

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7463527号
(P7463527)

(45)発行日 令和6年4月8日(2024.4.8)

(24)登録日 令和6年3月29日(2024.3.29)

(51)国際特許分類

F I

G 0 6 F 9/50 (2006.01) G 0 6 F 9/50 1 2 0 A

請求項の数 20 (全26頁)

(21)出願番号	特願2022-544250(P2022-544250)	(73)特許権者	503455363
(86)(22)出願日	令和2年3月18日(2020.3.18)		レイセオン カンパニー
(65)公表番号	特表2023-511153(P2023-511153 A)		アメリカ合衆国 マサチューセッツ州 0 2 4 5 1-1 4 4 9 ウォルサム ウィンター ストリート 8 7 0
(43)公表日	令和5年3月16日(2023.3.16)	(74)代理人	100107766
(86)国際出願番号	PCT/US2020/023305		弁理士 伊東 忠重
(87)国際公開番号	WO2021/154316	(74)代理人	100070150
(87)国際公開日	令和3年8月5日(2021.8.5)		弁理士 伊東 忠彦
審査請求日	令和4年7月21日(2022.7.21)	(74)代理人	100135079
(31)優先権主張番号	16/773,262		弁理士 宮崎 修
(32)優先日	令和2年1月27日(2020.1.27)	(72)発明者	メイヤー, ダグラス エー.
(33)優先権主張国・地域又は機関	米国(US)		アメリカ合衆国 アリゾナ州 8 5 7 4 7 ツーソン イースト・ゲットシンガー・レーン 8 6 5 6
		最終頁に続く	

(54)【発明の名称】 リザーベーションに基づいた高性能コンピューティングシステム及び方法

(57)【特許請求の範囲】

【請求項1】

高性能コンピューティング（H P C）管理機能によって、共有コンピューティングリソースの接続を検出することと、

第1プロジェクトに関連したコアコンピューティングリソースへ前記共有コンピューティングリソースを通信可能に結合することであり、前記第1プロジェクトに関連したコアコンピューティングリソースは、前記第1プロジェクトに関連したデータ処理動作を実行するために前記共有コンピューティングリソースを使用するよう構成される、ことと、

前記H P C管理機能によって、前記共有コンピューティングリソースをローカルコンピューティングリソースとしてエミュレートすることと、

前記共有コンピューティングリソースの電源を切り、前記第1プロジェクトに関連したコアコンピューティングリソースから前記共有コンピューティングリソースを切り離し、第2プロジェクトに関連したコアコンピューティングリソースへ前記共有コンピューティングリソースを通信可能に結合し、前記共有コンピューティングリソースの電源を入れることによって、前記共有コンピューティングリソースを前記第2プロジェクトに再割り当てすることであり、前記第2プロジェクトに関連したコアコンピューティングリソースは、前記第2プロジェクトに関連したデータ処理動作を実行するために前記共有コンピューティングリソースを使用するよう構成される、ことと、

前記H P C管理機能によって、前記第1プロジェクトが前記共有コンピューティングリソースにアクセスできなくなったことを決定することと

を有し、

前記共有コンピューティングリソースは、前記第1プロジェクト及び前記第2プロジェクトに関係があるデータを記憶する不揮発性メモリを欠いており、

前記共有コンピューティングリソース及び前記第1プロジェクトに関連したコアコンピューティングリソースは、集合的に、第1コンピューティングクラスタの少なくとも部分を形成し、

前記共有コンピューティングリソース及び前記第2プロジェクトに関連したコアコンピューティングリソースは、集合的に、第2コンピューティングクラスタの少なくとも部分を形成する、

方法。

10

【請求項2】

前記共有コンピューティングリソースは、前記第1プロジェクト及び前記第2プロジェクトに関係がある前記データを記憶するために使用される揮発性メモリを有し、

前記共有コンピューティングリソースの電源を切るとは、前記揮発性メモリをクリアし、前記第1プロジェクトに関連したデータが前記第2プロジェクトに関連したコアコンピューティングリソースによってアクセスされるのを防ぐ、

請求項1に記載の方法。

【請求項3】

前記共有コンピューティングリソースは、ディスクレスコンピューティングノードを有し、

20

前記第2プロジェクトに関連したコアコンピューティングリソースは、ブート情報を前記ディスクレスコンピューティングノードへ供給する少なくとも1つの管理サーバを有する、

請求項1に記載の方法。

【請求項4】

前記第2プロジェクトに関連したコアコンピューティングリソースは、情報を前記共有コンピューティングリソースへ供給することと、処理結果を前記共有コンピューティングリソースから受け取ることと、のうちの少なくとも1つを行う少なくとも1つのスクラッチサーバを有する、

請求項1に記載の方法。

30

【請求項5】

前記第2プロジェクトに関連したコアコンピューティングリソースは、前記第2コンピューティングクラスタによって実行される処理ジョブをサブミットしモニタすることと、前記少なくとも1つのスクラッチサーバにある前記処理結果にアクセスすることと、をユーザに可能にする少なくとも1つのセッションサーバを更に有する、

請求項4に記載の方法。

【請求項6】

前記第2プロジェクトに関連したコアコンピューティングリソースは、前記第2プロジェクトに関連したコアコンピューティングリソースによって使用されるプールに前記共有コンピューティングリソースを組み入れる少なくとも1つの管理サーバを有し、

40

前記少なくとも1つの管理サーバは、ジョブスケジューラ及びバッチキューイングシステムを有し、

前記ジョブスケジューラは、ユーザによってサブミットされた処理ジョブを受け取り、前記処理ジョブを前記バッチキューイングシステムへ供給し、

前記バッチキューイングシステムは、実行のために前記共有コンピューティングリソースを含む特定のコンピューティングリソースへ前記処理ジョブを割り当てる、

請求項1に記載の方法。

【請求項7】

前記第2プロジェクトに関連したコアコンピューティングリソースは、前記第2プロジェクトに関係があるデータを記憶する不揮発性メモリを欠いている少なくとも1つの専用

50

コンピューティングノードを有する、

請求項 1 に記載の方法。

【請求項 8】

前記第 2 プロジェクトに関連したコアコンピューティングリソースにある不揮発性ストレージに前記共有コンピューティングリソースからの処理結果を記憶することを更に有する、

請求項 1 に記載の方法。

【請求項 9】

前記共有コンピューティングリソースは、共有コンピューティングリソースの第 1 セット内の複数の共有コンピューティングリソースのうちの 1 つを有し、

当該方法は、前記共有コンピューティングリソースの第 1 セット及び共有コンピューティングリソースの少なくとも 1 つの更なるセットを、前記第 1 プロジェクトに関連したコアコンピューティングリソースと、前記第 2 プロジェクトに関連したコアコンピューティングリソースとへ通信可能に結合することを更に有し、

共有コンピューティングリソースの異なるセットは、異なるタイプのコンピューティングノードを有する、

請求項 1 に記載の方法。

【請求項 10】

前記共有コンピューティングリソースは、1 つ以上のサーバ及び 1 つ以上のグラフィックス処理ユニットのうちの少なくとも 1 つを有する、

請求項 1 に記載の方法。

【請求項 11】

共有コンピューティングリソースと、

第 1 プロジェクトに関連したコアコンピューティングリソースであり、前記共有コンピューティングリソースへの接続を検出し、前記第 1 プロジェクトに関連したデータ処理動作を実行するために前記共有コンピューティングリソースをローカルコンピューティングリソースとしてエミュレートするよう構成される第 1 高性能コンピューティング (HPC) 管理機能を有する、前記第 1 プロジェクトに関連したコアコンピューティングリソースと、

第 2 プロジェクトに関連したコアコンピューティングリソースであり、前記共有コンピューティングリソースへの接続を検出し、前記第 2 プロジェクトに関連したデータ処理動作を実行するために前記共有コンピューティングリソースをローカルコンピューティングリソースとしてエミュレートするよう構成される第 2 HPC 管理機能を有する、前記第 2 プロジェクトに関連したコアコンピューティングリソースと

を有し、

前記共有コンピューティングリソースは、前記第 1 プロジェクトに関連したコアコンピューティングリソースへ通信可能に結合され、電源を切られ、前記第 1 プロジェクトに関連したコアコンピューティングリソースから切り離され、そして、前記共有コンピューティングリソースを再割り当てするよう、前記第 2 プロジェクトに関連したコアコンピューティングリソースへ通信可能に結合されて電源を入れられる、よう構成され、

前記共有コンピューティングリソースが前記第 2 プロジェクトに関連したコアコンピューティングリソースへ通信可能に結合されているとき、前記第 1 HPC 管理機能は、前記第 1 プロジェクトが前記共有コンピューティングリソースにアクセスできなくなったことを決定するよう構成され、

前記共有コンピューティングリソースは、前記第 1 プロジェクト及び前記第 2 プロジェクトに関係があるデータを記憶する不揮発性メモリを欠いており、

前記共有コンピューティングリソース及び前記第 1 プロジェクトに関連したコアコンピューティングリソースは、集合的に、第 1 コンピューティングクラスタの少なくとも部分を形成し、

前記共有コンピューティングリソース及び前記第 2 プロジェクトに関連したコアコンピ

10

20

30

40

50

ューティングリソースは、集合的に、第2コンピューティングクラスタの少なくとも部分を形成する、
システム。

【請求項12】

前記共有コンピューティングリソースは、前記第1プロジェクト及び前記第2プロジェクトに関係がある前記データを記憶するために使用される揮発性メモリを有し、

前記共有コンピューティングリソースは、前記共有コンピューティングリソースの電源を切ることが、前記揮発性メモリをクリアし、前記第1プロジェクトに関連したデータが前記第2プロジェクトに関連したコアコンピューティングリソースによってアクセスされるのを防ぐように、構成される、

請求項11に記載のシステム。

【請求項13】

前記共有コンピューティングリソースは、ディスクレスコンピューティングノードを有し、

前記第2プロジェクトに関連したコアコンピューティングリソースは、ブート情報を前記ディスクレスコンピューティングノードへ供給する少なくとも1つの管理サーバを有する、

請求項11に記載のシステム。

【請求項14】

前記第2プロジェクトに関連したコアコンピューティングリソースは、情報を前記共有コンピューティングリソースへ供給することと、処理結果を前記共有コンピューティングリソースから受け取ることと、のうちの少なくとも1つを行う少なくとも1つのスクラッチサーバを有する、

請求項11に記載のシステム。

【請求項15】

前記第2プロジェクトに関連したコアコンピューティングリソースは、前記第2コンピューティングクラスタによって実行される処理ジョブをサブミットしモニタすることと、前記少なくとも1つのスクラッチサーバにある前記処理結果にアクセスすることと、をユーザに可能にするよう構成される少なくとも1つのセッションサーバを更に有する、

請求項14に記載のシステム。

【請求項16】

前記第2プロジェクトに関連したコアコンピューティングリソースは、前記第2プロジェクトに関連したコアコンピューティングリソースによって使用されるプールに前記共有コンピューティングリソースを組み入れるよう構成される少なくとも1つの管理サーバを有し、

