

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号
特許第5567147号
(P5567147)

(45) 発行日 平成26年8月6日 (2014. 8. 6)

(24) 登録日 平成26年6月27日 (2014. 6. 27)

(51) Int. Cl.

F I

G O 6 F 3 / 0 6 (2 0 0 6 . 0 1)

G O 6 F 1 3 / 1 0 (2 0 0 6 . 0 1)

G O 6 F 3 / 0 6 3 O 1 A

G O 6 F 3 / 0 6 3 O 1 M

G O 6 F 3 / 0 6 5 4 O

G O 6 F 1 3 / 1 0 3 4 O B

請求項の数 12 (全 108 頁)

(21) 出願番号	特願2012-547196 (P2012-547196)	(73) 特許権者	000005108
(86) (22) 出願日	平成22年7月16日 (2010. 7. 16)		株式会社日立製作所
(65) 公表番号	特表2013-527942 (P2013-527942A)		東京都千代田区丸の内一丁目6番6号
(43) 公表日	平成25年7月4日 (2013. 7. 4)	(74) 代理人	110000279
(86) 国際出願番号	PCT/JP2010/004621		特許業務法人ウィルフォート国際特許事務所
(87) 国際公開番号	W02012/007999	(72) 発明者	松井 佑光
(87) 国際公開日	平成24年1月19日 (2012. 1. 19)		神奈川県小田原市中里322番2号 株式会社日立製作所 ITプラットフォーム事業本部内
審査請求日	平成24年10月11日 (2012. 10. 11)	(72) 発明者	川野 浩史
			東京都千代田区丸の内一丁目6番1号 株式会社日立製作所 知的財産権本部内

最終頁に続く

(54) 【発明の名称】 記憶制御装置、又は複数の当該記憶制御装置を有する記憶システム

(57) 【特許請求の範囲】

【請求項 1】

複数の記憶制御装置ユニットを有し、
各記憶制御装置ユニットは、1以上のホストに接続された1以上の記憶制御装置で構成され、それら1以上の記憶制御装置が、それら1以上のホストに提供される複数の論理ボリュームを含んだ複数のボリュームを有し、
記憶制御装置ユニット同士が、下記（a1）乃至（a3）を満たす接続形態である第3の接続形態に従って、互いに接続されている、
（a1）1つの記憶制御装置ユニット内の1つの記憶制御装置ともう1つの記憶制御装置ユニット内の1つの記憶制御装置が、記憶制御装置の内部バスと同じ種類の接続媒体である第1種の接続媒体よりも最大通信距離が長く記憶制御装置の内部バスと異なる種類の接続媒体である第2種の接続媒体を介して接続されている、
（a2）前記1つの記憶制御装置ユニットが、それに接続されている1以上のホストに、前記もう1つの記憶制御装置ユニットが有する論理ボリュームを仮想化して提供する、
（a3）前記もう1つの記憶制御装置ユニットが、それに接続されている1以上のホストに、前記1つの記憶制御装置ユニットが有する論理ボリュームを仮想化して提供する、
記憶システムであって、
前記複数の記憶制御ユニットのそれぞれが、自身に接続されている1以上のホストに提供する論理ボリュームを、論理ボリュームのI/O性能に影響し得る1種類以上のファクターに基づいて定義されたティアに分類して管理するティア管理情報、を有し、

10

20

前記 1 種類以上のファクターは、接続形態種別というファクターを含み、
接続形態種別として、第 3 接続と、非第 3 接続があり、
第 3 接続に対応した論理ボリュームは、その論理ボリュームを提供する記憶制御装置ユニットに前記第 3 の接続形態で接続されている記憶制御装置ユニット内の論理ボリュームがマッピングされた仮想的な論理ボリュームであり、
非第 3 接続に対応した論理ボリュームは、前記第 3 接続に対応した論理ボリューム以外の論理ボリュームである、
記憶システム。

【請求項 2】

請求項 1 記載の記憶システムであって、
前記 1 種類以上のファクターが、経由する記憶制御装置ユニットの段数、接続距離、またはコントローラ性能のいずれかを含み、
コントローラ性能は、論理ボリュームの制御を担当する前記記憶制御装置のコントローラの性能または機能に基づき分類される値である、
記憶システム。

【請求項 3】

請求項 1 記載の記憶システムであって、
接続形態として、前記第 3 の接続形態の他に、第 1 及び／又は第 2 の接続形態があり、
前記第 1 の接続形態は、下記 (b 1) を満たす接続形態であり、
(b 1) 記憶制御装置ユニットにおいて、記憶制御装置同士が、前記第 1 種の接続媒体を介して互いに接続されている、
前記第 2 の接続形態は、下記 (c 1) 乃至 (c 3) を満たす接続形態であり、
(c 1) 内部記憶制御装置ユニット内の 1 つの記憶制御装置と、外部記憶制御装置ユニット内の 1 つの記憶制御装置が前記第 2 種の接続媒体を介して互いに接続されている、
(c 2) 前記内部記憶制御装置ユニットが、それに接続されている 1 以上のホストに、前記外部記憶制御装置ユニットが有する論理ボリュームを仮想化して提供する、
(c 3) 前記外部記憶制御装置ユニットが、前記内部記憶制御装置ユニットが有する論理ボリュームを仮想化しない、
前記内部記憶制御装置ユニットは、前記複数の記憶制御装置ユニット内の 1 つの記憶制御装置ユニットであり、
前記外部記憶制御装置ユニットは、前記複数の記憶制御装置ユニット外の 1 つの 1 つの記憶制御装置ユニットであり、
前記非第 3 接続として、自ユニット又は第 2 接続があり、
自ユニットに対応した論理ボリュームは、その論理ボリュームを提供する記憶制御装置ユニット内の論理ボリュームであって、その記憶制御装置ユニット以外の記憶制御装置ユニットが有する論理ボリュームがマッピングされた仮想的な論理ボリュームではない論理ボリュームであり、
第 2 接続に対応した論理ボリュームは、その論理ボリュームを提供する記憶制御装置ユニットに前記第 2 の接続形態で接続されている記憶制御装置ユニット内の論理ボリュームがマッピングされた仮想的な論理ボリュームである、
記憶システム。

【請求項 4】

請求項 3 記載の記憶システムであって、
前記複数の内部記憶制御装置ユニットが、第 3 の接続形態で直列に接続されており、
内部記憶制御装置ユニットが複数の記憶制御装置で構成されている場合、それら複数の記憶制御装置が、第 1 の接続形態で直列に接続されている、
記憶システム。

【請求項 5】

請求項 1 記載のシステムであって、
各記憶制御装置ユニットが、複数のポートを有し、

10

20

30

40

50

前記複数のポートは、論理ボリュームが割り当てられているポートである使用ポートと、論理ボリュームが割り当てられていないポートである空きポートとを含み、

第1の記憶制御装置ユニットから第2の記憶制御装置ユニットへのポート単位の負荷分散が行われ、

前記第1の記憶制御装置ユニットは、前記複数の記憶制御装置ユニットにおける或る記憶制御装置ユニットであり、

前記第2の記憶制御装置ユニットは、前記第1の記憶制御装置ユニットに1以上の前記第2種の接続媒体を介して第3の接続形態で接続されている記憶制御装置ユニットであり、

前記ポート単位の負荷分散では、以下の(F)及び(G)が行われ、

(F) 前記第1の記憶制御装置ユニットが有する複数のポートから使用ポートが選択される、

(G) 前記(F)で選択された使用ポートである選択ポートに割り当てられている論理ボリュームが、前記選択ポートから前記第2の記憶制御装置ユニットの空きポートに移動する、

前記(G)では、以下の(G1)乃至(G5)が行われる、

(G1) 前記選択ポートに割り当てられている論理ボリュームである移動元ボリュームが属するティアが特定される、

(G2) 前記第2の記憶制御装置ユニットが有する複数の論理ボリュームから、前記(G)で特定したティアと同じティアに属する論理ボリュームである移動先ボリュームが選択される、

(G3) 前記移動元ボリュームから前記移動先ボリュームにデータがコピーされる、

(G4) 前記空きポートに前記移動先ボリュームが割り当てられる、

(G5) 前記選択ポートの識別情報が前記空きポートに割り当てられ、且つ、前記移動元ボリュームの識別情報が前記移動先ボリュームに割り当てられる、
記憶システム。

【請求項6】

請求項5記載の記憶システムであって、

前記第1の記憶制御装置ユニットが、複数の第1のプロセッサを有し、

前記第1の記憶制御装置ユニットが有する1以上の使用ポートに割り当てられている各論理ボリュームに、その論理ボリュームを担当する第1のプロセッサが割り当てられており、

前記選択ポートは、負荷が第1の負荷よりも高い第1のプロセッサである高負荷の第1のプロセッサが担当する論理ボリュームが割り当てられているポートである、
記憶システム。

【請求項7】

請求項6記載の記憶システムであって、

前記複数の第1のプロセッサに前記高負荷の第1のプロセッサが含まれている場合、

(X) 前記複数の第1のプロセッサに、負荷が第2の負荷よりも低い第2のプロセッサである低負荷の第1のプロセッサが含まれていれば、前記高負荷の第1のプロセッサが担当する論理ボリュームの担当が、前記高負荷の第1のプロセッサから前記低負荷の第1のプロセッサに変わる、

(Y) 前記複数の第1のプロセッサに前記低負荷の第1のプロセッサが含まれていなければ、前記ポート単位の負荷分散が行われる、
記憶システム。

【請求項8】

請求項5記載の記憶システムであって、

前記選択ポートは、そのポートに割り当てられている全ての論理ボリュームについての総負荷である対象負荷を前記第1の記憶制御装置ユニットの負荷から減算することにより得られる第1の負荷と、前記第2の記憶制御装置ユニットの負荷に前記対象負荷を加算す

10

20

30

40

50

ることにより得られる第2の負荷とが平均化されるようなポートである、
記憶システム。

【請求項9】

請求項5記載の記憶システムであって、

前記第2の記憶制御装置ユニットが、複数の第2のプロセッサを有し、

前記第2の記憶制御装置ユニットが有する1以上の使用ポートに割り当てられている各論理ボリュームに、その論理ボリュームを担当する第2のプロセッサが割り当てられており、

前記複数の第2のプロセッサの負荷が平均化されるように、前記移動先ボリュームを担当する第2のプロセッサが決定される、

記憶システム。

【請求項10】

請求項1記載の記憶システムであって、

前記複数の記憶制御装置ユニットのうちの2以上の記憶制御装置ユニットである2以上の対象ユニットが、それぞれ、シンプロビジョニングが適用された論理ボリュームであるシンプロビジョニングボリュームを有し、

各対象ユニットが有する1以上のボリュームとそれが属するティアとを基に、前記2以上の対象ユニットに共有されるプールが構成され、

前記プールは、前記2以上の対象ユニットにそれぞれ関連付けられる2以上のプール領域に分割され、

各プール領域が、複数のページを含み、

前記各対象ユニットは、その対象ユニットが有するシンプロビジョニングボリュームに、その対象ユニットに関連付けられているプール領域から、ページを割り当てる、
記憶システム。

【請求項11】

請求項10記載の記憶システムであって、

各プール領域が、各第1のボリュームの一部の記憶領域と各第2のボリュームの一部の記憶領域とで構成され、2以上のページを含み、

前記プールを構成する全てのボリュームが同じティアに属し、

前記ファクターとして、距離又はコマンド応答時間というファクターがあり、

論理ボリュームについて、距離というファクターは、その論理ボリュームを有する記憶制御装置ユニットと、その論理ボリュームにマッピングされている論理ボリュームを有する記憶制御装置ユニットとを接続する1以上の前記第2種の接続媒体の長さに従うファクターであり、

論理ボリュームについて、コマンド応答時間というファクターは、その論理ボリュームに対するI/OのためのI/Oコマンドを記憶制御装置ユニットが出してからそのI/Oコマンドに対する応答をその記憶制御装置ユニットが受けるまでの時間に従うファクターである、

記憶システム。

【請求項12】

請求項10記載の記憶システムであって、

(J) プール領域毎に、シンプロビジョニングボリュームに割り当てられていないがシンプロビジョニングボリュームに割り当てられ得るページである未割当ページの数に特定され、

(K) 前記(J)で特定された未割当ページ数を基に、1つの対象ユニットに関連付けられているプール領域である対象プール領域について、移動すべき未割当ページの数である移動ページ数が決定され、

(L) 前記対象プール領域と別のプール領域との間で、前記移動ページ数分の未割当ページが移動する、

記憶システム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、記憶制御装置、又は複数の当該記憶制御装置を有する記憶システムに関する。

【背景技術】

【0002】

例えば、特許文献1には、低性能ディスクアレイ装置を高性能ディスクアレイ装置に接続し、高性能ディスクアレイ装置が低性能ディスクアレイ装置内の論理ボリュームを仮想化してホスト装置に提供する技術が記載されている。

10

【0003】

また、特許文献2には、相互接続される複数のRAID (Redundant Array of Independent (or Inexpensive) Disks) コントローラや、複数のRAIDコントローラ間でのデータ転送技術について記載されている。

【先行技術文献】

【特許文献】

【0004】

【特許文献1】 米国特許第7441095号明細書

【特許文献2】 米国特許第6658478号明細書

【発明の概要】

20

【発明が解決しようとする課題】

【0005】

しかるに、これらの文献においては、複数のストレージ装置間の性能の違いや、複数のストレージ装置間の距離、等について考慮されていない。

【0006】

しかしながら、例えば、既存の記憶制御装置（以下、既存装置）に対してその既存装置と同様なレベルの性能及び／又は機能を有する記憶制御装置を増設する場合と、その既存装置と異なるレベルの性能及び／又は機能を有する記憶制御装置を増設する場合と、を同様な設計で増設してしまえば、個別の記憶制御装置の性能及び／又は機能に必要な以上の負荷や無駄が生じることになり、複数の記憶制御装置を有するシステム全体としての適切な性能及び／又は機能実現ことが出来ない事態が生じる恐れがある。

30

【0007】

また、例えば、設置スペースの制限を考慮した場合、既存装置が配置されている第一のデータセンタと異なるサイトに記憶制御装置を増設することにより第二のデータセンタを構築する必要性も生じる。この場合、サイト間の距離が性能に影響を及ぼす場合があるため、同一サイトに増設する場合と同様な設計で増設してしまえば、ユーザの要求する性能を満たすことが出来ない事態が生じる恐れがある。

