

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7154261号
(P7154261)

(45)発行日 令和4年10月17日(2022.10.17)

(24)登録日 令和4年10月6日(2022.10.6)

(51)国際特許分類

F I

G 0 6 F	13/14	(2006.01)	G 0 6 F	13/14	3 1 0 F
G 0 6 F	13/10	(2006.01)	G 0 6 F	13/10	3 4 0 A
G 0 6 F	9/455	(2006.01)	G 0 6 F	13/10	3 3 0 C
G 0 6 F	16/188	(2019.01)	G 0 6 F	9/455	1 5 0
			G 0 6 F	16/188	

請求項の数 9 (全38頁)

(21)出願番号	特願2020-161152(P2020-161152)	(73)特許権者	000005108 株式会社日立製作所 東京都千代田区丸の内一丁目6番6号
(22)出願日	令和2年9月25日(2020.9.25)	(74)代理人	110000279 特許業務法人ウィルフォート国際特許事務所
(65)公開番号	特開2022-54132(P2022-54132A)	(72)発明者	山本 彰 東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内
(43)公開日	令和4年4月6日(2022.4.6)	(72)発明者	江原 寛人 東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内
審査請求日	令和3年5月20日(2021.5.20)	(72)発明者	中村 隆喜 東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内

最終頁に続く

(54)【発明の名称】 複合型ストレージシステム

(57)【特許請求の範囲】

【請求項1】

複数のストレージ装置を有するストレージボックスと、1以上の仮想計算機を実行可能な複数のサーバとを備える複合型ストレージシステムであって、

前記ストレージボックスは、論理ボリュームを格納し、

前記サーバが実行可能な仮想計算機には、論理ボリュームを用いてアプリケーションを実行するアプリケーションVM(仮想計算機)と、制御権を有する論理ボリュームに対する読み書きを制御可能なコントローラを構成するコントローラVMとがあり、

前記複合型ストレージシステムの前記複数のサーバの少なくとも1つのプロセッサは、

移動元のサーバが有するアプリケーションVMを所定の移動先のサーバに移動させる所定の状況が発生した場合に、前記アプリケーションVMを前記移動先のサーバに移動し、

前記アプリケーションVMが使用する論理ボリュームの制御権を前記移動先のサーバのコントローラVMに対して移動する

複合型ストレージシステム。

【請求項2】

前記所定の状況は、新たなサーバを前記複合型ストレージシステムに追加する時であり、前記移動先のサーバは、前記新たなサーバである

請求項1に記載の複合型ストレージシステム。

【請求項3】

前記所定の状況は、アプリケーションVMを有する移動元のサーバを削除する時であり

、前記移動先のサーバは、前記複合型ストレージシステムの他のサーバである
請求項 1 に記載の複合型ストレージシステム。

【請求項 4】

前記所定の状況は、アプリケーション VM を有する移動元のサーバを新たなサーバに置き換える時であり、前記移動先のサーバは、置き換えるサーバである

請求項 1 に記載の複合型ストレージシステム。

【請求項 5】

前記複合型ストレージシステムの前記 1 以上のサーバの少なくとも 1 つのプロセッサは、
移動元のサーバが有するアプリケーション VM を所定の移動先のサーバに移動させる所
定の状況が発生した場合に、

前記アプリケーション VM が使用する論理ボリュームの前記移動元のコントローラ VM
のキャッシュメモリが記憶するキャッシュデータを、前記移動先のサーバのコントローラ
VM のキャッシュメモリに対して移動する

請求項 1 に記載の複合型ストレージシステム。

【請求項 6】

複数のストレージ装置を有するストレージボックスと、1 以上の仮想計算機を実行可能
な複数のサーバとを備える複合型ストレージシステムであって、

前記ストレージボックスは、論理ボリュームを格納し、

前記サーバが実行可能な仮想計算機には、論理ボリュームを用いてアプリケーションを
実行するアプリケーション VM と、制御権を有する論理ボリュームに対する読み書きを制
御可能なコントローラを構成するコントローラ VM とがあり、

前記複合型ストレージシステムの前記複数のサーバの少なくとも 1 つのプロセッサは、

前記複合型ストレージシステムの前記サーバの構成を変更する所定の状況が発生した場
合に、第 1 サーバが有する前記ストレージボックスの記憶領域の割り当て権限を、第 2 サ
ーバのコントローラ VM に対して移動する

複合型ストレージシステム。

【請求項 7】

前記所定の状況は、前記第 2 サーバを前記複合型ストレージシステムに追加する状況で
ある

請求項 6 に記載の複合型ストレージシステム。

【請求項 8】

前記所定の状況は、前記第 1 サーバを前記複合型ストレージシステムから削除する状況
である

請求項 6 に記載の複合型ストレージシステム。

【請求項 9】

前記所定の状況は、前記第 1 サーバを前記第 2 サーバに置き換える状況である

請求項 6 に記載の複合型ストレージシステム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、1 以上のストレージ装置を有するストレージボックスと、複数のサーバとを
備える複合型ストレージシステム等に関する。

【背景技術】

【0002】

複数のストレージ装置を収容したストレージボックスを、複数のストレージコントロー
ラで共有するシステムが知られている（例えば、特許文献 1 参照）。特許文献 1 に記載の
技術では、複数のストレージコントローラが、複数のストレージボックスを共有している
。特許文献 1 においては、新たなストレージコントローラを含むシステムが追加された場
合などに、データの読み書きの際に、サーバから指定される論理ボリューム LU (L o g
i c a l U n i t) の制御権を、既存のストレージコントローラを含むシステムから、

10

20

30

40

50

新たに追加されたストレージコントローラを含むシステムに移動する技術が開示されている。また、特許文献 1 には、新たにストレージボックスが追加された場合などに、新しく追加されたストレージボックスの記憶領域の割り当て権限、すなわち、どの L U のどの記憶領域に割り当てるかを決めることに関して、どの記憶領域をどのストレージコントローラを含むシステムに割り当てるかを決定する権限を、どのストレージコントローラに割り当てるかを決める技術が開示されている。

【 0 0 0 3 】

また、ストレージシステムに関する技術として、ハイパーコンバインドアーキテクチャと呼ばれる技術が知られている（例えば、特許文献 2 参照）。本アーキテクチャにおいては、サーバ上に、アプリケーションが動作する VM（仮想計算機）とストレージコントローラが動作する VM が搭載されており、サーバは内蔵ストレージをもっている。複数のサーバ上のストレージコントローラは、ネットワーク経由で接続されている。特許文献 2 には、アプリケーションに対しては、個々のサーバ上のストレージコントローラと内蔵ストレージとを併せて、一つの仮想ストレージシステムに見せる技術が開示されている。また、特許文献 2 には、サーバ上の VM で動作しているアプリケーションを、一度、停止させ、別のサーバに移動し、別のサーバの VM で動作を再開させる技術が開示されている。

【 0 0 0 4 】

また、ストレージに関する技術としては、シン・プロビジョニングと呼ばれる容量の仮想化技術が知られている（例えば、特許文献 3 参照）。特許文献 3 が開示された技術においては、仮想ストレージが容量の仮想化機能も持っている。容量の仮想化機能では、容量プールと呼ばれる比較的大容量の記憶領域を、ページと呼ばれる区画に分割する。従来は、L U を定義する際、あわせて、その容量も指定し、ストレージでは、その容量に相当する記憶領域を確保していた。一方、容量の仮想化機能をもつ仮想ストレージでは、L U の定義時には、記憶領域を確保せず、実際に書き込み発生した場合、該当する領域にページを割り当てる。これにより、実際に、割り当てる記憶領域を削減できる。また、記憶領域を割り当てる契機が、書き込みが発生した契機であるため、管理者は L U の容量を厳密に計算する必要がなく、比較的大きい容量を定義すればよいので、管理コストも低減できる。

【先行技術文献】

【特許文献】

【 0 0 0 5 】

【文献】特表 2 0 1 6 - 5 1 0 4 4 0 号公報

国際公開第 2 0 1 8 / 0 2 9 8 2 0 号

特開 2 0 0 8 - 1 9 8 2 2 1 号公報

【発明の概要】

【発明が解決しようとする課題】

【 0 0 0 6 】

例えば、ストレージシステムに、新たなストレージコントローラが追加された場合などには、サーバから指定される論理ボリュームの制御権を、既存のストレージコントローラから、新たに追加されたストレージコントローラに移動することが考えられる。例えば、サーバが新たに追加された場合に、既存のサーバの VM で動作するストレージコントローラが制御権をもつ論理ボリュームの制御権を、新たに追加したサーバの VM で動作させるストレージコントローラに移動することが考えられる。

【 0 0 0 7 】

ここで、論理ボリュームを使用しているのは、既存のサーバの VM 上で動作しているアプリケーションであり、アプリケーションが論理ボリュームを使用する場合には、アプリケーションが動作しているサーバ上のストレージコントローラに対して論理ボリュームに対する I O 要求（リード・ライト要求）を行い、このストレージコントローラが追加したサーバのストレージコントローラに対して論理ボリュームの I O 要求を送信し、結果として論理ボリュームへの I O 処理が行われることとなる。このため、アプリケーションによる論理ボリュームへの I O 効率が悪くなるという問題がある。

【 0 0 0 8 】

本発明は、上記事情に鑑みなされたものであり、その目的は、論理ボリュームへの I O 効率を向上することのできる技術を提供することにある。

【課題を解決するための手段】

【 0 0 0 9 】

上記目的を達成するため、一観点に係る複合型ストレージシステムは、複数のストレージ装置を有するストレージボックスと、1以上の仮想計算機を実行可能な複数のサーバとを備える複合型ストレージシステムであって、前記ストレージボックスは、論理ボリュームを格納し、前記サーバが実行可能な仮想計算機には、論理ボリュームを用いてアプリケーションを実行するアプリケーション VM (仮想計算機) と、制御権を有する論理ボリューム 10 に対する読み書きを制御可能なコントローラを構成するコントローラ VM とがあり、前記複合型ストレージシステムの前記1以上のサーバの少なくとも1つのプロセッサは、移動元のサーバが有するアプリケーション VM を所定の移動先のサーバに移動させる所定の状況が発生した場合に、前記アプリケーション VM を前記移動先のサーバに移動し、前記アプリケーション VM が使用する論理ボリュームの制御権を前記移動先のサーバのコントローラ VM に対して移動する。

【発明の効果】

【 0 0 1 0 】

本発明によれば、論理ボリュームへの I O 効率を向上することができる。

【図面の簡単な説明】

【 0 0 1 1 】

【図1】図1は、一実施形態に係る複合型ストレージシステムの全体構成図である。

【図2】図2は、一実施形態に係るアプリケーション VM 管理情報の構成図である。

【図3】図3は、一実施形態に係るコントローラ VM の構成図である。

【図4】図4は、一実施形態に係るキャッシュメモリの構成図である。

【図5】図5は、一実施形態に係るコントローラ VM 管理情報の構成図である。

【図6】図6は、一実施形態に係る仮想ストレージシステム情報の構成図である。

【図7】図7は、一実施形態に係る他ストレージシステム情報の構成図である。

【図8】図8は、一実施形態に係る仮想論理ボリューム情報の構成図である。

【図9】図9は、一実施形態に係る論理ボリューム情報の構成図である。

【図10】図10は、一実施形態に係るキャッシュ管理情報の構成図である。

【図11】図11は、一実施形態に係る空きキャッシュ管理情報管理キューを説明する図である。

【図12】図12は、一実施形態に係るストレージボックス情報の構成図である。

【図13】図13は、一実施形態に係るストレージグループ情報の構成図である。

【図14】図14は、一実施形態に係る実ページ情報の構成図である。

【図15】図15は、一実施形態に係る空きページ情報管理キューを説明する図である。

【図16】図16は、一実施形態に係る未管理実ページ情報管理キューを説明する図である。

【図17】図17は、一実施形態に係るストレージ装置情報の構成図である。

【図18】図18は、一実施形態に係る管理プログラムにより構成される機能部の構成図である。

【図19】図19は、一実施形態に係るリード処理のフローチャートである。

【図20】図20は、一実施形態に係るライト受付処理のフローチャートである。

【図21】図21は、一実施形態に係るライトアフタ処理のフローチャートである。

【図22A】図22Aは、一実施形態に係るサーバ追加処理の追加サーバ側処理のフローチャートである。

【図22B】図22Bは、一実施形態に係るサーバ追加処理の他サーバ側処理のフローチャートである。

【図23A】図23Aは、一実施形態に係るサーバ削除処理の削除サーバ側処理のフロー

10

20

30

40

50

チャートである。

【図 2 3 B】図 2 3 B は、一実施形態に係るサーバ削除処理の他サーバ側処理のフローチャートである。

【図 2 4 A】図 2 4 A は、一実施形態に係るサーバ移動処理の移動元サーバ側処理のフローチャートである。

【図 2 4 B】図 2 4 B は、一実施形態に係るサーバ移動処理の移動先サーバ側処理のフローチャートである。

【図 2 5】図 2 5 は、一実施形態に係るキャッシュデータ移動管理処理のフローチャートである。

【図 2 6 A】図 2 6 A は、一実施形態に係るキャッシュデータ移動処理の移動先サーバ側処理のフローチャートである。

10

【図 2 6 B】図 2 6 B は、一実施形態に係るキャッシュデータ移動処理の移動元サーバ側処理のフローチャートである。

【図 2 7】図 2 7 は、一実施形態に係る移動 VM 判定実行処理のフローチャートである。

【図 2 8】図 2 8 は、一実施形態に係る移動 VM 受領処理のフローチャートである。

【図 2 9 A】図 2 9 A は、一実施形態に係る第 1 のアプリケーション起動制御処理のフローチャートである。

【図 2 9 B】図 2 9 B は、一実施形態に係る第 2 のアプリケーション起動制御処理のフローチャートである。

【発明を実施するための形態】

20

【0012】

実施形態について、図面を参照して説明する。なお、以下に説明する実施形態は特許請求の範囲に係る発明を限定するものではなく、また実施形態の中で説明されている諸要素及びその組み合わせの全てが発明の解決手段に必須であるとは限らない。

【0013】

図 1 は、一実施形態に係る複合型ストレージシステムの全体構成図である。

【0014】

複合型ストレージシステム 10 は、1 つ以上のストレージボックス 130 と、1 つ以上のサーバ 110 と、ネットワーク 120 と、ストレージ管理サーバ 115 とを備える。ネットワーク 120 は、ストレージボックス 130 と、サーバ 110 と、ストレージ管理サーバ 115 とを接続する。サーバ 110 は、1 つ以上のサーバポート 195 を介してネットワーク 120 に接続される。ストレージボックス 130 は、ボックスポート 170 を介して、ネットワーク 120 に接続される。