前記少なくとも1つの管理サーバは、ジョブスケジューラ及びバッチキューイングシステムを有し、

前記ジョブスケジューラは、ユーザによってサブミットされた処理ジョブを受け取り、前記処理ジョブを前記バッチキューイングシステムへ供給するよう構成され、

前記バッチキューイングシステムは、実行のために前記共有コンピューティングリソースを含む特定のコンピューティングリソースへ前記処理ジョブを割り当てるよう構成される、

請求項11に記載のシステム。

【請求項17】

前記第2プロジェクトに関連したコアコンピューティングリソースは、前記第2プロジェクトに関係があるデータを記憶する不揮発性メモリを欠いている少なくとも1つの専用コンピューティングノードを有する、

請求項11に記載のシステム。

【請求項18】

前記第2プロジェクトに関連したコアコンピューティングリソースは、記共有コンピュ

10

20

30

40

50

ーティングリソースからの処理結果を記憶するよう構成される不揮発性ストレージを有する、

請求項 11 に記載のシステム。

【請求項 19】

前記共有コンピューティングリソースは、1 つ以上のサーバ及び 1 つ以上のグラフィクス処理ユニットのうちの少なくとも 1 つを有する、

請求項 11 に記載のシステム。

【請求項 20】

異なるプロジェクトに関連しかつ専用であるコアコンピューティングリソースの複数のセットであり、コアコンピューティングリソースの各セットが、関連するプロジェクトのためのデータ処理動作を実行するよう構成される、前記コアコンピューティングリソースの複数のセットと、

前記異なるプロジェクトのうちの 1 つへ選択的に割り当てられ、該割り当てられたプロジェクトのためのデータ処理動作を実行するよう夫々構成される複数の共有コンピューティングリソースと

を有し、

コアコンピューティングリソースの各セットは、前記複数の共有コンピューティングリソースのうちの 1 つ以上の共有コンピューティングリソースの接続を検出し、前記割り当てられたプロジェクトのためのデータ処理動作を実行するために前記 1 つ以上の共有コンピューティングリソースをローカルコンピューティングリソースとしてエミュレートするよう構成される高性能コンピューティング（HPC）管理機能を有し、

各共有コンピューティングリソースは、前記プロジェクトのうちの第 1 プロジェクトに関連したコアコンピューティングリソースのセットへ通信可能に結合され、電源を切られ、前記第 1 プロジェクトに関連したコアコンピューティングリソースのセットから切り離され、そして、当該共有コンピューティングリソースを再割り当てするよう、前記プロジェクトのうちの第 2 プロジェクトに関連したコアコンピューティングリソースのセットへ通信可能に結合されて電源を入れられる、よう構成され、

各共有コンピューティングリソースは、前記プロジェクトに関係があるデータを記憶する不揮発性メモリを欠いており、

コアコンピューティングリソースの各セット及びコアコンピューティングリソースの当該セットへ通信可能に結合されている前記共有コンピューティングリソースは、コンピューティングクラスタを形成し、各コンピューティングクラスタは、前記コアコンピューティングリソースの当該セットへ通信可能に結合されている前記共有コンピューティングリソースを変えることによって再設定可能である、

システム。

【発明の詳細な説明】

【技術分野】

【0001】

本開示は、概して、コンピューティングシステムを対象とする。より具体的には、本開示は、リザーベーションに基づいた高性能コンピューティングシステム及び方法を対象とする。

【背景技術】

【0002】

異なるパーティによって異なる時点で高性能コンピューティングシステム（例えば、高性能サーバ又はグラフィクス処理ユニット）が使用されることがしばしば必要であるか、又は望ましいことがある。例えば、高性能コンピューティングリソースが購入し作動させるにはしばしば高価であるという事実によってコンピューティングリソースの共有が余儀なくされるので、時間にわたってコンピューティングリソースの使用を共有することは、それらの費用を複数のパーティの間で分配するのに役立ち得る。また、高性能コンピューティングリソースの使用を必要とするか又は望んでいる多くのパーティは、常にコンピュ

10

20

30

40

50

ーティングリソースを使用する必要があるわけではないので、時間にわたってコンピューティングリソースの使用を共有することは、コンピューティングリソースの全体的な使用量を増やすのに役立ち得る。

【発明の概要】

【0003】

本開示は、リザーベーションに基づいた高性能コンピューティングシステム及び方法に係る。

【0004】

第1実施形態では、方法は、第1プロジェクトに関連したコアコンピューティングリソースへ共有コンピューティングリソースを通信可能に結合することを含む。第1プロジェクトに関連したコアコンピューティングリソースは、第1プロジェクトに関連したデータ処理動作を実行するために共有コンピューティングリソースを使用するよう構成される。方法はまた、(i) 共有コンピューティングリソースの電源を切り、(ii) 第1プロジェクトに関連したコアコンピューティングリソースから共有コンピューティングリソースを切り離し、(iii) 第2プロジェクトに関連したコアコンピューティングリソースへ共有コンピューティングリソースを通信可能に結合し、(iv) 共有コンピューティングリソースの電源を入れることによって、共有コンピューティングリソースを第2プロジェクトに再割り当てすることを含む。第2プロジェクトに関連したコアコンピューティングリソースは、第2プロジェクトに関連したデータ処理動作を実行するために共有コンピューティングリソースを使用するよう構成される。共有コンピューティングリソースは、第1プロジェクト及び第2プロジェクトに関係があるデータを記憶する不揮発性メモリを欠いている。共有コンピューティングリソース及び第1プロジェクトに関連したコアコンピューティングリソースは、集合的に、第1コンピューティングクラスタの少なくとも部分を形成する。共有コンピューティングリソース及び第2プロジェクトに関連したコアコンピューティングリソースは、集合的に、第2コンピューティングクラスタの少なくとも部分を形成する。

【0005】

第2実施形態では、システムは、共有コンピューティングリソースを含む。システムはまた、第1プロジェクトに関連したデータ処理動作を実行するために共有コンピューティングリソースを使用するよう構成される、第1プロジェクトに関連したコアコンピューティングリソースを含む。システムは更に、第2プロジェクトに関連したデータ処理動作を実行するために前記共有コンピューティングリソースを使用するよう構成される、第2プロジェクトに関連したコアコンピューティングリソースを含む。共有コンピューティングリソースは、(i) 第1プロジェクトに関連したコアコンピューティングリソースへ通信可能に結合され、(ii) 電源を切られ、第1プロジェクトに関連したコアコンピューティングリソースから切り離され、(iii) 共有コンピューティングリソースを再割り当てするよう、第2プロジェクトに関連したコアコンピューティングリソースへ通信可能に結合されて電源を入れられる、よう構成される。共有コンピューティングリソースは、第1プロジェクト及び第2プロジェクトに関係があるデータを記憶する不揮発性メモリを欠いている。共有コンピューティングリソース及び第1プロジェクトに関連したコアコンピューティングリソースは、集合的に、第1コンピューティングクラスタの少なくとも部分を形成する。共有コンピューティングリソース及び第2プロジェクトに関連したコアコンピューティングリソースは、集合的に、第2コンピューティングクラスタの少なくとも部分を形成する。

【0006】

第3実施形態では、システムは、異なるプロジェクトに関連しかつ専用であるコアコンピューティングリソースの複数のセットを含み、コアコンピューティングリソースの各セットは、関連するプロジェクトのためのデータ処理動作を実行するよう構成される。システムはまた、異なるプロジェクトのうちの1つへ選択的に割り当てられ、その割り当てられたプロジェクトのためのデータ処理動作を実行するよう夫々構成される複数の共有コン

ピューティングリソースを含む。各共有コンピューティングリソースは、(i) プロジェクトのうちの第1プロジェクトに関連したコアコンピューティングリソースのセットへ通信可能に結合され、(ii) 電源を切られ、第1プロジェクトに関連したコアコンピューティングリソースのセットから切り離され、(iii) その共有コンピューティングリソースを再割り当てするよう、プロジェクトのうちの第2プロジェクトに関連したコアコンピューティングリソースのセットへ通信可能に結合されて電源を入れられる、よう構成される。各共有コンピューティングリソースは、プロジェクトに関係があるデータを記憶する不揮発性メモリを欠いている。コアコンピューティングリソースの各セット及びコアコンピューティングリソースのそのセットへ通信可能に結合されている共有コンピューティングリソースは、コンピューティングクラスタを形成する。各コンピューティングクラスタは、コアコンピューティングリソースのセットへ通信可能に結合されている共有コンピューティングリソースを変えることによって再設定可能である。

10

【0007】

他の技術的な特徴は、次の図面、明細書、及び特許請求の範囲から当業者には容易に明らかになり得る。

【0008】

本開示のより完全な理解のために、これより、添付の図面と併せて、以下の説明を参照する。

【図面の簡単な説明】

【0009】

20

【図1】本開示に係る、リザベーションに基づいた高性能コンピューティングシステムで使用されるコンピューティングクラスタの例を表す。

【図2】本開示に係る、1つ以上の再設定可能なコンピューティングクラスタを含むリザベーションに基づいた高性能コンピューティングシステムの例を表す。

【図3】本開示に係る、図2のリザベーションに基づいた高性能コンピューティングシステムの再設定の例を表す。

【図4】本開示に係る、リザベーションに基づいた高性能コンピューティングシステムをサポートする機能アーキテクチャの例を表す。

【図5】本開示に係る、リザベーションに基づいた高性能コンピューティングシステムにおけるコンピューティングノードの例を表す。

30

【図6】本開示に係る、リザベーションに基づいた高性能コンピューティングの方法の例を表す。

【発明を実施するための形態】

【0010】

後述される図1～6、及び本明細書中で本発明の原理を説明するために使用される様々な実施形態は、単なる例示であり、発明の範囲を制限するものとして決して解釈されるべきではない。当業者は、本発明の原理が、あらゆるタイプの適切に配置されたデバイス又はシステムで実装されてもよい、と理解するだろう。

【0011】

上述されたように、異なるパーティによって異なる時点で高性能コンピューティングシステム（例えば、高性能サーバ又はグラフィクス処理ユニット）が使用されることがしばしば必要であるか、又は望ましいことがある。数ある理由の中でも特に、これは、複数のパーティの間で高性能コンピューティングリソースの費用を分散するのを助けるために、又は高性能コンピューティングリソースの全体的な使用量を増やすのを助けるために、行われ得る。しかし、共有された高性能コンピューティングリソース又は他のコンピューティングリソースの使用により生じる1つの問題は、データセキュリティである。つまり、政府及び民間組織は、常に彼らの機密情報を保護しようと努めており、そうすることをしてしばしば求められるので、コンピューティングリソースが共有される場合に問題が発生する。これは、適切な予防措置がなければ、共有コンピューティングリソースに記憶されている1つのパーティに関連したデータが、共有コンピューティングリソースを用いて他のパ

40

50

ーティによってアクセスされる可能性があるからである。これはまた、データを処理するために使用される実際のコンピューティングノードが時間とともに変化する可能性があるクラウドコンピューティング環境でも一般的な問題である。

【0012】

いくつかの場合に、1つのパーティによって使用されている共有コンピューティングリソースが他のパーティに移動されて使用され得る前にどのような手順が行われる必要があるかに関して、厳密なガイドラインが用意されている。例えば、政府機関のために働く様々な商業請負業者がコンピューティングリソースを共有することは一般的であり、異なる ”need-to-know” (NTK) 環境の間での分類されたシステムの移動に関してしばしばセキュリティ規則が適用される。このようなタイプの規則はまた、同じ会社の異なる部署又は他のユニットが異なる政府プロジェクトのためにコンピューティングリソースを共有する場合など、単一の組織に影響を与える場合もある。