【0008】

また、このような課題は、増設時のみでなく、複数の記憶制御装置を利用した初期構成の際にも考慮しなければならない共通の課題である。

40

【0009】

また、クラウド環境においても、同一又は異なるサイトに設置された複数の記憶制御装置を柔軟且つ効率的に連携させることは、柔軟で且つ効率的な記憶システムの構築、制御及び／又は管理において重要視すべき事柄である。

【0010】

したがって、本発明の目的は、複数の記憶制御装置間の性能及び／又は機能の違いや、複数の記憶制御装置間の距離、等を考慮した記憶システムを提供することにある。

【課題を解決するための手段】

【0011】

記憶システムが、複数の記憶制御装置ユニットを有する。各記憶制御装置ユニットは、

50

1以上のホストに接続された1以上の記憶制御装置で構成され、それら1以上の記憶制御装置が、それら1以上のホストに提供される複数の論理ボリュームを含んだ複数のボリュームを有する。記憶制御装置ユニット同士が、下記(a1)乃至(a3)を満たす接続形態である第3の接続形態に従って、互いに接続されている、

(a1) 1つの記憶制御装置ユニット内の1つの記憶制御装置ともう1つの記憶制御装置内の1つの記憶制御装置が、記憶制御装置の内部バスと同じ種類の接続媒体である第1種の接続媒体よりも最大通信距離が長く記憶制御装置の内部バスと異なる種類の接続媒体である第2種の接続媒体を介して接続されている、

(a2) 上記1つの記憶制御装置ユニットが、それに接続されている1以上のホストに、上記もう1つの記憶制御装置ユニットが有する論理ボリュームを仮想化して提供する、

(a3) 上記もう1つの記憶制御装置ユニットが、それに接続されている1以上のホストに、上記1つの記憶制御装置ユニットが有する論理ボリュームを仮想化して提供する。

【0012】

記憶制御装置が有するボリュームは、論理ボリュームであっても良いし、物理的な記憶デバイスであっても良い。

【0013】

複数の論理ボリュームが属する複数のティアを表すティア管理情報があって良い。ティア管理情報は、記憶制御装置ユニット毎にあって良い。記憶制御装置ユニット毎のティア管理情報は、記憶制御装置ユニットにあって良いし、記憶制御装置ユニットの外（例えば、複数の記憶制御装置ユニットに接続されている共通の記憶装置）にあって良い。

【0014】

ティアは、論理ボリュームのI/O性能に影響し得る1種類以上のファクターに基づいて定義されて良い。1種類以上のファクターが、接続形態種別というファクターを含んで良い。接続形態種別として、第3接続と、非第3接続があって良い。第3接続に対応した論理ボリュームは、仮想的な論理ボリュームで良い。ここでいう「仮想的な論理ボリューム」とは、その論理ボリュームを提供する記憶制御装置ユニットに前記第3の接続形態で接続されている記憶制御装置ユニット内の論理ボリュームがマッピングされた論理ボリュームである。非第3接続に対応した論理ボリュームは、第3接続に対応した論理ボリューム以外の論理ボリュームで良い。非第3接続に対応した論理ボリュームは、シンプロビジョニングが適用された論理ボリュームであっても良いし、物理的な記憶デバイスに基づく論理ボリュームであっても良い。

【図面の簡単な説明】

【0015】

【図1】本発明の実施例1に係る記憶システムの構成を示す。

【図2】第1の接続形態で接続された2つのHDKC（高性能及び／又は高機能の記憶制御装置（DKC（Disk Control Apparatus）））を示す。

【図3】第2の接続形態に従うLU（Logical Unit）マッピングの一例を示す。

【図4】第3の接続形態に従うLUマッピングの第1の例を示す。

【図5】第3の接続形態に従うLUマッピングの第2の例を示す。

【図6】第1の接続媒体、第2の接続媒体、及び第3の接続媒体の全てが接続されているHDKCを示す。

【図7】記憶システムが有する管理情報の一例を示す。

【図8】接続管理テーブルの構成を示す。

【図9】図9Aは、ティア管理テーブルの構成を示す。図9Bは、図9Aに示すテーブルに採用されている性能ファクターとそのポイント（ティアレベルの要素）との関係例を示す。

【図10】拡張されたティア管理テーブルを示す。

【図11】図10に示すテーブルに採用されている性能ファクターとそのポイントとの関係例を示す。

【図12】LU／PDV（Physical Device）変換テーブルの構成を示す。

10

20

30

40

50

【図 1 3】仮想ページ管理テーブルの構成を示す。

【図 1 4】図 1 4 A は、実ページ管理テーブルの構成を示す。図 1 4 B は、キュー先頭管理テーブルの構成を示す。

【図 1 5】ダイナミックプロビジョニング（シンプロビジョニング）が適用された H D K C を示す。

【図 1 6】H D K C が行う D K C 拡張処理の流れを示す。

【図 1 7】I / O コマンドの処理の一部（H D K C がサーバから I / O コマンドを受ける際の処理）の流れを示す。

【図 1 8】I / O コマンド処理の一部（第 1 の接続形態に従う L U に対する I / O の処理）の流れを示す。

10

【図 1 9】第 1 の接続形態に従う構成の概要の一例を示す。

【図 2 0】D P - L U（ダイナミックプロビジョニング（シンプロビジョニング）に従う L U）に関するリード処理の流れを示す。

【図 2 1】D P - L U に関するライト処理の流れを示す。

【図 2 2】I / O コマンド処理の一部（第 2 の接続形態に従う L U に対する I / O の処理）の流れを示す。

【図 2 3】第 2 の接続形態に従う構成の概要の一例を示す。

【図 2 4】I / O コマンドの処理の一部（H D K C が第 3 の接続媒体を介して別の H D K C から I / O コマンドを受ける際の処理）の流れを示す。

【図 2 5】I / O コマンドの処理の一部（第 3 の接続形態（ケース 3 - 1）に従う I / O の処理）の流れを示す。

20

【図 2 6】第 3 の接続形態（ケース 3 - 1）に従う構成の概要の一例を示す。

【図 2 7】I / O コマンドの処理の一部（第 3 の接続形態（ケース 3 - 2）に従う I / O の処理）の流れを示す。

【図 2 8】第 3 の接続形態（ケース 3 - 2）に従う構成の概要の一例を示す。

【図 2 9】I / O コマンドの処理の一部（第 3 の接続形態（ケース 3 - 3）に従う I / O の処理）の流れを示す。

【図 3 0】第 3 の接続形態（ケース 3 - 3）に従う構成の概要の一例を示す。

【図 3 1】担当 C P U テーブルの構成を示す。

【図 3 2】ポート単位の負荷分散（第 2 種の負荷分散処理）の開始前における構成例を示す。

30

【図 3 3】ポート単位の負荷分散（第 2 種の負荷分散処理）の概要を示す。

【図 3 4】図 3 4 A は、実施例 2 に係る担当 C P U テーブルの構成を示す。図 3 4 B は、負荷閾値管理テーブルの構成を示す。

【図 3 5】ポート負荷管理テーブルの構成を示す。

【図 3 6】テンポラリ L U 管理テーブルの構成を示す。

【図 3 7】負荷分散制御処理の一部（負荷分散判断処理）の流れを示す。

【図 3 8】負荷分散制御処理の一部（第 1 種の負荷分散処理）の流れを示す。

【図 3 9】図 3 9 A は、第 1 種の負荷分散処理における、負荷分散前の担当 C P U テーブルの一例、を示す。図 3 9 B は、第 1 種の負荷分散処理における、負荷分散後の担当 C P U テーブルの一例、を示す。

40

【図 4 0】負荷分散制御処理の一部（第 2 種の負荷分散処理の一部）の流れを示す。

【図 4 1】負荷分散制御処理の一部（第 2 種の負荷分散処理の一部）の流れを示す。

【図 4 2】負荷分散制御処理の一部（第 2 種の負荷分散処理の一部）の流れを示す。

【図 4 3】図 4 1 の S 4 1 0 7 の処理の一部を示す。

【図 4 4】図 4 1 の S 4 1 0 7 の処理の残りを示す。

【図 4 5】図 4 5 A は、第 2 種の負荷分散処理の開始前におけるポート負荷管理テーブル # H 0 を示す。図 4 5 B は、第 2 種の負荷分散処理の開始前におけるポート負荷管理テーブル # H 1 を示す。

【図 4 6】図 4 6 A は、第 2 種の負荷分散処理の開始前における担当 C P U テーブル # H

50

0を示す。図46Bは、第2種の負荷分散処理の開始前における担当CPUテーブル#H1を示す。

【図47】図47Aは、第2種の負荷分散処理における更新後のポート負荷管理テーブル#H0を示す。図47Bは、第2種の負荷分散処理における更新後のポート負荷管理テーブル#H1を示す。

【図48】図48Aは、第2種の負荷分散処理における更新後の担当CPUテーブル#H0を示す。図48Bは、第2種の負荷分散処理における更新後の担当CPUテーブル#H1を示す。

【図49】図49Aは、第2種の負荷分散処理における更新前のテンポラリLU管理テーブル#H1を示す。図49Bは、第2種の負荷分散処理における更新後のテンポラリLU管理テーブル#H1を示す。

10

【図50】本発明の実施例3に係るHDKCユニット間でのプール共有の一例を示す。

【図51】未割当ページ使用量管理テーブルの構成を示す。

【図52】図52Aは、プールをHDKC#H0及び#H1で共有して差し支えない状況に係るティア管理テーブル#H0を示す。図52Bは、プールをHDKC#H0及び#H1で共有することが好ましくない状況に係るティア管理テーブル#H0を示す。

【図53】実施例3に係るプール作成処理の流れを示す。

【図54】図53のS5303で更新された後の実ページ管理テーブル#H0を示す。

【図55】図53のS5311で更新された後の実ページ管理テーブル#H1を示す。

【図56】図56Aは、図53のS5307で更新される前のキュー先頭管理テーブル#H0を示す。図56Bは、図53のS5314で更新される前のキュー先頭管理テーブル#H1を示す。

20

【図57】図53のS5306で更新された後の実ページ管理テーブル#H0を示す。

【図58】図53のS5313で更新された後の実ページ管理テーブル#H1を示す。

【図59】図59Aは、図53のS5307で更新された後のキュー先頭管理テーブル#H0を示す。図59Bは、図53のS5314で更新された後のキュー先頭管理テーブル#H1を示す。

【図60】プールメンテナンス処理の流れの一部を示す。

【図61】プールメンテナンス処理の流れの一部を示す。

【図62】プールメンテナンス処理の流れの一部を示す。

30

【図63】図63Aは、図60のS6002で更新された後の未割当ページ使用量管理テーブル#H0を示す。図63Bは、図60のS6052で更新された後の未割当ページ使用量管理テーブル#H1を示す。

【図64】第2の接続形態に従うLUマッピングの変形例を示す。

【図65】図65Aは、HDKC#H0から第3接続HDKC#H1へのLU移動前の構成例を示す。図65Bは、HDKC#H0から第3接続HDKC#H1へLU#0を移動するとLU#0が属するティアが変わることを示す。

【図66】実施例3において、プール作成処理によってプール領域5011が作成された後に、DP-LU#160の仮想ページ(0x02)に、プール領域5011から、RAIDグループ#H15に基づく実ページ(0x01)が割り当てられたときの、仮想ページ管理テーブル#H0を示す。

40

【図67】実施例3において、プール作成処理によってプール領域5011が作成された後に、DP-LU#160の仮想ページ(0x02)に、プール領域5011から、RAIDグループ#H15に基づく実ページ(0x01)が割り当てられたときの、実ページ管理テーブル#H0を示す。

【図68】実施例3において、プール作成処理によってプール領域5012が作成された後に、そのプール領域5012に関連付けられているDP-LUに初めてデータが書き込まれた場合の、仮想ページ管理テーブル#H1を示す。

【図69】実施例3において、プール作成処理によってプール領域5012が作成された後に、そのプール領域5012に関連付けられているDP-LUに初めてデータが書き込

50

まれた場合の、実ページ管理テーブル#H1を示す。

【発明を実施するための形態】

【0016】

以下、本発明の幾つかの実施例を説明する。

【0017】

なお、以下の説明では、「xxxテーブル」の表現にて各種情報を説明することがあるが、各種情報は、テーブル以外のデータ構造で表現されていてもよい。データ構造に依存しないことを示すために「xxxテーブル」を「xxx情報」と呼ぶことができる。

【0018】

また、以下の説明では、何らかの対象を識別するための情報として、ID（識別子）、番号などが用いられるが、それに限らず、他種の識別情報が用いられても良い。

10

【0019】

また、以下の説明では、「プログラム」を主語として処理を説明する場合があるが、プログラムは、プロセッサ（例えばCPU（Central Processing Unit））によって実行されることで、定められた処理を、適宜に記憶資源（例えばメモリ）及び／又は通信インタフェース装置（例えば通信ポート）を用いながら行うため、処理の主語がプロセッサとされてもよい。プロセッサは、プロセッサが行う処理の一部又は全部を行うハードウェア回路を含んでも良い。コンピュータプログラムは、プログラムソースから各計算機にインストールされても良い。プログラムソースは、例えば、プログラム配布サーバ又は記憶メディアであっても良い。

20

【実施例1】

【0020】

＜本発明の実施例1に係る記憶システムの構成＞。

【0021】

図1は、本発明の実施例1に係る記憶システムの構成を示す。

【0022】

記憶システム110に、1以上のサーバ101が接続されている。各サーバ101は、例えば、通信ネットワークを介して、記憶システム110に接続されている。通信ネットワークは、例えば、FC（Fibre Channel）ネットワーク、LAN（Local Area Network）、或いはインターネットである。

30

【0023】

サーバ101は、ホスト装置の一例である。サーバ101は、例えば、記憶システム110に接続される通信インタフェース装置と、コンピュータプログラムを記憶する記憶資源と、通信インタフェース装置及び記憶資源に接続されたプロセッサ（典型的にはCPU（Central Processing Unit））を有する。プロセッサが、記憶資源からコンピュータプログラムを読み出し実行する。コンピュータプログラムとして、典型的には、OS（Operating System）、及び、アプリケーションプログラム111がある。アプリケーションプログラム111は、記憶システム110から提供されるLU（Logical Unit）にアクセスする。LUは、論理的な記憶デバイスであり、論理ボリュームと呼ばれても良い。なお、1つ又は複数のサーバ101は、1つ又は複数のサーバ101内の1つ又は複数の仮想サーバを有することも良い。

