識別子決定ｍｏｄ３０６が、動的インスタンス化である要求元のセットのテナント識別子のセットを決定する。代替の実施形態では、テナント識別子決定ｍｏｄ３０６が、仮想コンテナのセットのテナント識別子のセットを決定する。さらに別の実施形態では、テナント識別子決定ｍｏｄ３０６が、ソフトウェア・ライブラリ・フレームワークのセットのテナント識別子のセットを決定する。代替として、テナント識別子決定ｍｏｄ３０６が、ユーザのセットのテナント識別子のセットを決定する。一部の実施形態では、テナント識別子決定ｍｏｄ３０６が、テナントのセットのインスタンスのセットのテナント識別子のセットを決定する。一部の実施形態では、テナント識別子決定ｍｏｄ３０６が、動作Ｓ２５５で要求受信ｍｏｄ３０２によって受信された要求のセットに対応するテナント識別子のセットを決定する。代替として、テナント識別子決定ｍｏｄ３０６が、動作Ｓ２６０でディレクトリ決定ｍｏｄ３０４によって決定されたディレクトリのセットに対応するテナント識別子のセットを決定する。

【００４１】

処理が動作Ｓ２７０に進み、コネクタ・サービス割り当てｍｏｄ３０８がコネクタ・サービスを割り当てる。本発明の一部の実施形態では、コネクタ・サービス割り当てｍｏｄ３０８がコネクタ・サービスを割り当てる。さらに別の実施形態では、コネクタ・サービスはコンピュータ・システム上の唯一のコネクタ・サービスである。代替として、コネクタ・サービスは、第１の分散ファイル・システムおよび第２の分散ファイル・システムに関連付けられた唯一のコネクタ・サービスである。これらの実施形態の一部では、コネクタ・サービスが、第１の分散ファイル・システム上の要求元のセットからの要求を、第２の分散ファイル・システムに向けて方向付ける。他の実施形態では、コネクタ・サービス割り当てｍｏｄ３０８が、少なくとも一部において、テナント識別子のセットに基づいて、コネクタ・サービスを割り当てる。さらに別の実施形態では、コネクタ・サービス割り当てｍｏｄ３０８が、少なくとも一部において、ディレクトリのセットに基づいて、コネクタ・サービスを割り当てる。コネクタ・サービスは、接続サーバと呼ばれることもある。コネクタ・サービスは、適切なチャネルのセットを介して、要求のセットを方向付ける。接続サーバは、（ｉ）ユーザのセットを認証すること、（ｉｉ）リソースのセットに対する資格をユーザのセットに与えること、（ｉｉｉ）パッケージのセットをリソースのセットに割り当てること、（ｉｖ）ローカル・セッションもしくはリモート・セッションまたはその両方を管理すること、（ｖ）セキュリティで保護された接続のセットを確立する

こと、または（ｖｉ）ポリシーを適用すること、あるいはその組み合わせを実行することを含むが、これらに限定されない、機能を実行してもよい。一部の実施形態では、コネクタ・サービス割り当てｍｏｄ３０８が、少なくとも一部において、動作Ｓ２５５で要求受信ｍｏｄ３０２によって受信された要求のセットの要求元のセットに基づいて、コネクタ・サービスを割り当てる。他の実施形態では、コネクタ・サービス割り当てｍｏｄ３０８が、少なくとも一部において、動作Ｓ２５５で要求受信ｍｏｄ３０２によって受信された要求のセットに基づいて、コネクタ・サービスを割り当てる。さらに別の実施形態では、コネクタ・サービス割り当てｍｏｄ３０８が、少なくとも一部において、動作Ｓ２６０でディレクトリ決定ｍｏｄ３０４によって決定されたディレクトリのセットに基づいて、コネクタ・サービスを割り当てる。代替の実施形態では、コネクタ・サービス割り当てｍｏｄ３０８が、少なくとも一部において、動作Ｓ２６５でテナント識別子決定ｍｏｄ３０６によって決定されたテナント識別子のセットに基づいて、コネクタ・サービスを割り当てる。

【００４２】

処理が動作Ｓ２７５に進み、ノード決定ｍｏｄ３１０が、要求元のセットに対応するノードを決定する。本発明の一部の実施形態では、ノード決定ｍｏｄ３１０が、要求元のセットに対応するノードを決定する。これらの実施形態の一部では、ノード決定ｍｏｄ３１０が、第１のノードが要求元のセット内の各要求元に対応するということを決定する。これらの実施形態の一部では、ノード決定ｍｏｄ３１０が、物理ノードが要求元のセットに対応するということを決定する。他の実施形態では、ノード決定ｍｏｄ３１０が、仮想ノードが要求元のセットに対応するということを決定する。代替の実施形態では、ノード決定ｍｏｄ３１０が、要求元のセット内の各要求元を第１のノードに割り当てることによって、要求元のセットに対応するノードを決定する。一部の実施形態では、ノード決定ｍｏｄ３１０が、要求のセットに対応するノードを決定する。さらに別の実施形態では、ノード決定ｍｏｄ３１０が、テナント識別子のセットに対応するノードを決定する。他の実施形態では、ノード決定ｍｏｄ３１０が、少なくとも一部において、コネクタ・サービスに基づいて、ノードを決定する。代替の実施形態では、ノード決定ｍｏｄ３１０が、少なくとも一部において、ノードとコネクタ・サービスの間の１対１の関係に基づいて、ノードを決定する。他の実施形態では、ノード決定３１０が、コネクタ・サービスとノードの間の経路をマッピングする。一部の実施形態では、ノード決定ｍｏｄ３１０が、要求受信ｍｏｄ３０２が動作Ｓ２５５で受信した要求のセットの送信元である、要求元のセットに対応するノードを決定する。他の実施形態では、ノード決定ｍｏｄ３１０が、動作Ｓ２５５で要求受信ｍｏｄ３０２によって受信された要求のセットに対応するノードを決定する。さらに別の実施形態では、ノード決定ｍｏｄ３１０が、動作Ｓ２６０でディレクトリ決定ｍｏｄ３０４によって決定されたディレクトリのセットに対応するノードを決定する。代替の実施形態では、ノード決定ｍｏｄ３１０が、動作Ｓ２６５でテナント識別子決定ｍｏｄ３０６によって決定されたテナント識別子のセットに対応するノードを決定する。代替として、ノード決定ｍｏｄ３１０が、少なくとも一部において、動作Ｓ２７０でコネクタ・サービス割り当てｍｏｄ３０８によって割り当てられたコネクタ・サービスに基づいて、ノードを決定する。