30

【0015】

サーバ 110 は、例えば、プロセッサや、メモリや、記憶デバイス等を備えた、計算機であり、アプリケーションを搭載した 1 つ以上の VM 140 (アプリケーション VM) と、ストレージコントローラを搭載した VM 100 (コントローラ VM) と、VM を管理する VM 管理部 150 とを備える。なお、アプリケーションやストレージコントローラがコンテナとして動作するようにしてもよい。

【0016】

40

アプリケーション VM 140 は、アプリケーションを実行し、同一サーバ 110 上で動作するコントローラ VM 100 に、データを読み書きする I/O 要求 (リード・ライト要求) を発行する。

【0017】

コントローラ VM 100 は、アプリケーション VM 140 から受け取った I/O 要求で指定されたデータを格納したストレージボックス 130 に対して、ネットワーク 120 を介して、要求を伝える。各サーバ 110 の各コントローラ VM 100 は、互いに、ネットワーク 120 を介して、データの送受信を行う。本実施形態では、コントローラ VM 100 は、いわゆる容量仮想化機能を持っている例を示すが、コントローラ VM 100 が容量仮想化機能をもっていなくてもよい。

50

【 0 0 1 8 】

各サーバ 1 1 0 の VM 管理部 1 5 0 は、互いに、ネットワーク 1 2 0 を介して、例えば、VM に関する情報等の情報交換を行う。

【 0 0 1 9 】

ネットワーク 1 2 0 は、例えば、NVM-over-ethernet (ethernet は登録商標) 等のプロトコルに従っていてもよい。

【 0 0 2 0 】

本実施形態では、各サーバ 1 1 0 上の複数のコントローラ VM 1 0 0 によって仮想ストレージシステム 1 9 0 が構成されている。各サーバ 1 1 0 のアプリケーション VM 1 4 0 からは、仮想ストレージシステム 1 9 0 が、1 つのストレージシステムに見える。なお、
10

【 0 0 2 1 】

ストレージボックス 1 3 0 は、1 つ以上のストレージ装置 1 6 0 と、1 つ以上のボックスポート 1 7 0 と、1 つ以上のボックスプロセッサ 1 8 0 と、ボックスメモリ 1 8 1 とを含む。

【 0 0 2 2 】

ボックスポート 1 7 0 は、ストレージボックス 1 3 0 をネットワーク 1 2 0 に接続するためのポートである。ストレージ装置 1 6 0 は、データを記憶する記憶媒体を備える装置であり、例えば、HDD (Hard Disk Drive) や、フラッシュメモリを記憶媒体とするフラッシュストレージなどである。ストレージ装置 1 6 0 に使用されるフラッシュストレージには、いくつかの種類があり、高価格、高性能、消去可能回数の多い SLC と、これに対して、低価格、低性能、消去可能回数の少ない MLC がある。また、ストレージ装置 1 6 0 の記憶媒体としては、相変化メモリなどの新しい記憶媒体が含まれていてもよい。本実施形態では、ストレージボックス 1 3 0 のストレージ装置 1 6 0 は、複数のコントローラ VM 1 0 0 によって、共有される。したがって、ストレージボックス 1 3 0 の中のストレージ装置 1 6 0 を、共有ストレージ装置と呼んでもよい。なお、本実施例では、ストレージ装置 1 6 0 の中の 1 台の装置が故障しても、その装置のデータを回復できる Redundancy Array Independent Device (RAID) 機能をもっているものとする。RAID 機能を持っている場合には、複数の同一種類のストレージ装置が、1 つの RAID 構成をとることとなる。この RAID 構成をストレージグループと呼ぶ。本実施形態では、例えば、RAID 構成は、1 つのストレージボックス 1 3 0 内のストレージ装置 1 6 0 の集合から構成されているものとしている。なお、必ずしも RAID 機能を持っていなくてもよい。
20
30

【 0 0 2 3 】

ボックスプロセッサ 1 8 0 は、コントローラ VM 1 0 0 の指示に従ってストレージ装置 1 6 0 との間での転送処理を実行する。ボックスメモリ 1 8 1 は、ボックスプロセッサ 1 8 0 が使用するプログラムや情報を格納する。

【 0 0 2 4 】

ストレージボックス 1 3 0 は、ネットワーク 1 2 0 を介して、仮想ストレージシステム 1 9 0 の中の 1 つ以上のコントローラ VM 1 0 0 に接続される。ストレージボックス 1 3 0 は、仮想ストレージシステム 1 9 0 の中のすべてのコントローラ VM 1 0 0 と接続されている必要はない。また、或るコントローラ VM 1 0 0 と別のコントローラ VM 1 0 0 とに接続されているストレージボックス 1 3 0 の集合が同じであってもよく、異なってもよい。ストレージボックス 1 3 0 は、ネットワーク 1 2 0 を介して、いくつかのコントローラ VM 1 0 0 によって共有されている。
40

【 0 0 2 5 】

ストレージ管理サーバ 1 1 5 は、複合型ストレージシステム 1 0 の管理者が管理のために使用する装置であり、ネットワーク 1 2 0 などを介して、コントローラ VM 1 0 0 や、ストレージボックス 1 3 0 との間で通信を行い、各種情報を収集したり、各装置の設定を行ったりする。
50

【 0 0 2 6 】

次に、アプリケーションVM 140が管理するアプリケーションVM管理情報について説明する。

【 0 0 2 7 】

図2は、一実施形態に係るアプリケーションVM管理情報の構成図である。

【 0 0 2 8 】

アプリケーションVM管理情報は、第1プロセッサ使用量250と、第2プロセッサ使用量260と、仮想論理ボリューム番号270と、を含む。

【 0 0 2 9 】

第1プロセッサ使用量250と、第2プロセッサ使用量260とは、アプリケーションVM管理情報を格納しているアプリケーションVM 140のプロセッサの使用量を表している。第1プロセッサ使用量250は、所定の時点から現在までのプロセッサ使用量を示し、所定の時間が経過した場合に、0にクリアされて、その時点からのプロセッサの使用量を示す。第2プロセッサ使用量260は、所定の時間におけるアプリケーションVM 140のプロセッサの使用量を表しており、所定の時間が経過した場合における第1プロセッサ使用量250の値がコピーされる。

10

【 0 0 3 0 】

仮想論理ボリューム番号270は、アプリケーションVM 140が使用する論理ボリュームの識別子であり、1つ以上存在する。アプリケーションVM 140は、I/O要求(リード要求やライト要求)を発行する際には、仮想論理ボリューム番号270と論理ボリューム内の相対アドレスとを指定する。本実施形態においては、各サーバ110のVM管理部150が互いに、アプリケーションVM管理情報について情報交換を行って、アプリケーションVM 140を、サーバ110間で移動させる処理を行う。

20

【 0 0 3 1 】

次に、コントローラVM 100の構成について説明する。

【 0 0 3 2 】

図3は、一実施形態に係るコントローラVMの構成図である。

【 0 0 3 3 】

コントローラVM 100は、キャッシュメモリ210と、制御メモリ220と、プロセッサ230とを持つ。キャッシュメモリ210と、制御メモリ220とは、サーバ110がもつメモリの一部が、コントローラVM 100用に割り当てられたものである。また、プロセッサ230は、サーバ110がもつプロセッサの一部又はプロセッサによる一部の処理時間が、コントローラVM 100用に割り当てられたものである。本実施形態では、アプリケーションVM 140を、サーバ間110で移動させる場合には、そのアプリケーションVM 140が使用している仮想論理ボリュームの制御権を、コントローラVM 100間で移動するようにしている。

30

【 0 0 3 4 】

キャッシュメモリ210は、ストレージボックス130のストレージ装置160に格納されるデータの中で、コントローラVM 100からよくアクセスされるデータを格納する。コントローラVM 100は、アプリケーションVM 140から、I/O要求を受け付ける。コントローラVM 100は、アプリケーションVM 140からリード要求を受け取った場合、リード要求に対応するストレージ装置160から、データを読み込んで、キャッシュメモリ210に格納して、そのデータをアプリケーションVM 140に渡す。なお、読み込んだデータをキャッシュメモリ210に格納しなくてもよい。一方、コントローラVM 100は、アプリケーションVM 140からライト要求を受け取った場合、受け取ったライトデータをキャッシュメモリ210に書き込んだ段階で、ライト要求を完了させ、すなわち、ライト要求の応答を返し、後からライト要求に対応するストレージ装置160にライトデータを書き込む。なお、受け取ったライトデータを、ライト要求に対応するストレージ装置160に書き込んでから、ライト要求を完了させてもよい。

40

【 0 0 3 5 】

50

制御メモリ 220 は、コントローラ VM 100 による各種処理に必要なプログラムや、情報（コントローラ VM 管理情報）を格納する。

【0036】

プロセッサ 230 は、制御メモリ 220 に格納されたプログラムを実行することにより、各種処理を実行する。

【0037】

次にキャッシュメモリ 210 の構成について説明する。

【0038】

図 4 は、一実施形態に係るキャッシュメモリの構成図である。

【0039】

キャッシュメモリ 210 は、例えば、固定長のスロット 21100 に分割されて管理されている。スロット 21100 は、リード・ライトデータの割り当て単位となる。

【0040】

次に、コントローラ VM 100 の制御メモリ 220 に管理されているコントローラ VM 管理情報について説明する。

【0041】

図 5 は、一実施形態に係るコントローラ VM 管理情報の構成図である。

【0042】

コントローラ VM 管理情報は、仮想ストレージシステム情報 2050、他ストレージシステム情報 2070、1 以上の仮想論理ボリューム情報 2084、1 以上の論理ボリューム情報 2000、1 以上のストレージボックス情報 2060、1 以上のストレージグループ情報 2300、1 以上のストレージ装置情報 2500、1 以上のキャッシュ管理情報 2750、空きキャッシュ管理情報ポインタ 2650、1 以上の実ページ情報 2100、空きページ管理情報ポインタ 2400、未管理ページ情報ポインタ 2200、及び仮想ページ容量 2600 を含む。これら各情報について以下に詳細に説明する。

【0043】

まず、仮想ストレージシステム情報 2050 について説明する。

【0044】

図 6 は、一実施形態に係る仮想ストレージシステム情報の構成図である。

【0045】

仮想ストレージシステム情報 2050 は、仮想ストレージシステム 190 に関する情報であり、仮想ストレージシステム識別子 2051 と、コントローラ VM 2080 とを含む。仮想ストレージシステム識別子 2051 は、仮想ストレージシステム情報 2050 を格納する制御メモリ 220 を有するコントローラ VM 100（以下、制御メモリ 220 の情報の説明において、当該コントローラ VM 100 という）が含まれる仮想ストレージシステム 190 の識別子である。コントローラ VM 識別子 2080 は、当該コントローラ VM 100 を搭載しているサーバ 110（以下、制御メモリ 220 の情報の説明において、当該サーバ 110 という）上における当該コントローラ VM 100 の識別子である。

【0046】

次に、他ストレージシステム情報 2070 について説明する。

【0047】

図 7 は、一実施形態に係る他ストレージシステム情報の構成図である。

【0048】

他ストレージシステム情報 2070 は、仮想ストレージシステム識別子 2051 と、1 以上の他コントローラ VM 識別子 2072 とを含む。仮想ストレージシステム識別子 2051 は、図 6 の仮想ストレージシステム情報 2050 の仮想ストレージシステム識別子 2051 と同じであり、当該コントローラ VM 100 が含まれる仮想ストレージシステム 190 の識別子である。他コントローラ VM 識別子 2072 は、当該コントローラ VM 100 を含む仮想ストレージシステム 190 に含まれる各サーバ 100 上の各コントローラ VM 100 の識別子である。

10

20

30

40

50

【 0 0 4 9 】

次に、仮想論理ボリューム情報 2 0 8 4 について説明する。

【 0 0 5 0 】

図 8 は、一実施形態に係る仮想論理ボリューム情報の構成図である。

【 0 0 5 1 】

仮想論理ボリューム情報 2 0 8 4 は、仮想論理ボリューム及び論理ボリュームに対応して設けられる情報である。仮想論理ボリューム情報 2 0 8 4 は、仮想論理ボリューム識別子 2 0 8 5 と、制御権情報 2 0 8 6 と、コントローラ VM 識別子 2 0 8 7 と、ストレージポート識別子 2 0 8 8 と、論理ボリューム識別子 2 0 8 9 とを含む。

【 0 0 5 2 】

仮想論理ボリューム識別子 2 0 8 5 は、仮想論理ボリューム情報 2 0 8 4 に対応する仮想論理ボリューム（対応仮想論理ボリュームという）の識別子である。本実施形態では、仮想論理ボリュームに対しては、いずれか 1 つのコントローラ VM 1 0 0 が読み書きを行う権限（制御権）をもつ。制御権情報 2 0 8 6 は、当該コントローラ VM 1 0 0 が対応仮想論理ボリュームの制御権をもっているか、否かを表す。コントローラ VM 識別子 2 0 8 7 は、当該コントローラ VM 1 0 0 が対応仮想論理ボリュームの制御権をもっていない場合において、対応仮想論理ボリュームに対して制御権を持っているコントローラ VM 1 0 0 の識別子を表す。ストレージポート識別子 2 0 8 8 は、対応仮想論理ボリュームにアクセスする際のストレージポート 1 7 0 の識別子を表す。論理ボリューム識別子 2 0 8 9 は、対応仮想論理ボリュームの制御権を有している場合には、当該コントローラ VM 1 0 0 内の対応する論理ボリュームの識別子を表し、制御権を有していない場合には、制御権をもつコントローラ VM 1 0 0 内の対応する論理ボリュームの識別子を表す。

【 0 0 5 3 】

次に、論理ボリューム情報 2 0 0 0 について説明する。

【 0 0 5 4 】

図 9 は、一実施形態に係る論理ボリューム情報の構成図である。

【 0 0 5 5 】

論理ボリューム情報 2 0 0 0 を説明する前に、本実施形態におけるデータへの I O について説明する。本実施形態では、アプリケーション VM 1 4 0 がデータの I O（リード・ライト）の対象とする記憶装置は、仮想論理ボリュームである。アプリケーション VM 1 4 0 は、I O の対象となる仮想論理ボリュームの識別子、仮想論理ボリューム内の対象のアドレス、I O の対象のデータ長を指定した I O 要求（リード要求、ライト要求）を発行する。コントローラ VM 1 0 0 は、アプリケーション VM 1 4 0 から I O 要求を受け取ると、仮想論理ボリューム情報 2 0 8 4 に基づいて、対応する論理ボリュームの識別子を認識する。本実施形態では、論理ボリュームの識別子は、コントローラ VM 1 0 0 内でユニークな情報である。