【0013】

しばしば、これらのセキュリティ規則は、ハードディスクドライブ若しくは他の不揮発性記憶デバイスの「サニタイゼーション」(sanitization) (通常、記憶されたデータを上書きにより一掃することによる) 又はコンピューティングノードでのハードドライブ若しくは他の不揮発性記憶デバイスの物理的な交換を求める。サニタイゼーションレコードはまた、記憶デバイスをサニタイズするために行われる特定のステップを識別するために、通常、様々な不揮発性記憶デバイスについて作られる必要がある。更に、コンピューティングリソースを移動する許可を得ることは、しばしば、複数レベルのセキュリティ認証とともに、移動されるコンピューティングリソースを取得及び喪失するコンピューティングシステムのハードウェアインベントリへの変更を必要とする。セキュリティレビュープロセス全体は、ワークロード及び関連するプロジェクト又はデータなどの様々な要因に応じて、延長される場合がある(場合により数週間又は数ヶ月)。よって、多数の手続き要件、長い文書化、及び多段階の承認プロセスは、閉鎖区域の境界を越えた装置の再利用を妨げる可能性がある。そのような再利用は、非常に努力を要し、かつ、変化するニーズへの迅速な対応を妨げる。更に悪いことには、異なる政府期間は、共有コンピューティングリソースに関して言えば、異なる要件をしばしば有している。更には、たとえ異なるプロジェクトを単一ネットワークに統合できたとしても、監査目的で必要とされる膨大な量の生データが共有リソースに関して生成される可能性がある。

【0014】

コンピューティングリソースを共有するための他のアプローチは、仮想化又は仮想ローカルエリアネットワーク(Virtual Local Area Networks, VLAN)の構築によるコンピューティングリソースの論理的な分離を軸に展開する。しかし、仮想化は、コンピューティングリソースの物理的な分離に関連した保護を欠いている。また、仮想化は、政府規則又は規制がコンピューティングリソースの物理的な分離を求めている場合など、いくつかのアプリケーションでの利用には適切でない場合がある(これは、分類されたデータ又は機密データの処理プロジェクトに関連したコンピューティングリソースでは一般的である。))。

【0015】

これら及び他の問題の結果として、コンピューティングリソースは、コンピューティングリソースが特定のプロジェクトによって完全に利用されているかどうかにかかわらず、通常は、その特定のプロジェクトの存続期間中にその特定のプロジェクトに割り当てられ得る。また、人々及び組織は、たとえコンピューティングリソースが完全には利用されていないとしても、将来的にコンピューティングリソースが再び必要とされる場合に大幅な遅延が発生する可能性があるため、しばしば、他者による使用のためにそれらのコンピューティングリソースを解放することを躊躇する。

【0016】

本開示は、リザーベーションに基づいた高性能コンピューティングのための様々な技術を提供する。以下で更に詳細に記載されるように、コンピューティングシステムアーキテク

10

20

30

40

50

チャは、少なくとも1つの情報システムをサポートする1つ以上の共有コンピューティングリソースを含み、これは、容易にサニタイズされ得、かつ、異なるプロジェクトに瞬時に再割り当てされ得る。とりわけ、コンピューティングシステムアーキテクチャは、1つ以上の共有コンピューティングリソースからハードディスクドライブ（Hard Disk Drives, HDD）及び任意の他のユーザ書き換え可能な不揮発性記憶デバイスを削除又は除外する。いくつかの場合に、揮発性ストレージのみが、1つ以上の共有コンピューティングリソースで使用される。コンピューティングシステムアーキテクチャの他のコンポーネントは、オペレーティングシステムファイル及び他のローカルファイルのためといった、データのローカルストレージをエミュレートしながら、1つ以上の共有コンピューティングリソースとの間でデータにアクセスし記憶又は読み出すことができる高速サーバ又は他の処理デバイスを使用する。1つ以上の共有コンピューティングリソースの揮発性ストレージ内のデータは、容易にクリアされ得る。これは、様々な政府セキュリティ規則又は他のデータセキュリティ要件の順守を促進する。

【0017】

いくつかの実施形態で、このようなコンピューティングシステムアーキテクチャは、複数のコンピューティングクラスタを含み、各コンピューティングクラスタは、一般的に、2つタイプのコンポーネントに分けられる。第1タイプのコンポーネントは、コンピューティングクラスタの「ブラウン」（brawn）を表し、サーバブレード、コンピューティングエンジン、及び異なるコンピューティングクラスタ間で共有可能である他のコンピューティングリソースを含む。ハードドライブ及び他のユーザ書き換え可能な不揮発性記憶デバイスは、共有コンピューティングリソースには含まれない。これらは、異なるプロジェクトの間で共有され得、かつ、必要な場合にデータに関して容易にクリアされ得るコンポーネントである。第2タイプのコンポーネントは、コンピューティングクラスタの「ブレイン」（brain）又はコアを表し、ファイルサーバ、管理サーバ、ディスクレス画像サーバ、又は他のコアコンピューティングリソースを含んでよく、これらの少なくとも一部は、共有コンピューティングリソースの使用と相互作用するか、又は共有コンピューティングリソースの使用をサポートする。これらは、個々のプロジェクトに割り当てられ得、プロジェクト間で共有されないコンポーネントであるから、これらのコンポーネントは、データをより永続的に記憶することができるハードドライブ又は他のユーザ書き換え可能な不揮発性記憶デバイスを含むことができる。コアコンピューティングリソースの少なくとも一部は、共有コンピューティングリソースへの低レイテンシ高バンド幅接続を有することができる。これらの接続は、異なるプロジェクトのコアコンピューティングリソースに共有コンピューティングリソースを割り当てよう必要に応じて変更され得る。

【0018】

しばしば、共有コンピューティングリソース及びコアコンピューティングリソースは、異なるラックに物理的に置かれ、特定の共有コンピューティングリソースは、コアコンピューティングリソースのラックへの共有コンピューティングリソースのラックの適切な結合によって、特定のプロジェクトに（一時的に）割り当てられ得る。例えば、共有コンピューティングリソースを含むラックが、特定のプロジェクトの情報システムへの割り当てのために、識別される場合に、従うべき標準化されたプロシージャは、そのラックにある共有コンピューティングリソースの電源を切り、共有コンピューティングリソースを調べて共有コンピューティングリソースが実際に電源を切られたことを確かめ（従って、不揮発性メモリしか含まれないので全てのデータが失われる）、共有コンピューティングリソースから前のプロジェクトのコアコンピューティングリソースへのネットワーク接続を切断し、共有コンピューティングリソースから新しいプロジェクトのコアコンピューティングリソースへのネットワーク接続を接続することを含むことができる。いくつかの場合に、新しいプロジェクトにおける少なくとも1つのディスクレス画像サーバは、共有コンピューティングリソースにおけるサーバ又は他のコンピューティングノードのリモートブートをサポートすることができ、新しいプロジェクトにおける管理サーバは、新しいプロジェクトのコアコンピューティングリソースによって使用されるプールに共有コンピューテ

イングリソースを組み入れることができる。このようなプロセスは、他のアプローチと比較してよりずっと速く完了され得る。また、共有コンピューティングリソースが異なるプロジェクトによる使用のために割り当て及び再割り当てされるべき場合を識別するために、予想されるワークロードに基づく（場合により、パフォーマンスメトリクスによってサポートされる）リザーベーションシステムが使用されてもよい。

【0019】

このようにして、共有コンピューティングリソースは、たとえデータの共有を防ぐためにプロジェクトが厳密なセキュリティ規則を有しているとしても、異なるプロジェクトをサポートするよう必要に応じて割り当て及び再割り当てされ得る（従って、NTK環境での使用に適している）。分類された、機密の、又は他のプロジェクトで使用される共有コンピューティングリソースは、例えば数時間内といったように、よりずっと速く移動され得る。とりわけ、これは、新しい若しくは短期のプロジェクトへの、又は重要だが一時的な「急な」ニーズへの共有コンピューティングリソースの即時のかつ容易な割り当て及び再割り当てを可能にする。また、これは、ハードドライブを共有コンピューティングリソースに出し入れする必要なしに達成され得、共有コンピューティングリソースがそれら自体の情報システムを含み得るということで、共有コンピューティングリソースを使用する各プロジェクトのハードウェアインベントリを変更する必要がなくなる。更に、これは、仮想化（通常は、性能低下に付随し、有意な監査要件を有している）の使用を必要とせずに、かつ、基礎をなすシステムデータ処理動作の変更を必要とせずに（ローカルデータ記憶のエミュレーションが提供されるため）、達成され得る。更に、「テナンシー」（tenancy）（共有可能であることを意味する）に利用可能な共有コンピューティングリソースのベースラインは、専用の情報システムの部分として追跡され得、共有コンピューティングリソースが必要に応じて監視、解放、及び再請求されることを可能にする。更に、このアプローチは、例えば、ワークステーションレベルで又はサーバレベルで（又はそれら両方のレベル）といったように、コンピューティングシステム内の様々なレベルで使用され得る。これは、例えば、ディスクレス動作が、分類されたデスクトップ及びコンピューティングサーバのようなコンピューティングノードを含むシステムをサポートするために使用される場合など、様々な状況で有用であり得る。

【0020】

図1は、本開示に係るリザーベーションに基づいた高性能コンピューティングシステムで使用される例示的なコンピューティングクラスタ100を表す。図1に示されるように、コンピューティングクラスタ100は、多数のラック102に收容されている様々なコンピューティングリソースを含む。各ラック102は、如何なる適切なサイズ、形状、及び寸法も有することができ、各ラック102は、コンピューティングリソースをいくつでも保持するよう構成され得る。また、各ラック102は、ラック102内のコンピューティングリソースへ電力を分配するために使用され得る少なくとも1つの電力分配ユニットを含んでもよい。各電力分配ユニットは、15kWから24kWの電力といった、如何なる適切な量の電力も供給するよう構成されてよい。いくつかの実施形態で、各ラック102は、複数の余剰の電力分配ユニットを含んでもよい。なお、コンピューティングリソースは、如何なる適切な様態でも收容及び給電されてよいことに留意されたい。

【0021】

コンピューティングクラスタ100は、コアコンピューティングリソース104及び共有コンピューティングリソース106に分けられる。コアコンピューティングリソース104は、一般的に、特定のプロジェクトに割り当てられ、プロジェクト間で（少なくとも容易には）共有されないコンピューティングノードを表す。結果として、コアコンピューティングリソース104は、プロジェクト関連データを永続的に記憶するために使用されるハードディスクドライブ又は他のユーザ書き換え可能な不揮発性記憶デバイスを含む。対照的に、共有コンピューティングリソース106は、一般的に、異なる時点で異なるプロジェクトに割り当てられ得るコンピューティングリソースを表す。結果として、共有コンピューティングリソース106には、ハードディスクドライブ及び他のユーザ書き換え