【００４３】

処理が動作Ｓ２８０に進み、要求処理ｍｏｄ３１２が要求のセットを処理する。本発明の一部の実施形態では、要求処理ｍｏｄ３１２が要求のセットを処理する。一部の実施形態では、要求処理ｍｏｄ３１２が、少なくとも一部において、テナント識別子のセットに基づいて、要求のセットを処理する。他の実施形態では、要求処理ｍｏｄ３１２が、少なくとも一部において、ノードに基づいて、要求のセットを処理する。さらに別の実施形態では、要求処理ｍｏｄ３１２が、少なくとも一部において、ディレクトリに基づいて、要求のセットを処理する。一部の実施形態では、要求処理ｍｏｄ３１２が、第１の分散ファイル・システムを第２の分散ファイル・システムにマウントする。代替の実施形態では、要求処理ｍｏｄ３１２が、少なくとも一部において、コネクタ・サービスに基づいて、要

求のセットを処理する。読み取り要求の場合、要求処理ｍｏｄ３１２が、データのセットをストレージから読み取る。書き込み要求の場合、要求処理ｍｏｄ３１２が、ストレージ内のデータのセットを変更する。入力要求の場合、要求処理ｍｏｄ３１２が、データのセットを受信する。出力要求の場合、要求処理ｍｏｄ３１２が、データのセットを送信する。一部の実施形態では、要求処理ｍｏｄ３１２が、動作Ｓ２５５で要求受信ｍｏｄ３１２によって受信された要求のセットを処理する。他の実施形態では、要求処理ｍｏｄ３１２が、少なくとも一部において、動作Ｓ２６５でテナント識別子決定ｍｏｄ３０６によって決定されたテナント識別子のセットに基づいて、要求のセットを処理する。さらに別の実施形態では、要求処理ｍｏｄ３１２が、少なくとも一部において、動作Ｓ２７５でノード決定ｍｏｄ３１０によって決定されたノードに基づいて、要求のセットを処理する。他の実施形態では、要求処理ｍｏｄ３１２が、少なくとも一部において、動作Ｓ２６０でディレクトリ決定ｍｏｄ３０４によって決定されたディレクトリのセットに基づいて、要求のセットを処理する。代替の実施形態では、要求処理ｍｏｄ３１２が、少なくとも一部において、動作Ｓ２７０でコネクタ・サービス決定ｍｏｄ３０８によって決定されたコネクタ・サービスに基づいて、要求のセットを処理する。

【００４４】

動作Ｓ２８５で処理が終了し、結果生成ｍｏｄ３１４が結果のセットを生成する。本発明の一部の実施形態では、結果生成ｍｏｄ３１４が、要求のセットの結果のセットを生成する。一部の実施形態では、結果生成ｍｏｄ３１４が、データのセットを含んでいるメッセージのセットを生成することによって、読み取り要求のセットに対する結果のセットを生成する。一部の実施形態では、結果生成ｍｏｄ３１４が、新しいデータ・エントリのセットを生成することによって、書き込み要求のセットに対する結果のセットを生成する。一部の実施形態では、結果生成ｍｏｄ３１４が、受信されたデータのセットを格納することによって、入力要求のセットに対する結果のセットを生成する。一部の実施形態では、結果生成ｍｏｄ３１４が、メッセージのセットを生成することによって、出力要求のセットに対する結果のセットを生成する。他の実施形態では、結果生成ｍｏｄ３１４が、ＰＯＳＩＸ互換ではない第１の分散ファイル・システムの結果を生成する。さらに別の実施形態では、結果生成ｍｏｄ３１４が、Ｈａｄｏｏｐである第１の分散ファイル・システムの結果のセットを生成する。他の実施形態では、結果は、新しいデータ・エントリまたはデータのセットを含むメッセージあるいはその両方を含むが、これらに限定されない。一部の実施形態では、結果生成ｍｏｄ３１４が、動作Ｓ２５５で要求受信ｍｏｄ３０２によって受信された要求のセットに対する結果のセットを生成する。

【００４５】

ＩＩＩ．追加のコメントまたは実施形態あるいはその両方

本発明の一部の実施形態は、現在の技術に対して改善を行うために、次の要因、潜在的な問題、または潜在的領域、あるいはこれらすべてを認識する。（ｉ）テナント識別子のセットに対応するノードのセット、コネクタ・サービスのセット、またはディレクトリのセット、あるいはその組み合わせを管理することが、リソースにおける指数関数的増加につながる、（ｉｉ）さまざまなオペレーティング・システムが、ノードのセット、コネクタ・サービスのセット、またはディレクトリのセット、あるいはその組み合わせを多数の方法で処理する、（ｉｉｉ）一部の分散ファイル・システム（ＤＦＳ：distributed file systems）が、ポータブル・オペレーティング・システム・インターフェイス（ＰＯＳＩＸ：portable operating system interface）互換ではない、（ｉｖ）一部のＤＦＳをマウントできない、または（ｖ）超集中型インフラストラクチャが、リソースの使用量を減らそうとする、あるいはその組み合わせ。テナント識別子のセットに対応するノードのセット、コネクタ・サービスのセット、またはディレクトリのセット、あるいはその組み合わせを管理する従来の手段では、各テナント識別子に対応する個別のノードおよび個別のディレクトリが必要になる。