【 0 0 5 6 】

論理ボリューム情報 2 0 0 0 は、論理ボリューム毎に対応して設けられる情報である。論理ボリューム情報 2 0 0 0 は、論理ボリューム識別子 2 0 0 1、論理容量 2 0 0 2、論理ボリューム種別 2 0 0 5、論理ボリューム R A I D 種別 2 0 0 3、実ページポインタ 2 0 0 6、第 1 リード回数 2 0 0 7、第 1 ライト回数 2 0 0 8、第 2 リード回数 2 0 0 9、第 2 ライト回数 2 0 1 0、L U N 移動中フラグ 2 0 1 5、L U N 移動元識別子 2 0 1 6、移動ポインタ 2 0 1 7、移動待ちフラグ 2 0 1 8、閉塞移動中フラグ 2 0 1 9、閉塞移動元 L U N 2 0 2 0、及びキャッシュ管理ポインタ 2 0 2 2 を含む。

【 0 0 5 7 】

論理ボリューム識別子 2 0 0 1 は、論理ボリューム情報 2 0 0 0 に対応する論理ボリューム（論理ボリューム情報 2 0 0 0 の説明において対応論理ボリュームという）の識別子を示す。論理容量 2 0 0 2 は、対応論理ボリュームの論理的な容量である。論理ボリューム種別 2 0 0 5 は、論理ボリュームの種別を表す。論理ボリューム R A I D 種別 2 0 0 3 は、対応論理ボリュームが構成する R A I D の種別（R A I D レベル）、例えば、R A I

10

20

30

40

50

D0、RAID1などを示す。例えば、RAID5のように、N台の容量に対し、1台の容量の冗長データを格納する場合、Nの具体的な数値を示すようにする。ただし、論理ボリュームRAID種別2003としては、任意のRAIDレベルを指定できるわけではなく、少なくとも1つRAIDグループが対応するRAIDレベルである必要がある。

【0058】

本実施形態の特徴の1つは、コントローラVM100が容量仮想化機能を持っていることである。容量仮想化機能とは、論理ボリュームを定義したときに実際の記憶領域を割り当てずに、ライト要求があったときに始めて、ライトされる領域を含むページを割り当てる機能である。論理ボリュームは、ページごとに分割されており、論理ボリュームの空間を仮想ページといい、仮想ページに対して実際に割り当てる領域を実ページという。このため、論理ボリューム情報2000は、論理容量2002を、仮想ページのサイズにわたった数だけの実ページポイントを持ち、仮想ページに対して実ページを割り当てたときに、実ページポイントに、割り当てた実ページのアドレスを設定する。

10

【0059】

例えば、容量仮想化機能において、記憶領域の割り当て単位は、ページと呼ばれる。論理ボリュームは、アプリケーションVM140が読み書きをする論理的な記憶装置である。論理ボリュームの空間は、仮想ページを単位として分割され、実際のストレージグループの記憶領域は、実ページという単位で分割されている。容量仮想化機能においては、論理ボリュームの記憶容量を、実際の記憶媒体の容量よりも大きく見せる。このため、仮想ページの数のほうが、実ページの数より大きいのが、一般的である。容量仮想化機能を実現した場合、コントローラVM100がアプリケーションVM140からのライト要求により書き込みを指示されたアドレスを含む仮想ページに対して実ページが割り当てていないとき、実ページを割り当てる。図5に示すコントローラVM管理情報の仮想ページ容量2600は、仮想ページの容量を示す。本実施形態では、仮想ページ容量2600と実ページの容量とは等しいというわけではない。これは、実ページの容量は、RAIDのレベルにより異なってくる冗長データを含むためである。したがって、実ページの容量は、その実ページが割り当てられたストレージグループのRAIDレベルにより決まる。例えば、RAID1のようにデータを2重に書き込む場合、実ページの容量は、仮想ページ容量2600の2倍になる。また、RAID5のように、N台のストレージ装置の容量に対し、1台分のストレージ装置の容量の冗長データを格納する場合、実ページの容量は、仮想ページ容量2600の $(N+1)/N$ の容量となる。また、RAID0のように、冗長性がない場合、仮想ページ容量2600と等しい容量が実ページの容量となる。なお、本実施形態においては、仮想ページ容量2600は仮想ストレージシステム190の中で共通であるが、コントローラVM100の中で、仮想ページ容量2600に異なったものがあったてもよい。なお、本実施形態では、それぞれのストレージグループは、例えば、RAID5で構成されているものとするが、任意のRAIDレベルで構成されていてもよい。

20

30

【0060】

図9の説明に戻り、第1リード回数2007、第1ライト回数2008、第2リード回数2009、第2ライト回数2010は、対応論理ボリュームのリード処理及びライト処理の実行回数に関する情報である。本実施形態では、対応論理ボリュームに対するリード要求又はライト要求を受け付けると、第1リード回数2007又は第1ライト回数2008に加算する。さらに、所定の間隔ごとに、第1リード回数2007を第2リード回数2009にコピーし、第1ライト回数2008を第2ライト回数2010にコピーし、第1リード回数2007及び第1ライト回数2008を0にする。このように、第2リード回数2009及び第2ライト回数2010には、所定の期間におけるリード回数及びライト回数が格納されることとなり、対応論理ボリュームのIOについての性能特性を把握できる。なお、リードやライトの回数以外に、リードやライトのデータ量などを計測して管理するようにしてもよい。本実施形態では、対応論理ボリュームのリード処理及びライト処理の実行回数に関する情報に基づいて、論理ボリュームをコントローラVM100の間で再配置する。

40

50

【 0 0 6 1 】

LUN移動中フラグ2015は、対応論理ボリュームを移動中であることを示すフラグである。LUN移動元識別子2016は、対応論理ボリュームを移動中の場合、移動元となる論理ボリュームのコントローラVM100、ストレージパス195、及び論理ボリュームの識別子を表す。移動ポインタ2017は、移動処理（コピー処理）が、どのブロック（アドレスでもよい）まで進んだかを示す。移動待ちフラグ2018は、対応論理ボリュームが移動を待っている状態にあることを示す。閉塞移動中フラグ2019は、他のコントローラVM100に障害があり、障害を起こしたコントローラVM100のキャッシュメモリ210及び制御用メモリ220の引継ぎデータをコピーしている状態を表す。

【 0 0 6 2 】

閉塞移動元LUN2020は、移動元となる論理ボリュームの論理ボリューム情報2000に設定される情報を示し、移動元の論理ボリュームを管理するコントローラVM100の識別子及び移動元の論理ボリュームの識別子を示す。

【 0 0 6 3 】

キャッシュ管理ポインタ2022は、対応論理ボリュームをスロット21100の容量で分割したそれぞれの領域に対して、スロット21100が割り当てられている（すなわち、キャッシュメモリ210に格納されている）か否かを表す。キャッシュ管理ポインタ2022は、キャッシュメモリ210のスロット2110が割り当てられている場合、対応するキャッシュ管理情報2750へのポインタであり、キャッシュメモリ210のスロット21100が割り当てられていない場合、ヌル状態である。

【 0 0 6 4 】

次に、キャッシュ管理情報2750について説明する。

【 0 0 6 5 】

図10は、一実施形態に係るキャッシュ管理情報の構成図である。

【 0 0 6 6 】

キャッシュ管理情報2750は、キャッシュメモリ210のスロット21100毎に対応して設けられる情報である。キャッシュ管理情報2750は、次キャッシュ管理情報ポインタ2751、割り当て論理ボリュームアドレス2752、ブロックビットマップ2753、及び更新ビットマップ2754を含む。

【 0 0 6 7 】

次キャッシュ管理情報ポインタ2751は、データを格納していない状態であるスロットにおいて有効な情報であり、データを格納していない状態である次のスロット21100に対応するキャッシュ管理情報2750を示すポインタである。割り当て論理ボリュームアドレス2752は、キャッシュ管理情報2750に対応するスロット21100に格納されているデータが、どの論理ボリュームのどのアドレスから開始する領域のデータであるかを示す。ブロックビットマップ2753は、キャッシュ管理情報2750に対するスロット21100の領域の中で、キャッシュメモリ210に格納されたブロック（ブロックは、読み書きの最小単位）を示すビットマップである。ブロックビットマップ2753は、各ブロックに対応するビットを有し、キャッシュメモリ210に格納されているブロックのビットがONに設定される。更新ビットマップ2754は、サーバ110から書き込み要求を受けて、サーバ110から受け取ったデータをキャッシュメモリ210に格納しているブロックであって、まだ、ストレージ装置160に書き込んでいないブロックを示すビットマップである。本実施形態では、更新ビットマップ2754は、各ブロックに対応するビットを有し、ストレージ装置160に書き込んでいないブロックのビットがONに設定される。

【 0 0 6 8 】

次に、空きキャッシュ管理情報管理キュー2201について説明する。

【 0 0 6 9 】

図11は、一実施形態に係る空きキャッシュ管理情報管理キューを説明する図である。

【 0 0 7 0 】

10

20

30

40

50

空きキャッシュ管理情報管理キュー 2 2 0 1 は、空きキャッシュ管理情報ポインタ 2 6 5 0 によって、データを格納されていないスロット（空きスロット）に対応するキャッシュ管理情報 2 7 5 0 を管理する。

【 0 0 7 1 】

空きキャッシュ管理情報ポインタ 2 6 5 0 は、空きスロットに対応する先頭のキャッシュ管理情報 2 7 5 0 をポイント（指し示す）する。先頭のキャッシュ管理情報 2 7 5 0 は、その中の次キャッシュ管理情報ポインタ 2 7 5 1 により、次の空きスロットに対応するキャッシュ管理情報 2 7 5 0 をポイントする。なお、最後の空きスロットに対応するキャッシュ管理情報 2 7 5 0 の次キャッシュ管理情報ポインタ 2 7 5 1 は、空きキャッシュ管理情報ポインタ 2 6 5 0 をポイントする。この空きキャッシュ管理情報管理キュー 2 2 0 1 によると、空きキャッシュ管理情報ポインタ 2 6 5 0 から空きスロットに対応するキャッシュ管理情報 2 7 5 0 を順次迎えることができ、空きスロットを特定することができる、

10

【 0 0 7 2 】

次に、ストレージボックス情報 2 0 5 0 について説明する。

【 0 0 7 3 】

図 1 2 は、一実施形態に係るストレージボックス情報の構成図である。

【 0 0 7 4 】

ストレージボックス情報 2 0 5 0 は、ストレージボックス 1 3 0 毎に対応して設けられる情報である。ストレージボックス情報 2 0 5 0 は、ストレージボックス識別子 7 0 0 0 、接続情報 7 0 0 1 、ストレージ装置台数 7 0 0 2 、接続ストレージ装置台数 7 0 0 3 、

20

パス数 7 0 0 4 、パス識別子 7 0 0 5 、ストレージグループ数 7 0 0 6 、及びストレージグループ識別子 7 0 0 7 を含む。

【 0 0 7 5 】

ストレージボックス識別子 7 0 0 0 は、ストレージボックス情報 2 0 5 0 に対応するストレージボックス（ストレージボックス情報の説明において対応ストレージボックスという）1 3 0 の識別子である。接続情報 7 0 0 1 は、対応ストレージボックス 1 3 0 が、当該コントローラ V M 1 0 0 に接続されているか否かを示す情報である。ストレージ装置台数 7 0 0 2 は、対応ストレージボックス 1 3 0 に接続可能なストレージ装置 1 6 0 の台数である。接続ストレージ装置台数 7 0 0 3 は、対応ストレージボックス 1 3 0 に実際に接続されているストレージ装置 1 6 0 の台数である。パス数 7 0 0 4 は、対応ストレージボックス 1 3 0 のパスの数である。パス識別子 7 0 0 5 は、対応ストレージボックス 1 3 0 のパス毎の識別子である。ストレージグループ数 7 0 0 6 は、対応ストレージボックス 1 3 0 に含まれるストレージグループの数である。本実施形態では、ストレージグループは、1 つのストレージボックス 1 3 0 の中のストレージ装置 1 6 0 で構成されるものとするが、本発明はこれに限られず、1 つのストレージグループが、異なるストレージボックス 1 3 0 のストレージ装置 1 6 0 で構成されていてもよい。ストレージグループ識別子 7 0 0 7 は、対応ストレージボックス 1 3 0 に含まれるストレージグループの識別子である。

30

【 0 0 7 6 】

次に、ストレージグループ情報 2 3 0 0 について説明する。

【 0 0 7 7 】

図 1 3 は、一実施形態に係るストレージグループ情報の構成図である。

40

【 0 0 7 8 】

ストレージグループ情報 2 3 0 0 は、ストレージグループ毎に対応して設けられる情報である。ストレージグループ情報 2 3 0 0 は、ストレージグループ識別子 2 3 0 1 、ストレージグループ種別 2 3 0 6 、ストレージグループ R A I D 種別 2 3 0 2 、空き実ページ情報ポインタ 2 3 0 3 、未管理実ページ情報ポインタ 2 3 0 4 、ストレージ装置ポインタ 2 3 0 5 、第 1 R 回数 3 2 1 0 、第 2 R 回数 3 2 1 1 、第 1 W 回数 3 2 1 2 、及び第 2 W 回数 3 2 1 3 を含む。

【 0 0 7 9 】

ストレージグループ識別子 2 3 0 1 は、ストレージグループ情報 2 3 0 0 に対応するス

50

ストレージグループ（ストレージグループ情報 2300 の説明において、対応ストレージグループという。）2300 の識別子である。ストレージグループ種別 2306 は、対応ストレージグループ 2300 が、ストレージ装置 160 で構成されるか、サーバ 110 の内部のストレージ装置で構成されるかを示す情報である。ストレージグループ種別 2306 には、ストレージボックス 130 のストレージ装置 160 を含んで構成される場合、そのストレージ装置 160 を含むストレージボックス 130 の識別子も含む。ストレージグループ RAID 種別 2302 は、対応ストレージグループの RAID レベルである。本実施形態における RAID レベルは、図 9 の論理ボリューム RAID 種別 2003 を説明したときに述べたとおりである。