40

【0024】

記憶システム110は、複数の記憶制御装置を有する。記憶制御装置は、例えば、RAID（Redundant Array of Inexpensive Disks）制御を利用して複数の物理記憶媒体にデータを記憶させることを制御するコントローラ（以下、内部コントローラ）を含んだ装置（例えば、ディスクアレイ装置）である。物理記憶媒体は、物理記憶デバイスと言うこともできるので、それを略して、以下、「PDEV」（Physical Device）と言うことができる。PDEVは、例えば、ディスク型の記憶媒体（例えば、HDD（Hard Disk Drive））である。そこで、以下の説明では、記憶制御装置を、ディスク制御装置を略して「DKC」（Disk Control Apparatus）と言うことがある。そして、特に、高性能及び／又

50

は高機能のDKCを、「HDKC」(High Performance and/or High functionality DKC)と言い、低性能及び／又は低機能のDKC(例えば、HDKCより性能及び／又は機能の低いDKC)を、「LDKC」(Low Performance and/or Low functionality DKC)と言うことがあり、HDKC及びLDKCを特に区別しない場合には、単に「DKC」と言うことがある。PDEVは、HDKC及び／又はLDKCに内蔵されても良いし、HDKC及び／又はLDKCに外付けされても良い。また、ディスク型記憶媒体は、例えば、FC(Fibre Channel)、SAS(Serial Attached SCSI)、ATA(Advanced Technology Attachment)、SATA(Serial ATA)等のドライブで良い。また、PDEVは、ディスク型記憶媒体に代えて、他種の物理記憶媒体、例えば、SSD(Solid State Drive)、半導体メモリ格納装置／ボード等でも良い。

10

【0025】

記憶システム110は、HDKCユニット群104と、HDKCユニット群104より下位にあるLDKC群105とを有する。

【0026】

HDKCユニット群104は、複数のHDKCユニット141を有する。

【0027】

HDKCユニット141は、1以上のHDKCで構成されている。つまり、HDKCユニット141は、HDKCそれ自体でも良いし、複数のHDKCの集合でも良い。HDKCユニット141が複数のHDKCの集合の場合、それら複数のHDKCは、例えば、直列に接続されている。HDKCユニット141内において、2つのHDKCは、互いに、近距離接続可能な接続媒体(以下、内部バス)で接続される。内部バスは、例えば、最大でも数メートル程度と短い。内部バスは、例えば、PCIバス、又は、PCI-Express(PCIe)バスである。HDKCユニット141における、HDKCとHDKCとの接続形態を、本実施例では、「第1の接続形態」と言う。また、第1の接続形態に従う内部バスは、「第1の接続媒体」(或いは、第1のデータ転送用ライン)と呼ばれても良い。

20

【0028】

複数のHDKCユニット141は、例えば、直列に接続されている。HDKCユニット141内のHDKCと、別のHDKCユニット141内のHDKCが、互いに、長距離接続可能な接続媒体(以下、外部バス)で接続される。外部バスは、本実施例では、FC(Fibre Channel)ケーブルであるが、それに代えて、FCoE(Fibre Channel over Ethernet)のような別な接続媒体が採用されても良い。HDKCユニット141内のHDKCと、別のHDKCユニット141内のHDKCとの接続形態を、本実施例では、「第3の接続形態」と言う。第3の接続形態に従う外部バスは、「第3の接続媒体」(或いは、第3のデータ転送用ライン)と呼ばれても良い。また、以下、第1のHDKCに、他のHDKCを介することなく、外部バス(第3の接続媒体)を介して第2のHDKCが接続されている場合、第1のHDKCに第2のHDKCが直接的に接続されているということであり、第1のHDKCに、他のHDKC及び外部バス(第3の接続形態)を介して第2のHDKCが接続されている場合、第1のHDKCに第2のHDKCが間接的に接続されているということである。

30

40

【0029】

LDKC群105は、1以上のLDKC151を有する。LDKC151は、HDKCに外部バスで接続されている。LDKC151とHDKCとの接続形態を、本実施例では、「第2の接続形態」と言う。第2の接続形態に従う外部バスは、「第2の接続媒体」(或いは、第2のデータ転送用ライン)と呼ばれても良い。LDKC151は、例えば、複数のPDEVを有し、それらに基づく1以上のLU(Logical Unit)を管理する。LUは、論理的な記憶デバイス(論理ボリューム)である。LDKC内のLUは、仮想的にHDKC内のLUとして管理される。

【0030】

サーバ101は、1又は複数のHDKCに外部バス(通信ネットワーク)を介して接続

50

されるが、記憶システム 110 は、そのサーバ 101 には、仮想的に、1 台の D K C が接続されているように見せることができる。サーバ 101 には、そのサーバ 101 に接続されている H D K C (図 1 の説明において「対象 H D K C」と言う) から複数の L U が提供されるが、それら複数の L U は、例えば、下記 (x 1) 乃至 (x 5) のうちの少なくとも 1 つを含んで良い。

(x 1) 対象 H D K C が有する L U。

(x 2) 第 1 の接続媒体を介して提供される L U。すなわち、対象 H D K C を含んだ H D K C ユニット (図 1 の説明において「対象 H D K C ユニット」と言う) 内の他の H D K C が有する L U、

(x 3) 第 3 の接続媒体を介して提供される L U。すなわち、対象 H D K C ユニットに直接的又は間接的に接続されている他の H D K C ユニットが有する L U がマッピングされている仮想的な L U。

(x 4) 第 2 の接続媒体を介して提供される L U。すなわち、対象 H D K C ユニットに接続されている L D K C が有する L U がマッピングされている仮想的な L U、

(x 5) 第 3 の接続媒体及び第 2 の接続媒体を介して提供される L U。すなわち、他の H D K C ユニットが有する仮想的な L U (他の H D K C ユニットに接続されている L D K C が有する L U がマッピングされた仮想的な L U) がマッピングされている仮想的な L U。

【0031】

D K C に搭載される P D E V の容量及び／又は搭載台数は、一般に、増加している。そのため、D K C がサポートする記憶容量も、一般に、増加している。

【0032】

しかし、D K C の内部コントローラの性能及び／又は機能は、D K C がサポートする記憶容量の増加に比べて、向上していない。

【0033】

また、記憶システムを利用するユーザによって、D K C に要求されるニーズ (記憶容量、性能、ポートの数) に幅がある。

【0034】

記憶システムが、ハイエンドユーザのニーズに合わせて構築された場合、ローエンドユーザにとって、記憶システムは、過剰な性能及び／又は機能を有し、且つ、高価である。また、小規模から徐々に規模が大きくなるよう記憶システムを構築するようにしたとしても、初期導入費用が高くなってしまう傾向にある。

【0035】

前述した特許文献 2 によれば、複数の D K C 又は内部コントローラを内部バスにより相互接続させ、スケーラビリティを向上させることができる。しかし、この場合、例えば、下記の問題点がある。

(1) D K C 間及び／又は内部コントローラ間が、近距離 (例えば最大で数メートル) でなければならない。記憶システムの規模を大きくしようとすると、同一サイトに大きなスペースが必要となる。スペースが足りない場合、拡張ができない。

(2) インタフェースの種類によっては、接続できる D K C 及び／又は内部コントローラの数に制限がある。

【0036】

本実施例では、H D K C が、第 1 の接続媒体に代えて又は加えて、長距離接続が可能な第 3 の接続媒体 (外部バス) で接続され、第 3 の接続媒体で接続された複数の H D K C が仮想的に 1 つの D K C とされる。このため、複数の H D K C をそれぞれ離れた場所に設置することが可能であり、同一サイトに大きなスペースを必要としない。また、接続できる D K C の数に特に制限が無いので、スケーラビリティが向上する。

【0037】

これにより、初期導入費用を削減しつつ、小規模構成から大規模な構成へと柔軟な且つ効率的なスケールアウトを実現できる技術を提供できる。また、同一又は異なるサイトに設置された複数の記憶制御装置 (例えば、H D K C ユニット 141、H D K C 及び／又は

10

20

30

40

50

L D K C 1 5 1) を柔軟且つ効率的に連携させることは、クラウド環境において柔軟で且つ効率的な記憶システムを構築、制御及び／又は管理する上で必要且つ重要な技術となる。

【0038】

記憶システム110は、H D K C又はL D K Cを有するが、サーバ101に、複数のL Uを有する単一のD K Cとして把握させることができる。この場合、H D K Cが増設されたとしても、H D K Cの上位のデバイス（例えば、サーバ、又は、通信ネットワークを構成するスイッチ）は、H D K Cの増設としてではなく、フロントエンドの通信ポート（以下、F Eポート）の増設としてしか把握しない。F Eポートは、フロントエンドのデバイスに接続される通信ポートであり、H D K Cが有する。フロントエンドのデバイスとは、H D K Cの外にあるデバイスであり、例えば、サーバ、通信ネットワーク内のスイッチ、第1の接続形態で接続されるH D K C、第3の接続形態で接続されるH D K C、或いは、第2の接続形態で接続されるL D K Cである。

10

【0039】

なお、記憶システムは、単一のD K Cではなく、複数のD K Cとして、サーバ101から把握されて良い。また、H D K Cが増設された場合、サーバ101から、H D K Cが増設されたことが把握されて良い。

【0040】

H D K Cについて、「高性能及び／又は高機能」とは、第1の性能閾値より高い性能、及び／又は、所定種類の機能でも良いし、他のD K Cの性能及び／又は機能に比べて高い性能及び／又は機能（他のD K Cとほぼ同じレベルの性能及び／又は機能）でも良い。また、H D K Cは、ディスクアレイ装置に限らず、他種の記憶制御装置（例えば、ストレージ仮想化機能を有するインテリジェントなスイッチ）でも良い。

20

【0041】

H D K Cに関する技術的な目的は、例えば、下記の（a）乃至（d）、
（a）複数のH D K Cを接続する際の制御方法や接続後の利用方法等を提供すること、
（b）或るH D K Cと異なるサイトに、同じくH D K Cを接続する際の制御方法や接続後の利用方法等を提供すること、
（c）複数のH D K Cを接続する場合に、各D K Cの間の接続関係や距離に応じて、接続されるH D K Cに作成されるL Uのティアレベル等の属性を適切に管理し、接続後の適切な利用方法等を提供すること（「ティア」について後述する）、
（d）或るH D K Cと他の複数のH D K Cとを接続する場合に、当該他の複数のH D K C間の性能の違いや、当該或るH D K Cからの距離に応じて、他のH D K C毎に作成されるL Uのティアレベル等の属性を適切に管理し、接続後の適切な利用方法等を提供すること、
がある。

30

【0042】

L D K Cについて、「低性能及び／又は低機能」とは、第2の性能閾値より低い性能、及び／又は、所定種類の機能でも良いし、他のD K Cの性能及び／又は機能に比べて低い性能及び／又は機能（他のD K Cとほぼ同じレベルの性能及び／又は機能）でも良い。第2の性能閾値は、第1の性能閾値以下でよい。また、L D K Cは、ディスクアレイ装置に限らず、他種の記憶制御装置でも良い。

40

【0043】

L D K Cに関する技術的な目的は、例えば、下記（p）及び（q）、
（p）或るH D K CとL D K Cとを接続する場合、各D K Cの間の性能の違いや距離に応じて、L D K Cに作成されるL Uのティアレベル等の属性を適切に管理し、接続後の適切な利用方法等を提供すること、
（q）或るH D K Cと複数のL D K Cとを接続する場合に、当該複数のL D K C間の性能の違いや、H D K Cからの距離に応じて、L D K C毎に作成されるL Uのティアレベル等の属性を適切に管理し、接続後の適切な利用方法等を提供すること、

50

である。

【0044】

次に、各接続形態を説明する。

【0045】

＜第1の接続形態＞。

【0046】

図2は、第1の接続形態で接続された2つのHDKCを示す。

【0047】

なお、図2には、ハードウェア要素に、参照符号と、IDが付されている。以下の説明では、同種の要素を区別する場合には、IDを用いて要素を表し（例えば、HDKC#H0）、同種の要素を特に区別しない場合には、参照符号を用いて要素を表す（例えば、HDKC201）。

10

【0048】

また、HDKCのIDは、アルファベット「H」と1桁の番号との組合せ（2桁の符号）であるが、HDKCに複数の同種のハードウェア要素がある場合、同種の複数のハードウェア要素は、そのHDKCのIDを上位に含んだ3桁の符号（ID）で区別されて良い。具体的には、例えば、HDKC#H0が有する2つのCPUのIDは、H00、H01である。なお、「CPU#H0」と包括表記した場合、CPU#H0は、CPU#H00及び#H01のうちの少なくとも1つである。これは、CPUに限らず、他種のハードウェア要素についても同様である。

20

【0049】

HDKC201は、PDEV群と、内部コントローラとに大別することができる。

【0050】

PDEV群は、1以上のRG231を含む。RG231は、RAID（Redundant Array of Independent（or Inexpensive）Disks）グループであり、2以上のPDEVで構成されている。

【0051】

内部コントローラは、例えば、2つのコントローラモジュール211で構成されている。1つのHDKC201に存在するコントローラモジュール211の数は、1でも良いし、3以上でも良い。

30

【0052】

コントローラモジュール211は、フロントインターフェース（F-I/F）221と、バックエンドインターフェース（B-I/F）225と、キャッシュメモリ（CM）224と、CPU222と、スイッチ（SW）226と、ローカルメモリ（LM）223とを有する。

【0053】

F-I/F221は、サーバ101に接続される通信インターフェース装置である。F-I/F221は、1又は複数の上位ポートを有する。上位ポートは、サーバ又はスイッチ等の上位のデバイスに接続される通信ポートである。

【0054】

B-I/F225は、1以上のRG231に接続される通信インターフェース装置である。

40

【0055】

CM224は、RG231に書き込まれるデータ、及び、RG231から読み出されたデータを一時記憶する。少なくとも1つのCM224は、キャッシュメモリとして使用される記憶領域（CM領域）に加えて、複数のコントローラモジュールの共有メモリ（SM）として使用される記憶領域（SM領域）を有しても良い。

【0056】

LM223は、CPU222に使用される情報や、CPU222で実行されるコンピュータプログラムを記憶して良い。LM223は、CPU222に接続されており、SW2

50

2 6には接続されていないが、SW 2 2 6に接続されていても良い。

【0 0 5 7】

CPU 2 2 2は、LM 2 2 3に記憶されているコンピュータプログラム（以下、制御プログラム）を実行することができる。制御プログラムは、LM 2 2 3に代えて又は加えて、他の記憶資源に記憶されていても良い。