【００４６】

図４は、本発明に従う方法を表すフローチャート４００を示している。処理が動作Ｓ４

０５で開始し、マルチテナント構成サブシステムが、Ｉ／Ｏ要求をＨａｄｏｏｐコンテナ・インスタンスから受信する。処理が動作Ｓ４１０に進み、マルチテナント構成サブシステムが、Ｈａｄｏｏｐコンテナ・インスタンスのテナント識別子のセットを分離する。処理が動作Ｓ４１５に進み、マルチテナント構成サブシステムが、少なくとも一部において、テナント識別子のセットに基づいて、Ｈａｄｏｏｐコンテナ・インスタンスを認識する。処理が動作Ｓ４２０に進み、マルチテナント構成サブシステムが、Ｈａｄｏｏｐコンテナ・インスタンスの許可のセットをチェックする。処理が動作Ｓ４２５で終了し、マルチテナント構成サブシステムが、Ｉ／Ｏ要求を処理する。

【００４７】

図５は、Ｈａｄｏｏｐインスタンス５０２、Ｈａｄｏｏｐインスタンス５０４、Ｈａｄｏｏｐインスタンス５０６、コネクタ・サービス５０８、分散ファイル・システム５１０、および物理ノード５１２を含むシステム５００の機能ブロック図を示している。Ｈａｄｏｏｐインスタンス５０２、Ｈａｄｏｏｐインスタンス５０４、およびＨａｄｏｏｐインスタンス５０６の各々と、分散ファイル・システム５１０との間の通信は、コネクタ・サービス５０８を通って横断する。分散ファイル・システム５１０は、物理ノード５１２上に存在することによって、コネクタ・サービス５０８を介してすべての通信を処理できる。

【００４８】

本発明の一部の実施形態は、次の機能、特徴、または長所、あるいはその組み合わせのうちの１つまたは複数を含んでよい。（ｉ）ＤＦＳインスタンス・データのセットを分離すること、（ｉｉ）Ｈａｄｏｏｐインスタンス・データのセットを分離すること、（ｉｉｉ）マルチテナント認識モジュールをＤＦＳコネクタ・サービスに導入すること、または（ｉｖ）マルチテナント能力を超集中型ＤＦＳに提供すること、あるいはその組み合わせ。超集中型ＤＦＳは、マルチテナントＤＦＳと呼ばれることもある。本発明の一部の実施形態では、マルチテナント認識モジュールは、図４の動作Ｓ４１０およびＳ４１５を組み込む。他の実施形態では、図５のコネクタ・サービス５０８は、図４の動作Ｓ４１０または動作Ｓ４１５あるいはその両方を実行する。さらに別の実施形態では、マルチテナント構成サブシステムは、コネクタ・サービスおよび物理ノードを１対１の関係で提供する。代替の実施形態では、マルチテナント構成サブシステムは、プライベート・ネットワーク・アドレスのセットを使用してＤＦＳインスタンスのセットを構成する。代替として、マルチテナント構成サブシステムは、プライベート・ネットワーク・アドレスを使用してＤＦＳインスタンスのセットを構成する。一部の実施形態では、マルチテナント構成サブシステムは、ディレクトリ内のＤＦＳインスタンスを分離する。さらに別の実施形態では、マルチテナント構成サブシステムは、少なくとも一部において、テナントに基づいてディレクトリ内のＤＦＳインスタンスを分離する。他の実施形態では、マルチテナント構成サブシステムは、ディレクトリ内のＤＦＳインスタンスの動作のセットを分離する。

【００４９】

図６は、２つのテーブルを示している。図６の第１のテーブルは、インスタンス・コンテナ・マッピング・リスト（instance container mapping list）である。３つのコンテナを含む２つのインスタンスが示されており、そのため、６つのテナントＩＤが含まれている。これらの６つのテナントＩＤは、すべて１つのノードにマッピングされている。図６の第２のテーブルは、逆インスタンス・コンテナ・マッピング・リスト（reverse instance container mapping list）である。同じ６つのテナントＩＤが示されている。ただし第２のテーブルは、対応するインスタンスを決定するために分類される。

【００５０】

図７は、本発明に従う方法を表すフローチャート７００を示している。処理が動作Ｓ７０５で開始し、マルチテナント構成サブシステムが、Ｉ／Ｏ読み取り／書き込み要求をコンテナ内のＨａｄｏｏｐジョブから受信する。処理が動作Ｓ７１０に進み、マルチテナント構成サブシステムが、コンテナのＩＰアドレスをＩ／Ｏ要求から取得する。処理が動作Ｓ７１５に進み、マルチテナント構成サブシステムが、物理ノードのＩＰアドレスを取得する。処理が動作Ｓ７２０に進み、マルチテナント構成サブシステムが、コンテナのＩＰ

およびノードのＩＰに基づいて、インスタンス・コンテナ・マッピング・リストを照会する。処理が動作Ｓ７２５に進み、マルチテナント構成サブシステムが、インスタンスＩＰを取得する。処理が動作Ｓ７３０に進み、マルチテナント構成サブシステムが、インスタンスのディレクトリを取得する。処理が動作Ｓ７３５に進み、マルチテナント構成サブシステムが、Ｉ／Ｏ経路のセットを変換する。処理が動作Ｓ７４０で終了し、マルチテナント構成サブシステムが、Ｉ／Ｏ要求のセットを処理する。