【0080】

空き実ページ情報ポインタ 2303 は、対応ストレージグループ 2300 の中で、当該コントローラ VM 100 が割り当て権限をもっている仮想ページに割り当てられていない状態である空き実ページの中で、当該コントローラ VM 100 が割り当て権限をもっている空き実ページ（使用可能実ページ）の実ページ管理情報の先頭を示すポインタである。未管理実ページ情報ポインタ 2304 は、対応ストレージグループ 2300 の中で、当該コントローラ VM 100 が割り当て権限をもっている仮想ページに割り当てられていない状態である空き実ページの中で、当該コントローラ VM 100 が割り当て権限をもっていない空き実ページ（使用不可実ページ）の実ページ管理情報の先頭を示すポインタである。

【0081】

第 1 R 回数 3210、第 2 R 回数 3211、第 1 W 回数 3212、第 2 W 回数 3212 は、対応ストレージグループでのリード処理及びライト処理の実行回数に関する情報である。本実施形態では、対応ストレージグループに対するリード要求又はライト要求を受け付けると、受け付けた要求に従って第 1 R 回数 3210、又は第 1 W 回数 3212 を加算する。また、所定の間隔ごとに、第 1 R 回数 3210 を第 2 R 回数 3211 にコピーするとともに、第 1 W 回数 3212 を第 2 W 回数 3213 にコピーし、第 1 R 回数 3210 及び第 1 W 回数 3212 を 0 にする。これによって、第 2 R 回数 3211 及び第 2 W 回数 3213 には、所定の間隔におけるリード回数及びライト回数が格納されることとなり、このリード回数及びライト回数により、対応ストレージグループの性能特性を把握できる。なお、IO に関する回数以外に、IO に関するデータ量などを計測してストレージグループ情報 2300 に格納するようにしてもよい。本実施形態では、これらの情報を用いて、論理ボリュームをコントローラ VM 100 の間で再配置する。

【0082】

次に、実ページ情報 2100 について説明する。

【0083】

図 14 は、一実施形態に係る実ページ情報の構成図である。

【0084】

実ページ情報 2100 は、実ページ毎に対応して設けられる情報である。実ページ情報 2100 は、ストレージグループ識別子 2101、実ページアドレス 2102、空きページポインタ 2103、及び未管理実ページポインタ 2104 を含む。

【0085】

ストレージグループ識別子 2101 は、実ページ情報に対応する実ページ（対応実ページという）が、割り当てられているストレージグループの識別子を示す。実ページアドレス 2102 は、対応実ページが、ストレージグループ識別子 2101 に対応するストレージグループの中で、どの相対的なアドレスに割り当てられているかを示す情報である。空きページポインタ 2103 は、対応実ページが、仮想ページに割り当てられていない場合、すなわち、空きページである場合に有効な値となる。この場合、空きページポインタ 2103 は、ストレージグループ識別子 2101 に対応するストレージグループの中で、当該コントローラ VM 100 が割り当て権限をもっており、且つ、仮想ページが割り当てられていない、次の実ページに対応する実ページ情報 2100 をさす。なお、対応実ページが仮想ページに割り当てられている場合、空きページポインタ 2103 は、ヌル値となる

10

20

30

40

50

。本実施形態では、当該コントローラVM100が、割り当て権限をもたない実ページが存在する。未管理実ページポインタ2104は、当該コントローラVM100が割り当て権限をもっておらず、且つ、仮想ページに割り当てられていない、次の実ページに対応する実ページ情報2100をさす。なお、割り当て権限をもっている、次の実ページに対応する実ページ情報2100の場合、未管理実ページポインタ2104は、ヌル値となる。

【0086】

次に、空きページ情報管理キュー2900について説明する。

【0087】

図15は、一実施形態に係る空きページ情報管理キューを説明する図である。

【0088】

空きページ情報管理キュー2900は、空き実ページ情報ポインタ2303によって、空き実ページに対応する実ページ情報2100を管理する。ここで、空きページ情報ポインタ2303は、ストレージグループごとに設けられる情報である。また、空き実ページとは、仮想ページに割り当てられていない実ページを意味する。

【0089】

空き実ページ情報ポインタ2303は、先頭の空き実ページに対応する実ページ情報2100をポイント（指し示す）する。先頭の実ページ情報2100は、その中の空きページポインタ2103により、次の空き実ページに対応する実ページ情報2100をポイントする。なお、最後の空き実ページに対応する実ページ情報2100の空きページポインタ2103は、空き実ページ情報ポインタ2303をポイントする。なお、最後の実ページ情報2100の空きページポインタ2103は、空き実ページ情報ポインタ2303をポイントせずに、ヌル値としてもよい。

【0090】

コントローラVM100は、実ページが割り当てられていない仮想ページに書き込み要求を受け付けると、論理ボリュームRAID種別2003と、実ページポインタ2006とに該当する、ストレージグループのいずれか（例えば、該当するストレージグループの中の空き実ページ数の最も多いストレージグループ）に対応する空き実ページ情報ポインタ2303から、空き実ページを探し、仮想ページに割り当てる。

【0091】

次に、未管理実ページ情報管理キュー2910について説明する。

【0092】

図16は、一実施形態に係る未管理実ページ情報管理キューを説明する図である。

【0093】

未管理実ページ情報管理キュー2910は、未管理実ページ情報ポインタ2304によって、未管理実ページに対応する実ページ情報2100を管理する。ここで、未管理実ページ情報ポインタ2304は、ストレージグループごとに設けられる情報である。本実施形態では、当該コントローラVM100が割り当て権限をもたない実ページが存在する。未管理実ページとは、当該コントローラVM100が割り当て権限をもたない実ページを意味する。

【0094】

未管理実ページ情報ポインタ2304は、先頭の未管理実ページに対応する実ページ情報2100をポイント（指し示す）する。先頭の実ページ情報2100は、その中の未管理実ページポインタ2104により、次の未管理実ページに対応する実ページ情報2100をポイントする。なお、最後の未管理実ページに対応する実ページ情報2100の未管理実ページポインタ2104は、未管理実ページ情報ポインタ2304をポイントする。なお、最後の実ページ情報2100の未管理実ページポインタ2104は、未管理実ページ情報ポインタ2304をポイントせずに、ヌル値としてもよい。

【0095】

コントローラVM100は、ストレージボックス130が追加されると、他のコントローラVM100と連携し、容量などから、実ページ数を計算し、必要な実ページ情報を生

10

20

30

40

50

成し、実ページごとに、どのコントローラVM100が割り当て権限をもつかを決定する。各コントローラVM100は、当該コントローラVM100が、割り当て権限をもつ実ページの実ページ情報2100を、対応するストレージグループの空きページ情報キュー2900に接続し、割り当て権限をもたない実ページの実ページ情報2100を、対応するストレージグループの未管理実ページ情報キュー2910に接続する。

【0096】

次に、ストレージ装置情報2500について説明する。

【0097】

図17は、一実施形態に係るストレージ装置情報の構成図である。

【0098】

ストレージ装置情報2500は、ストレージ装置毎に対応して設けられる情報である。ストレージ装置情報2500は、ストレージ装置識別子2500、接続形態2501、接続パス識別子2502、ストレージタイプ2503、及び容量2504を含む。

【0099】

ストレージ装置識別子2500は、ストレージ装置情報2500に対応するストレージ装置160（対応ストレージ装置という）の識別子である。接続形態2501は、対応ストレージ装置が、ストレージボックス130のストレージ装置160であるか、サーバ110の内部のストレージ装置であるかを示す。接続パス2502は、ストレージボックス130のストレージ装置160の場合には、接続されているパスの識別子を示す。ストレージタイプ2503は、対応ストレージ装置における、HDD、フラッシュメモリ、などの記憶媒体の種類を示す。容量2504は、対応ストレージ装置の容量を示す。なお、本実施形態では、ストレージグループを構成するストレージ装置のストレージタイプ2503と、容量2504とは、等しいものとする。

【0100】

次に、コントローラVM100と、VM管理部150とが、上記した各種管理情報を用いて実行する処理動作について説明する。

【0101】

コントローラVM100と、VM管理部150とは、サーバ110のプロセッサがサーバ110のメモリに格納されたプログラムを実行することにより、各種機能部を構成する。

【0102】

図18は、一実施形態に係る管理プログラムにより構成される機能部の構成図である。

【0103】

サーバ110のメモリは、コントローラVM100の機能を実現するリード処理実行部4000、ライト要求受付部4100、ライトアフタ実行部4200、サーバ追加処理部4300、サーバ削除処理部4400、サーバ移動処理部4500、キャッシュデータ移動スケジューラ部4910、及びキャッシュデータ移動処理部4920を構成するプログラムと、VM管理部150の機能を実現する、移動VM判定・実行部4600、VM受領部4700、及びVM起動部4800を構成するプログラムとを格納している。なお、各機能部の処理動作については、後述する。

【0104】

まず、リード処理実行部4000によるリード処理について説明する。

【0105】

図19は、一実施形態に係るリード処理のフローチャートである。

【0106】

リード処理は、アプリケーションVM140から、コントローラVM100が、リード要求を受け付けたときに実行される。

【0107】

ステップ5000：コントローラVM100のリード処理実行部4000は、受け取ったリード要求で指定された仮想論理ボリューム及びリード要求のアドレス（関連アドレス）を、仮想論理ボリューム情報2084に基づいて、論理ボリューム及び論理ボリューム

10

20

30

40

50

でのリード要求のアドレス（リード要求アドレス、関連アドレス：R A）に変換し、この論理ボリュームに対応する論理ボリューム情報 2 0 0 0 を獲得する。ここで、変換して得られた論理ボリュームを対象論理ボリュームということとする。

【 0 1 0 8 】

ステップ 5 0 0 1：リード処理実行部 4 0 0 0 は、受け取ったリード要求のアドレスから、キャッシュ管理ポインタ 2 0 2 2、キャッシュ管理情報 2 7 5 0 のブロックビットマップ 2 7 5 3 などに基づいて、リード要求で指定されたデータがキャッシュメモリ 2 1 0 でヒットするか否かをチェックする。この結果、キャッシュミスであれば（ステップ 5 0 0 1：N）、リード処理実行部 4 0 0 0 は、処理をステップ 5 0 0 2 へ進める。一方、キャッシュヒットであれば（ステップ 5 0 0 1：Y）、リード処理実行部 4 0 0 0 は、処理をステップ 5 0 1 5 へ進める。

10

【 0 1 0 9 】

ステップ 5 0 0 2：リード処理実行部 4 0 0 0 は、空きキャッシュ管理情報ポインタ 2 6 5 0 が示す先頭のキャッシュ管理情報 2 7 5 0 を、対象論理ボリュームに対応する論理ボリューム情報 2 0 0 0 の関連アドレスに対応するキャッシュ管理ポインタ 2 0 2 2 にセットすることで、スロット 2 1 1 0 0 を割り当てる。さらに、リード処理実行部 4 0 0 0 は、キャッシュ管理情報 2 7 5 0 の割り当て論理ボリュームアドレス 2 7 5 2 に、対象論理ボリュームの識別子と、アドレスとを設定する。

【 0 1 1 0 】

ステップ 5 0 0 3：リード処理実行部 4 0 0 0 は、対象論理ボリュームに対応する論理ボリューム情報 2 0 0 0 の L U N 移動中フラグ 2 0 1 5 がオンであるか否かをチェックする。L U N 移動中フラグ 2 0 1 5 がオフであれば（ステップ 5 0 0 3：N）、対象論理ボリュームのマイグレーションが実行されていないことを示しているので、リード処理実行部 4 0 0 0 は、処理をステップ 5 0 0 7 へ進める。一方、L U N 移動中フラグ 2 0 1 5 がオンであれば、リード処理実行部 4 0 0 0 は、さらに、リード要求で指定されていたアドレスと、移動ポインタ 2 0 1 7（すなわち、対象論理ボリュームの先頭からコピーを開始し、現在コピーが完了しているアドレスを示す）とを比較して、リード要求アドレスが移動ポインタよりも大きければ（ステップ 5 0 0 3：Y）、リード要求アドレスはマイグレーションが実行されていないアドレスであることを示しているので、処理をステップ 5 0 0 4 へ進め、リード要求アドレスが移動ポインタ 2 0 1 7 よりも大きくなければ（ステップ 5 0 0 3：N）、リード要求で指定されたアドレスは、マイグレーションが既に実行されているアドレスへのアクセスであるので、処理をステップ 5 0 0 7 へ進める。

20

30

【 0 1 1 1 】

ステップ 5 0 0 4：リード処理実行部 4 0 0 0 は、移動元 L U N 識別子 2 0 1 6 に格納されたコントローラ V M 1 0 0 に対して、移動元 L U N 識別子 2 0 1 6 に格納されたストレージパス、論理ボリュームの識別子とリード要求で指定されたアドレスとを指定して、リード要求を発行する。なお、リード要求の発行先のコントローラ V M 1 0 0 が、自コントローラ V M 1 0 0 の場合もある。この場合、リード処理実行部 4 0 0 0 は、論理ボリュームの識別子から、対応する論理ボリューム情報 2 0 0 0 を認識し、論理ボリューム情報 2 0 0 0 において、対応するアドレスの実ページポインタ 2 0 0 6 を参照して、対応するストレージグループを認識し、R A I D レベルなどから、要求されたデータを含むストレージ装置 1 6 0 とそのアドレスを認識して、該当するストレージ装置 1 6 0 に対して、リード要求を発行する。

40

【 0 1 1 2 】

ステップ 5 0 0 5：リード処理実行部 4 0 0 0 は、リード要求が完了するのを待つ。

【 0 1 1 3 】

ステップ 5 0 0 6：リード処理実行部 4 0 0 0 は、リード要求の応答としてデータを受け取り、受け取ったデータをキャッシュメモリ 2 1 0 に格納する。さらに、リード処理実行部 4 0 0 0 は、受け取ったデータに対応するブロックビットマップ 2 7 5 3 のビットをセットし、この後、処理をステップ 5 0 1 5 に進める。