【0 0 5 8】

SW 2 2 6には、F-I/F 2 2 1、CPU 2 2 2、CM 2 2 4、B-I/F 2 2 5、及び他の直近のコントローラモジュール 2 1 1内のSW 2 2 6が接続されている。SW 2 2 6は、これらの要素間の接続を切り替える。SW 2 2 6と各要素との間のインタフェースは、例えば、PCI-Express (PCIe) である。

10

【0 0 5 9】

1つのHDKCにおいて、2つのコントローラモジュール 2 1 1内の2つのSW 2 2 6（例えばSW#H 0 0及び#H 0 1）が、内部バス 2 2 7で接続されている。同様に、第1の接続形態で接続されている別々のHDKC#H 0 及び#1内の2つのSW 2 2 6（SW#H 0 1及び#H 1 0）が、内部バス 2 2 7で接続されている。

【0 0 6 0】

このような構成により、HDKCユニット 1 4 1では、複数のコントローラモジュール 2 1 1内のハードウェア要素や、複数のHDKC 2 0 1内のRG 2 3 1を共有することができる。また、複数のHDKC 2 0 1内の管理情報（例えば、後述する1以上のテーブル）も共有できる。このため、1つの内部コントローラ（或いは、1つのコントローラモジュール）で、複数のHDKC 2 0 1の情報を管理することができ、HDKCユニット 1 4 1は、1つのHDKCのように振る舞うことが可能である。例えば、CPU#H 0 1が、CM#H 0 0、#H 1 0及び#H 1 1のいずれにも、CM#H 0 1にアクセスするのと同じようにアクセスすることができる。また、CPU#H 0 1は、RG#H 0 0～#1 4に対してI/O (Input/Output)を行うのと同様に、RG#H 1 0～#H 1 4のいずれに対してもI/Oを行うことができる。

20

【0 0 6 1】

具体的には、例えば、サーバ 1 0 1から指定されたLUN (Logical Unit Number) に対応するLUに対してライトコマンドを受けたコントローラモジュール 2 1 1と、そのライトコマンドに従うライト先のRG 2 3 1とが、同一のHDKC 2 0 1にある場合、一点鎖線に示す流れで、データがRG 2 3 1に書き込まれて良い。すなわち、ライトコマンドを受けたコントローラモジュール#H 0 0（別のコントローラモジュールでも良い）が、CM#H 0 0（別のCMでも良い）にデータを書込み、CM#H 0 0内のデータを、ライト先RG#H 0 1に書き込んでも良い。なお、それに限らず、I/Oコマンドを受けたコントローラモジュールとは別のコントローラモジュールが、I/O先のRGに対するI/Oを行っても良い。

30

【0 0 6 2】

また、例えば、ライトコマンドを受けたコントローラモジュール 2 1 1と、そのライト先のRG 2 3 1とが、異なるHDKC 2 0 1にある場合、点線に示す流れで、データがRG 2 3 1に書き込まれて良い。すなわち、ライトコマンドを受けたコントローラモジュール#H 0 1（別のコントローラモジュールでも良い）が、SW#H 0 1及び#H 1 0を介して、CM#H 1 0（他のCMでも良い）にデータを書込み、コントローラモジュール#H 1 0（他のコントローラモジュールでも良い）が、CM#H 1 0内のデータを、SW#H 1 0及び#1 1を介して、ライト先RG#H 1 3に書き込んでも良い。

40

【0 0 6 3】

また、HDKCユニット 1 4 1がサーバ 1 0 1に提供するLUは、シンプロビジョニング (Thin Provisioning)に従う仮想的なLUであっても良い。

【0 0 6 4】

或るHDKC内のコントローラモジュールは、別のHDKC内のコントローラモジュール或いはRGにアクセスする際、後述のLUマッピングテーブルを参照する必要は無い。

50

なぜなら、HDKCユニットにおいて、各コントローラモジュールは、そのコントローラモジュールを有するHDKCとは別のHDKC内のハードウェア要素（例えば、CM、RG）に、そのコントローラモジュールを有するHDKC内のハードウェア要素であるかのようにアクセスすることができるからである。

【0065】

なお、前述したように、HDKCユニット141において、コントローラモジュール同士、及び、HDKC同士は、内部バスで接続されているため、コントローラモジュール間の距離、及び、HDKC間の距離は、外部バス接続に比べて短い。

【0066】

以上が、第1の接続形態に関する説明である。なお、LDKCも、内部コントローラと、PDEV群とを有して良い。LDKCの構成は、HDKCと同様の構成であっても良い。LDKCの内部コントローラの構成は、HDKCの内部コントローラの構成と異なっても良い。

【0067】

＜第2の接続形態＞。

【0068】

図3は、第2の接続形態に従うLUマッピングの一例を示す。なお、以下、HDKC及びそれが有する要素や、LDKC及びそれが有する要素を、参照符号に代えて、要素のIDを用いて説明することがある。また、以下の説明では、DKC#xが有する内部コントローラを、「内部コントローラ#x」と言うことがある。

【0069】

DKCの内部コントローラは、複数のFEポート（フロントエンドの通信ポート）501を有する。複数のFEポート501は、イニシエータになるFEポート（イニシエータポート）、ターゲットになるFEポート（ターゲットポート）、及び、イニシエータにもターゲットにもなるFEポート（イニシエータ／ターゲットポート）の少なくとも1つを含んで良い。以下、図では、イニシエータポートは「IP」と表され、ターゲットポートは「TP」と表され、イニシエータ／ターゲットポートは、「ITP」と表されることがある。HDKCについて言えば、FEポートは、F-I/F（図2参照）にある。

【0070】

HDKC#H0のターゲットポート（例えば#H00）には、サーバ（例えば#0）が接続される。1つのターゲットポートに接続されるサーバの数は、2以上でも良い。

【0071】

HDKC#H0のイニシエータポート#H03が、LDKC#L0（そのターゲットポート#L01）に接続される。LDKC#L0は、LU#A及び#Bを有する。LU#A及び#Bのうちの少なくとも1つが、シンプロビジョニングに従う仮想的なLUであっても良い。シンプロビジョニングは、ダイナミックプロビジョニングと呼ばれることもある。そこで、以下、シンプロビジョニングに従う仮想的なLUを、「DP-LU」と言うことがある。

【0072】

ターゲットポート#H02に、LU#4及び#5が関連付けられており、サーバ#02が、ターゲットポート#H02に接続されている。サーバ#02は、LU#4及び／又は#5を指定したI/Oコマンドを発行することができる。

【0073】

LU#4には、外部のLU#Aがマッピングされており、LU#5には、外部のLU#Bがマッピングされている。つまり、LU#4及び#Bは、仮想的なLUである。

【0074】

第2の接続形態によれば、HDKC#H0が、イニシエータポート#H03に接続されたLDKC#L0内の外部LU#A及び#Bを、仮想的にLU#4及び#5として扱うことができる。仮想LU#4（#5）と外部LU#A（#B）との対応関係は、LUマッピングテーブル322（#H0）から特定可能である。LUマッピングテーブル322（#

10

20

30

40

50

H0)は、そのテーブルを有するHDKC内の仮想LUにマッピングされている外部LUに関する情報を有する。具体的には、LUマッピングテーブル#0は、仮想LU毎に、下記の情報、

(*) 仮想LUにマッピングされている外部LUを有するDKCのID、

(*) その外部LUのID、

を有する。LUマッピングテーブル#H0は、内部コントローラ#H0内の記憶資源（例えば、LM、CM）に記憶される。内部コントローラ#H0（CPU#H0（CPU#H00又は#H01））は、サーバ#2からのI/OコマンドでLU#4が指定されていた場合、LU#4にLU#AがマッピングされていることをLUマッピングテーブル#H0から特定し、特定したLU#Aを指定したI/OコマンドをLDKC#L0に送信する。それにより、LU#4（LU#A）に対するI/Oが行われる。

10

【0075】

なお、1つの仮想LUにマッピングされる外部LUは、1つLDKCが有する1つの外部LUに限られることは無く、1つLDKCが有する複数の外部LUや、LDKCの各々から1つ又は複数つつ提供される外部LU（複数のLDKCにまたがって提供される複数の外部LU）が、1つの仮想LUにマッピングされることも良い。これらの場合、上記外部LUのIDには、複数の外部LUのIDが含まれることになる。

【0076】

第2の接続形態によれば、HDKC#H0は、LDKC#L0を、HDKC#H0が有するPDEVと同等に扱って良い。LDKC#L0が有するFEポートには、サーバが接続されても良いし、接続されなくても良い。第2の接続形態によれば、HDKC#H0が、LDKC#L0内のLUがマッピングされた仮想LUを有することはあっても、LDKC#L0が、HDKC#H0内のLUがマッピングされた仮想LUを有することは無い。

20

【0077】

第2の接続形態によれば、仮想LU#4及び#5は、HDKC#H0内のRGに基づくLU（例えばLU#0）に比べ、アクセスが外部バス501を経由するというオーバーヘッドがある（外部バス501は、外部のスイッチ（例えば、FC（Fibre Channel）スイッチ）を経由しても良い）。このため、仮想LU#4及び#5は、HDKC#H0内のRGに基づくLUよりも低いティア（例えば、中位ティア又は下位ティア）として使用されて良い。なお、後に説明する実施例2では、負荷分散が行われるが、第2の接続形態に従う仮想LUは、例えば、中位ティア及び／又は下位ティアに属する負荷分散元LU（データの移動元LU）、及び／又は、中位ティア及び／又は下位ティアに属する負荷分散先LU（データの移動先LU）で良い。負荷分散において、例えば、第1のHDKCから第3の接続媒体を介して第2のHDKCに仮想LUが移動された場合、その仮想LUにマッピングされている第1の外部LU（第1のLDKC内のLU）内のデータが、第2のHDKCに接続されている第2のLDKC内の第2のLU（第2のHDKC内の仮想LUにマッピングされるLU）に移動されても良い。

30

【0078】

また、例えば、第2の接続形態に従いマッピングされた外部LU（LDKC内のLU）は、図64に示すように、HDKC#H0内のプール#H0を構成する少なくとも1つのLUにマッピングされても良い。従って、プール#H0を構成する1以上のLU（プールLU）のうちの少なくとも1つが、仮想LUで良い。HDKC#H0は、DPLU（ダイナミックプロビジョニングに従う仮想LU）#0を有する。HDKC#H0は、DPLU#0内の或る仮想ページ61にデータが書き込まれる場合、その仮想ページ61に、プール#0から実ページ62を割り当てることができる。HDKC#H0は、実ページ62にデータを書き込むことができる。LDKC#L0内の外部LU#Dがマッピングされた仮想LUが、その実ページ62を含む場合、実ページ62に書き込まれるデータは、その実ページ62に対応する、外部LU#D内の記憶領域63に、書き込まれる。

40

【0079】

また、LDKCに代えて又は加えて、HDKCが、第2の接続媒体を介してHDKCに

50

接続されて良い。

【0080】

また、同一の1又は複数のLDKCが1つのHDKCユニット141内の複数のHDKC201に接続されても良い。この場合、1つのHDKCユニット141内の複数のHDKCの各々が、同一の1又は複数のLDKC内の同一のLU（第一のLU）を利用することにより、この同一のLU（第一のLU）がマッピングされた複数の仮想ボリューム（第一の仮想LU、第二の仮想LU）を1つ又は複数のサーバ101に提供することも良い。また、1つのHDKCユニット141内の複数のHDKCの各々が、同一の1又は複数のLDKC内の別々のLU（第二のLU、第三のLU）を利用することにより、この別々のLU（第二のLU、第三のLU）の各々がマッピングされた別々の仮想ボリューム（第三の仮想LU、第四の仮想LU）を1つ又は複数のサーバ101に提供することも良い。

10

【0081】

これらのうち、1つのHDKCユニット141内の複数のHDKCの各々が、同一のLU（第一のLU）がマッピングされた複数の仮想ボリューム（第一の仮想LU、第二の仮想LU）を1つ又は複数のサーバ101に提供する場合、この複数の仮想ボリューム（第一の仮想LU、第二の仮想LU）は、フェールオーバ、又は負荷分散のために利用される。

【0082】

このうち、フェールオーバへの利用の具体例としては、1つのHDKCユニット141内の複数のHDKC及び／又は管理装置301が、1つ又は複数のサーバ101から第一の仮想LUを介して第一のLUのデータをリード及び／又はライトするアクティブな通常パスと、この1つ又は複数のサーバ101から第二の仮想LUを介してこの第一のLUのデータをリード及び／又はライトするアクティブ又はパッシブな交代パスと、の両方を管理し、上記通常パスの障害に応じて上記交代パスを利用して1つ又は複数のサーバ101から送信されるライトコマンド及び／又はリードコマンドを処理して、第一のLUへのデータライト及び／又は第一のLUからのデータリードを実施して、1つ又は複数のサーバ101へライト完了応答及び／又はリード完了応答を返す。このとき、1つのHDKCユニット141内の複数のHDKCの各々はキャッシュメモリ224を有するので、上記通常パス及び／又は交代パスにおいては、ライトコマンド及び／又はリードコマンドを処理して、HDKC内のキャッシュメモリ224へライトデータを書き込み及び／又はデータをリードして、1つ又は複数のサーバ101へライト完了応答及び／又はリード完了応答を返した後、第一のLUへのデータライト及び／又は第一のLUからのデータリードを実施することも良い。米国においては、この具体的な処理について、米国特許第7337350号（Mimatsu et al.）の明細書及び図面を援用することができる。この場合、当該明細書及び図面のHost Computerが本願の1つ又は複数のサーバ101、当該明細書及び図面のStorage System 2及びStorage System 3の各々が本願の1つのHDKCユニット141内の複数のHDKC、当該明細書及び図面のStorage System 1が本願の同一のLU（第一のLU）を有するLDKC、にそれぞれ対応付けて本願技術を適用する。