【００５１】

本発明の一部の実施形態は、次の機能、特徴、または長所、あるいはその組み合わせのうちの１つまたは複数を含んでよい。（ｉ）ＤＦＳが、さまざまなホストからのファイルのセットへのアクセスを可能にするか、（ｉｉ）ＤＦＳが、ユーザのセットが、デバイスのセットにわたってファイルのセットを共有できるようにするか、または（ｉｉｉ）ＤＦＳが一般的なストレージ・システムであるか、あるいはその組み合わせである。ＤＦＳの例としては、ＩＢＭ Ｇｅｎｅｒａｌ Ｐａｒａｌｌｅｌ Ｆｉｌｅ Ｓｙｓｔｅｍ（「ＧＰＦＳ（ＴＭ）」）Ｆｉｌｅ Ｐｌａｃｅｍｅｎｔ Ｏｐｔｉｍｉｚｅｒ （「ＦＰＯ」）、Ｒｅｄ Ｈａｔ Ｌｉｎｕｘ（Ｒ）、ＧｌｕｓｔｅｒＦＳ、Ｌｕｓｔｒｅ、Ｃｅｐｈ、およびＡｐａｃｈｅ Ｈａｄｏｏｐ Ｄｉｓｔｒｉｂｕｔｅｄ Ｆｉｌｅ Ｓｙｓｔｅｍ（「ＨＤＦＳ」）が挙げられる。ＩＢＭおよびＧＰＦＳは、世界中の多くの管轄区域で登録されている、Ｉｎｔｅｒｎａｔｉｏｎａｌ Ｂｕｓｉｎｅｓｓ Ｍａｃｈｉｎｅｓ Ｃｏｒｐｏｒａｔｉｏｎの商標である。Ｌｉｎｕｘは、米国またはその他の国あるいはその両方における、Ｌｉｎｕｓ Ｔｏｒｖａｌｄｓの登録商標である。

【００５２】

本発明の一部の実施形態は、次の機能、特徴、または長所、あるいはその組み合わせのうちの１つまたは複数を含んでよい。（ｉ）ＤＦＳをマウントすること、（ｉｉ）データをＤＦＳから読み取ること、（ｉｉｉ）データをＤＦＳに書き込むこと、（ｉｖ）ＰＯＳＩＸアプリケーションを使用してデータをＤＦＳから読み取ること、（ｖ）ＰＯＳＩＸアプリケーションを使用してデータをＤＦＳに書き込むこと、（ｖｉ）ＤＦＳのエコシステム内でＰＯＳＩＸアプリケーションを使用してデータをＤＦＳから読み取ること、または（ｖｉｉ）ＤＦＳのエコシステム内でＰＯＳＩＸアプリケーションを使用してデータをＤＦＳに書き込むこと、あるいはその組み合わせ。本発明の一部の実施形態は、次の機能、特徴、または長所、あるいはその組み合わせのうちの１つまたは複数を含んでよい。（ｉ）少なくとも一部において、ユーザＩＤに基づいて許可のセットを決定すること、（ｉｉ）少なくとも一部において、グループＩＤに基づいて許可のセットを決定すること、（ｉｉｉ）動作環境に関する許可のセットを決定すること、または（ｉｖ）オペレーティング・システムに関する許可のセットを決定すること、あるいはその組み合わせ。

【００５３】

本発明の一部の実施形態は、次の機能、特徴、または長所、あるいはその組み合わせのうちの１つまたは複数を含んでよい。（ｉ）ＰＯＳＩＸアプリケーションを使用してＤＦＳを実行すること、（ｉｉ）単一のコネクタ・サービスを経由してファイルのセットを転送すること、（ｉｉｉ）ＰＯＳＩＸアプリケーションを使用してＤＦＳ上で単一のコネクタ・サービスを経由してファイルのセットを転送すること、または（ｉｖ）ＰＯＳＩＸアプリケーションを使用して超集中型ＤＦＳを実行すること、あるいはその組み合わせ。本発明の一部の実施形態は、次の機能、特徴、または長所、あるいはその組み合わせのうちの１つまたは複数を含んでよい。（ｉ）非ＰＯＳＩＸアプリケーションを使用してＤＦＳを実行すること、（ｉｉ）単一のコネクタ・サービスを経由してファイルのセットを転送すること、（ｉｉｉ）非ＰＯＳＩＸアプリケーションを使用してＤＦＳ上で単一のコネクタ・サービスを経由してファイルのセットを転送すること、または（ｉｖ）非ＰＯＳＩＸアプリケーションを使用して超集中型ＤＦＳを実行すること、あるいはその組み合わせ。本発明の一部の実施形態は、次の機能、特徴、または長所、あるいはその組み合わせのうちの１つまたは複数を含んでよい。（ｉ）ＤＦＳインスタンスのセットのクラスタのセットを作成すること、（ｉｉ）ユーザのセットのＤＦＳインスタンスのセットのクラスタの

セットを作成すること、（ｉｉｉ）ネットワーク・アドレスのセットをクラスタのセットに割り当てること、（ｉｖ）テナント識別子のセットをクラスタのセットに割り当てること、（ｖ）ネットワーク・アドレスのセットをクラスタのセットに割り当てることであって、ネットワーク・アドレスのセットがＤＦＳに関連していない、こと、または（ｖｉ）テナント識別子のセットをクラスタのセットに割り当てることであって、ネットワーク・アドレスのセットがＤＦＳに関連していない、こと、あるいはその組み合わせ。

【００５４】

本発明の一部の実施形態は、次の機能、特徴、または長所、あるいはその組み合わせのうちの１つまたは複数を含んでよい。（ｉ）コネクタ・サービスの数を減らすこと、（ｉｉ）単一のコネクタ・サービスを使用すること、（ｉｉｉ）マルチテナント構成を維持するために必要なコネクタ・サービスの数を減らすこと、（ｉｖ）指数関数的レベルでマルチテナント構成を維持するために必要なコネクタ・サービスの数を減らすこと、（ｖ）ＤＦＳ上のクライアントの数に対応するテナント識別子の数を減らすこと、または（ｖｉ）ＤＦＳ上のクライアントの数に対応するＩＰアドレスの数を減らすこと、あるいはその組み合わせ。