可能な不揮発性記憶デバイスがない。いくつかの実施形態で、共有コンピューティングリソース106は、プロジェクト関連データを記憶するために使用される揮発性記憶デバイスしか含まなくてよい。この分割によれば、共有コンピューティングリソース106は、割り当てられたプロジェクトのためのデータ処理計算のかなりの部分又は全部を実行するために使用されてよく、コアコンピューティングリソース104は、共有コンピューティングリソース106の動作を支援するようインテリジェンス及び制御を提供するために、かつ、関連するプロジェクトのためのデータ保持能力を提供するために使用されてよい。

【0022】

コアコンピューティングリソース104及び共有コンピューティングリソース106で使用されるコンピューティングノードの具体的なタイプは、アプリケーションに基づいて多種多様であることができる。以下は、コアコンピューティングリソース104及び共有コンピューティングリソース106で使用され得るコンピューティングノードのタイプの例を表す。しかし、他の又は更なるタイプのコンピューティングノードが、コアコンピューティングリソース104で及び共有コンピューティングリソース106で使用されてもよい。

【0023】

コアコンピューティングリソース104は、1つ以上の管理サーバ108a～108bを含み得る。管理サーバ108a～108bは、コアコンピューティングリソース104及びコンピューティングクラスタ100全体の動作を管理するのを助ける様々な機能を提供するよう構成される。例えば、管理サーバ108a～108bは、エンジニア又は他の者によってサブミットされた処理ジョブを受け取って、処理ジョブをバッチキューイングシステムへ供給することができる処理ジョブスケジューラを実行又は別なふうに提供し得る。バッチキューイングシステムは、実行のために特定のコンピューティングリソース（例えば、共有コンピューティングリソース106）へ処理ジョブを割り当てる。バッチキューイングシステムは、如何なる適切な様態でも、例えば、ユーザによって定義された要件に基づいて、処理ジョブをコンピューティングリソースに割り当ててよい。とりわけ、これは、深夜まで開始せず、人が朝に仕事場に着く前に完了し得る処理ジョブを含む、ジョブの「24時間体制」（around the clock）処理を可能にする。管理サーバ108a～108bはまた、様々なコアコンピューティングリソース104及び共有コンピューティングリソース106のためのメトリクス監視も実行又は別なふうに提供し得る。これは、管理サーバ108a～108b（及び管理サーバ108a～108bと相互作用するエンジニア又は他の者）が時間にわたってコンピューティングリソース104及び106のパフォーマンスを監視することを可能にする。

【0024】

管理サーバ108a～108bは、「ディスクレス」（diskless）コンピューティングノードのためのブートサービスを更に提供し得る。ディスクレスコンピューティングノードは、一般的に、必要とされるブート情報が管理サーバ108a～108bで保持されているコンピューティングデバイスを表す。これは、例えば、ディスクレスコンピューティングノードが、ブート情報のための不揮発性ストレージがない共有コンピューティングリソース106に相当する場合に、起こり得る。各ディスクレスコンピューティングノードのために管理サーバ108a～108bで保持されているブート情報は、システムアドレス、オペレーティングシステムの共有された読み出し専用のコピー、及びディスクレスコンピューティングノードに必要な一意のファイル（例えば、ログ及び設定ファイル）などの情報を含み得る。各ディスクレスコンピューティングノードは、管理サーバ108a～108bと相互作用することによって、自動的にアドレスを要求して、完全動作状態にブートし得る。これは、ディスクレスコンピューティングノードでのローカルハードドライブの必要性を取り除くことができる。

【0025】

この例では、コアコンピューティングリソース104で使用される複数の管理サーバ108a～108bがある。より具体的には、2つの管理サーバ108a～108bがこの

例では使用されている。複数の管理サーバ108a～108bの存在は、第1管理サーバ108a～108bが一次モードで作動することを可能にし得る一方で、第2管理サーバ108b～108aは、バックアップ又は二次モードで作動して、第1管理サーバが機能しなくなるか又は何らかの他の不具合を経験する場合に主たる役割を引き継ぐよう準備され得る。当然、単一の管理サーバが使用されてもよく、あるいは、複数の管理サーバは、如何なる他の適切な様態でも作動してよい。

【0026】

コアコンピューティングリソース104はまた、1つ以上のスクラッチサーバ110a～110b、112を含んでもよい。各スクラッチサーバ110a～110b、112は、多数の不揮発性記憶デバイス（例えば、ハードディスクドライブ又はソリッドステートドライブ）を含むファイルサーバを表し、各ファイルサーバは、通常は、高い信頼性を有しながら大規模である。スクラッチサーバ110a～110b、112はまた、通常は、高速ネットワーク接続を用いて作動して、スクラッチサーバ110a～110b、112からのデータの高速な出し入れを可能にする。とりわけ、スクラッチサーバ110a～110b、112は、共有コンピューティングリソース106によって実行されるコンピューティング動作により使用され、生成され、収集され、又はそれと別なふうに関連するデータを記憶しかつその読み出しを助けるために使用され得る。具体例として、スクラッチサーバ110a～110b、112は、共有コンピューティングリソース106によって使用されるシミュレーション入力ファイル及び共有コンピューティングリソース106によって生成されたシミュレーション出力ファイルを記憶しかつそれらの読み出しを助けるために使用され得る。

【0027】

この例では、スクラッチサーバ110a～110b、112は、write-heavy（WH）スクラッチサーバ110a～110b及びread-heavy（RH）スクラッチサーバ112を含む。その名の通り、write-heavyスクラッチサーバ110a～110bは、より頻繁に書き込まれるデータを記憶するために使用され、一方、read-heavyスクラッチサーバ112は、より頻繁に読み出されるデータを記憶するために使用される。よって、例えば、read-heavyスクラッチサーバ112は、処理動作を実行するために共有コンピューティングリソース106によって繰り返し読み出されて使用される共通データのような、めったに変化しないデータを記憶し得る。このタイプのデータは、データがアーカイブ内にあり、read-heavyスクラッチサーバ112が機能しなくなる場合に容易に置換され得る場合など、ほとんど又は全くバックアップを必要としない可能性がある。対照的に、write-heavyスクラッチサーバ110a～110bは、処理動作の実行中に共有コンピューティングリソース106によって生成されるデータのような、より頻繁に変化するデータを記憶し得る。スクラッチサーバ間のこのタイプのデータ分離は、write-heavyスクラッチサーバ110a～110bにかかる負荷を軽減するのに役立ち得る。2つのwrite-heavyスクラッチサーバ110a～110b及び1つのread-heavyスクラッチサーバ112が図1には示されているが、これは単なる例示であることに留意されたい。また、write-heavyスクラッチサーバ110a～110b及びread-heavyスクラッチサーバ112でのデータの分離は不要であり、1つ以上のスクラッチサーバが全てのデータのために使用されてもよいことにも留意されたい。更に、複数のスクラッチサーバは、如何なる適切な様態でも、例えば、独立して、又は冗長的な構成で、使用されてもよいことに留意されたい。

【0028】

コアコンピューティングリソース104は、少なくとも1つのセッションサーバ114を更にも含んでもよい。各セッションサーバ114は、エンジニア又は他の者のリモートログインをサポートするコンピューティングノードを表す。各セッションサーバ114はまた、スクラッチサーバ110a～110b、112の少なくとも一部への高速なアクセスも支援することができる。各セッションサーバ114は、作業者が処理ジョブの実行をセ

10

20

30

40

50

ットアップ、サブミット、及び監視すること並びに生成されたデータを用いて後処理動作（例えば、データ削減及びレビュー）に従事することを可能にし得る。いくつかの場合に、セッションサーバ114は、複数の同時のユーザをサポートするよう構成されたワークステーション又は他のコンピューティングノードに相当し得る。セッションサーバ114は、大量のメモリを含み、高い処理ワークロードをサポートし、スクラッチサーバ110a～110b、112への極めて高速なアクセスを提供し得る。

【0029】

いくつかのアプリケーションで、コンピューティングクラスタ100は、24時間体制で高い負荷の下で作動することを期待される場合があり、write-heavyスクラッチサーバ110a～110bは、同様にアクティブであり得る。そのようなものとして、write-heavyスクラッチサーバ110a～110bは、他のサーバの従来のアイドル時間を有さない可能性がある。write-heavyスクラッチサーバ110a～110bは、災害復旧バックアップ用に構成されない可能性がある（サーバ110a～110bの速度が低下したり、サーバ110a～110bの通常のアクティビティによって妨げられたりする）。そのような状況では、重大なシステム障害の発生時にデータが失われる危険性がある。よって、エンジニア又は他の者は、セッションサーバ114を介してスクラッチサーバ110a～110bで処理中のジョブデータをレビューし、データを後処理し、結果をより永続的な記憶デバイス（ここでは図示せず）にコピーすることができる。

【0030】

コアコンピューティングリソース104はまた、少なくとも1つの仮想デスクトップインフラストラクチャ（Virtual Desktop Infrastructure, VDI）サーバ116を含んでもよい。VDIサーバ116は、コンピューティングクラスタ100に関連したユーザのワークステーションで使用されるデスクトップ環境などの、仮想化されたデスクトップ環境をホストすることができる。いくつかの実施形態で、VDIサーバ116は、Windows（登録商標）及びLinux（登録商標）デスクトップ環境などの、様々なタイプの仮想化デスクトップ環境をホストすることができる。仮想化の使用を通じて、従来のセッションサーバで利用されるだろうより大きい大域的メモリ空間及び複数の処理コアへのアクセスなしで、複数のユーザセッションが分離され得る。

【0031】

コアコンピューティングリソース104は、任意に、1つ以上の光ディスクからデータを読み出すよう（また、場合により、1つ以上の光ディスクにデータを書き込むよう）構成されたドライブを表す少なくとも1つの光学ドライブ118を含んでもよい。コアコンピューティングリソース104はまた、任意に、1つ以上の作業コンピューティングノード120を含んでもよい。作業コンピューティングノード120は、共有コンピューティングリソース106で使用する同じタイプのコンピューティングノードに相当し得る。よって、作業コンピューティングノード120には、ハードディスクドライブ及び任意の他のユーザ書き換え可能な不揮発性記憶デバイスがなくてもよい。いくつかの実施形態で、作業コンピューティングノード120は、揮発性記憶デバイスしか含まなくてもよい。作業コンピューティングノード120は、プロジェクトによって必要とされるデータ処理動作を実行するために使用されてよく、作業コンピューティングノード120は、上述されたように、管理サーバ108a～108bを用いてブートされ得る。いくつかの場合に、作業コンピューティングノード120は、1つ以上のプロセッサ及び適切な量の揮発性メモリを夫々含むサーバブレードに相当し得る。各作業コンピューティングノード120は、作業コンピューティングノード120のリモートシステム管理を可能にするインテリジェントプラットフォーム管理インターフェース（Intelligent Platform Management Interface, IPMI）又は他の適切なインターフェースを含むか又はそれをサポートし得る。具体例として、IPMIは、作業コンピューティングノード120のためのオペレーティングシステムの画像をマウントするために使用されてよく、画像は、管理サーバ108a～108bによって供給される。他の場合には、作業コンピューティングノード