50

【 0 1 1 4 】

ステップ 5 0 0 7 : リード処理実行部 4 0 0 0 は、閉塞移動中フラグ 2 0 1 9 をチェックする。閉塞移動中フラグ 2 0 1 9 がオフであれば (ステップ 5 0 0 7 : N)、リード処理実行部 4 0 0 0 は、処理をステップ 5 0 1 2 へ進める。一方、閉塞移動中フラグ 2 0 1 9 がオンであれば、他のコントローラ VM に障害が発生して、当該コントローラ VM がキャッシュや制御用メモリをコピーしている状態であることを示しているため、リード処理実行部 4 0 0 0 は、さらに、リード要求アドレスが移動ポインタ 2 0 1 7 より大きい場合 (ステップ 5 0 0 7 : Y)、処理をステップ 5 0 0 8 へ進め、リード要求アドレスが移動ポインタ 2 0 1 7 より大きくない場合 (ステップ 5 0 0 7 : N)、処理をステップ 5 0 1 2 へ進める。

10

【 0 1 1 5 】

ステップ 5 0 0 8 : リード処理実行部 4 0 0 0 は、閉塞移動元 L U N 2 0 2 0 に格納されたコントローラ VM 1 0 0 に、閉塞移動元 L U N 2 0 2 0 に格納されたストレージパス及び論理ボリューム識別子と、ライト要求で指定されたアドレスとを指定して、キャッシュメモリ 2 1 0 のスロットからデータを読み出す要求 (キャッシュリード要求) を発行する。

【 0 1 1 6 】

ステップ 5 0 0 9 : リード処理実行部 4 0 0 0 は、キャッシュリード要求が完了するのを待つ。キャッシュリード要求の応答において、要求したスロットにデータがないという旨を受け取った場合、リード処理実行部 4 0 0 0 は、処理をステップ 5 0 1 2 へ進める一方、キャッシュリード要求の対象のデータを受け取った場合には、処理をステップ 5 0 1 0 へ進める。

20

【 0 1 1 7 】

ステップ 5 0 1 0 : リード処理実行部 4 0 0 0 は、受け取ったデータをキャッシュメモリ 2 1 0 に格納する。さらに、リード処理実行部 4 0 0 0 は、受け取ったデータに対応するブロックビットマップ 2 7 5 3 のビットをセットし、この後、処理をステップ 5 0 1 1 へ進める。

【 0 1 1 8 】

ステップ 5 0 1 1 : リード処理実行部 4 0 0 0 は、セットしたブロックビットマップ 2 7 5 3 を参照し、要求されたデータが、キャッシュメモリ 2 1 0 に格納されているか否かをチェックする。この結果、要求されたデータがキャッシュメモリ 2 1 0 に存在すれば (ステップ 5 0 1 1 : Y)、リード処理実行部 4 0 0 0 は処理をステップ 5 0 1 5 へ進める。一方、要求されたデータがキャッシュメモリ 2 1 0 に存在しなければ (ステップ 5 0 1 1 : N)、リード処理実行部 4 0 0 0 は処理をステップ 5 0 1 2 へ進める。

30

【 0 1 1 9 】

ステップ 5 0 1 2 : リード処理実行部 4 0 0 0 は、論理ボリュームの識別子から、対応する論理ボリューム情報 2 0 0 0 を認識し、論理ボリューム情報 2 0 0 0 において、対応するアドレスの実ページポインタ 2 0 0 6 を参照して、対応するストレージグループを認識し、RAID レベルなどから、要求されたデータを含むストレージ装置 1 6 0 とそのアドレスを認識して、該当するストレージ装置 1 6 0 に対して、リード要求を発行する。

40

【 0 1 2 0 】

ステップ 5 0 1 3 : リード処理実行部 4 0 0 0 は、リード要求が完了するのを待つ。

【 0 1 2 1 】

ステップ 5 0 1 4 : リード処理実行部 4 0 0 0 は、リード要求の応答としてデータを受け取り、受け取ったデータをキャッシュメモリ 2 1 0 に格納する。さらに、リード処理実行部 4 0 0 0 は、受け取ったデータに対応するブロックビットマップ 2 7 5 3 のビットをセットし、この後、処理をステップ 5 0 1 5 へ進める。

【 0 1 2 2 】

ステップ 5 0 1 5 : リード処理実行部 4 0 0 0 は、要求されたデータを、アプリケーション VM 1 4 0 に転送する。次いで、リード処理実行部 4 0 0 0 は、対象論理ボリューム

50

の第1リード回数2007、対象論理ボリュームに対応するストレージグループのストレージグループ情報2300の第1R回数3210を1増やして、処理を完了する。

【0123】

次に、ライト要求受付部4100によるライト受付処理について説明する。

【0124】

図20は、一実施形態に係るライト受付処理のフローチャートである。ライト受付処理は、コントローラVM100が、アプリケーションVM140からライト要求を受け付けたときに実行される。

【0125】

ステップ6000：コントローラVM100のライト受付処理部4100は、受け取ったライト要求で指定された仮想論理ボリューム及びライト要求のアドレス（関連アドレス）を、仮想論理ボリューム情報2084に基づいて、論理ボリューム及び論理ボリュームでのライト要求のアドレス（関連アドレス：RA）に変換し、この論理ボリュームに対応する論理ボリューム情報2000を獲得する。ここで、変換して得られた論理ボリュームを対象論理ボリュームということとする。

10

【0126】

ステップ6001：ライト受付処理部4100は、受け取ったライト要求のアドレスから、キャッシュ管理ポインタ2022、キャッシュ管理情報2750のブロックビットマップ2753などに基づいて、ライト要求で指定されたデータがキャッシュメモリ210でヒットするか否かをチェックする。この結果、キャッシュミスであれば（ステップ6001：N）、ライト受付実行部4100は、処理をステップ6002へ進める。一方、キャッシュヒットであれば（ステップ6001：Y）、ライト受付処理部4100は、処理をステップ6010へ進める。

20

【0127】

ステップ6002：ライト受付実行部4100は、対象論理ボリュームに対応する論理ボリューム情報2000のLUN移動中フラグ2015がオンであるか否かをチェックする。LUN移動中フラグ2015がオフであれば（ステップ6002：N）、ライト受付実行部4100は、処理をステップ6004へ進める。一方、LUN移動中フラグ2015がオンであれば、ライト受付実行部4100は、さらに、ライト要求アドレスと移動ポインタ2017とを比較して、ライト要求アドレスが移動ポインタ2017よりも大きければ（ステップ6002：Y）、処理をステップ6003へ進め、ライト要求アドレスが移動ポインタ2017よりも大きくなければ（ステップ6002：N）、処理をステップ6004へ進める。

30

【0128】

ステップ6003：ライト受付実行部4100は、移動元の論理ボリュームが当該コントローラVM100の論理ボリュームであれば、論理ボリュームに対応する論理ボリューム情報2000の実ページポインタ2006を参照し、実ページが割当てられていなければ、適当なストレージグループを選択して、ストレージグループに対応するストレージグループ情報2300の空き実ページ情報ポインタ2303から、空き実ページの実ページ情報2100を特定し、特定した実ページ情報2100の空き実ページを割り当てる。その後、ライト受付実行部4100は、処理をステップS6005へ進める。

40

【0129】

ステップ6004：ライト受付実行部4100は、対応論理ボリュームが当該コントローラVM100の論理ボリュームであれば、論理ボリュームに対応する論理ボリューム情報2000の実ページポインタ2006を参照し、実ページが割当てられていなければ、適当なストレージグループを選択して、ストレージグループに対応するストレージグループ情報2300の空き実ページ情報ポインタ2303から、空き実ページの実ページ情報2100を特定し、特定した実ページ情報2100の空き実ページを割り当てる。

【0130】

ステップ6005：ライト受付実行部4100は、空きキャッシュ管理情報ポインタ2

50

650の先頭のキャッシュ管理情報2750を、対象論理ボリュームに対応する論理ボリューム情報2000のキャッシュ管理ポイント2022にセットすることで、対象論理ボリュームにスロット21100を割り当てる。さらに、ライト受付実行部4100は、スロット管理情報2750の割り当て論理ボリュームアドレス2752に対応論理ボリュームの識別子と、ライト対象のアドレスを設定する。

【0131】

ステップ6006：ライト受付処理部4100は、閉塞移動中フラグ2019をチェックする。閉塞移動中フラグ2019がオフであれば（ステップ6006：N）、ライト受付処理部4100は、処理をステップ6010へ進める。一方、閉塞移動中フラグ2019がオンであれば、ライト受付処理部4100は、さらに、ライト要求アドレスが移動ポイント2017よりも大きい場合（ステップ6006：Y）、処理をステップ6007へ進め、ライト要求アドレスが移動ポイント2017よりも大きくない場合（ステップ6006：N）、処理をステップ6010へ進める。

10

【0132】

ステップ6007：ライト受付処理部4100は、閉塞移動元LUN2020に格納されたコントローラVM100に、閉塞移動元LUN2020に格納されたストレージパス及び論理ボリューム識別子と、ライト要求で指定されたアドレスとを指定して、キャッシュメモリ210のスロットからデータを読み出す要求（キャッシュリード要求）を発行する。

【0133】

ステップ6008：ライト受付処理部4100は、キャッシュリード要求が完了するのを待つ。キャッシュリード要求の応答において、要求したスロットにデータがないという旨を受け取った場合、ライト受付処理部4100は、処理をステップ6010へ進める一方、キャッシュリード要求の対象のデータを受け取った場合には、処理をステップ6009に進める。

20

【0134】

ステップ6009：ライト受付処理部4100は、受け取ったデータに対応するブロックビットマップ2753及び更新ビットマップ2754のビットを設定し、受け取ったデータをキャッシュメモリ210に格納する。この後、処理をステップ6010に進める。

【0135】

ステップ6010：ライト受付処理部4100は、アプリケーションVM140からのライトデータを受け取り、キャッシュ210に格納する。この後、ライト受付処理部4100は、受け取ったデータに対応するキャッシュ管理情報2750のブロックビットマップ2753及び更新ビットマップ2754のビットを設定する。

30

【0136】

ステップ6011：ライト受付処理部4100は、対応論理ボリュームの論理ボリューム情報2000のLUN移動中フラグ2015が、オンか否かをチェックする。LUN移動中フラグ2015がオフであれば、ライト受付処理部4100は、処理をステップ6015へ進める。一方、LUN移動中フラグ2015がオンであれば、ライト受付処理部4100は、さらに、ライト要求アドレスと移動ポイント2017とを比較して、ライト要求アドレスが移動ポイント2017よりも大きければ（ステップ6011：Y）、処理をステップ6012へ進め、ライト要求アドレスが移動ポイント2017よりも大きくなければ（ステップ6011：N）、処理をステップ6015へ進める。

40

【0137】

ステップ6012：ライト受付処理部4100は、移動元LUN2016に格納されたコントローラVM100が、当該コントローラVM100か否かをチェックする。移動元LUN2016に格納されたコントローラVM100が当該コントローラVM100でなければ、ライト受付処理部4100は、移動元LUN2016に格納されたコントローラVM100、ストレージパス、及び論理ボリューム識別子と、ライト要求で指定されたアドレスとを指定して、ライト要求で受け取ったデータを書き込むようライト要求を発行し

50

、キャッシュ 210 内に格納したデータを送信する。

【0138】

ステップ 6013 : ライト受付処理部 4100 は、ライト要求が完了するのを待つ。

ステップ 6014 : ライト受付処理部 4100 は、ライト要求が完了すると、ライトを行ったデータに対応するキャッシュ管理情報 2750 を空きキャッシュ管理情報ポインタ 2650 が示すようにする。

【0139】

ステップ 6015 : ライト受付処理部 4100 は、アプリケーション VM 140 にライト要求の完了報告を行い、対象論理ボリュームの論理ボリューム情報 2000 の第 1 ライト回数 2008 と、対象論理ボリュームに対応するストレージグループのストレージグループ情報 2300 の第 1 W 回数 3212 を 1 増やして、処理を完了する。

10

【0140】

次に、ライトアフタ実行部 4200 によるライトアフタ処理について説明する。

【0141】

図 21 は、一実施形態に係るライトアフタ処理のフローチャートである。ライトアフタ処理は、任意のタイミングで適宜実行される。

【0142】

ステップ 7000 : ライトアフタ実行部 4200 は、キャッシュ管理情報 2750 をサーチして、更新ビットマップ 2754 がオンになっているキャッシュ管理情報 2750 を見つける。

20

【0143】

ステップ 7001 : ライトアフタ実行部 4200 は、見つけたキャッシュ管理情報 2750 の割り当て論理ボリュームアドレス 2752 を確認して、キャッシュ管理情報 2750 に対応するスロットに対応する論理ボリュームを認識する。ここで、認識した論理ボリュームをライトアフタ処理において、対象論理ボリュームということとする。

【0144】

ステップ 7002 : ライトアフタ実行部 4200 は、対象論理ボリュームに対応する論理ボリューム情報 2000 の LUN 移動中フラグ 2015 がオンか否かをチェックする。LUN 移動中フラグ 2015 がオフであれば (ステップ 7002 : N)、ライトアフタ実行部 4200 は、処理をステップ 7009 へ進める。一方、オンの場合には、ライトアフタ実行部 4200 は、移動元論理ボリューム 2016 の中のコントローラ VM 100 が、当該コントローラ VM 100 であり、且つ、キャッシュ管理情報 2750 に対応する論理ボリュームのアドレスと、移動ポインタ 2017 と、を比較して、論理ボリュームのアドレスが移動ポインタ 2017 より大きければ (ステップ 7002 : Y)、処理をステップ 7003 へ進め、論理ボリュームのアドレスが移動ポインタ 2017 よりも大きくなれば (ステップ 7002 : N)、処理をステップ 7003 へ進める。

30

【0145】

ステップ 7003 : ライトアフタ実行部 4200 は、LUN 移動元識別子 2016 に格納されている論理ボリュームの識別子から、この論理ボリュームに対応する論理ボリューム情報 2000 を認識する。さらに、ライトアフタ実行部 4200 は、実ページポインタ 2006 を認識し、書き込みを行うアドレスを認識する。さらに、ライトアフタ実行部 4200 は、論理ボリューム RAID 種別 2003 などから、スロットのデータに対応する冗長データ (パリティデータ) の生成に必要な情報を読み込むため、ストレージ装置 160 とそのアドレスを認識して、認識したストレージ装置 160 にリード要求を発行する。

40

【0146】

ステップ 7004 : ライトアフタ実行部 4200 は、リード要求による読み出しが完了するのを待つ。

【0147】

ステップ 7005 : ライトアフタ実行部 4200 は、リード要求により読み出したデータを用いて冗長データを生成する。

50

【 0 1 4 8 】

ステップ 7 0 0 6 : ライトアフタ実行部 4 2 0 0 は、更新ビットマップ 2 7 5 4 の更新されたことを示すビットに対応するブロックのデータと、生成した冗長データとについて、格納すべきストレージ装置 1 6 0 とそのアドレスとを認識して、認識したストレージ装置 1 6 0 に対してライト要求を発行する。