【００５５】

本発明の一部の実施形態では、マルチテナント構成サブシステムがテナントのＤＦＳクラスタを生成する。さらに別の実施形態では、マルチテナント構成サブシステムは、ＤＦＳクラスタに対応するテナントＩＤを生成する。ＤＦＳクラスタは、複数の要求元または複数のテナントあるいはその両方を含む第１の分散ファイル・システムと呼ばれることもある。これらの実施形態の一部では、マルチテナント構成サブシステムは、テナントＩＤをノードに割り当てる。

【００５６】

本発明の一部の実施形態は、次の機能、特徴、または長所、あるいはその組み合わせのうちの１つまたは複数を含んでよい。（ｉ）ＤＦＳ内のディレクトリのセットを構成すること、（ｉｉ）ＤＦＳ内のディレクトリのセットを構成し、コネクタ・サービスを再開すること、（ｉｉｉ）ＤＦＳインスタンスのソフトウェア・ライブラリ・フレームワーク・インスタンスのセットを作成すること、（ｉｖ）テナント情報のセットを超集中型ＤＦＳ内のディレクトリに格納すること、（ｖ）再開せずにＤＦＳのディレクトリを認識すること、（ｖｉ）新しいＤＦＳインスタンスを作成せずにＤＦＳを再開すること、（ｖｉｉ）ＤＦＳクラスタをテナントに提供すること、（ｖｉｉｉ）テナントのＤＦＳクラスタを維持すること、または（ｉｘ）少なくとも一部において、ハードウェア・リソースのセットに基づいてＤＦＳを分離すること、あるいはその組み合わせ。本発明の一部の実施形態は、次の機能、特徴、または長所、あるいはその組み合わせのうちの１つまたは複数を含んでよい。（ｉ）ソフトウェア・ライブラリ・フレームワークを構築するときに、ユーザＩＤを生成すること、（ｉｉ）ソフトウェア・ライブラリ・フレームワークをコンパイルするときに、ユーザＩＤを生成すること、（ｉｉｉ）ソフトウェア・ライブラリ・フレームワークを構築するときに、グループＩＤを生成すること、または（ｉｖ）ソフトウェア・ライブラリ・フレームワークをコンパイルするときに、グループＩＤを生成すること、あるいはその組み合わせ。

【００５７】

本発明の一部の実施形態は、次の機能、特徴、または長所、あるいはその組み合わせのうちの１つまたは複数を含んでよい。（ｉ）超集中型ビッグ・データＤＦＳを管理すること、（ｉｉ）マルチテナント・ビッグ・データＤＦＳを管理すること、（ｉｉｉ）クラウド・システム内の超集中型ＤＦＳを管理すること、または（ｉｖ）仮想システム内の超集中型ＤＦＳを管理すること、あるいはその組み合わせ。

【００５８】

ＩＶ．定義

「本発明」は、説明された対象が、出願時の初期の一連の請求項によって、審査中に書かれた修正された一連の請求項によって、または特許審査によって許可され、交付済み特

許に含まれる最終的な一連の請求項によって、あるいはその組み合わせによって、カバーされるということの、絶対的な指示または暗示あるいはその両方を生み出さない。「本発明」という用語は、最先端の技術を超える１つまたは複数の進歩を含んでいる可能性のある本開示の１つまたは複数の部分を示すことを補助するために使用される。「本発明」という用語およびその指示または暗示あるいはその両方についてのこの理解は、一時的かつ暫定的であり、関連する情報が開発されたとき、および請求項が修正されたときに、特許審査の過程で変更される可能性がある。

【００５９】

「実施形態」については、「本発明」の定義を参照する。

【００６０】

「または～あるいはその組み合わせ（And/or）」は、包含的離接であり、論理離接とも呼ばれ、一般に「包含的論理和」と呼ばれる。例えば、「Ａ、Ｂ、またはＣ、あるいはその組み合わせ」という語句は、ＡまたはＢまたはＣのうちの少なくとも１つが真であるということを意味しており、「Ａ、Ｂ、またはＣ、あるいはその組み合わせ」は、ＡおよびＢおよびＣの各々が偽である場合にのみ、偽になる。

【００６１】

項目「のセット」は、１つまたは複数の項目が存在しており、少なくとも１つの項目が存在しなければならないが、２つ、３つ、またはそれ以上の項目が存在する可能性もあるということを意味する。項目「のサブセット」は、共通の特徴を含んでいる項目のグループ内に１つまたは複数の項目が存在するということを意味する。

【００６２】

「複数の」項目は、２つ以上の項目が存在しており、少なくとも２つの項目が存在しなければならないが、３つ、４つ、またはそれ以上の項目が存在する可能性もあるということを意味する。

【００６３】

「含む（Includes）」およびその変形（例えば、含んでいる（including）、含む（include）など）は、特に明示的に示されない限り、「～を含むが、必ずしもそれに限定されない」ということを意味する。

【００６４】

「ユーザ」または「サブスクライバ」は、（ｉ）１人の個人、（ｉｉ）１人の個人または２人以上の人間の代わりに行動するための十分な知能を有する人工知能の実体、（ｉｉｉ）１人の個人または２人以上の人間によって活動が行われている企業実体、または（ｉｖ）１つの「ユーザ」または「サブスクライバ」として行動している任意の１つまたは複数の関連する「ユーザ」または「サブスクライバ」の組み合わせ、あるいはその組み合わせを含むが、必ずしもこれらに限定されない。

【００６５】

「受信する」、「提供する」、「送信する」、「入力する」、「出力する」、および「報告する」という用語は、特に明示的に指定されない限り、（ｉ）物体と対象の間の関係に関する直接性の特定の程度、または（ｉｉ）物体と対象の間に挿入された中間的なコンポーネント、中間的な動作、もしくは物、またはその組み合わせのセットの存在もしくは不在、あるいはその両方を示しているか、または暗示していると受け取られるべきではない。