20

30

【0083】

また、負荷分散への利用の具体例としては、次のようなものが挙げられる。1つのHDKCユニット141内の複数のHDKC及び／又は管理装置301が、1つ又は複数のサーバ101から第一の仮想LUを介して第一のLUのデータをリード及び／又はライトするパスと、この1つ又は複数のサーバ101から第二の仮想LUを介してこの第一のLUのデータをリード及び／又はライトするパスと、の両方を管理し、1つのHDKCユニット141内の複数のHDKCの各々の負荷に応じて、上記複数のパスの一方のパスを利用して実施していた処理を上記複数のパスの他方のパスを利用して実施するように変更する。ここでいう処理とは、1つ又は複数のサーバ101から送信されるライトコマンド及び／又はリードコマンドを処理して、第一のLUへのデータライト及び／又は第一のLUからのデータリードを実施して、1つ又は複数のサーバ101へライト完了応答及び／又はリード完了応答を返すという処理である。このとき、1つのHDKCユニット141内の

40

50

複数のHDKCの各々はキャッシュメモリ224を有するので、上記複数のパスにおいては、ライトコマンド及び／又はリードコマンドを処理して、HDKC内のキャッシュメモリ224へライトデータを書き込み及び／又はデータをリードして、1つ又は複数のサーバ101へライト完了応答及び／又はリード完了応答を返した後、第一のLUへのデータライト及び／又は第一のLUからのデータリードを実施することも良い。米国においては、この具体的な処理について、米国特許公開第2006/0271758号(Innan et al.)の明細書及び図面を援用することができる。この場合、当該明細書及び図面のHost Deviceが本願の1つ又は複数のサーバ101、当該明細書及び図面のVirtualization Storage Device 1及びVirtualization Storage Device 2の各々が本願の1つのHDKCユニット141内の複数のHDKC、当該明細書及び図面のExternal Storage Deviceが本願の同一のLU（第一のLU）を有するLDKC、にそれぞれ対応付けて本願技術を適用する。なお、フェールオーバー又は負荷分散への利用の具体例において複数のサーバ101が利用される場合には、当該複数のサーバ101がクラスタ関係を有するサーバに該当することが良く、この場合、上記フェールオーバーの交代パスの利用又は上記負荷分散の他方のパスの利用に際して複数のサーバ101間で処理が引き継がれる。

【0084】

このとき、1つ又は複数のサーバ101は、1つ又は複数のサーバ101内の1つ又は複数の仮想サーバに該当することも良い。1つ又は複数のサーバ101間で1つの仮想サーバが利用される場合には、上記フェールオーバーの交代パスの利用又は上記負荷分散の他方のパスの利用に際して1つ又は複数のサーバ101間で仮想サーバが移行される。1つ又は複数のサーバ101間で複数の仮想サーバが利用される場合には、上記フェールオーバーの交代パスの利用又は上記負荷分散の他方のパスの利用に際して1つ又は複数のサーバ101の複数の仮想サーバ間で処理が引き継がれる。この場合、HDKCユニット141や管理装置301は、1つのHDKCユニット141内の複数のHDKCの各々の性能、状態（障害の有無、程度等）及び／又は負荷、又は複数のHDKC間の性能の違い、状態（障害の有無、程度等）の違い、距離、及び／又は負荷の違いに応じて、複数のHDKCの何れをフェールオーバーや負荷分散のSourceやDestinationにするかを判断や決定して、フェールオーバーや負荷分散が実施されることが良い。なお、上記の距離は、実際の距離に限らず、コマンド送受信時間、コマンド応答時間及び／又はコマンド処理時間（詳細については図10に対応する明細書部分で説明）でも良い。これらの技術は、本願で始めて明らかにされる技術の一つである。

【0085】

また、同一の1又は複数のLDKCが複数のHDKCユニット141に接続されても良い。この場合も、上記に記載の、同一の1又は複数のLDKCが1つのHDKCユニット141内の複数のHDKC201に接続される場合と同様な態様、具体例及び援用を適用する。ただし、1つのHDKCユニット141内の複数のHDKCは、複数のHDKCユニット141と読み替えて適用される。この場合、複数のHDKCユニット141や管理装置301は、複数のHDKCユニット141の各々の性能、状態（障害の有無、程度等）及び／又は負荷、又は複数のHDKCユニット141間の性能の違い、状態（障害の有無、程度等）の違い、距離、及び／又は負荷の違いに応じて、複数のHDKCユニット141の何れをフェールオーバーや負荷分散のSource（フェールオーバー元や負荷分散元）やDestination（フェールオーバー先や負荷分散先）にするか否かを判断や決定して、フェールオーバーや負荷分散が実施されることが良い。なお、上記の距離は、実際の距離に限らず、コマンド送受信時間、コマンド応答時間及び／又はコマンド処理時間（詳細については図10に対応する明細書部分で説明）でも良い。また、1つ又は複数のHDKCユニット141や管理装置301は、上記の同一の1又は複数のLDKCが1つのHDKCユニット141内の複数のHDKC201に接続される場合では、1つのHDKCユニット141内の複数のHDKCの各々の状態（障害の有無、程度等）及び／又は負荷、又は複数のHDKC間の状態（障害の有無、程度等）の違い及び／又は負荷の違いに応じて、複数のHDKCの何れをフェールオーバーや負荷分散のSourceやDestinationにするかを判断や決定

する（性能や距離を考慮しない）のに対して、同一の1又は複数のLDKCが複数のHDKCユニット141に接続される場合には、複数のHDKCユニット141の各々の性能、状態（障害の有無、程度等）及び／又は負荷、又は複数のHDKCユニット141間の性能の違い、状態（障害の有無、程度等）の違い、距離、及び／又は負荷の違いに応じて、複数のHDKCユニット141の何れをフェールオーバや負荷分散のSourceやDestinationにするかを判断や決定する、ように制御することも良い。これらの技術も、本願で始めて明らかにされる技術の一つである。

【0086】

＜第3の接続形態＞。

【0087】

図4は、第3の接続形態に従うLUマッピングの第1の例を示す。

【0088】

HDKC#H0及び#1は、FEポートとして、ターゲットポート、イニシエータ／ターゲットポートを有する。FEポートは、前述したように、F-I/F（図2参照）にある。

【0089】

HDKC#H0のターゲットポート（例えば#H01）にサーバ（例えば#0）が接続されており、HDKC#H1のターゲットポート（例えば#H12）にサーバ（例えば#2）が接続されている。

【0090】

HDKC#H0のイニシエータ／ターゲットポート#H03と、HDKC#H1のイニシエータ／ターゲットポート#H11が、外部バス601で接続されている（外部バス601は、外部のスイッチ（例えば、FC（Fibre Channel）スイッチ）を経由しても良い）。つまり、第3の接続形態によれば、HDKC#H0がHDKC#H1にI/Oコマンドを発行することもあれば、HDKC#H1がHDKC#H0にI/Oコマンドを発行することもある。なぜなら、HDKC#H0内のLUがHDKC#H1内の仮想LUにマッピングされることもあれば、HDKC#H1内のLUがHDKC#H0内の仮想LUにマッピングされることがあるからである。図4によれば、LU#1が仮想LU#Aにマッピングされ、LU#3が仮想LU#Cにマッピングされている。また、LU#Bが仮想LU#0にマッピングされ、LU#Dが仮想LU#2にマッピングされている。図4によれば、仮想LUも、仮想LUにマッピングされるLUも、ターゲットポートに関連付けられている。

【0091】

HDKC#H0は、LUマッピングテーブル#H0を有し、HDKC#H1は、LUマッピングテーブル#H1を有する。LUマッピングテーブル#H0は、HDKC#H0が有する仮想LU#0（#2）にマッピングされている外部LU#B（#D）に関する情報を有する。LUマッピングテーブル#H1は、HDKC#H1が有する仮想LU#A（#C）にマッピングされている外部LU#1（#3）に関する情報を有する。仮想LU#0（#2）にマッピングされている外部LU#B（#D）は、LUマッピングテーブル#H0から特定可能であり、仮想LU#1（#3）にマッピングされている外部LU#A（#C）は、LUマッピングテーブル#H1から特定可能である。

【0092】

HDKC#H0及び#H1に接続されているサーバには、1台のHDKCであるかのように見せることができる。例えば、サーバ#0は、HDKC#H0内のLU#0及び#1を認識することができるが、LU#0の実体は、LU#Bであり、HDKC#H0ではなくHDKC#H1に存在する。しかし、サーバ#0は、LU#0を認識し、そのLU#0にマッピングされている、HDKC#H1内のLU#Bを認識しない。例えば、内部コントローラ#H0（CPU#H0）は、サーバ#0からのI/OコマンドでLU#0が指定されていた場合、LU#0にLU#BがマッピングされていることをLUマッピングテーブル#H0から特定し、特定したLU#Bを指定したI/OコマンドをLDKC#H1に

10

20

30

40

50

送信する。それにより、LU#0 (LU#B) に対する I/O が行われる。

【0093】

第3の接続形態に従う仮想LU#0、#2、#A及び#Cは、HDKC内のLUにマッピングされることから、上位ティアに属して良い。第3の接続形態に従う仮想LUは、例えば、上位ティアに属する負荷分散元LU（データの移動元LU）、及び／又は、上位ティアに属する負荷分散先LU（データの移動先LU）で良い。負荷分散において、例えば、仮想LUにマッピングされているLU内のデータが、そのLUを有するHDKCから第3の接続媒体を介して別のHDKCのLUに移動されても良い。

【0094】

図5は、第3の接続形態に従うLUマッピングの第2の例を示す。

10

【0095】

図5に示すように、仮想LU（例えばLU#A）にマッピングされる外部LU（例えばLU#4）は、サーバに接続されているFEポート（ターゲットポート）に関連付けられているLUではなく、サーバに接続されていないFEポート（イニシエータ／ターゲットポート）（例えば#H03）に関連付けられているLUであっても良い。

【0096】

<第1の接続媒体、第2の接続媒体、及び第3の接続媒体の全てに接続されているHDKC>。

【0097】

図6は、第1の接続媒体、第2の接続媒体、及び第3の接続媒体の全てが接続されているHDKC#H0を示す。

20

【0098】

HDKC#H0に、第1の接続媒体（内部バス）227を介してHDKC#H1が接続されており、第2の接続媒体（外部バス）501を介してLDKC#L0が接続されており、第3の接続媒体（外部バス）601を介してHDKC#HAが接続されている。

【0099】

HDKC#H0は、下記のLU、

(*) HDKC#H0が有するRGに基づくLU#0、

(*) 第1の接続形態に従うLU#x（HDKC#H0は、HDKC#H1内のLU#xに直接アクセスできる）、

30

(*) 第2の接続形態に従う仮想LU#3（第2の接続形態に従い外部LU#z（LDKC#L0内の外部LU）がマッピングされた仮想LU）、

(*) 第3の接続形態に従う仮想LU#2（第3の接続形態に従い外部LU#y（HDKC#HA内の外部LU）がマッピングされた仮想LU）、

を有する。上述した複数のLUについて、例えば、下記のことが言える。

(*) LU#0、#x、#2及び#3のうちの少なくとも1つは、サーバから認識されるLU（例えば、DP-LU）であっても良いし、サーバから認識されないLU（例えばプールLU）であっても良い。

(*) LU#x、#y及び#zのうちの少なくとも1つは、DP-LUであっても良い。

(*) LU#x及び#yのうちの少なくとも1つは、第2の接続形態に従う仮想LU（第2の接続形態に従い外部LUがマッピングされた仮想LU）であっても良い。

40

【0100】

<管理情報>。

【0101】

図7は、記憶システム110が有する管理情報の一例を示す。

【0102】

管理情報は、記憶システム110が管理する情報である。全てのHDKCが、全ての種類の管理情報を記憶しても良い。また、管理情報の内容は、全てのHDKCで同一であっても良いし、HDKC毎に異なっても良い。例えば、第1のHDKCが記憶する第1の種類の管理情報は、第1のHDKCに関する情報のみを有していて、第1のHDKC

50

に関係しない情報を有していなくて良い。

【0103】

また、複数のHDKCで共有可能な記憶領域が記憶しても良い管理情報があっても良いし、アクセス性能の低下を防ぐために個々のHDKCが記憶した方が良い管理情報があっても良い。

【0104】

図7によれば、複数のHDKCが共有可能な記憶領域を含んだ装置（共有情報管理装置）301が、複数のHDKCに接続されている。共有情報管理装置301は、以下の管理情報、

（＊）第3の接続形態に関する情報（例えば、HDKCの順序、HDKC間毎の距離の長さ、各HDKCの状態（正常或いは閉塞等））を有する第3接続管理テーブル311、

（＊）仮想LUに関する情報（例えば、仮想LU毎に、第2と第3の接続形態のどちらに従う外部LUがマッピングされているか、及び、その外部LUを有するDKCのIDを有する情報）を有する仮想LU管理テーブル312、

（＊）各HDKCの内部コントローラに関する情報（例えば、内部コントローラの性能（例えば、高速、中速或いは低速）を表す情報）を有するコントローラ管理テーブル313、

（＊）複数のLUの種別に関する情報（例えば、LU毎に、LUの基になっているPDEVの種類（例えば、SSD、SAS、或いはSATA）を表す情報）を有するLU種別管理テーブル314、

を記憶することができる。共有情報管理装置301は、様々な管理情報を管理及び制御する管理ソフトウェアを有する。各HDKCは、必要なタイミング（例えば、HDKCにLUが追加されたとき、第2又は第3の接続形態に従う仮想LUがHDKCに追加されたとき、第2又は第3の接続形態で接続されるHDKCが増えたとき）のみ、共有情報管理装置301にアクセスして良い。

【0105】

また、図7によれば、個々のHDKCが、例えば、以下の管理情報、

（＊）ティアに関する情報を有するティア管理テーブル321、

（＊）LUの対応関係を表す情報を有するLUマッピングテーブル322、

（＊）LUとPDEVとの対応関係を表す情報を有するLU/PDEV変換テーブル323、

（＊）どのLUに対するI/OをどのCPUが担当するかを表す情報を有する担当CPUテーブル324、

（＊）DP-LUを構成する仮想ページに関する情報を有する仮想ページ管理テーブル325、

（＊）プールを構成する実ページに関する情報を有する実ページ管理テーブル326、

（＊）未割当ての実ページを管理するキューの先頭エントリのポインタを有するキュー先頭管理テーブル327、

（＊）DKC同士の接続に関する情報を有する接続管理テーブル328、

を記憶することができる。これらのテーブル321乃至328は、共有情報管理装置301が記憶するテーブル311乃至314に比べて、頻繁にHDKC内のCPUによってアクセスされて良い。HDKC内のCPUは、そのHDKCが記憶するテーブル321乃至328にアクセスすることができる。また、HDKC内のCPUは、そのHDKCに第1の接続媒体を介して接続されているHDKC内のテーブルにアクセスできて良いが、そのHDKCに第3の接続媒体を介して接続されているHDKC内のテーブルにアクセスできないで良い。