【 0 1 4 9 】

ステップ 7 0 0 7 : ライトアフタ実行部 4 2 0 0 は、ライト要求の書き込みが完了するのを待つ。

【 0 1 5 0 】

ステップ 7 0 0 8 : ライトアフタ実行部 4 2 0 0 は、書き込みを行ったブロックに対応する更新ビットマップ 2 7 5 4 のビットをオフに設定し、この後、処理を終了する。

10

【 0 1 5 1 】

ステップ 7 0 0 9 : ライトアフタ実行部 4 2 0 0 は、対象論理ボリュームの識別子から、この論理ボリュームに対応する論理ボリューム情報 2 0 0 0 を認識する。さらに、ライトアフタ実行部 4 2 0 0 は、実ページポインタ 2 0 0 6 を認識し、書き込みを行うアドレスを認識する。さらに、ライトアフタ実行部 4 2 0 0 は、論理ボリューム R A I D 種別 2 0 0 3 などから、スロットのデータに対応する冗長データ(パリティデータ)の生成に必要な情報を読み込むため、ストレージ装置 1 6 0 とそのアドレスを認識して、認識したストレージ装置 1 6 0 にリード要求を発行する。

【 0 1 5 2 】

ステップ 7 0 1 0 : ライトアフタ実行部 4 2 0 0 は、リード要求による読み出しが完了するのを待つ。

20

【 0 1 5 3 】

ステップ 7 0 1 1 : ライトアフタ実行部 4 2 0 0 は、リード要求により読み出したデータを用いて冗長データを生成する。

【 0 1 5 4 】

ステップ 7 0 1 2 : ライトアフタ実行部 4 2 0 0 は、更新ビットマップ 2 7 5 4 の更新されたことを示すビットに対応するブロックのデータと、生成した冗長データとについて、格納すべきストレージ装置 1 6 0 とそのアドレスとを認識して、認識したストレージ装置 1 6 0 に対してライト要求を発行する。

30

【 0 1 5 5 】

ステップ 7 0 1 3 : ライトアフタ実行部 4 2 0 0 は、ライト要求の書き込みが完了するのを待つ。

【 0 1 5 6 】

ステップ 7 0 1 4 : ライトアフタ実行部 4 2 0 0 は、書き込みを行ったブロックに対応する更新ビットマップ 2 7 5 4 のビットをオフに設定し、この後、処理を終了する。

【 0 1 5 7 】

次に、新たなサーバ 1 1 0 が追加される際に実行されるサーバ追加処理について説明する。ここで、この処理の説明において、追加されるサーバ 1 1 0 を追加サーバ(第 2 のサーバ)ということとする。なお、統合移動元のサーバが有するアプリケーション VM を所定の移動先のサーバに移動させる所定の状況が発生した場合や、統合型ストレージシステムのサーバの構成を変更する状況が発生した場合の一例として、新たなサーバ 1 1 0 を追加する状況が該当するが、他の例についても後述する。

40

【 0 1 5 8 】

図 2 2 A は、一実施形態に係るサーバ追加処理の追加サーバ側処理のフローチャートである。図 2 2 A のサーバ追加処理は、追加サーバのコントローラ VM 1 0 0 で実行される。

【 0 1 5 9 】

ステップ 8 0 0 0 : コントローラ VM 1 0 0 のサーバ追加処理部 4 3 0 0 は、仮想ストレージシステム 1 9 0 を構成する各コントローラ VM 1 0 0 に、新しいサーバ 1 1 0 が追加されたこと、及び当該コントローラ VM 1 0 0 がアクセス可能なストレージボックス 1

50

30を通報する。

【0160】

ステップ8001：サーバ追加処理部4300は、通報に対する回答が返ってくるのをまつ。通報には、このサーバ110のコントローラVM100に対して制御権を渡す仮想論理ボリュームの集合等が含まれる。

【0161】

ステップ8002：サーバ追加処理部4300は、制御権を譲りうけた仮想論理ボリュームの集合を、仮想論理ボリューム情報2084、論理ボリューム情報2000に設定する。特に、サーバ追加処理部4300は、譲り受けた論理ボリュームに割り当てていた実ページの集合に対し、実ページ情報2100を生成し、必要な情報の設定を行って、論理ボリューム情報2000の実ページポインタ2006が、対応する実ページ情報2100を示すようにする。さらに、サーバ追加処理部4300は、割り当て権限が与えられた実ページに関して、実ページ情報2100を生成して、対応するストレージグループの空きページ情報管理キュー2900に登録する。

10

【0162】

ステップ8003：制御権を移動させる論理ボリュームについては、移動元のコントローラVM100のキャッシュメモリ210に残っているライトデータを、ストレージ装置160に書き込む必要がある。ここでは、サーバ追加処理部4300は、移動元の論理ボリュームについて、移動元のコントローラVM100のキャッシュメモリ210にストレージ装置160に書き込むライトデータがあれば、移動先のコントローラVM100のキャッシュメモリ210に移動する処理を開始させる。このため、サーバ追加処理部4300は、キャッシュ移動スケジュール処理部4900をコールする。ただし、本実施形態では、移動元のコントローラVMのサーバ追加処理部4300が、キャッシュメモリ210に残っているライトデータを、直接ストレージ装置160に書き込んでよい。これにより、論理ボリュームの制御権を譲り受けたコントローラVM100は、この論理ボリュームのキャッシュメモリ210に残っているライトデータをキャッシュメモリ210で管理することができ、I/O要求に対して迅速に対応することができることとなる。

20

【0163】

ステップ8004：次に、サーバ追加処理部4300は、VM管理部150のVM起動部4800と通信を行い、仮想論理ボリュームの移動が完了したことを通知し、VM起動部4800からの応答を待つ。なお、応答がかえってきたら、サーバ追加処理部4300は、処理を終了する。

30

【0164】

図22Bは、一実施形態に係るサーバ追加処理の他サーバ側処理のフローチャートである。図22Bのサーバ追加処理は、追加サーバからの要求を受けた他のサーバ110のコントローラVM100で実行される。

【0165】

ステップ8005：他のサーバ110のコントローラVM100のサーバ追加処理部4300は、追加サーバのコントローラVM100がアクセス権限をもっていて、かつ、自コントローラVM100がアクセス権限をもっているストレージボックス130を検出する。ここで、以下の処理は、検出したストレージボックス130を対象に実行される。なお、条件を満たすストレージボックス130が1つもない場合には、追加サーバのコントローラVM100に対して、その旨を通知する。

40

【0166】

ステップ8006：サーバ追加処理部4300は、条件を満たすストレージボックス130が存在する場合、まず、そのストレージボックス130上に、定義されている仮想論理ボリュームをすべて把握し、これらの仮想論理ボリュームを制御権の移動候補とする。

【0167】

ステップ8007：サーバ追加処理部4300は、移動候補とした仮想論理ボリュームと、その仮想論理ボリュームに対応する論理ボリュームの第1リード回数2007、第1

50

ライト回数2008、第2リード回数2009、第2ライト回数2010を、VM管理部150の移動VM判定・実行部4600に通知し、移動する仮想論理ボリュームが決定されるのを待つ。

【0168】

ステップ8008：サーバ追加処理部4300は、VM管理部150から制御権を移動する仮想論理ボリュームの集合の通知を受け取ると、通知された仮想論理ボリューム対応する論理ボリュームの集合、及び実ページポイントを参照して、仮想論理ボリュームに対して割り当てている実ページの実ページ情報2100から実ページアドレス2102を認識し、対応する実ページ情報を消去する。さらに、サーバ追加処理部4300は、そのストレージボックス130に属するストレージグループ190の中で、割り当て権限を移譲する空き実ページを決める。決定した実ページに対応する実ページ情報2100から、実ページのアドレスを認識し、さらに、実ページ情報2100を空きページ情報管理キュー2900から外し、実ページ情報2100を消去する。

10

【0169】

ステップ8009：サーバ追加処理部4300は、追加サーバのコントローラVM100に、制御権を渡す仮想論理ボリューム及び論理ボリュームの集合と、この集合に対応する論理ボリュームに割り当てている実ページのアドレスの情報、割り当て権限を渡す実ページのアドレスの集合を通知し、処理を終了する。

【0170】

次に、サーバ110を削除する際に実行されるサーバ削除処理について説明する。ここで、この処理の説明において、削除されるサーバ110を削除サーバ(第1のサーバ)ということとする。

20

【0171】

図23Aは、一実施形態に係るサーバ削除処理の削除サーバ側処理のフローチャートである。図23Aのサーバ削除処理は、削除サーバのコントローラVM100で実行される。

【0172】

ステップ9000：削除を行うサーバ110上のコントローラVM100のサーバ削除処理部4400は、仮想ストレージシステム190を構成する他のコントローラVM100から、それぞれのコントローラVM100に接続されたストレージボックス130の集合の情報をネットワーク120経由で取得する。

30

【0173】

ステップ9001：サーバ削除処理部4400は、すべての論理ボリューム情報2000を参照して、それぞれの論理ボリュームがどのストレージボックス130に定義されているかを認識する。さらに、サーバ削除処理部4400は、各ストレージボックス130のストレージグループごとに、空き実ページを認識する。

【0174】

ステップ9002：サーバ削除処理部4400は、移動VM判定・実行部4600に、各コントローラVM100に接続されているストレージボックス130の情報、すべての論理ボリュームがどのストレージボックス130に定義されているかについての情報、その論理ボリュームに対応する仮想論理ボリュームの集合、第1リード回数2007、第1ライト回数2008、第2リード回数2009、及び第2ライト回数2010を通知し、それぞれの論理ボリュームについての制御権の移動先のコントローラVM100の指示が帰ってくるのを待つ。

40

【0175】

ステップ9003：サーバ削除処理部4400は、割り当て権限をもっている実ページのそれぞれについて、どのコントローラVM100に移動するかを決定する。サーバ削除処理部4400は、論理ボリュームに割当てられている実ページのアドレスの把握と、対応する実ページ情報2100の消去、割り当て権限を移す実ページのアドレスの把握と、対応する実ページ情報2100の消去とを行う。

【0176】

50

ステップ9004：サーバ削除処理部4400は、制御権を移すと決定した仮想論理ボリュームと、制御権を移す論理ボリュームの集合と、対応する論理ボリュームに割当てた実ページのアドレス情報と、割り当て権限を移す実ページのアドレスの集合とを、移動先のコントローラVM100に通知する。

【0177】

ステップ9005：サーバ削除処理部4400は、通知したコントローラVM100からの回答が返ってくるのを待ち、回答が返ってきた場合には処理を完了する。

【0178】

次に、サーバ削除処理における削除サーバのコントローラVM100から要求を受けた他のサーバ110側の処理について説明する。

【0179】

図23Bは、一実施形態に係るサーバ削除処理の他サーバ側処理のフローチャートである。図23Bのサーバ削除処理は、他のサーバ110のコントローラVM100で実行される。

【0180】

ステップ9006：コントローラVM100のサーバ削除処理部4400は、削除されるコントローラVM100から受け取った仮想論理ボリュームの情報、論理ボリュームの情報を、仮想論理ボリューム情報2084、論理ボリューム情報2000に設定する。ただし、論理ボリュームの識別子は、このコントローラVM100が決定して、仮想論理ボリューム情報2084の論理ボリューム識別子2089と、論理ボリューム情報2000の論理ボリューム識別子2001に設定する。また、サーバ削除処理部4400は、論理ボリュームに割当てた実ページごとに、実ページ情報2100を生成して、実ページアドレスなどの必要な情報を設定する。また、サーバ削除処理部4400は、対応する実ページポインタ2006に、実ページ情報2100のアドレスを設定する。また、サーバ削除処理部4400は、このコントローラVM100に割り当て権限を移すと決定した実ページに対応する実ページ情報2100を生成して、必要な情報を設定し、生成した実ページ情報2100を、空きページ情報管理キュー2900に登録する。これにより、他のサーバ110に対して、実ページの割り当て権限を適切に移すことができる。

【0181】

ステップ9007：サーバ110を削除して、コントローラVM100を閉塞しようとする場合、削除サーバのコントローラVM100のキャッシュメモリ210に残っているライトデータを、他のコントローラVM100に移動する必要がある。特に、キャッシュメモリ210が二重化されていて、一方のキャッシュメモリ210が閉塞された場合など、ライトデータを他のコントローラVM100に移し、二重化した状態にし、障害を起こしたコントローラVM100を閉塞したほうが、高信頼化を図るうえで有効である。したがって、コントローラVM100を閉塞しようとする場合、キャッシュメモリ210に残っているライトデータを、ストレージ装置160に書き込む必要がある。ここでは、閉塞するコントローラVM100の論理ボリュームに、ストレージ装置160に書き込むライトデータがあれば、移動先のコントローラVM100のキャッシュメモリ210に移動する処理を実行する。このため、サーバ削除処理部4400は、キャッシュデータ移動スケジューラ部4900をコールする。ただし、移動元のコントローラVM100が、キャッシュメモリ210に残っているライトデータを、ストレージ装置160に直接書き込むようにしてもよい。

【0182】

ステップ9008：サーバ削除処理部4400は、この後、閉塞するコントローラVM100に処理の完了を報告する。

ステップ9009：サーバ削除処理部4400は、論理ボリュームの制御権の移動が完了したことを、VM起動部4800に連絡する。

ステップ9010：サーバ削除処理部4400は、VM起動部4800からの完了報告を待って、完了後に処理を終了する。

10

20

30

40

50

【 0 1 8 3 】

次に、移動元のサーバ 1 1 0 から移動先のサーバ 1 1 0 へアプリケーション VM を移動する際に実行されるサーバ移動処理について説明する。このようなケースは、例えば、移動元のサーバ（第 1 のサーバ）を移動先のサーバ（第 2 のサーバ）に置き換える場合等に該当する。

【 0 1 8 4 】

図 2 4 A は、一実施形態に係るサーバ移動処理の移動元サーバ側処理のフローチャートである。図 2 4 A のサーバ移動処理は、移動元のサーバ 1 1 0 のコントローラ VM 1 0 0 のサーバ移動処理部 4 5 0 0 に実行される。