【００６６】

「モジュール」は、ハードウェア、ファームウェア、またはソフトウェア、あるいはその組み合わせの任意のセットであり、モジュールが次のいずれの状態であるかに関わらず、機能を実行するように動作可能である。（ｉ）１つにまとまって局所的に近接している、（ｉｉ）広い領域にわたって分散している、（ｉｉｉ）大きい一群のソフトウェア・コード内で１つにまとまって近接している、（ｉｖ）一群のソフトウェア・コード内に存在する、（ｖ）１つのストレージ・デバイス、メモリ、または媒体内に存在する、（ｖｉ）機械的に接続されている、（ｖｉｉ）電気的に接続されている、または（ｖｉｉｉ）デー

タ通信で接続されている、あるいはその組み合わせの状態にある。「サブモジュール」は、「モジュール」内の「モジュール」である。

【００６７】

「コンピュータ」は、大きいデータ処理能力または機械可読命令読み取り能力あるいはその両方を備える任意のデバイスであり、デスクトップ・コンピュータ、メインフレーム・コンピュータ、ラップトップ・コンピュータ、フィールドプログラマブル・ゲート・アレイ（ＦＰＧＡ）ベース・デバイス、スマートフォン、パーソナル・デジタル・アシスタント（ＰＤＡ）、人体に装着または挿入されるコンピュータ、組み込みデバイス型コンピュータ、または特定用途向け集積回路（ＡＳＩＣ：application-specific integrated circuit）ベース・デバイス、あるいはその組み合わせを含むが、必ずしもこれらに限定されない。

【００６８】

「電気的に接続される」とは、介在する要素が存在するように間接的に電気的に接続されるか、または直接的に電気的に接続されることを意味する。「電気接続」は、コンデンサ、インダクタ、変圧器、真空管などの要素を含んでよいが、これらに限定される必要はない。

【００６９】

「機械的に接続される」とは、中間的なコンポーネントを介して行われる間接的な機械的接続または直接的な機械的接続のいずれかを意味する。「機械的に接続される」は、固定された機械的接続、および機械的に接続されコンポーネント間の相対運動を許容する機械的接続を含む。「機械的に接続される」は、溶接接続、半田接続、締め具（例えば、くぎ、ボルト、ねじ、ナット、面ファスナ、結び目、リベット、簡易脱着接続、掛け金、または磁気的接続、あるいはその組み合わせ）による接続、圧力ばめ接続、摩擦適合接続、重力によって引き起こされるかみ合いによって固定される接続、旋回可能もしくは回転可能な接続、またはスライド可能な機械的接続、あるいはその組み合わせを含むが、これらに限定されない。

【００７０】

「データ通信」は、現在知られているか、または将来開発される任意の種類のデータ通信方式を含むが、必ずしもこれらに限定されない。「データ通信」は、無線通信、有線通信、または無線部分および有線部分を含む通信経路、あるいはその組み合わせを含むが、必ずしもこれらに限定されない。「データ通信」は、（ｉ）直接データ通信、（ｉｉ）間接データ通信、または（ｉｉｉ）形式、パケット化の状態、媒体、暗号化の状態、もしくはプロトコル、またはその組み合わせが、データ通信の過程全体にわたって一定のままであるデータ通信、あるいはその組み合わせに、必ずしも限定されない。

【００７１】

「実質的に人間が介入しない」という語句は、人間による入力がほとんど、または全く存在せずに（多くの場合、ソフトウェアなどの機械論理の動作によって）自動的に発生するプロセスを意味する。「実質的に人間が介入しない」ことを含む例としては、（ｉ）コンピュータが複雑な処理を実行しており、電力系統の停止に起因して処理の継続が中断されないように、人間がコンピュータを代替電源に切り替えること、（ｉｉ）コンピュータが、リソースを大量に使用する処理を実行しようとしており、人間が、リソースを大量に使用する処理が本当に実行されるべきであるかどうかを確認すること（この場合、分離していると考えられる確認のプロセスは、実質的な人間の介入を伴うが、人間によって行われる必要のある単純なはい／いいえ形式の確認にもかかわらず、リソースを大量に使用する処理は、どのような実質的な人間の介入も含まない）、および（ｉｉｉ）コンピュータが、機械論理を使用して重要な決定（例えば、悪天候を予想して、すべての飛行機を飛行禁止にすることの決定）を行うが、重要な決定を実施する前に、コンピュータが、単純なはい／いいえ形式の確認を人間側から得なければならないこと、が挙げられる。