【0106】

また、各HDKCは、図示しないが、そのHDKCが有するLUに関する情報（例えば、LU毎に、LU ID、LU種別、記憶容量）を有するLU管理テーブルを記憶して良い。LU種別としては、例えば、通常のLU（HDKC内のRGに基づくLU）、仮想L

10

20

30

40

50

U、及び、D P - L U（ダイナミックプロビジョニング（シンプロビジョニング）に従う L U）があって良い。L U I Dは、例えば、サーバが使用する識別情報である L U N（Logical Unit Number）、サーバが使用しない識別情報（ボリューム I D）、又はそれらの組合せで良い。

【0107】

H D K CがD P - L Uを有しない場合、そのH D K C内の記憶資源は、テーブル325乃至327を記憶しないで良い。

【0108】

テーブル321乃至328の全てが、同一の記憶資源（例えば、C M（S M領域））に記憶されても良いし、テーブル321乃至328が、複数の記憶資源（例えば、L M及びC M（S M領域））に分散していても良い。また、H D K Cユニットにおいて、或るH D K C内の記憶資源が、テーブル321乃至328の一部を記憶し、別のH D K C内の記憶資源が、テーブル321乃至328の別の一部を記憶しても良い。

【0109】

また、L D K Cの記憶資源（例えば、L M、C M（S M領域））も、そのL D K Cにとって必要な管理情報、例えば、L U / P D E V変換テーブル（L D K C内のL Uと、L D K C内のP D E Vとの関係を表す情報を有するテーブル）を記憶して良い。また、L D K C内の外部L Uが、D P - L Uであれば、そのL D K C内の記憶資源は、仮想ページ管理テーブル及び実ページ管理テーブルを記憶することができる。

【0110】

D K Cが、複数のプールを有する場合、そのD K Cには、実ページ管理テーブルが、プール毎にあって良い。なお、H D K Cユニット141は、複数の内部ボリューム（自装置内の1つ又は複数の記憶媒体から提供される記憶領域を元にして構成されるボリューム）及び／又は論理ボリューム（他のH D K Cユニット141やL D K C151から提供されるL U）の複数の記憶領域を、プールとして一元的に管理している。

【0111】

以下、幾つかの管理情報を、図面を参照して詳細に説明する。

【0112】

<接続管理テーブル328>。

【0113】

図8は、接続管理テーブル328（例えば#H0）の構成を示す。

【0114】

接続管理テーブル#H0は、H D K C # H 0及びそれに接続されている各D K Cについて、例えば、以下の情報、

（＊）D K Cの識別子であるD K C I D 801、

（＊）H D K C # H 0に接続する媒体の種類を表す接続媒体802、

（＊）D K Cの種別（性能及び／又は機能）を表すD K C種別803、

（＊）H D K C # H 0への接続の形態を表す接続形態804、

を有する。このテーブル#H0が有する各情報は、管理者によって設定されても良いし、制御プログラム（C P U # H 0が実行するコンピュータプログラム）が行う処理（図16参照）によって設定されても良い。

【0115】

図8によれば、H D K C # H 0に対応した接続媒体802及び接続形態804がそれぞれは、それらに対応するD K CがH D K C # H 0（接続管理テーブル#H0に対応したD K Cそれ自体）であることを意味する「自装置」を表す。

【0116】

また、図8によれば、H D K C # H 0に、第1の接続形態でH D K C # H 1が接続されていて、第2の接続形態でL D K C # L 0が接続されていて、第3の接続形態でH D K C # H 3が接続されていることがわかる。

【0117】

10

20

30

40

50

＜ティア管理テーブル 3 2 1＞。

【0 1 1 8】

図 9 A は、ティア管理テーブル 3 2 1（例えば # H 0）の構成を示す。

【0 1 1 9】

ティア管理テーブル # H 0 は、L U 毎に、例えば、下記の情報、

(*) L U の基になっている R G の識別子である R G I D 9 0 1、

(*) L U の識別子である L U I D 9 0 2、

(*) どんな接続形態に従う L U であるかに対応したポイントである接続形態 9 0 3、

(*) どんな種別の P D E V の基になっている L U であるかに対応したポイントである P D E V 種別 9 0 4、

(*) 接続形態 9 0 3 と P D E V 種別 9 0 4 に基づいて算出された値（例えば、接続形態 9 0 3 と P D E V 種別 9 0 4 の合計）であるティアレベル 9 0 5、
を有する。

【0 1 2 0】

R G I D 9 0 1 は、例えば、L U が D P - L U の場合、そのことを意味する値であって良い。

【0 1 2 1】

P D E V 種別 9 0 4 は、例えば、L U が D P - L U の場合、そのことを意味する値であっても良いし、D P - L U が関連付けられているプールを構成する 1 以上のプール L U のうちの少なくとも 1 つのプール L U に対応した P D E V 種別 9 0 4 に基づくポイントであっても良い。また、L U が仮想 L U の場合、P D E V 種別 9 0 4 は、その仮想 L U にマッピングされている外部 L U の基になっている P D E V の種別に従うポイントであって良い。

【0 1 2 2】

サーバ 1 0 1 及び／又はアプリケーション 1 1 1 は、その種類によってデータの重要度や L U（例えば、通常の L U、D P - L U、又は外部 L U がマッピングされている仮想 L U）へのアクセス特性（例えば、I / O 頻度）が異なる。また、本願の L U（例えば、通常の L U、D P - L U、又は外部 L U がマッピングされている仮想 L U）は、ティア（例えば、以下にて説明するティアレベル、又はティアレベルの範囲）の違いによって、L U のアクセス性能及び／又はコストが異なる。したがって、サーバ 1 0 1 及び／又はアプリケーション 1 1 1 の種類に応じた適切なアクセス性能及び／又はコストを実現するために、管理者及び／又はコンピュータシステム（例えば H D K C ユニット 1 4 1、H D K C 及び／又は管理装置 3 0 1）は、サーバ 1 0 1 及び／又はアプリケーション 1 1 1 の種類に応じたティア（例えば、ティアレベル、又はティアレベルの範囲）の L U（例えば、通常の L U、D P - L U、又は外部 L U がマッピングされている仮想 L U）を選択して割り当てるのが好ましい。例えば、管理者は、実稼動メールサーバやアプリケーション（例えば、電子メールサーバやアプリケーション）、その他のクリティカルなオンラインアクセスを行うサーバやアプリケーションに対して、より上位のティア（例えば、ティアレベル、又はティアレベルの範囲）に属する L U を選択して割り当てるのが好ましく、よりクリティカルな程度の低いオンラインアクセスを行うサーバやアプリケーションに対して、中位のティアに属する L U を選択して割り当てるのが好ましく、メールアーカイブ／バックアップサーバやアプリケーションに対して、より下位のティアに属する L U を選択して割り当てるのが好ましい。

【0 1 2 3】

また、同一種類のサーバ 1 0 1 やアプリケーション 1 1 1 であっても、アクセスされるデータやファイルによって重要性及び／又はアクセス特性（例えば、I / O 頻度）が異なる。したがって、データやファイルの重要度及び／又はアクセス特性に応じた適切なアクセス性能及び／又はコストを実現するために、管理者やコンピュータシステム（例えば H D K C ユニット 1 4 1、H D K C 及び／又は管理装置 3 0 1）は、データやファイルの重要度及び／又はアクセス特性に応じた適切なティア（例えば、ティアレベル、又はティア

10

20

30

40

50

レベルの範囲)のLU(例えば、通常のLU、DP-LU、又は外部LUがマッピングされている仮想LU)を選択して割り当てることが好ましい。例えば、重要性及び／又はI／O頻度の高いデータはより上位のティアに属するLUを選択して割り当てることが好ましく、重要性及び／又はI／O頻度が中程度のデータは中位のティアに属するLUを選択して割り当てることが好ましく、重要性及び／又はI／O頻度の低いデータはより下位のティアに属するLUを選択して割り当てることが好ましい。

【0124】

なお、ここで言う上位のティア及び下位のティアとは、相対的に上位又は下位のティアであっても良いし、上位のティアは、第1のティアレベル又は第1のティアレベル範囲よりも高いティアレベル又はティアレベル範囲であっても良いし、下位のティアは、第2のティアレベル又は第2のティアレベル範囲よりも低いティアレベル又はティアレベル範囲であっても良い(第2のティアレベルは、第1のティアレベル以下であっても、第2のティアレベル範囲は、第1のティアレベル範囲以下であっても良い)。

10

【0125】

このテーブル#H0によれば、RG別に、そのRGが基になっている1以上のLUのティアレベルに基づくティアレベル(例えば、1以上のLUのティアレベルを合計)を算出することで、RG毎のティアレベルを算出することも可能である。なお、RGのティアレベルは、そのRGが基になっている1以上のLUのティアレベルを利用する方法とは別の方法で、決定されても良い。

【0126】

20

本実施例において、「ティア」とは、LUのティアである。本実施例では、LUについて「ティアレベル」と呼ばれる値が算出される。複数のティアのうちの、算出されたティアレベルに対応したティアが、LUが属するティアである。ティアレベルは、例えば、LUのI／O性能(例えば、単位時間当たりに入出力できるデータのサイズ、或いは、レスポンスタイム)の大きさを表す値と言うこともできる。

【0127】

1つのティアレベルにつき1つのティアが存在しても良いし、1つのティアレベル範囲につき1つのティアが存在しても良いし、或いは、それらの組合せであっても良い。複数の同一の大きさのティアレベル範囲があっても良いし、複数の異なる大きさのティアレベル範囲があっても良い。本実施例では、ティアレベルが小さいほど、ティアは上位である。しかし、ティアレベルの大きさとティアの高さの関係は、それに限られない。例えば、ティアレベルが小さいほど、ティアは下位となっても良い。

30

【0128】

LUのティアレベルは、そのLUのI／O性能に影響する1以上のファクター(以下、性能ファクター)に依存する。具体的には、例えば、本実施例では、ティアレベルは、1以上の性能ファクターに対応した1以上のポイントを用いて算出された値(例えば合計)である。LUの1以上の性能ファクターは、図9Aの例によれば、接続形態(すなわち、HDKC#H0にどんな接続形態で接続されているDKC内のLUがマッピングされているか)、及び、PDEV種別(どんな種別のPDEVに基づくLUであるか)である。

【0129】

40

図9Aに例示した性能ファクターとそれのポイントとの対応関係は、図9Bに例示されている。

【0130】

図9Bによれば、I／O性能として有利であるほど、ポイントが小さい。このため、同じPDEV種別であっても接続形態が異なっていたり、接続形態が同じでもPDEV種別が異なっていたりすれば、ティアレベルが異なる。

【0131】

図9Bによれば、第1乃至第3の接続形態のうち、第1の接続形態、第3の接続形態、第2の接続形態の順で、I／O性能が有利である。しかし、例えば、HDKCに第2の接続形態で接続されているDKCの性能及び／又は機能が、HDKCと同じである場合、第

50

2の接続形態のポイントは、第3の接続形態のポイントと同じであっても良い。

【0132】

また、図9Bによれば、PDEV種別904は、PDEVのI/O性能に応じた値である。例えば、SSDのポイントの方がSAS-HDDのポイントよりも小さく、SAS-HDDのポイントの方がSATA-HDDのポイントよりも小さい。

【0133】

なお、PDEV種別904は、それに対応するLUの基になっているRGを構成するPDEVの種別に加えて、そのRGのRAIDレベル（更に、3D+1Pといったコンビネーションを含んで良い）を基に決定されても良い。この場合、例えば、RAID5の方がRAID3よりも性能レベルが高いので、RAID5のポイントの方がRAID3のポイントよりも小さい。また、PDEV種別904は、それに対応するLUの基になっているRGを構成する媒体の種別に代えて、そのRGのRAIDレベル（更に、3D+1Pといったコンビネーションを含んで良い）を基に決定されても良い。

【0134】

（a）接続形態（DKCの性能を含んで良い）及びPDEV種別以外の性能ファクターについては、ティアレベルを違えるほどの差が存在しない、又は、（b）接続形態（DKCの性能を含んで良い）及びPDEV種別のみを基にしてティアレベルが決定される場合、LUのティアレベルは、以下の順番に小さい。すなわち、下記（1）乃至（3）のLUのうち、（1）のLUが、ティアレベルが最も小さく、より上位のティアとする、（2）のLUが、ティアレベルが2番目に小さく、中位のティアとする、（3）のLUが、ティアレベルが最も大きく、より下位のティアとする。ここでは、下記（1）乃至（3）のLUを有するDKCが、図6に示したHDKC#H0であることを例に採る。

（1）自DKC（HDKC#H0）内のRGに基づくLUであって、自DKCからサーバに提供されるLU（例えばLU#0）、

（2）他のHDKC（例えばHDKC#HA）内のLU（例えばLU#y）が仮想化されたLUであって、自DKCからサーバに提供されるLU（例えばLU#2）、

（3）LDKC（例えばLDKC#L0）内のLU（例えばLU#z）が仮想化されたLUであって、自DKCからサーバに提供される（例えばLU#3）。

【0135】

図6において、HDKC#H0と#H1は、同一のHDKCユニットを構成しているので、上記条件（a）又は（b）が満たされる場合、LU#0と、LU#xが仮想化されたLU#1は、ティアレベルが同じである。

【0136】

また、（2）のLUの方が（3）のLUよりもティアレベルが小さい理由（I/O性能の面で有利な理由）は、（2）のLU（仮想LU）にマッピングされている外部LUを有するDKCの方が、（3）のLU（仮想LU）にマッピングされている外部LUを有するDKCよりも、性能及び／又は機能が低いからである。

【0137】

上記条件（a）又は（b）が満たされる場合、本発明は、例えば、下記（A）乃至（G）のうちの少なくとも1つのケースに適用可能であって良い。また、少なくとも1つのHDKCが、後述の特有の処理（例えば、図16に記載の処理等）を実行するため、本発明は、下記（A）乃至（G）のうちの少なくとも1つのケースに代えて又は加えて、下記（H）乃至（M）のうちの少なくとも1つのケースに適用可能であって良い。