【 0 1 8 5 】

サーバ移動処理においては、移動元のサーバ 1 1 0 のコントローラ VM 1 0 0 のすべての論理ボリュームの制御権が、移動先のサーバ 1 1 0 のコントローラ VM 1 0 0 に移される。また、移動元のサーバ 1 1 0 のコントローラ VM 1 0 0 が割り当て権限をもつすべての実ページが移動先のサーバ 1 1 0 のコントローラ VM 1 0 0 に移される。また、移動元のサーバ 1 1 0 と移動先のサーバ 1 1 0 では、同じストレージボックス 1 3 0 にアクセス可能であるものとする。

【 0 1 8 6 】

ステップ 1 0 0 0 0 : サーバ移動処理部 4 5 0 0 は、移動 VM 判定・実行部 4 6 0 0 に連絡し、アプリケーション VM 1 4 0 を移動先のサーバ 1 1 0 に移動するように要求する。

【 0 1 8 7 】

ステップ 1 0 0 0 1 : サーバ移動処理部 4 5 0 0 は、要求の完了を待つ。
ステップ 1 0 0 0 2 : サーバ移動処理部 4 5 0 0 は、論理ボリューム情報 2 1 0 0 を参照して、ストレージボックス 1 3 0 に定義された論理ボリュームを認識する。さらに、サーバ移動処理部 4 5 0 0 は、各ストレージボックス 1 3 0 のストレージグループごとに、空き実ページを認識する。

【 0 1 8 8 】

ステップ 1 0 0 0 3 : サーバ移動処理部 4 5 0 0 は、すべての論理ボリュームの制御権、割り当て権限をもっている実ページのそれぞれについて、移動先のコントローラ VM 1 0 0 に移動するため、論理ボリュームに割り当てられている実ページのアドレスの把握と、実ページに対応する実ページ情報 2 1 0 0 の消去と、割り当て権限を移す実ページのアドレスの把握と、実ページに対応する実ページ情報 2 1 0 0 の消去とを行う。

【 0 1 8 9 】

ステップ 1 0 0 0 4 : サーバ移動処理部 4 5 0 0 は、すべての仮想論理ボリューム、論理ボリュームの情報、割り当て権限を移す実ページ、及び、空き実ページの情報を実移動先のコントローラ VM 1 0 0 に通知する。

【 0 1 9 0 】

ステップ 1 0 0 0 5 : サーバ移動処理部 4 5 0 0 は、回答が返ってくるのを待って、処理を完了する。

【 0 1 9 1 】

次は、サーバ移動処理における移動元のコントローラ VM 1 0 0 から要求を受けた、移動先となるサーバ 1 1 0 のコントローラ VM 1 0 0 の処理について説明する。

【 0 1 9 2 】

図 2 4 B は、一実施形態に係るサーバ移動処理の移動先サーバ側処理のフローチャートである。図 2 4 B のサーバ移動処理は、移動先のサーバ 1 1 0 のコントローラ VM 1 0 0 のサーバ移動処理部 4 5 0 0 により実行される。

【 0 1 9 3 】

ステップ 1 0 0 0 6 : サーバ移動処理部 4 5 0 0 は、移動元のコントローラ VM 1 0 0 から受け取った仮想論理ボリュームの情報及び論理ボリュームの情報を、仮想論理ボリューム情報 2 0 8 4、論理ボリューム情報 2 0 0 0 に設定する。ただし、論理ボリュームの識別子は、このコントローラ VM 1 0 0 が決定して、仮想論理ボリューム情報 2 0 8 4 の

10

20

30

40

50

論理ボリューム識別子 2089 と論理ボリューム情報 2000 の論理ボリューム識別子 2001 に設定する。また、サーバ移動処理部 4500 は、論理ボリュームに割当てた実ページごとに、実ページ情報 2100 を生成して、実ページアドレスなどの必要な情報を設定する。また、サーバ移動処理部 4500 は、対応する実ページポイント 2006 に、実ページ情報 2100 のアドレスを設定する。また、サーバ移動処理部 4500 は、このコントローラ VM 100 に割り当て権限を移すと決定した実ページに対応する実ページ情報 2100 を生成して、必要な情報を設定し、生成した実ページ情報 2100 を、空きページ情報管理キュー 2900 に登録する。

【0194】

ステップ 10007 : 移動を行う場合、キャッシュメモリ 210 に残っているライトデータを、ストレージ装置 160 に書き込む必要がある。ここでは、移動元の論理ボリュームについて、移動元のコントローラ VM 100 のキャッシュメモリ 210 にストレージ装置 160 に書き込むライトデータがあれば、移動先のコントローラ VM 100 のキャッシュメモリ 210 に移動する処理を行う。このため、サーバ移動処理部 4500 は、キャッシュデータ移動スケジュール部 4910 をコールする。

10

【0195】

ステップ 10008 : サーバ移動処理部 4500 は、移動元のコントローラ VM 100 に処理が完了したことを報告する。

ステップ 10009 : サーバ移動処理部 4500 は、VM 起動部 4800 に論理ボリュームの制御権の移動が完了したことを報告する。

20

ステップ 10010 : サーバ移動処理部 4500 は、VM 起動部 4800 からの完了報告をまって、処理を終了する。

【0196】

次に、キャッシュデータ移動管理処理について説明する。

【0197】

図 25 は、一実施形態に係るキャッシュデータ移動管理処理のフローチャートである。

【0198】

キャッシュデータ移動管理処理は、閉塞するコントローラ VM 100 から論理ボリュームを引き継ぐコントローラ VM 100 や、論理ボリュームの移動先のコントローラ VM 100 において、キャッシュデータ移動スケジュール部 4910 により実行される処理である。

30

【0199】

ステップ 15000 : キャッシュデータ移動スケジュール部 4910 は、引き継ぐ論理ボリューム及び移動先となる論理ボリュームの論理ボリューム情報 2000 の閉塞移動中フラグ 2019 をオンにして、閉塞移動元 LUN 2020 に移動元の論理ボリューム、移動元のコントローラ VM 100 の識別子を設定する。

【0200】

ステップ 15001 : キャッシュデータ移動スケジュール部 4910 は、論理ボリューム単位にコピーを行うキャッシュデータ移動処理部 4920 を起動する。

【0201】

ステップ 15002 : キャッシュデータ移動スケジュール部 4910 は、どれか一つの論理ボリュームの完了報告を待つ。

40

ステップ 15003 : キャッシュデータ移動スケジュール部 4910 は、引き継ぐ論理ボリューム及び移動先となる論理ボリュームについて未処理のものがあるか否かをチェックする。未処理のものがあれば (ステップ 15003 : Y)、キャッシュデータ移動スケジュール部 4910 は、処理をステップ 15002 へ進める。一方、未処理のものが無ければ (ステップ 15003 : N)、キャッシュデータ移動スケジュール部 4910 は、処理を完了する。

【0202】

次に、キャッシュデータ移動処理について説明する。

50

【0203】

図26Aは、一実施形態に係るキャッシュデータ移動処理の移動先サーバ側処理のフローチャートである。図26Aのキャッシュデータ移動処理は、論理ボリュームを引き継ぐコントローラVM100や、移動先のコントローラVM100で実行される処理である。

【0204】

ステップ17000：コントローラVM100のキャッシュデータ移動処理部4920は、移動対象の論理ボリューム情報2000の閉塞移動元LUN2020に含まれる、コントローラVM100及び論理ボリュームの識別子と、移動ポインタ2017のアドレスとを指定して、そのアドレス以降で、キャッシュメモリ210に残っているライトデータを転送するように、移動元のコントローラVM100に要求する。

10

【0205】

ステップ17001：キャッシュデータ移動処理部4920は、要求したデータを受信したか否かを判定する。この結果、データを受信しなかった場合（ステップ17001：N）には、キャッシュデータ移動処理部4920は、処理を終了する。データを受信した場合（ステップ17001：Y）には、キャッシュデータ移動処理部4920は、処理をステップ17002に進める。

【0206】

ステップ17002：キャッシュデータ移動処理部4920は、受信したデータをキャッシュメモリ210に格納し、コピーポインタ2017に、送られてきたデータの次のアドレスを設定して、処理をステップ17003に進める。

20

【0207】

ステップ17003：キャッシュデータ移動処理部4920は、論理ボリュームのキャッシュデータを移動する処理は終了したか否かを判定する。キャッシュデータを移動する処理が終了していない場合（ステップ17003：N）には、キャッシュデータ移動処理部4920は、処理をステップ17000に進める。一方、処理が終了した場合（ステップ17003：Y）には、すべてのデータを受け取ったこととなるので、キャッシュ移動スケジュール部4910に完了報告を行って、その後、処理を完了する。

【0208】

次に、キャッシュデータ移動処理における移動元のコントローラVM100の処理について説明する。

30

【0209】

図26Bは、一実施形態に係るキャッシュデータ移動処理の移動元サーバ側処理のフローチャートである。

【0210】

ステップ17004：移動元のコントローラVM100のキャッシュデータ移動処理部4920は、移動先のコントローラVM100から受け取った、論理ボリュームのアドレスから、ストレージ装置160に書き込むべきライトデータがキャッシュメモリ210に残っているかを探す。見つかった場合、キャッシュデータ移動処理部4920は、このライトデータと論理ボリューム上のアドレスとを、移動元のコントローラVM100に転送し、処理を終了する。

40

【0211】

次に、移動VM判定実行処理について説明する。

【0212】

図27は、一実施形態に係る移動VM判定実行処理のフローチャートである。

【0213】

移動VM判定実行処理においては、移動VM判定・実行部4600は、コントローラVM100から受け取った情報と、アプリケーションVM140が持つアプリケーションVM管理情報の第1プロセッサ使用量250、第2プロセッサ使用量260等に基づいて、他のサーバ110に移動するアプリケーションVM140を決定する。

【0214】

50

ステップ18000：移動VM判定・実行部4600は、コントローラVM100から受け取った指示が、すべてのアプリケーションVM140の移動の指示であるか否かをチェックする。この結果、すべてのアプリケーションVM140の移動の指示である場合（ステップ18000：Y）には、移動VM判定・実行部4600は処理をステップ18001に進める。一方、すべてのアプリケーションVM140の移動の指示でない場合（ステップ18000：N）には、移動VM判定・実行部4600は処理をステップ18002へ進める。

【0215】

ステップ18001：移動VM判定・実行部4600は、移動対象をすべてのアプリケーションVMとし、処理をステップ18003へ進める。

10

【0216】

ステップ18002：移動VM判定・実行部4600は、コントローラVM100から受け取った指示に含まれる移動候補となる論理ボリュームに関する情報と、アプリケーションVM140が持つアプリケーションVM管理情報である第1プロセッサ使用量250、第2プロセッサ使用量260、及び仮想論理ボリューム番号270に基づいて、移動するアプリケーションVM140と移動先のサーバ110とを決める。さらに、移動VM判定・実行部4600は、そのアプリケーションVM140が使用する仮想論理ボリューム番号270に対応する論理ボリュームを移動対象とし、移動先として決めたサーバ110のコントローラVM100を移動先として決定する。これにより、アプリケーションVM140と、そのアプリケーションVM140により使用される論理ボリュームの制御権が移動先のサーバ110への移動対象とされることとなる。この結果、移動先のサーバ110において、アプリケーションVM140は、使用する論理ボリュームに対して迅速にI/O処理を行うことができることが担保される。

20

【0217】

ステップ18003：移動VM判定・実行部4600は、コントローラVM100に処理の完了を報告する。なお、この際、一部の論理ボリュームの制御権を移す場合、移動VM判定・実行部4600は、制御権を移すと決めた仮想論理ボリューム、論理ボリューム、及び移動先のコントローラVM100を、コントローラVM100に通知する。

【0218】

ステップ18004：移動VM判定・実行部4600は、移動すると決めたアプリケーションVM140を停止させる。

30

【0219】

ステップ18005：移動VM判定・実行部4600は、移動先となるサーバ110のVM受領部4700に、移動先のサーバ110でアプリケーションVM140を実行するために必要な情報を送る。この後、移動VM判定・実行部4600は、処理を完了する。

【0220】

次に、移動VM受領処理について説明する。

【0221】

図28は、一実施形態に係る移動VM受領処理のフローチャートである。

【0222】

移動VM受領処理は、VM受領部4700により実行される処理である。

40

【0223】

ステップ19000：VM受領部4700は、移動VM判定・実行部4600から、自サーバ110上でアプリケーションVM140を起動するために、必要な情報を受け取る。

ステップ19001：VM起動部4800に、アプリケーションVM140の起動準備できたことを通知する。この後、VM起動部4800は、処理を終了する。

【0224】

次に、アプリケーションVM起動制御処理について説明する。

【0225】

図29Aは、一実施形態に係る第1のアプリケーションVM起動制御処理のフローチャ

50

ートである。第1のアプリケーションVM起動制御処理は、コントローラVM100から論理ボリュームの制御権の移動が完了したことの通知(ステップ10009)の受領時に実行される処理を示している。

【0226】

ステップ20000: VM起動部4800は、アプリケーションVM140が起動可能となっているか否かをチェックする。具体的には、VM起動部4800は、後述するステップ20005によって、アプリケーションVM140が起動可能となっていることが記憶されているか否かをチェックする。この結果、アプリケーションVM140が起動可能となっている場合(ステップ20000: Y)には、VM起動部4800は、処理をステップ20001に進める。一方、アプリケーションVM140が起動可能となっていない

10

【0227】

ステップ20001: 既にアプリケーションVM140が再開可能となっているので、VM起動部4800は、アプリケーションVM140を再開させ、処理を完了させる。

【0228】

ステップ20002: VM起動部4800は、論理ボリュームの制御権の移動が完了したことを記憶し、処理を終了する。

【0229】

次に、第2のアプリケーションVM起動制御処理について説明する。

20

【0230】

図29Bは、一実施形態に係る第2のアプリケーションVM起動制御処理のフローチャートである。第2のアプリケーションVM起動制御処理は、VM受領部4700からのアプリケーションVM140が起動可能となっていることの通知(ステップ19001)の受領時に実行される処理を示している。

【0231】

ステップ20003: VM起動部4800は、論理ボリュームの制御権の移動が終了したか否かをチェックする。具体的には、VM起動部4800は、ステップ20002によって、論理ボリュームの制御権の移動が終了したことが記憶されているか否かをチェックする。この結果、論理ボリュームの制御権の移動が終了した場合(ステップ20003: Y)には、VM起動部4800は、処理をステップ20004に進める。一方、論理ボリュームの制御権の移動が終了していない場合(ステップ20003: N)には、VM起動部4800は、処理をステップ20005に進める。