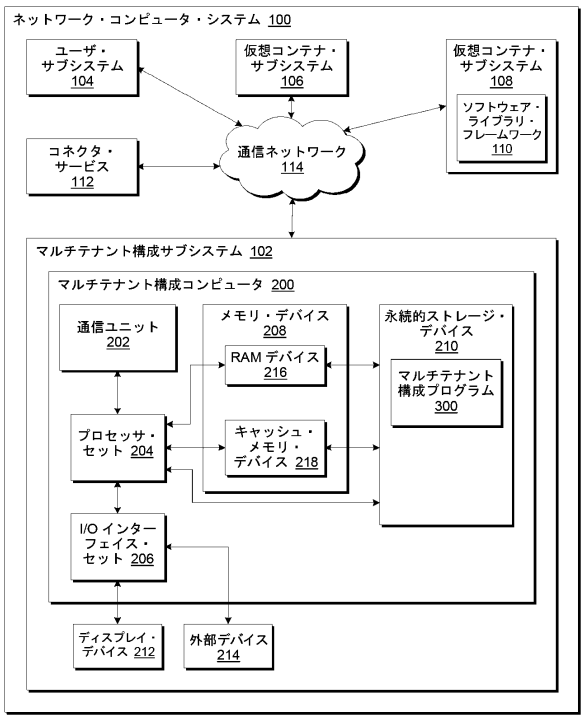
【００７２】

「自動的に」とは、「人間による介入なしで」ということを意味する。

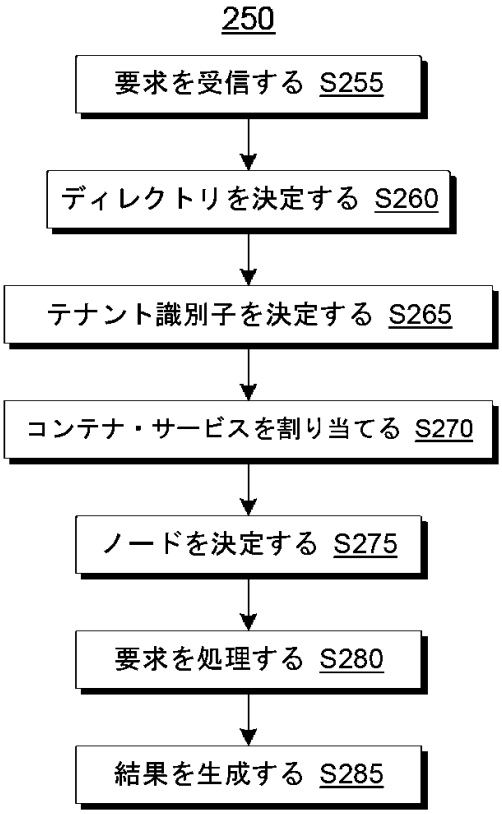
【００７３】
「リアルタイム」（および形容詞の「リアルタイムの」）という用語は、前述した情報処理で妥当な応答時間を実現するために十分短い期間である、任意の時間枠を含む。さらに、「リアルタイム」（および形容詞の「リアルタイムの」）という用語は、一般に「ほぼリアルタイム」と呼ばれる、前述したオンデマンドの情報処理で妥当な応答時間を実現するために十分短い期間である、任意の（例えば、１秒未満または数秒以内の）時間枠を通常は含む。これらの用語は、正確に定義するのは困難であるが、当業者によって十分理解されている。
【符号の説明】
【００７４】
１００　ネットワーク・コンピュータ・システム
１０２　マルチテナント構成サブシステム
１０４　ユーザ・サブシステム
１０６，１０８　仮想コンテナ・サブシステム
１１０　ソフトウェア・ライブラリ・フレームワーク
１１１　コネクタ・サービス
１１４　通信ネットワーク
２００　マルチテナント構成コンピュータ
２０１　通信ユニット
２０４　プロセッサ・ユニット
２０６　Ｉ／Ｏインターフェイス・セット
２０８　メモリ・デバイス
２１０　永続的ストレージ・デバイス
２１２　ディスプレイ・デバイス
２１４　外部デバイス
２１６　ＲＡＭデバイス
２１８　キャッシュ・メモリ・デバイス
３００　マルチテナント構成プログラム

【図面】

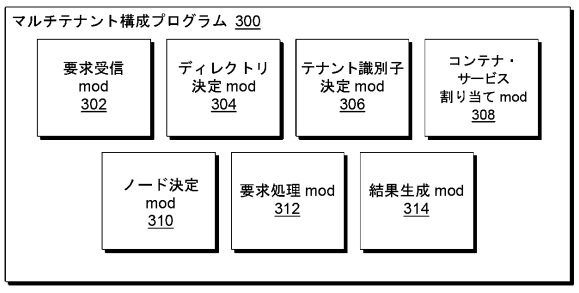
【図１】


ネットワーク・コンピュータ・システム 100
ユーザ・サブシステム 104
仮想コンテナ・サブシステム 106
仮想コンテナ・サブシステム 108
ソフトウェア・ライブラリ・フレームワーク 110
コネクタ・サービス 112
通信ネットワーク 114
マルチテナント構成サブシステム 102
マルチテナント構成コンピュータ 200
通信ユニット 202
メモリ・デバイス 208
RAM デバイス 216
永続的ストレージ・デバイス 210
マルチテナント構成プログラム 300
プロセッサ・セット 204
キャッシュ・メモリ・デバイス 218
I/O インターフェイス・セット 206
ディスプレイ・デバイス 212
外部デバイス 214

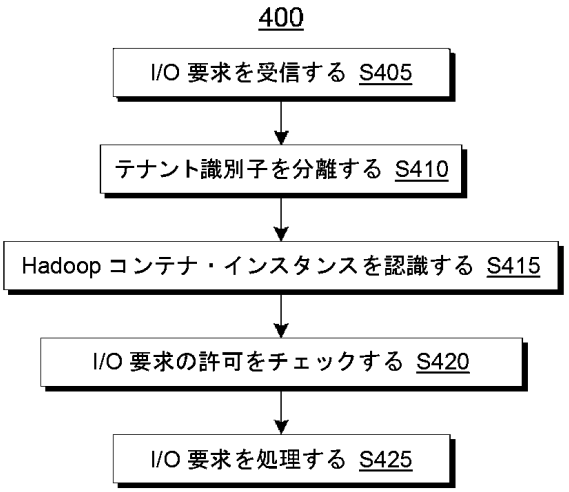
【図２】


250
要求を受信する S255
ディレクトリを決定する S260
テナント識別子を決定する S265
コンテナ・サービスを割り当てる S270
ノードを決定する S275
要求を処理する S280
結果を生成する S285