120は、特殊な画像関連データ処理動作を提供するよう設計されたグラフィクス処理ユニットに相当し得る。

【0032】

コアコンピューティングリソース104での作業コンピューティングノード120の存在は、コアコンピューティングリソース104へ結合されている共有コンピューティングリソース106があるかどうかにかかわらず、何らかのデータ処理機能が提供されることを可能にする。しばしば、作業コンピューティングノード120は、コアコンピューティングリソース104を含むラック102が空いた又は利用可能な空間を有する場合に、コアコンピューティングリソース104で使用され得る。しかし、作業コンピューティングノード120は、例えば、作業コンピューティングノード120の機能が不要である場合、又はコアコンピューティングリソース104を保持しているラックに利用可能な空間がない場合など、コアコンピューティングリソース104から省略される場合があることに留意されたい。

【0033】

更に、コアコンピューティングリソース104は、1つ以上のネットワーキングスイッチ122a~122bを含んでもよい。ネットワーキングスイッチ122a~122bは、コアコンピューティングリソース104を互いに及び（もしあれば）共有コンピューティングリソース106へ通信可能に結合するために使用され得る。この例では、コアコンピューティングリソース104により使用される2つのネットワーキングスイッチ122a~122bが存在する。第1ネットワーキングスイッチ122aは、多数のEthernet（登録商標）ポートを含んでよく、これらのポートは、Ethernet接続を用いて様々なコンポーネント108a~108b、110a~110b、112、114、116、118、120へ結合され得る。第1ネットワーキングスイッチ122aはまた、いくつかの光ポートを含んでもよく、これらのポートは、少なくとも1つの光ファイバ接続を用いて第2ネットワーキングスイッチ122bへ結合され得る。第2ネットワーキングスイッチ122bは、第1ネットワーキングスイッチ122aへ及び共有コンピューティングリソース106へ光ファイバ接続を用いて結合され得る多数の光ポートを含んでよい。いくつかの場合に、少なくとも、第2ネットワーキングスイッチ122bと共有コンピューティングリソース106との間の接続は、冗長的な光ファイバ接続又は他の冗長接続を用いて形成され得ることに留意されたい。第1ネットワーキングスイッチ122aと様々なコンポーネント108a~108b、110a~110b、112、114、116、118、120との間の接続、及びネットワーキングスイッチ122a~122b間の接続は、冗長接続を用いて形成されてもされなくてもよい。

【0034】

ネットワーキングスイッチ122a~122bの数及びタイプは、スイッチに結合されるノードの数及び使用される接続のタイプなどの多数の要因に基づいて様々であることができることに留意されたい。また、光接続及びEthernet接続の使用は、単なる例示のためであって、如何なる適切な接続も、コアコンピューティングリソース104のノードに伴って行われてもよいことに留意されたい。

【0035】

共有コンピューティングリソース106を含む各ラック102は、1つ以上のネットワーキングスイッチ124a~124bを含んでよく、ネットワーキングスイッチ124a~124bは、ラック102内の共有コンピューティングリソース106をコアコンピューティングリソース104へ（及び場合により互いに）通信可能に結合するために使用され得る。この例では、共有コンピューティングリソース106の各ラック102で 사용되는2つのネットワーキングスイッチ124a~124bが存在する。各ネットワーキングスイッチ124a~124bは、多数のEthernetポートを含んでよく、これらのポートは、Ethernet接続を用いてラック102内の共有コンピューティングリソース106へ結合され得る。各ネットワーキングスイッチ124a~124bはまた、いくつかの光ポートを含んでもよく、これらのポートは、光ファイバ接続を用いてコアコ

ンピューティングリソース104へ結合され得る。いくつかの場合に、ネットワーキングスイッチ124a～124bは冗長ペアとして構成され、それにより、ネットワーキングスイッチ124a～124bの1つ又は光ファイバ接続の1つが機能しなくなるか又は何らかの他の不具合を経験する場合に、コアコンピューティングリソース104との接続は失われない。先と同じく、ここでの光接続及びEthernet接続の使用は、単なる例示のためであって、如何なる適切な接続も、共有コンピューティングリソース106のノードに伴って行われてもよいことに留意されたい。

【0036】

共有コンピューティングリソース106を含む各ラック102はまた、異なるプロジェクトに関連したコアコンピューティングリソース104の異なるセット間で共有可能であるコンピューティングノードも含む。この例では、共有コンピューティングリソース106を含むラック102の全てが、同じタイプのコンピューティングノード、つまり、作業コンピューティングノード126を有する。作業コンピューティングノード126には、ハードディスクドライブ及び任意の他のユーザ書き換え可能な不揮発性記憶デバイスがなくてもよい。いくつかの場合に、作業コンピューティングノード126は、揮発性記憶デバイスしか含まなくてもよい。作業コンピューティングノード126は、プロジェクトによって必要とされるデータ処理動作を実行するために使用されてよく、作業コンピューティングノード126は、上述されたように、管理サーバ108a～108bを用いてブートされ得る。いくつかの場合に、作業コンピューティングノード126は、1つ以上のプロセッサ及び適切な量の揮発性メモリを夫々含むサーバブレードに相当し得る。各作業コンピューティングノード126は、作業コンピューティングノード126のリモートシステム管理を可能にするIPMI又は他の適切なインターフェースを含み得る。他の場合には、作業コンピューティングノード126は、特殊な画像関連データ処理動作を提供するよう設計されたグラフィクス処理ユニットに相当し得る。作業コンピューティングノード126の数は、必要に応じて又は望まれるように様々であることができることに留意されたい。

【0037】

この例では、様々なケーブル128が、コアコンピューティングリソース104と共有コンピューティングリソース106との間の接続を形成するために使用される。例えば、ケーブル128は、コアコンピューティングリソース104と共有コンピューティングリソース106との間の光ファイバ接続又は他のタイプの高速度接続を含んでよい。上述されたように、コアコンピューティングリソース104を含むラック102と共有コンピューティングリソース106を含むラック102との間に冗長なケーブル128はあってもなくてもよい。

【0038】

図1から分かるように、コアコンピューティングリソース104は、共有コンピューティングリソース106へ容易に接続され、また、それから容易に切り離され得る。例えば、共有コンピューティングリソース106を含むラック102が、現在、異なるプロジェクトのためのコアコンピューティングリソースを含むラック（図1に図示せず）へ結合されている、と仮定する。共有コンピューティングリソース106を含むラック102は、電源を切られ得る。そして、共有コンピューティングリソース106がもはや動作電力を受け取っていないことを確かめるために、検査が行われ得る。共有コンピューティングリソース106に如何なるタイプのユーザ書き換え可能な不揮発性ストレージもない場合（例えば、共有コンピューティングリソース106がランダムアクセスメモリのような揮発性メモリしか含まない場合）に、このことは、共有コンピューティングリソース106内の全データを失わせる。共有コンピューティングリソース106を含むラック102は、異なるプロジェクトのためのコアコンピューティングリソースを含むラックから切り離され得る。そして、1つ以上のケーブル128が、共有コンピューティングリソース106を含むラック102を、図1におけるコアコンピューティングリソース104を含むラック102へ接続するために使用され得る。共有コンピューティングリソース106は、電

源を入れ直され得る。それから、共有コンピューティングリソース 106 は、新たに割り当てられたプロジェクトのためのコアコンピューティングリソース 104 による使用のために利用可能になる。

【0039】

このアプローチは、プロジェクト間のデータセキュリティを依然として維持しながら、共有コンピューティングリソース 106 が、先行アプローチと比較して、よりずっと速く割り当て及び再割り当てされることを可能にする。更に、このアプローチは、多数のハードドライブ又は他の不揮発性記憶デバイスを繰り返しサニタイズする必要性を回避する。例えば、ここでの共有コンピューティングリソース 106 が、単一のハードドライブを夫々備えた 128 個の作業コンピューティングノード 126 を含む場合に、共有コンピューティングリソース 106 を 1 つのプロジェクトから他へ移すことは、128 個のハードドライブのための適切なサニタイゼーションレコードの生成とともに、これらのハードドライブをワイピング又は置換することを必要とする。作業コンピューティングノード 126 から不揮発性メモリを排除又は省略し、作業コンピューティングノード 126 を使用するようにコアコンピューティングリソース 104 を構成することによって、このアプローチは、共有コンピューティングリソース 106 内のハードドライブをワイピング又は置換する必要性を除くことができる。

【0040】

図 1 は、リザベーションに基づいた高性能コンピューティングシステムで使用されるコンピューティングクラスタ 100 の一例を表すが、様々な変更が図 1 に対して行われてよい。例えば、コアコンピューティングリソース 104 及び共有コンピューティングリソース 106 の夫々は、ラック 102 をいくつでも占有してよい。また、コアコンピューティングリソース 104 及び共有コンピューティングリソース 106 を形成する実際のコンピューティングノードは、必要に応じて又は望まれるように様々であってよく、上記のコンピューティングノードのタイプ以外の又はそれらに追加したコンピューティングノードが、ここで使用されてもよい。具体例として、コアコンピューティングリソース 104 は、内蔵キーボード、ビデオディスプレイ、及びマウスを含む「フリップ KVM」モジュールを含んでもよく、KVM モジュールは、ローカルメンテナンスアクティビティをサポートするようにコアコンピューティングリソース 104 内の如何なるノードへも接続され得る。

【0041】

図 2 は、本開示に係る、1 つ以上の再設定可能なコンピューティングクラスタを含む例示的なリザベーションに基づいた高性能コンピューティングシステム 200 を表す。図 2 に示されるように、コンピューティングシステム 200 は、多数のラック 202 に収容されている様々なコンピューティングリソースを含む。先と同じく、各ラック 202 は、如何なる適切なサイズ、形状、及び寸法も有することができ、各ラック 202 は、コンピューティングリソースをいくつでも保持するよう構成され得る。また、各ラック 202 は、ラック 202 内のコンピューティングリソースへ電力を分配するために使用され得る少なくとも 1 つの電力分配ユニットを含んでよい。各電力分配ユニットは、如何なる適切な量の電力も供給するよう構成されてよく、各ラック 202 は、複数の冗長的な電力分配ユニットを含んでよい。しかし、コンピューティングリソースは、如何なる適切な様態でも収容及び給電されてよいことに留意されたい。

【0042】

この例では、コアコンピューティングリソースの 3 つのセット 204 a ~ 204 c と、共有コンピューティングリソースの 3 つのセット 206 a ~ 206 c が存在する。コアコンピューティングリソースの異なるセット 204 a ~ 204 c は、一般的に、特定のプロジェクトに割り当てられ、プロジェクト間で共有されないコンピューティングノードを表す。よって、例えば、コアコンピューティングリソース 204 a は、第 1 プロジェクトと関連付けられ得、コアコンピューティングリソース 204 b は、第 2 プロジェクトと関連付けられ得、コアコンピューティングリソース 204 c は、第 3 プロジェクトと関連付けられ得る。政府セキュリティ規則又は他の要件に従うために、コアコンピューティングリ

ソース204a～204cの間に接続はなくてもよい。また、コアコンピューティングリソース204a～204cは空隙をあけられてよい。つまり、コアコンピューティングリソース204a～204cは、如何なる安全でないネットワーク（例えば、インターネット）へも接続を有さない。