30

【0232】

ステップ20004: 既に論理ボリュームの制御権の移動が終了しているため、VM起動部4800は、アプリケーションVM140を再開させ、処理を完了させる。

ステップ20005: VM起動部4800は、アプリケーションVM140が起動可能となっていることを記憶し、処理を終了する。

【0233】

なお、本発明は、上述の実施形態に限定されるものではなく、本発明の趣旨を逸脱しない範囲で、適宜変形して実施することが可能である。

40

【0234】

例えば、上記実施形態において、プロセッサが行っていた処理の一部又は全部を、ハードウェア回路で行うようにしてもよい。また、上記実施形態におけるプログラムは、プログラムソースからインストールされてよい。プログラムソースは、プログラム配布サーバ又は記憶メディア(例えば可搬型の記憶メディア)であってもよい。

【符号の説明】

【0235】

10...複合型ストレージシステム、100...コントローラVM、110...サーバ、120...ネットワーク、130...ストレージボックス、140...アプリケーションVM、15

50

0 ... VM管理部、160 ... ストレージ装置、210 ... キャッシュメモリ、220 ... 制御メモリ、230 ... プロセッサ

【図面】

【図1】

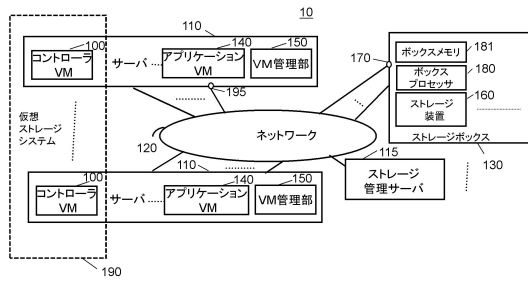


Fig. 1

【図2】

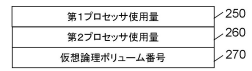


Fig. 2

10

【図3】

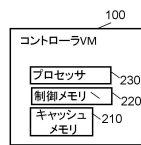


Fig. 3

【図4】

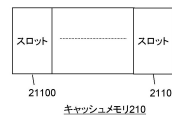


Fig. 4

20

30

40

50

【 図 5 】

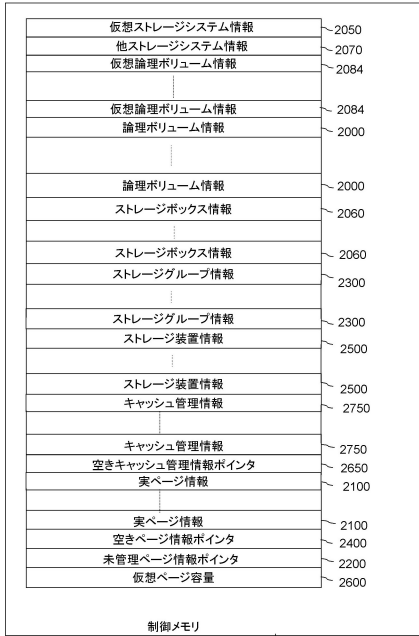


Fig. 5 220

【 図 6 】

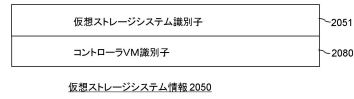


Fig.6

10

20

【 図 7 】

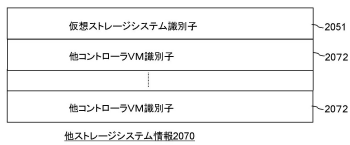
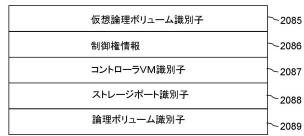


Fig.7

【 図 8 】



仮想論理ボリューム情報 2084

Fig.8

30

40

50

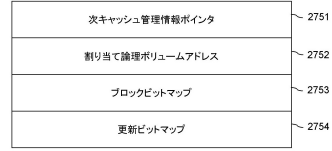
【 図 9 】



論理ボリューム情報 2000

Fig. 9

【 図 1 0 】

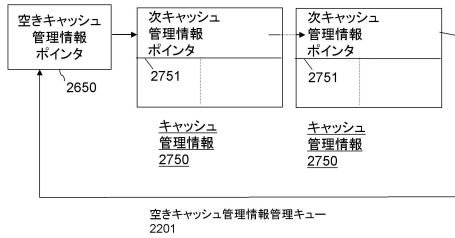


キャッシュ管理情報 2750

Fig.10

10

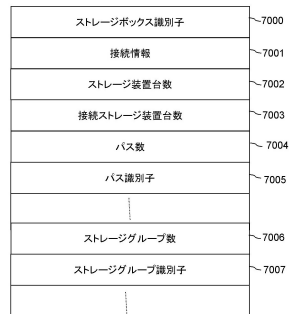
【 図 1 1 】



空きキャッシュ管理情報管理キュー 2201

Fig. 11

【 図 1 2 】



ストレージボックス情報 2050

Fig.12

20

30

40

50

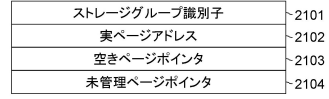
【 図 1 3 】



ストレージグループ情報 2300

Fig. 13

【 図 1 4 】

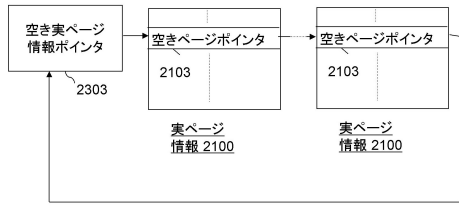


実ページ情報 2100

Fig. 14

10

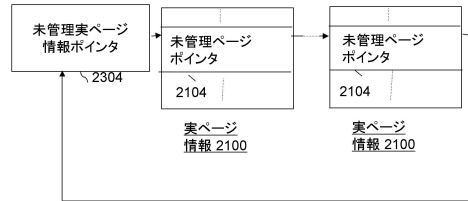
【 図 1 5 】



空きページ情報管理キュー 2900

Fig. 15

【 図 1 6 】



未管理実ページ情報管理キュー 2910

Fig. 16

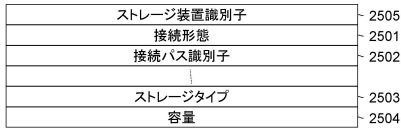
20

30

40

50

【 図 1 7 】



ストレージ装置情報 2500

Fig. 17

【 図 1 8 】

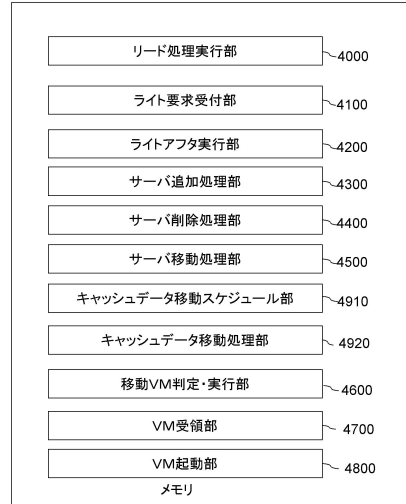


Fig. 18

10

【 図 1 9 】

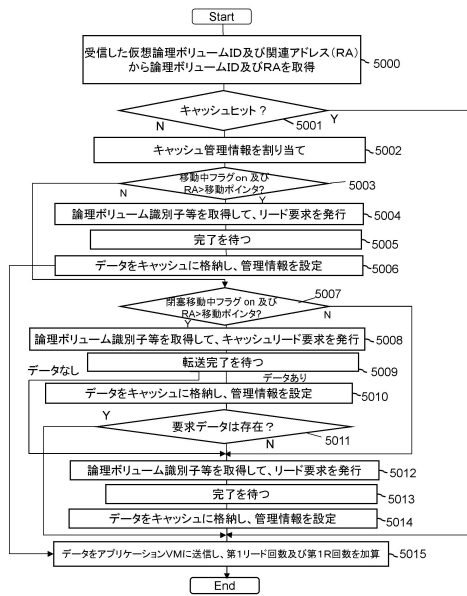


Fig. 19

【 図 2 0 】

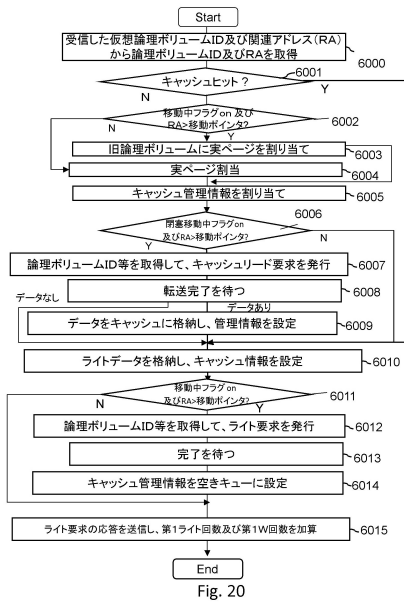


Fig. 20

20

30

40

50

【 図 2 1 】

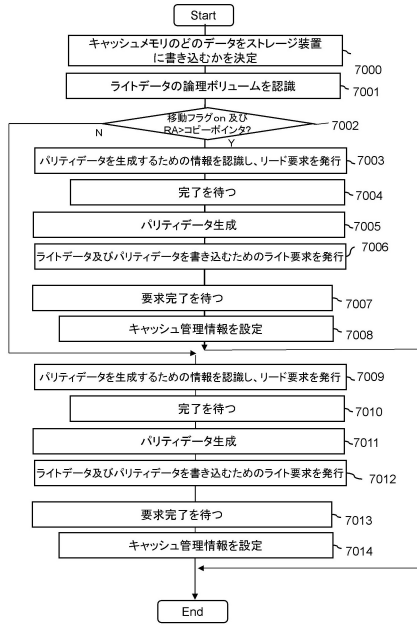


Fig. 21

【 図 2 2 A 】

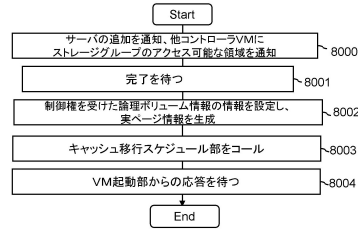


Fig. 22A

10

20

【 図 2 2 B 】

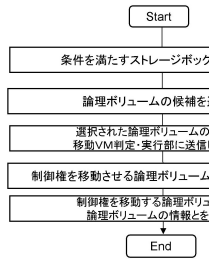


Fig. 22B

【 図 2 3 A 】

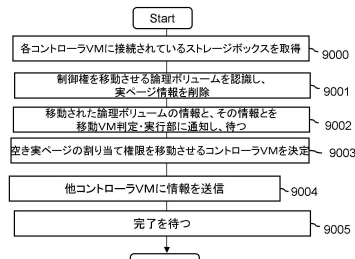


Fig. 23A

30

40

50

【図 2 3 B】

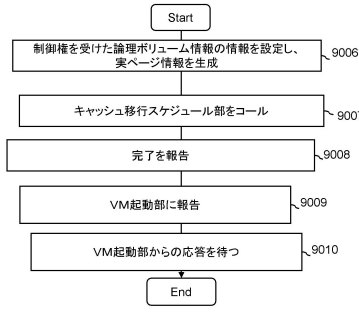


Fig. 23B

【図 2 4 A】

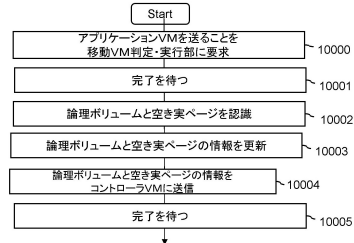


Fig. 24A

【図 2 4 B】

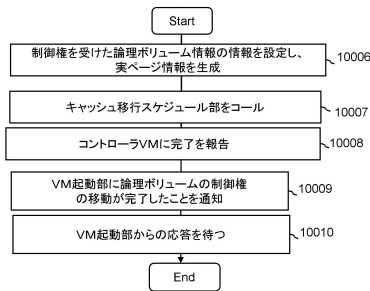


Fig. 24B

【図 2 5】

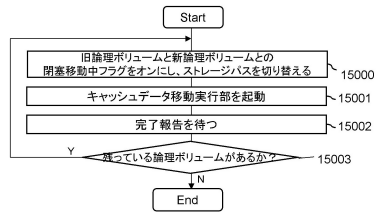


Fig. 25

【図 2 6 A】

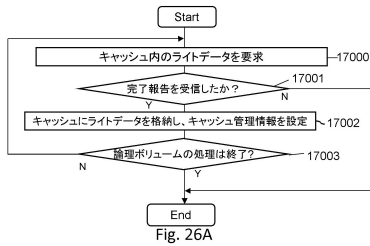


Fig. 26A

【図 2 6 B】



Fig. 26B

10

20

30

40

50

【 図 27 】

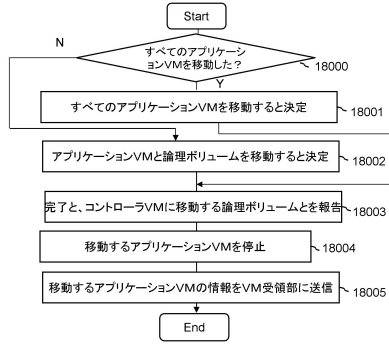


Fig. 27

【 図 28 】

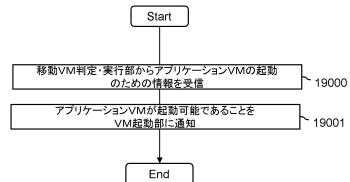


Fig. 28

10

【 図 29 A 】

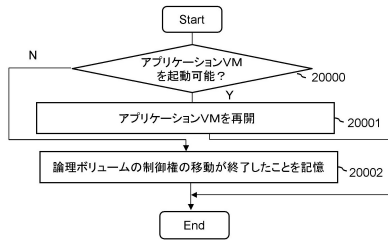


Fig. 29A

【 図 29 B 】

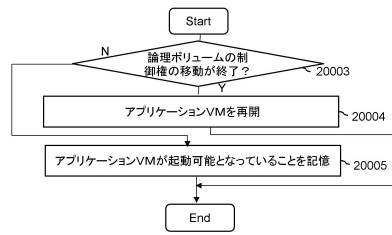


Fig. 29B

20

30

40

50

フロントページの続き

(72)発明者 大平 良徳

東京都千代田区丸の内一丁目6番6号 株式会社日立製作所内

審査官 局 成矢

(56)参考文献 国際公開第17/145223(WO, A1)

国際公開第18/029820(WO, A1)

特開2020-154990(JP, A)

(58)調査した分野 (Int.Cl., DB名)

G06F 13/14

G06F 13/10

G06F 9/455

G06F 16/188