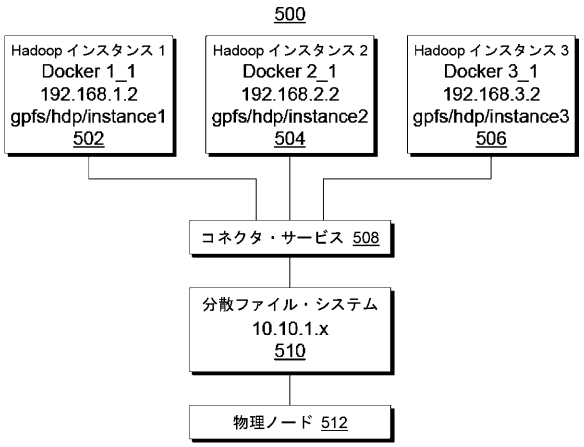
【図３】


マルチテナント構成プログラム 300
要求受信 mod 302
ディレクトリ決定 mod 304
テナント識別子決定 mod 306
コンテナ・サービス割り当て mod 308
ノード決定 mod 310
要求処理 mod 312
結果生成 mod 314

【図４】


400
I/O 要求を受信する S405
テナント識別子を分離する S410
Hadoop コンテナ・インスタンスを認識する S415
I/O 要求の許可をチェックする S420
I/O 要求を処理する S425

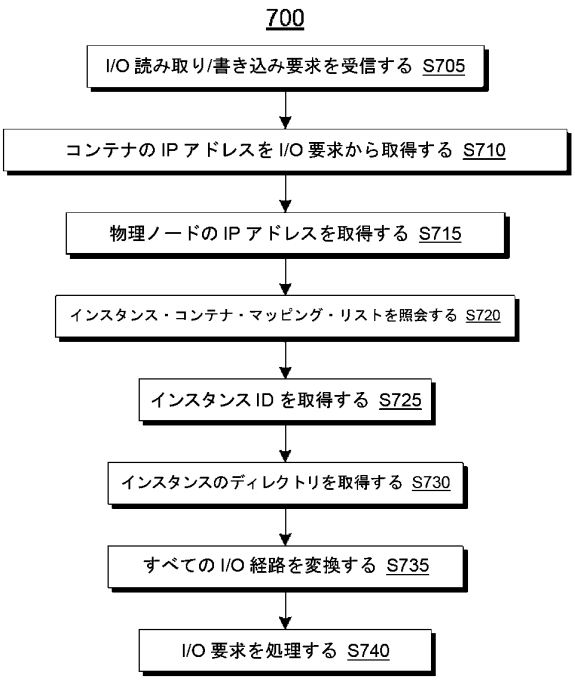
【図５】



【図６】

インスタンス	コンテナ	テナント ID	ノード
インスタンス 1	コンテナ 1_1	192.168.1.2	ノード 1
インスタンス 1	コンテナ 2_1	192.168.1.3	ノード 1
インスタンス 1	コンテナ 3_1	192.168.1.4	ノード 1
インスタンス 2	コンテナ 1_2	192.168.2.2	ノード 1
インスタンス 2	コンテナ 2_2	192.168.2.3	ノード 1
インスタンス 2	コンテナ 3_2	192.168.2.4	ノード 1
ノード	コンテナ	テナント ID	インスタンス
ノード 1	コンテナ 1_1	192.168.1.2	インスタンス 1
ノード 1	コンテナ 2_1	192.168.1.3	インスタンス 1
ノード 1	コンテナ 3_1	192.168.1.4	インスタンス 1
ノード 1	コンテナ 1_2	192.168.2.2	インスタンス 2
ノード 1	コンテナ 2_2	192.168.2.3	インスタンス 2
ノード 1	コンテナ 3_2	192.168.2.4	インスタンス 2

【図７】



フロントページの続き

(33)優先権主張国・地域又は機関
米国(US)
(72)発明者 チェン、ヨン
中華人民共和国１１１００１９３ ベイジン ドンベイワン ウエスト・ロード８ ジョングァンツン・ソフトウェア・パーク２８
(72)発明者 ユアン、チェン、カイ
中華人民共和国１１１００１９３ ベイジン ドンベイワン ウエスト・ロード８ ジョングァンツン・ソフトウェア・パーク２８
(72)発明者 フォン、ティエン
中華人民共和国１０００１９３ ベイジン チャン・ピン・ディストリクト １５－１－３０２ ロン・ユエ・ヤン４
(72)発明者 ワン、シン
中華人民共和国１１１００１９３ ベイジン ドンベイワン ウエスト・ロード８ ジョングァンツン・ソフトウェア・パーク２８
(72)発明者 バオ、シャオ、ミン
中華人民共和国１１１００１９３ ベイジン ドンベイワン ウエスト・ロード８ ジョングァンツン・ソフトウェア・パーク２８
審査官 北村 学

(56)参考文献
特開２０００－１４８５６５（ＪＰ，Ａ）
特開２００７－１８８２０９（ＪＰ，Ａ）
特開２００７－２８７１８０（ＪＰ，Ａ）
特開２０１１－１３４０３７（ＪＰ，Ａ）
特表２０１４－５１７９４１（ＪＰ，Ａ）
特開２０１５－２１９８５２（ＪＰ，Ａ）
米国特許第０９０６９７７８（ＵＳ，Ｂ１）
米国特許出願公開第２０１４／０００６７０８（ＵＳ，Ａ１）
杵渕 雄樹，検証解剖！ジャーナリングファイルシステムＥｘｔ３，ＵＮＩＸ ＵＳＥＲ，日本，ソフトバンクパブリッシング株式会社，2005年07月01日，第14巻 第7号，pp. 62-68
水吉 俊幸，Ｌｉｎｕｘビジネス・ソリューション 第３回，日経バイト，日本，日経ＢＰ社，1999年11月22日，第198号，pp. 182～189
Apache Hadoop 2.7.2 - Introduction，The Apache Software Foundation，2016年03月11日，https://web.archive.org/web/20160311231004/https://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/filesystem/introduction.html

(58)調査した分野 (Int.Cl.，ＤＢ名)
ＩＰＣ Ｇ０６Ｆ １６／００ － １６／９５８
Ｇ０６Ｆ ９／５０