（A）図6において、HDKC#H0が提供するLU#0、LU#3及びLU#2（上記（1）乃至（3）のLU）が、いずれも、サーバからのI/Oコマンドに従うI/O先（データのリード及び／又はライトのターゲット）となり得るケース。

（B）図6において、HDKC#H0が、LU#0のようなLU（HDKC#H0内のRGに基づくLU）を有しないケース。ケース（B）では、例えば、HDKC#H0が、LU#0～#3のうちLU#0を有しない。なお、このケース（B）及び以下に記載の少なくとも1つのケース（例えば、LU#0のようなLUを有しない少なくとも1つのケース

）において、HDKC#H0は、PDEVを有しない記憶制御装置、例えば、ストレージの仮想化機能を有するインテリジェントスイッチでも良い。

（C）図6において、HDKC#H0が、HDKC#H1に接続されていない又はHDKC#H1に接続されていたとしてもLU#1のような仮想LU（第1の接続形態で接続されているHDKC内のLUがマッピングされた仮想LU）を有しないケース。ケース（C）では、例えば、HDKC#H0が、LU#0～#3のうちLU#1を有しない。

（D）図6において、次の（d1）及び（d2）に該当するケース。（d1）HDKC#H0が、LU#0のようなLUを有しない。（d2）HDKC#H0が、HDKC#H1に接続されていない又はHDKC#H1に接続されていたとしてもLU#1のような仮想LUを有しない。ケース（D）では、例えば、HDKC#H0が、LU#0～#3のうちLU#0及び#1を有しない。

10

（E）図6において、HDKC#H0が、LDKC#L0に接続されていない又はLDKC#L0に接続されていたとしてもLU#3のような仮想LU（第2の接続形態で接続されているLDKC内のLUがマッピングされた仮想LU）を有しないケース。ケース（E）では、例えば、HDKC#H0が、LU#0～#3のうちLU#3を有しない。

（F）図6において、次の（f1）及び（f2）に該当するケース。（f1）HDKC#H0が、LU#0のようなLUを有しない。（f2）HDKC#H0が、LDKC#L0に接続されていない又はLDKC#L0に接続されていたとしてもLU#3のような仮想LUを有しない。ケース（F）では、例えば、HDKC#H0が、LU#0～#3のうちLU#0及び#3を有しない。

20

（G）図6において、次の（g1）及び（g2）に該当するケース。（g1）HDKC#H0が、HDKC#H1に接続されていない又はHDKC#H1に接続されていたとしてもLU#1のような仮想LUを有しない。（g2）HDKC#H0が、LDKC#L0に接続されていない又はLDKC#L0に接続されていたとしてもLU#3のような仮想LUを有しない。ケース（G）では、例えば、HDKC#H0が、LU#0～#3のうちLU#1及び#3を有しない。

（H）図6において、HDKC#H0が、HDKC#HAに接続されていない又はHDKC#HAに接続されていたとしてもLU#2のような仮想LU（第3の接続形態で接続されているHDKC内のLUがマッピングされた仮想LU）を有しないケース。ケース（H）では、例えば、HDKC#H0が、LU#0～#3のうちLU#2を有しない。

30

（I）図6において、次の（i1）及び（i2）に該当するケース。（i1）HDKC#H0が、HDKC#HAに接続されていない又はHDKC#HAに接続されていたとしてもLU#2のような仮想LUを有しない。（i2）HDKC#H0が、HDKC#H1に接続されていない又はHDKC#H1に接続されていたとしてもLU#1のような仮想LUを有しない。ケース（I）では、例えば、HDKC#H0が、LU#0～#3のうちLU#1及び#2を有しない。

（J）図6において、次の（j1）及び（j2）に該当するケース。（j1）HDKC#H0が、HDKC#HAに接続されていない又はHDKC#HAに接続されていたとしてもLU#2のような仮想LUを有しない。（j2）HDKC#H0が、LDKC#L0に接続されていない又はLDKC#L0に接続されていたとしてもLU#3のような仮想LUを有しない。ケース（J）では、例えば、HDKC#H0が、LU#0～#3のうちLU#2及び#3を有しない。

40

（K）図6において、次の（k1）乃至（k3）に該当するケース。（k1）HDKC#H0が、HDKC#HAに接続されていない又はHDKC#HAに接続されていたとしてもLU#2のような仮想LUを有しない。（k2）HDKC#H0が、HDKC#H1に接続されていない又はHDKC#H1に接続されていたとしてもLU#1のような仮想LUを有しない。（k3）HDKC#H0が、LDKC#L0に接続されていない又はLDKC#L0に接続されていたとしてもLU#3のような仮想LUを有しない。ケース（K）では、例えば、HDKC#H0が、LU#0～#3のうちLU#1、#2、及び#3を有しない。

50

(L) 図6において、次の(11)及び(12)に該当するケース。(11) HDKC#H0が、LU#0のようなLUを有しない。(12) HDKC#H0が、HDKC#HAに接続されていない又はHDKC#HAに接続されていたとしてもLU#2のような仮想LUを有しない。ケース(L)では、例えば、HDKC#H0が、LU#0～#3のうちLU#0及び#2を有しない。

(M) 図6において、次の(m1)及び(m2)に該当するケース。(m1) HDKC#H0が、LU#0のようなLUを有しない。(m2) HDKC#H0が、DKC#H1、#HA及び#L0のいずれか2つに接続されていない又は、DKC#H1、#HA及び#L0のいずれか2つに接続されていたとしてもLU#1～#3のうちのいずれか2つのような2つの仮想LUを有しない。ケース(M)では、例えば、HDKC#H0が、LU#0～#3のうちLU#0を有さず、且つ、LU#1～#3のうちのいずれか2つを有しない。

10

【0138】

したがって、本実施例に拠れば、以下の記憶システム、第一の記憶コントローラ及び／又は第一の記憶制御ユニットを新たに提供できる。この記憶システムは、第一の記憶コントローラ(例えば、プロセッサ222、コントローラ211又はHDKC201)又は第一の記憶制御ユニット(例えば、HDKCユニット141)と、第二の記憶コントローラ(例えば、プロセッサ222、コントローラ211又はHDKC201)又は第二の記憶制御ユニット(例えば、HDKCユニット141)と、を有する。この第一の記憶制御ユニットは、上位装置から第一のロジカルユニット(LU)宛に受信されるデータを制御する上記第一の記憶コントローラを有する。また、上記第二の記憶制御ユニットは、上記第一の記憶コントローラ又は第一の記憶制御ユニットから第二のLU宛に受信されるデータを制御する上記第二の記憶コントローラと、この第二の記憶コントローラの制御に応じてデータを格納する複数の記憶媒体(例えば、RG231)とを有する。上記第一の記憶コントローラは、上記第二の記憶コントローラの性能に応じて前記第一のLUのティアのレベルを管理し、上記第一のLUのティアのレベルに適合したタイプのアプリケーションによるデータの参照及び／又は更新などの処理に応じて上記上位装置から上記第一のLU宛に受信されるデータを、上記第二のLUに宛てて送信するように制御する。

20

【0139】

図6において、HDKC#H0がサーバへ提供するLU#0～#3の少なくとも1つは、通常のLUでも良いし、DP-LUでも良い。

30

【0140】

「通常のLU」は、LU作成時にLUの容量と等しい容量の記憶領域がRGから割り当てられたLUであって良い。通常のLUは、外部LUがマッピングされた仮想LUでも良い。

【0141】

「DP-LU」は、複数の仮想ページに分割された仮想的なLUである。未割当ての仮想ページ(実ページが割り当てられていない仮想ページ)に属するアドレスを指定したライトコマンドをDKCが受信した場合に、DKCは、その未割当ての仮想ページにプールから未割当ての実ページ(仮想ページに割り当てられていない実ページ)を割り当てることができる。プールは、例えば、1以上のRGに基づく通常のLU(プールLU)で構成されて良い。プールは、複数存在してよい。各プールLUは、2以上の実ページに分割されて良い。仮想ページは、仮想的な記憶領域であり、実ページは、プールを構成する記憶領域である。

40

【0142】

したがって、DP-LUを構成する全ての仮想ページにそれぞれ実ページが割り当てられた状態を除いて、DP-LUに割り当てられた実ページのトータル容量は、DP-LUの容量(仮想的な容量)よりも小さい。DKCは、複数の実ページをプールという形式で一元管理することができる。1つの実ページは、1以上のRGを構成する複数のPDEVに基づいていて良い。実ページに書き込まれるデータは、その実ページの基になっている

50

R GのR A I Dレベルに応じて、そのR Gを構成する複数のP D E Vに書き込まれて良い。

【0143】

なお、1つのプールに、1又は複数のD P-L Uが関連付けられ、D P-L Uには、そのD P-L Uに関連付けられているプールから実ページが割り当てられて良い。また、プールから割り当てられる実ページは、n回前（nは自然数）に割り当てられた実ページの基になっているR Gとは異なるR Gが基になっている実ページで良い（例えば、n=1で良い）。具体的には、例えば、ラウンドロビン形式で、割り当てられる実ページの基になっているR Gが切り替わって良い。これにより、特定のR GにI/Oが集中してしまうことを回避でき、以って、D K CのI/O性能の向上が期待できる。1つのプールに関連付けられるD P-L U又はD P-L Uのグループ（D P-L Uグループ）がD P-L U又はD P-L Uグループ毎に異なるティアに属している場合（例えば、第一のD P-L U又は第一のD P-L Uグループは第一のティアに属し、第二のD P-L U又は第二のD P-L Uグループは第二のティアに属している場合）、ティア別に、n回前に割り当てられた実ページの基になっているR Gと異なるR Gから実ページが割り当てられて良い（例えば、第一のD P-L U又は第一のD P-L Uグループは第一のティアを構成する複数のR Gの範囲内において、ラウンドロビン形式で、割り当てられる実ページの基になっているR Gが切り替わり、第二のD P-L U又は第二のD P-L Uグループは第二のティアを構成する複数のR Gの範囲内において、ラウンドロビン形式で、割り当てられる実ページの基になっているR Gが切り替わる）。また、ラウンドロビン形式の適用に際しては、割り当てられる実ページの基になっているR Gが1つのR G単位に切り替わることに限られず、R Gの一部を単位に又は複数のR G単位（R Gグループ単位）に切り替わることも良い。複数のR G単位に切り替わる場合、H D K Cユニット1 4 1、H D K CやL D K C 1 5 1のプロセッサ2 2 2は、ある複数のR G単位（R Gグループ単位）から1又は複数の実ページを割り当てた場合には、次に他の複数のR G単位（R Gグループ単位）から1又は複数の実ページを割り当てることにより、特定の複数のR G単位（R Gグループ単位）にI/Oが集中してしまうことを回避でき、以って、D K CのI/O性能の向上が期待できる。

【0144】

また、1又は複数のD P-L Uの各仮想ページは、各仮想ページのアクセス特性に応じて、このアクセス特性に適する記憶媒体に対応付けられた実ページにマッピングさせることもできる。又は、各仮想ページのアクセス特性に応じて、このアクセス特性に適していない記憶媒体に対応付けられた実ページから、このアクセス特性に適する記憶媒体に対応付けられた実ページにマッピングを変更することもできる。ここでいうアクセス特性とは、例えば、各仮想ページへのアクセスの種類（ランダムアクセスやシーケンスアクセスの別、又はアクセス送信元のサーバ1 0 1やアプリケーション1 1 1の上述のような種類）、各仮想ページへのI/O頻度又はライト頻度、又は各仮想ページに書き込まれたデータがキャッシュメモリ2 2 4から仮想ページにマッピングされた実ページに対応する記憶媒体へ書き込まれる頻度、又は各仮想ページに書き込まれたり読み出されたりするデータがキャッシュメモリ2 2 4と仮想ページにマッピングされた実ページに対応する記憶媒体との間で書き込まれたり読み出されたりする頻度である。この場合、本実施例でいうところの複数のティア（例えば、ティアレベル又はティアレベル範囲）は、1又は複数のD P-L Uの各仮想ページ毎に管理されることになり、上記のようにL Uのティアという技術ではなく、1又は複数のD P-L Uの各仮想ページ毎のティアという技術と捉えることになる。ただし、各仮想ページのアクセス特性に応じてこのアクセス特性に適する記憶媒体に対応付けられた実ページにマッピングさせる場合には、各仮想ページ毎のティアという技術になるが、各仮想ページのアクセス特性に応じて仮想ページにマッピングされる実ページが変更する場合には、予めL Uとティアとの関係が設定されるため、D P-L Uの作成時等にはL Uのティアという技術が適用され、その後、アクセス特性に応じた仮想ページへマッピングされる実ページの変更に従って各仮想ページ毎のティアという技術が適用されることになる。

【0145】

なお、本実施例では、サーバからのライトコマンドに応じて、このライトコマンドで指定されたアドレスが属する仮想ページに実ページが割り当てられるが、本発明は、D P - L Uの作成時等にD P - L Uの全ての領域に実ページが割り当てられていない例に限られないで良い。例えば、下記の例も、本発明に適用されて良い。下記のいずれの例でも、実際に使用される容量を削減可能である。

【0146】

(第1の例)

【0147】

D P - L Uを構成する複数の仮想ページのうちの少なくとも1つ仮想ページ(又は、仮想ページに含まれる一部の領域)に、予め(そのD P - L Uについてサーバからライトコマンドを受信する前に)、記憶領域(例えば物理ページ)が割り当てられていて良い。つまり、プレアロケーションが行われていて良い。ライトコマンドに従うデータを格納するために、既に割当済みの記憶領域では不十分であり更なる記憶領域が必要な場合、D K C(内部コントローラ)は、記憶領域をプールからD P - L Uに追加的に割り当てて良い。

【0148】

(第2の例)

【0149】

D P - L Uを構成する複数の仮想ページのうちの少なくとも1つの仮想ページに、予め(そのD P - L Uについてサーバからライトコマンドを受信する前に)、ゼロデータ(例えば、ビット値「0」で構成されたデータ)が格納されている実ページ又は制御用の領域が仮に割り当てられていて良い。つまり、仮アロケーションが行われていて良い。ゼロ領域(ゼロデータが格納された実ページ又は制御用の領域)は、例えば、S S Dなどの記憶デバイス上の領域で良い。D K C(内部コントローラ)は、ライトコマンドを受信した場合、ライト先の仮想ページへ割り当てられる記憶領域を、ゼロ領域から、データが格納されるべき実ページへ変更して良い。つまり、本アロケーションが行われて良い。D K C(内部コントローラ)は、本アロケーションが済んでいない仮想ページに属するアドレスを指定したリードコマンドをサーバから受信した場合、その仮想ページに割り当てられているゼロ領域内のゼロデータをサーバに応答して良い。