【0043】

コアコンピューティングリソースの各セット204a～204cは、特定のプロジェクトに割り当てられている如何なる適切なコンピューティングノードも含む。この例では、コアコンピューティングリソースの各セット204a～204cは、一対のラック202に收容されている。コアコンピューティングリソースの各セット204a～204cにおける1つのラック202は、コアコンピューティングリソース104に関して上述されたコンピューティングノードの一部又は全部を含んでよく、コアコンピューティングリソースの各セット204a～204cにおける他のラック202は、特定のプロジェクトに専用の追加のコンピューティングノード（例えば、追加ノード120）を含んでよい。しかし、ラック202内のコンピューティングノードの如何なる適切な配置も使用されてよいことに留意されたい。

【0044】

コアコンピューティングリソースの異なるセット204a～204cは、関連するプロジェクトのニーズに応じて、異なる数及びタイプのコンピューティングノードを有してよい。例えば、コンピューティングリソースの1つのセット204aは、多数のサーバコンピューティングノードを含んでよく、コアコンピューティングリソースの他のセット204bは、より少ないサーバコンピューティングノードと、複数のグラフィクス処理ユニットとを含んでよく、コアコンピューティングリソースの更なる他のセット204cは、多数のグラフィクス処理ユニットを含んでよい。コアコンピューティングリソースの各セット204a～204cは、通常は、1つ以上の管理サーバ、1つ以上のスクラッチサーバ、1つ以上のセッションサーバ、及び場合により上記の他のコンポーネントを含んでよい（なお、これらのコンポーネントのいくつかは、特定の実施では省略されてもよい）。

【0045】

同様に、共有コンピューティングリソースの各セット206a～206cは、複数のプロジェクトの間で共有され得る如何なる適切なコンピューティングノードも含む。この具体例で、共有コンピューティングリソースの各セット206a～206cは3つのラック202に收容されている。なお、ラック202の数は、必要に応じて又は望まれるように様々であることができる。共有コンピューティングリソースの異なるセット206a～206cは、複数のプロジェクトの間で共有され得る異なるタイプのコンピューティングノードを含んでよい。例えば、共有コンピューティングリソースの1つのセット206aは、多数のサーバコンピューティングノードを含んでよく、共有コンピューティングリソースの他のセット206bは、より少ないサーバコンピューティングノードと、複数のグラフィクス処理ユニットとを含んでよく、共有コンピューティングリソースの更なる他のセット206cは、多数のグラフィクス処理ユニットを含んでよい。

【0046】

様々なケーブル208が、コアコンピューティングリソース204a～204cを所望の共有コンピューティングリソース206a～206cへ結合するために使用される。ケーブル208は、例えば、コンピューティングリソース間の光ファイバ接続又は他の適切な高速データ接続に相当し得る。先と同じく、コアコンピューティングリソース204a～204cを含むラック202と共有コンピューティングリソース206a～206cを含むラック202との間に冗長なケーブル208があってもなくてもよい。

【0047】

図2から分かるように、異なるプロジェクトは、異なるコンピューティングニーズを有するので、異なる共有コンピューティングリソースへ結合され得る。例えば、ここでのコアコンピューティングリソースのセット204aは、共有コンピューティングリソースのセット206aにおける2つのラック202へ、及び共有コンピューティングリソースの

セット 206b における単一のラック 202 へ結合されている。これは、多数のサーバコンピューティングノード及び限られた数のグラフィクス処理ユニットを備えたコアコンピューティングリソースのセット 204a をもたらす。ここでのコアコンピューティングリソースのセット 204b は、共有コンピューティングリソースのセット 206b における 2 つのラック 202 へ結合されている。これは、コアコンピューティングリソースのセット 204a と比べてサーバコンピューティングノードは少ないがグラフィクス処理ユニットは多いコアコンピューティングリソースのセット 204b をもたらす。ここでのコアコンピューティングリソースのセット 204c は、共有コンピューティングリソースのセット 206c における 2 つのラック 202 へ結合されている。これは、コアコンピューティングリソースのセット 204a ~ 204b と比較してサーバコンピューティングノードは更に少ないがグラフィクス処理ユニットは更に多いコアコンピューティングリソースのセット 204c をもたらす。

10

【0048】

図 3 は、本開示に係る、図 2 のリザーベーションに基づいた高性能コンピューティングシステム 200 の再構成の例を表す。この例では、コアコンピューティングリソース 204b に関連したプロジェクトが、コアコンピューティングリソース 204a によって前に使用されていた共有コンピューティングリソース 206b のラック 202 の使用を必要とする、ことが仮定される。これは、コアコンピューティングリソース 204b による一時的な必要性、又はコアコンピューティングリソース 204a の処理タスクの完了などの様々な理由で起こり得る。どのような理由であろうと、コアコンピューティングリソース 204a によって前に使用されていた共有コンピューティングリソース 206b は、シャットダウンされて、検査され得る。コアコンピューティングリソース 204a によって前に使用されていた共有コンピューティングリソース 206b は、コアコンピューティングリソース 204a から切り離され、コアコンピューティングリソース 204b へ接続され得、それらの共有コンピューティングリソース 206b は、コアコンピューティングリソース 204b による使用のために電源を入れられ得る。このようにして、共有コンピューティングリソースは、必要に応じて異なるプロジェクトに容易に割り当て及び再割り当てされ得る。

20

【0049】

図 2 及び図 3 で、各コンピューティングクラスタは、コアコンピューティングリソースのセット 204a ~ 204c のうちの 1 つに、コアコンピューティングリソースのそのセット 204a ~ 204c へ接続されている共有コンピューティングリソース 206a ~ 206c のいずれかをプラスしたものであることによって、定義される。結果として、各コンピューティングクラスタは再設定可能であり、コンピューティングクラスタにより使用される共有コンピューティングリソースが変化するにつれて時間とともに変化し得る。更に、異なるプロジェクトが必要に応じて特定の共有コンピューティングリソースをリザーブすることが可能であるので、コンピューティングシステム 200 は「リザーベーションに基づく」(reservation-based) ことができ、リザーブされた共有コンピューティングリソースは、そのようなりザーベーションに従ってコアコンピューティングリソース 204a ~ 204c の間で容易に移動され得る。いくつかの場合に、共有コンピューティングリソースのリザーベーションは、時間にわたるコアコンピューティングリソース 204a ~ 204c の予想されるワークロードに基づくことができ、コアコンピューティングリソース 204a ~ 204c のパフォーマンスメトリクスによってサポートされ得る。

30

40

【0050】

図 2 は、1 つ以上の再設定可能なコンピューティングクラスタを含むリザーベーションに基づいた高性能コンピューティングシステム 200 の一例を表し、図 3 は、図 2 のリザーベーションに基づいた高性能コンピューティングシステム 200 の一例となる再構成を表すが、様々な変更が図 2 及び図 3 に対して行われてよい。例えば、コンピューティングシステム 200 は、如何なる適切な数及びセットのコアコンピューティングリソースも、及び如何なる適切な数及びセットの共有コンピューティングリソースも含んでよい。また、コ

50

ンピューティングシステム 200 は、コアコンピューティングリソースと共有コンピューティングリソースとの間の如何なる適切な接続も含んでよい。

【0051】

図 4 は、本開示に係る、リザーベーションに基づいた高性能コンピューティングシステムをサポートする例示的な機能アーキテクチャ 400 を表す。説明を簡単にするために、図 4 に示される機能アーキテクチャ 400 は、図 1 のコンピューティングクラスタ 100 の複数のインスタンスを含み得る図 2 及び図 3 のコンピューティングシステム 200 で使用されているものとして説明され得る。しかし、図 4 に示される機能アーキテクチャ 400 は、如何なる適切なシステムにおいても、如何なる適切なコンピューティングクラスタによっても、使用されてよい。

【0052】

図 4 に示されるように、ここでの機能アーキテクチャ 400 は、1 つ以上のプロジェクト空間 402 a ~ 402 c 及び 1 つ以上の共有リソース空間 404 を含む。各プロジェクト空間 402 a ~ 402 c は、一般的に、特定のプロジェクトに関連した機能を表す。例えば、各プロジェクト空間 402 a ~ 402 c は、特定のプロジェクトのためのコアコンピューティングリソースの関連するセット 204 a ~ 204 c の機能を表し得る。各共有リソース空間 404 は、一般的に、共有コンピューティングリソースに関連した機能を表す。例えば、共有リソース空間 404 は、共有コンピューティングリソースの 1 つ以上のセット 206 a ~ 206 c の機能を表し得る。この例では、共有リソース空間 404 は、必要に応じて又は望まれるようにプロジェクト空間 402 a ~ 402 c の異なる 1 つによってリザーブされるか又はそれに割り当てられ得る様々なコンピューティングノード 406 によって少なくとも部分的に定義される。各プロジェクト空間 402 a ~ 402 c は、（等しかろうと等しくなからうと）任意の適切な数のコンピューティングノード 406 をリザーブするか又はそれらを割り当てられてよく、各プロジェクト空間 402 a ~ 402 c によってリザーブされるコンピューティングノード 406 の数は、時間とともに（場合により大幅に）変化し得ることに留意されたい。

【0053】

各プロジェクト空間 402 a ~ 402 c はローカルプロジェクト機能 408 を含む。ローカルプロジェクト機能 408 は、一般的に、プロジェクトに関連したエンジニア又は他の者へ提供されるか又はその者によって実行される機能を表す。例えば、ローカルプロジェクト機能 408 は、プロジェクトに関連したエンジニア又は他の者によって使用されるワークステーション又は他のローカルコンピュータの機能を表し得る。セッションホスト 410 は、作業者がコアコンピューティングリソースにリモートでログイン及びアクセスすることを可能にする機能を表す。例えば、セッションホスト 410 は、コアコンピューティングリソースの部分形成するセッションサーバ 114 によって提供される機能を表し得る。

【0054】

高性能コンピューティング（High-Performance Computing, HPC）管理機能 412 は、特定のプロジェクト又はプロジェクト空間に割り当てられている共有リソース空間 404 内のいずれかのリザーブされているコンピューティングノード 406 との相互作用をサポートするために使用される。例えば、HPC 管理機能 412 は、コアコンピューティングリソースの部分形成する管理サーバ 108 a ~ 108 b 及びスクラッチサーバ 110 a ~ 110 b、112 によって提供される機能を表し得る。とりわけ、HPC 管理機能 412 は、リザーブされているコンピューティングノード 406 を、関連するプロジェクト空間 402 a ~ 402 c 内のローカルノードとして出現させるエミュレーション又は他の機能をサポートするか、あるいは、関連するプロジェクト空間 402 a ~ 402 c でのリザーブされているコンピューティングノード 406 の使用を別なふうにサポートすることができる。HPC 管理機能 412 はまた、追加のリザーブされているコンピューティングノード 406 がプロジェクト空間 402 a ~ 402 c に割り当てられて接続される場合を検出することもでき、プロジェクト空間 402 a ~ 402 c でのそれらの追加のリザ

10

20

30

40

50

ーブされているコンピューティングノード406の使用をサポートすることができる。更に、HPC管理機能412は、特定のリザーブされているコンピューティングノード406が再割り当てされ、プロジェクト空間402a~402cから切り離される場合を検出することができ、それらの追加のリザーブされているコンピューティングノード406をプロジェクト空間402a~402cから除くことができる。このようにして、HPC管理機能412は、プロジェクト空間402a~402cの他のコンポーネントに対する変更を必要とせずに、共有コンピューティングリソースの一時的な使用をサポートすることができる。

【0055】

HPC管理機能412はまた、HPCストレージ414へのアクセスを提供してもよく、かつ、1つ以上の専用のコンピューティングノード416の使用をサポートしてもよい。HPCストレージ414は、共有リソース空間404からの情報（又は共有リソース空間404からの情報に基づいたデータ）が記憶され得る1つ以上の記憶デバイスを表す。例えば、上述されたように、エンジニア又は他の者は、セッションサーバ114を介してスクラッチサーバ110a~110bで処理中のジョブデータをレビューし、データを後処理し、結果をより永続的なストレージにコピーすることができる。ここでのHPCストレージ414は、結果又は他のデータが移動され得る少なくとも1つの記憶デバイスを表す。また、上述されたように、いくつかのコアコンピューティングリソースは、例えば、コアコンピューティングリソース104が1つ以上の作業コンピューティングノード120を含む場合など、特定の機能を実行するための専用コンピューティングリソースを含んでもよい。これらのリソースは、専用のコンピューティングノード416に相当し、HPC管理機能412は、これらの専用のコンピューティングノード406へのアクセスを提供することができる。HPC管理機能412はまた、専用のコンピューティングノード406からの情報（又は専用のコンピューティングノード406からの情報に基づいたデータ）をHPCストレージ414に記憶することもできる。

【0056】

図4は、リザーベーションに基づいた高性能コンピューティングシステムをサポートする機能アーキテクチャ400の一例を表すが、様々な変更が図4に対して行われてよい。例えば、機能アーキテクチャ400は、如何なる適切な数のプロジェクト空間も、及び如何なる適切な数の共有リソース空間も含んでよい。また、異なるプロジェクト空間は、異なる機能コンポーネント及び機能コンポーネントの配置を有してよく、異なる共有リソース空間は、異なる数及びタイプの共有コンピューティングリソースを有してよい。

【0057】

図5は、本開示に係る、リザーベーションに基づいた高性能コンピューティングシステムにおける例示的なコンピューティングノード500を表す。説明を簡単にするために、図5に示されるコンピューティングノード500は、図1のコンピューティングクラスタ100の複数のインスタンスを含み得る図2及び図3のコンピューティングシステム200で使用されているものとして説明され得る。例えば、コンピューティングノード500の1つ以上のインスタンスは、上記のコアコンピューティングリソース及び共有コンピューティングリソースの夫々を少なくとも部分的に実装するために使用されてよい。しかし、それらのコンポーネントの夫々は、如何なる他の適切な様態でも実装されてよい。また、図5に示されるコンピューティングノード500は、如何なる適切なシステムにおいても、如何なる適切なコンピューティングクラスタによっても、使用されてよい。

【0058】

図5に示されるように、コンピューティングノード500は、少なくとも1つの処理デバイス502、少なくとも1つの記憶デバイス504、少なくとも1つの通信ユニット506、及び少なくとも1つの入出力（I/O）ユニット508を含むコンピューティングデバイス又はシステムを表す。処理デバイス502は、揮発性ストレージ510にロードされるか又は不揮発性ストレージ512から読み出され得る命令を実行してよい。処理デバイス502は、如何なる適切な配置でも、如何なる適切な数及びタイプのプロセッサ又

10

20

30

40

50

は他の処理デバイスも含んでよい。処理デバイス 502 のタイプの例として、1 つ以上のマイクロプロセッサ、マイクロコントローラ、デジタル信号プロセッサ (Digital Signal Processors, DSP)、特定用途向け集積回路 (Application Specific Integrated Circuits, ASIC)、フィールドプログラマブルゲートアレイ (Field Programmable Gate Arrays, FPGA)、又はディスクリート回路がある。

【0059】

揮発性ストレージ 510 及び不揮発性ストレージ 512 は、情報 (例えば、データ、プログラムコード、及び／又は一時的又は永続的な他の適切な情報) を記憶すること及びその読み出しを助けることが可能な如何なる構造も表す記憶デバイス 504 の例である。揮発性ストレージ 510 は、ランダムアクセスメモリ又は任意の他の適切な揮発性記憶デバイスを表し得る。不揮発性ストレージ 512 は、ハードドライブ、ソリッドステートドライブ、又は任意の他の適切な不揮発性記憶デバイスのような、データの長期記憶をサポートする 1 つ以上のコンポーネント又はデバイスを含んでよい。特定のコンポーネント (例えば、共有コンピューティングリソース) には不揮発性ストレージがないので、不揮発性ストレージ 512 は、任意であるものとしてここでは示されていることに留意されたい。

【0060】

通信ユニット 506 は、他のシステム又はデバイスとの通信をサポートする、例えば、通信ユニット 506 は、有線又は無線ネットワークを介した通信を助けるネットワークインターフェース又は無線トランシーバを含むことができる。通信ユニット 506 は、1 つ以上の電気、光、又は無線通信リンクなどの如何なる適切な物理又は無線通信リンクも通る通信をサポートしてよい。

【0061】

I/O ユニット 508 は、データの入力及び出力を可能にする。例えば、I/O ユニット 508 は、キーボード、マウス、キーパッド、タッチスクリーン、又は他の適切な入力デバイスを通じたユーザ入力のための接続を提供し得る。I/O ユニット 508 はまた、ディスプレイ又は他の適切な出力デバイスへ出力を送ってもよい。しかし、コンピューティングノード 500 が、リモートでアクセスされ得るサーバ又は他のデバイスに相当する場合など、コンピューティングノード 500 がローカル I/O を必要としない場合には、I/O ユニット 508 は省略されてもよいことに留意されたい。

【0062】

図 5 は、リザベーションに基づいた高性能コンピューティングシステムにおけるコンピューティングノード 500 の一例を表すが、様々な変更が図 5 に対して行われてよい。例えば、コンピューティング及び他のデータ処理デバイス及びシステムは、多種多様な構成で提供され、図 5 は、如何なる特定のコンピューティングデバイス又はシステムにも本開示を制限しない。

【0063】

図 6 は、本開示に係る、リザベーションに基づいた高性能コンピューティングのための例示的な方法 600 を表す。説明を簡単にするために、図 6 に示される方法 600 は、図 1 のコンピューティングクラスタ 100 の複数のインスタンスを含み得る図 2 及び図 3 のコンピューティングシステム 200 の使用を伴うものとして説明され得る。しかし、図 6 に示される方法 600 は、如何なる適切なシステムによっても、如何なる適切なコンピューティングクラスタによっても、使用されてよい。

【0064】

図 6 に示されるように、ステップ 602 で、共有コンピューティングリソースは、第 1 プロジェクトのコアコンピューティングリソースとともに作動する。これは、例えば、特定のプロジェクトのために共有コンピューティングリソース 106、206a~206c にある 1 つ以上の作業コンピューティングノード 120 (例えば、1 つ以上のサーバ又はグラフィクス処理ユニット) を作動させることを含んでよい。ここでの共有コンピューティングリソース 106、206a~206c は、第 1 プロジェクトのための如何なる適切なデータ処理動作も実行してよい。ステップ 604 で、共有コンピューティングリソースが

再割り当てされるべきかどうか判定される。これは、例えば、第1プロジェクトによる共有コンピューティングリソース106、206a~206cのリザーベーションが終了しているか、あるいは、他のプロジェクトによる共有コンピューティングリソース106、206a~206cのリザーベーションが開始しているかどうかを判定することを含んでよい。そうでない場合には、第1プロジェクトは、共有コンピューティングリソース106、206a~206cを使用し続けることができる。

【0065】

共有コンピューティングリソースが再割り当てされている場合には、ステップ606で、共有コンピューティングリソースは電源を切られる。これは、例えば、作業者が、共有コンピューティングリソース106、206a~206cを含むラック102、202の電源を切り、共有コンピューティングリソース106、206a~206cを検査して、それがもはや電力を受け取っていないことを確かめることを含んでよい。共有コンピューティングリソースは、ステップ608で第1プロジェクトのコアコンピューティングリソースから切り離され、ステップ610で第2プロジェクトのコアコンピューティングリソースへ接続される。これは、例えば、作業者が、共有コンピューティングリソース106、206a~206cを第1プロジェクトのコアコンピューティングリソース104、204a~204cから切り離し、共有コンピューティングリソース106、206a~206cを第2プロジェクトのコアコンピューティングリソース104、204a~204cへ接続するために、適切なケーブル128、208を切り替えるか又は移動することを含んでよい。ステップ612で、共有コンピューティングリソースは電源を入れ直される。これは、例えば、作業者が、共有コンピューティングリソース106、206a~206cを含むラック102、202の電源を入れ、共有コンピューティングリソース106、206a~206cを検査して、それが電源を受け取っていることを確かめることを含んでよい。

【0066】

共有コンピューティングリソースは、ステップ614で第2プロジェクトのコアコンピューティングリソースとともに作動し、ステップ616で、第2プロジェクトのデータ交換及びデータ処理動作が、共有コンピューティングリソースを用いてサポートされる。これは、例えば、第2プロジェクトのための如何なる適切なデータ処理動作も実行するために共有コンピューティングリソース106、206a~206cを作動させることを含んでよい。この時点で、第2プロジェクトは、共有コンピューティングリソース106、206a~206cが再割り当てされ（、方法600が、再び、共有コンピューティングリソース106、206a~206cを再割り当てするために繰り返され得）るまで、共有コンピューティングリソース106、206a~206cを使用し続けることができる。

【0067】

図6は、リザーベーションに基づいた高性能コンピューティングのための方法600の一例を表すが、様々な変更が図6に対して行われてよい。例えば、連続したステップとして示されているが、図6の様々なステップは、重なり合っても、同時に行われても、異なる順序で行われても、又は何回行われてもよい。また、共有コンピューティングリソースはいくつでも、いくつかのプロジェクトの間でも必要に応じて又は望まれるように割り当て及び再割り当てされてよい。

【0068】

いくつかの実施形態で、本明細書で説明されている様々な機能は、コンピュータ可読プログラムコードから形成され、コンピュータ可読媒体において具現化されるコンピュータプログラムによって、実装又はサポートされる。「コンピュータ可読プログラムコード」という表現は、ソースコード、オブジェクトコード、及び実行可能コードを含む如何なるタイプのコンピュータコードも含む。「コンピュータ可読媒体」という表現は、リードオンリーメモリ（Read Only Memory, ROM）、ランダムアクセスメモリ（Random Access Memory, RAM）、ハードディスクドライブ（Hard Disk Drive, HDD）、コンパクトディスク（Compact Disc, CD）、デジタルビデオディスク（Digital V

ideo Disc, DVD)、又は任意の他のタイプのメモリなどの、コンピュータによってアクセス可能な如何なるタイプの媒体も含む。「非一時的な」コンピュータ可読媒体は、一時的な電気又は他の信号を運ぶ有線、無線、光、又は他の通信リンクを除く。非一時的なコンピュータ可読媒体は、データが永続的に記憶され得る媒体と、書き換え可能な光ディスク又は消去可能な記憶デバイスのような、データが記憶されての地に上書きされ得る媒体とを含む。

【0069】

本明細書の全体を通して使用される特定の語及び語句の定義を説明することが有利であり得る。「アプリケーション」及び「プログラム」という用語は、適切なコンピュータコード（ソースコード、オブジェクトコード、又は実行可能コードを含む）での実施のために適応された1つ以上のコンピュータプログラム、ソフトウェアコンポーネント、命令の組み、プロシージャ、関数、オブジェクト、クラス、インスタンス、関連データ、又はその部分を指す。「通信する」(communicate)という用語及びその派生語は、直接通信及び間接通信の両方を包含する。「含む」(include)及び「有する」(comprise)という用語並びにそれらの派生語は、制限なしの包含を意味する。「又は」(or)という用語は、包括的であって、「及び／又は」(and/or)を意味する。「～に関連した」(associated with)という語句及びその派生語句は、含む(include)、～に含まれる(be included within)、～と相互接続する(interconnect with)、含む(contain)、～に含まれる(be contained within)、～へ又は～と接続する(connect to又はwith)、～へ又は～と結合する(couple to又はwith)、～と通信可能である(be communicable with)、～と協働する(cooperate with)、インターリーブする(interleave)、並列する(juxtapose)、～に近接する(be proximate to)、～へ又は～と結びつけられる(be bound to又はwith)、持つ(have)、～の性質を持つ(have a property of)、～と関係を持つ(have a relationship to又はwith)、などを意味し得る。「～のうちの少なくとも1つ」(at least one of)という語句は、アイテムのリストともに使用される場合に、リストアップされているアイテムのうちの1つ以上の異なる組み合わせが使用されてもよく、また、リスト内のただ1つのアイテムしか必要とされなくてもよいことを意味する。例えば、「A、B、及びCのうちの少なくとも1つ」は、次の組み合わせ：A、B、C、AとB、AとC、BとC、AとBとCのいずれかを含む。

【0070】

本願の説明は、いずれかの特定の要素、ステップ、又は機能が、請求項の範囲に含まなければならない必須又は重要な要素であることを暗示するものとして、読まれるべきではない。特許対象の範囲は、許可された請求項によってのみ定義される。更に、請求項のどれもが、「～する手段」(means for)又は「～するステップ」(step for)という厳密な語が、機能を特定する分詞句とともに、特定の請求項で明示的に使用されない限りは、添付の特許請求の範囲又は請求項のいずれに関しても、米国特許法(35 U.S.C)第122(f)条を行使しない。請求項内の「メカニズム」、「モジュール」、「デバイス」、「ユニット」、「コンポーネント」、「要素」、「メンバー」、「装置」、「マシン」、「システム」、「プロセッサ」、又は「コントローラ」などの(しかし、これらに限られない)用語の使用は、特許請求の範囲自体の特徴によって更に変更又は強調されるように、当業者に知られている構造を指すと理解及び意図され、米国特許法第122(f)条を行使する意図はない。

【0071】

本開示は、特定の実施形態及び一般的に関連する方法について説明しており、一般的に、それらの実施形態及び方法の代替及び置換は、当業者に明らかだろう。従って、例となる実施形態の上記の説明は、本開示を定義又は制約しない。他の変更、置換、及び代替も、続く特許請求の範囲によって定義される本開示精神及び範囲から逸脱せずに、可能である。

10

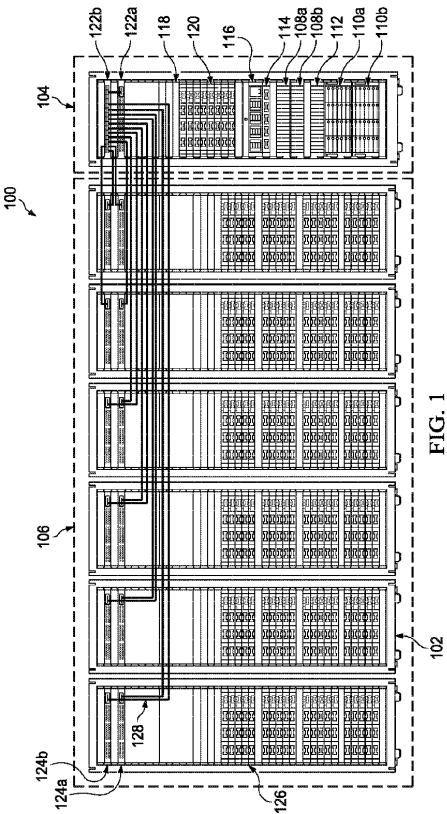
20

30

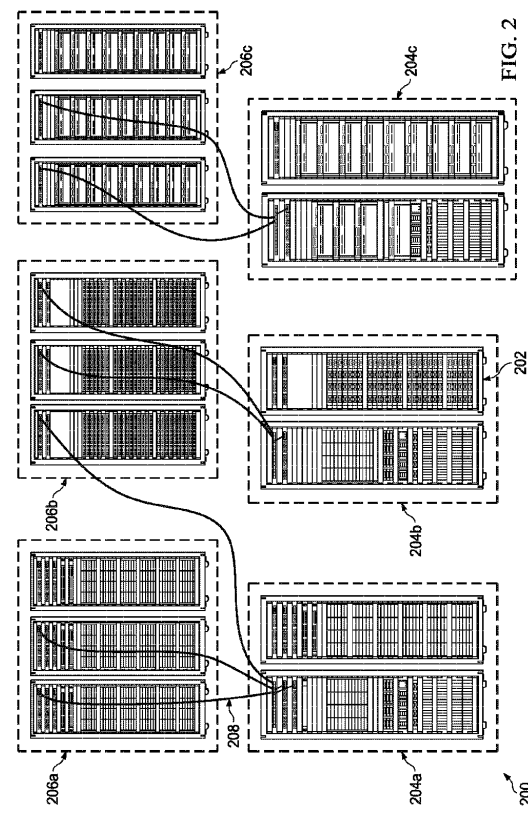
40

50

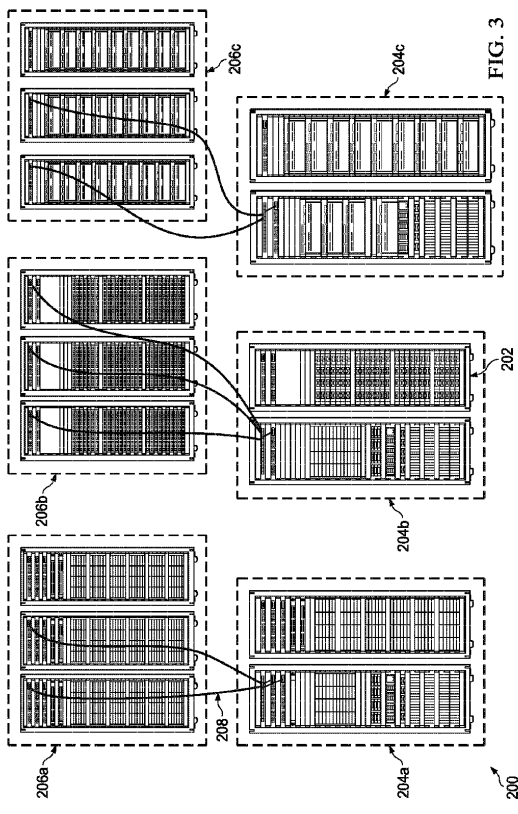
【図面】
【図 1】



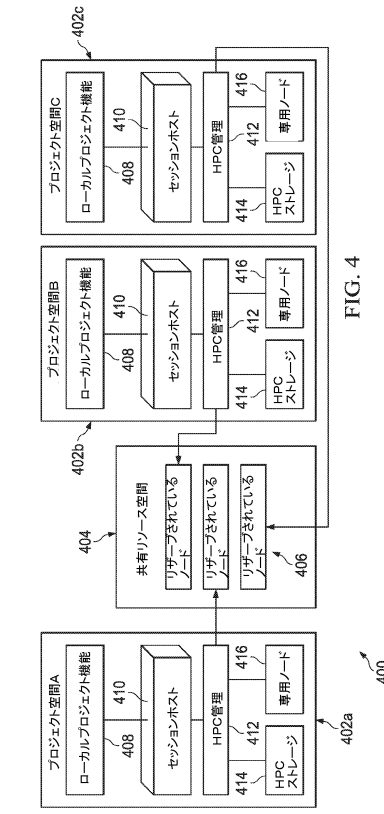
【図 2】



【図 3】



【図 4】



10

20

30

40

50

【図 5】

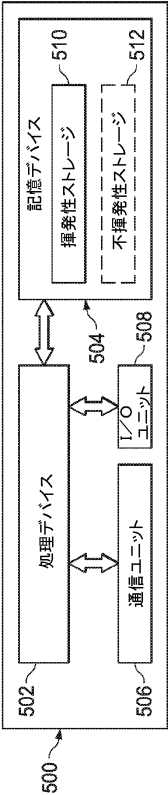


FIG. 5

【図 6】

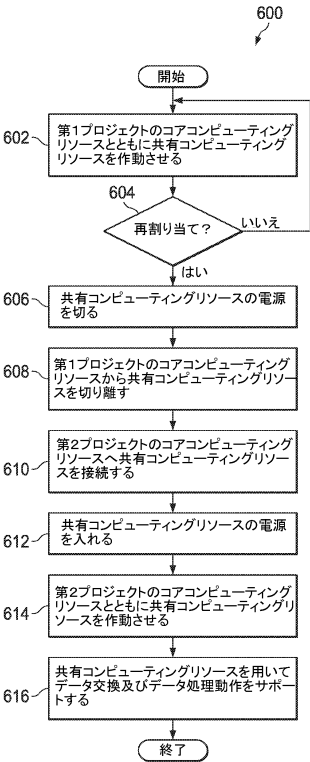


FIG. 6

フロントページの続き

- (72)発明者 ストーン, ジョン ディー.
 アメリカ合衆国 アリゾナ州 8 5 7 5 0 ツーソン ノース・サビーノ・キャニオン・ロード 3 5
 0 0 ユニット 7 2
- (72)発明者 スプーナー, ダッドリー エフ., ザ セカンド
 アメリカ合衆国 アリゾナ州 8 5 6 4 1 ベール イースト・ノーナ・レーン 1 2 6 3 8
- (72)発明者 バード, ライアン エル.
 アメリカ合衆国 アリゾナ州 8 5 6 4 1 ベール イースト・ドローダウン・トレイル 5 8 1
- (72)発明者 マクウォーター, アムジー エル.
 アメリカ合衆国 アリゾナ州 8 5 6 0 2 ベンソン サウス・スタリオン・ランチ・ロード 2 4 5
- 審査官 三坂 敏夫
- (56)参考文献 特開 2 0 0 6 - 2 6 8 7 1 1 (J P , A)
 特表 2 0 0 6 - 5 2 1 6 0 9 (J P , A)
 特表 2 0 0 7 - 5 3 3 0 3 2 (J P , A)
 特開 2 0 0 6 - 1 0 7 1 5 0 (J P , A)
 米国特許出願公開第 2 0 1 9 / 0 3 8 4 6 4 9 (U S , A 1)
- (58)調査した分野 (Int.Cl., D B 名)
 G 0 6 F 9 / 4 5 5 - 9 / 5 4