【0150】

<ティア管理テーブルの変形例>。

【0151】

ティアレベルに影響する性能ファクターとして、接続形態及びP D E V種別の少なくとも1つに代えて又は加えて、1以上の別の性能ファクターを採用することができる。以下、それら1以上の別の性能ファクターを説明する。

【0152】

図10は、拡張されたティア管理テーブル321(例えば#H0)を示す。図10に示すテーブル321は、図9Aに示したテーブル321の一変形例である。本実施例において、H D K Cが参照するテーブル321は、図9Aに示したテーブル321であっても良いし、図10に示すテーブル321であっても良いし、図9Aに示すテーブル321の別

【0153】

図10によれば、拡張されたティア管理テーブル#H0は、H D K C #H0が有するL U毎に、例えば、情報1001~1010を有する。情報1001、1002、1006は、図9Aに示した情報901、902及び904と同じである。情報1003~1005、及び1007~1010は、以下の通りである。図10に例示する性能ファクターとそれのポイントとの対応関係は、図11に例示されている。

(*) L Uが第3の接続形態に従う仮想L Uであるか否かに基づくポイントである第3の接続形態1003、

(*) L Uが第2の接続形態に従う仮想L Uであるか否かに基づくポイントである第2の

10

20

30

40

50

接続形態 1004、

(*) HDKC#H0がプレキャッシュ機能を利用しているか否かに基づくポイントであるプレキャッシュ1005、

(*) LUに対するI/Oのためにデータが経由しなければならないHDKCユニットの数(段数)に基づくポイントである、カスケードの違い1007、

(*) トータル距離(LUに対するI/Oのためにデータが経由する1以上の第2及び/又は第3の接続媒体の長さの合計)、コマンド送信時間(HDKC#H0が他のDKCに対してコマンドを発行してから他のDKCがコマンドを受信するまでにかかる時間)、コマンド応答時間(HDKC#H0が他のDKCに対してコマンドを発行してからその応答を受領するまでにかかる時間)、及び/又は、コマンド処理時間(HDKC#H0が他のDKCに対してコマンドを発行してから他のDKCがそのコマンドの処理を実行して完了応答を受領するまでにかかる時間)、に基づくポイントである、距離(コマンド応答時間)1008、

(*) 内部コントローラの性能及び/又は機能に基づくポイントであるコントローラ性能1009、

(*) 情報1003~1009に基づいて算出された値(例えば、情報1003~1009の合計)であるティアレベル1010。

【0154】

情報1003及び1004は、図9Aに示した接続形態903の拡張である。HDKC#H0が受けたI/Oコマンドから特定されるLUが仮想LUの場合、その仮想LUにマッピングされている外部LU(その外部LUも仮想LUの場合、その外部LUにマッピングされているLU)に対するI/Oのために、I/O対象のデータが、1以上の接続媒体(1以上の第1の接続媒体、1以上の第2の接続媒体、1以上の第3の接続媒体、或いは、それらのうちの2以上の異なる種類の接続媒体)を経由することがある。データが経由する接続媒体の種類(及び/又は数)によって、I/O性能が異なる。そのため、情報1003及び1004が設けられている。

【0155】

データがいずれの接続媒体も経由しない(自DKC)又は第1の接続媒体を経由する場合の第3の接続形態1003(ポイント)の方が、データが第3の接続媒体を経由する場合の第3の接続形態1003(ポイント)よりも小さい。データが第1及び第3の接続媒体の両方を経由する場合、第3の接続形態1003(ポイント)は、大きくて良い。

【0156】

データが第2の接続媒体を経由しない場合の第2の接続形態1004(ポイント)の方が、データが第2の接続媒体を経由する場合の第2の接続形態1004(ポイント)よりも、小さい。また、データが第2の接続媒体を経由する場合の第2の接続形態1004(ポイント)は、データが第3の接続媒体を経由する場合の第3の接続形態1003(ポイント)よりも大きくて良い。

【0157】

「プレキャッシュ機能」とは、リードコマンドの場合、第3の接続形態に従う仮想LU(HDKC#H0内のLU)を指定したサーバからのリードコマンドに対して、その仮想LUにマッピングされている外部LUを有するHDKCからリード対象のデータを受けることなく、HDKC#H0が有するCM#H0内のデータをサーバに返す機能である。また、ライトコマンドの場合、第3の接続形態に従う仮想LU(HDKC#H0内のLU)を指定したサーバからのライトコマンドに対して、その仮想LUにマッピングされている外部LUを有するHDKCに対してライト対象のデータを書き込むことなく、HDKC#H0が有するCM#H0内にデータを書き込むことに応じてサーバにライト処理完了の応答を返す機能である。プレキャッシュ機能が利用されれば、ライト性能及び/又はリード性能が高い。このため、プレキャッシュ機能が有る場合のプレキャッシュ1005(ポイント)の方が、プレキャッシュ機能が無い場合のプレキャッシュ1005(ポイント)よりも、小さい。

10

20

30

40

50

【0158】

LUのI/O性能（例えばリード性能）は、I/Oのためにデータが経由するHDKCユニットの数（段数）が多いほど、低下すると考えられる。このため、データが経由するHDKCユニットの数（段数）が多いほど、カスケードの違い1007（ポイント）が、大きくて良い。

【0159】

LUのI/O性能（例えばリード性能）は、トータル距離が長いほど、或いは、コマンド応答時間が長いほど、低いと考えられる。このため、トータル距離が長いほど（及び／又は、コマンド応答時間が長いほど）、距離（コマンド応答時間）1009（ポイント）は大きくて良い。

10

【0160】

LUの1以上の性能ファクターを基にそのLUのティアレベルを決定するためには、図9B又は図11に示したような情報（すなわち、性能ファクターの値に対応したポイントを性能ファクターの種別毎に表す情報）を管理する必要がある。しかし、記憶システム110に存在する全DKCに関する情報を各HDKCが管理することとした場合、各HDKCが多く情報を管理することとなり、且つ、情報の更新にもオーバーヘッドが大きくなると考えられる。

【0161】

このため、全HDKCが共有する方が好ましい情報（共有情報）と、HDKCが個々に管理する方が好ましい情報（個別情報）に分けて良い。共有情報は、前述したように、共有情報管理装置301（例えば、特定のHDKC、サーバ、或いは、HDKCの管理システム（1以上の管理計算機で構成されたシステム））が記憶して良い。HDKCは、共有情報が必要な場合にのみ、共有情報管理装置301から必要な共有情報を取得して良い。

20

【0162】

例えば、図6において、HDKC#H0とHDKC#HA間の距離が、HDKC#H0とLDKC#L0間の距離に比べて長い場合、LDKC#L0内の外部LUにマッピングされた仮想LUのI/O性能は、HDKC#HA内の外部LUにマッピングされた仮想LUのI/O性能と同じかそれ以上となるケースもあり得る。拡張されたティア管理テーブル321によれば、そのようなケースでも、適切なティアレベルを算出することができる。

30

【0163】

また、例えば、図6において、HDKC#H0とHDKC#HA間の距離が、HDKC#H0とHDKC#H1間の距離に比べてあまり長くなく、且つ、HDKC#HAがHDKC#H1に比べて非常に高性能及び／又は高機能である場合、HDKC#HA内の外部LUにマッピングされた仮想LUのI/O性能は、HDKC#H1内の外部LUにマッピングされた仮想LUのI/O性能と同じかそれ以上となるケースもあり得る。拡張されたティア管理テーブル321によれば、そのようなケースでも、適切なティアレベルを算出することができる。

【0164】

また、ティア管理テーブルの変形例は、図10に示した例に限られない。例えば、ティア管理テーブルは、接続形態（DKCの性能を含んで良い）903及びPDEV種別904という基本項目の他に、プレキャッシュ1005、カスケードの違い1007、距離（コマンド応答時間）1008、コントローラ性能1009の全てを含んで良い。

40

【0165】

また、プレキャッシュ1005、カスケードの違い1007、及び距離（コマンド応答時間）1008のうちの1又は2つだけを基に、ティアレベルが算出されても良い。

【0166】

また、プレキャッシュ1005、カスケードの違い1007及び距離（コマンド応答時間）1008のうちの少なくとも1つ（例えばいずれか1つ）と、PDEV種別904との組み合わせのみで、ティアレベルが算出されても良い。

50

【0167】

また、プレキャッシュ1005、カスケードの違い1007及び距離（コマンド応答時間）1008のうちの少なくとも1つ（例えばいずれか1つ）と、接続形態903、コントローラ性能1009及びPDEV種別904のうちの少なくとも1つ（例えばいずれか1つ）との組み合わせだけで、ティアレベルが算出されても良い。

【0168】

さらに、どのようにティアレベルが算出される場合でも、本発明は、前述したケース（A）乃至（M）の少なくとも1つに適用されて良い。例えば、ケース（M）において、距離（コマンド応答時間）1008のみを基にティアレベルが算出されても、距離（コマンド応答時間）1008に対応した適切なティアレベルのLU#1、#2、又は#3の少なくとも1つのような仮想LUを、データのリード及び／又はライトのターゲットとしてサーバに提供することができる。

10

【0169】

また、例えば、或るHDKCと他の複数のHDKCとを接続する場合に、当該或るHDKC内のプロセッサ（又は、例えば、当該或るHDKCに接続された管理システム内のプロセッサ）は、当該他の複数のHDKC間の性能の違いや、当該或るHDKCから他のHDKCまでの距離に応じて、他のHDKC毎に、LUのティアレベル等の属性を適切に管理することができて良い。

【0170】

また、例えば、或るHDKCと複数のLDKCとを接続する場合に、当該或るHDKC内のプロセッサ（又は、例えば、当該或るHDKCに接続された管理システム内のプロセッサ）は、当該複数のLDKC間の性能の違いや、当該或るHDKCからLDKCまでの距離に応じて、LDKC毎に、LUのティアレベル等の属性を適切に管理することができて良い。

20

【0171】

また、例えば、性能ファクターとして、下記の少なくとも1つが採用されて良い。

（＊）LUがDP-LUであるか否か。

（＊）そのDP-LUに関連付けられているプールに関する属性（例えば、プールを構成する内部ボリュームが属するティア（例えばティアレベル又はティアレベル範囲）、内部ボリュームについてのPDEV種別）。

30

【0172】

<LU/PDEV変換テーブル323>。

【0173】

図12は、LU/PDEV変換テーブル323（例えば#H0）の構成を示す。

【0174】

LU/PDEV変換テーブル#H0は、HDKC#H0が有するLU（典型的には通常のLU）毎に、例えば、下記の情報、

（＊）LUの識別子であるLU ID1201、

（＊）LUが基になっているRGの識別子であるRG ID1202、

（＊）LUが基になっているRGのRAIDレベル（及びコンビネーション）を表すRAIDレベル（コンビネーション）1203、

40

（＊）LUの記憶容量1204、

（＊）LUが基になっているRGの記憶領域におけるLUスタートアドレス（そのLUのスタートのアドレス）を表すスタートアドレス1205、

（＊）LUが基になっているRGを構成するPDEVの種別を表すPDEV種別1206、

（＊）LUが基になっているRGを構成する1以上のPDEVの識別子のリストであるPDEV ID1207、

を有する。

【0175】

50

<担当CPUテーブル324>。

【0176】

図31は、担当CPUテーブル324（例えば#H0）の構成を示す。

【0177】

担当CPUテーブルH0は、HDKC#H0（又は、HDKC#H0を含んだHDKCユニット）が有するLU毎に、例えば、下記の情報、

（＊）LUに対するI/Oコマンドの処理を担当するCPU（担当CPU）の識別子を表す担当CPU ID3101、

（＊）LUの担当CPUの使用率（負荷の高さ）を表すCPU使用率3102、

（＊）LUの識別子であるLU ID3103、

を有する。CPU使用率3102は、例えば、定期的に又は不定期的に、各CPU#H0又は特定のCPU#H0によって更新されて良い。

【0178】

<仮想ページ管理テーブル325>。

【0179】

図13は、仮想ページ管理テーブル325（例えば#H0）の構成を示す。

【0180】

仮想ページ管理テーブル#H0は、HDKC#H0が有する仮想ページ毎に、例えば、以下の情報、

（＊）仮想ページを有するDP-LUの識別子であるDP-LU ID1301、

（＊）仮想ページのアドレスを表す仮想ページアドレス1302、

（＊）仮想ページに物理ページが割り当てられているか否かを表す割当状態1303、

（＊）仮想ページに割り当てられている実ページの基になっているRG（プールRG）の識別子であるプールRG ID1304、

（＊）仮想ページに割り当てられている実ページのアドレスを表す実ページアドレス1305、
を有する。

【0181】

<実ページ管理テーブル326>。

【0182】

図14Aは、実ページ管理テーブル326（例えば#H0）の構成を示す。

【0183】

実ページ管理テーブル#H0は、HDKC#H0が有する実ページ毎に、例えば、以下の情報、

（＊）実ページに関する情報を有する行へのアドレスを表すテーブルポインタ1401、

（＊）実ページの基になっているプールRGの識別子であるプールRG ID1402、

（＊）実ページのアドレスを表す実ページアドレス1403、

（＊）実ページが仮想ページに割り当てられているか否かを表す割当状態1404、

（＊）実ページの割当先の仮想ページを有するDP-LUの識別子であるDP-LU1405、

（＊）別の未割当て実ページに関する行のテーブルポインタ1401を表す未割当キューポインタ1406、
を有する。

【0184】

未割当ての実ページは、キュー（以下、未割当キュー）で管理されている。未割当キューポインタ1406は、そのポインタ1406に対応する実ページの次の未割当て実ページに対応した行のテーブルポインタ1401を表す。CPU#H0（制御プログラム#H0）は、仮想ページに割り当てる実ページとして、未割当キューの先頭の実ページ（図14Bに示すキュー先頭管理テーブル327（#H0）が有するポインタと一致するテーブルポインタ1401に対応した実ページ）を見つける。

10

20

30

40

50

【0185】

なお、未割当キューにおいて、複数の未割当実ページは、内部ボリューム（RG）がラウンドロビンとなるように並んで良い（図14Aの複数の未割当キューポインタ1406を参照）。全ての内部ボリュームに対して均等にI/Oが行われ、それにより、特定の内部ボリュームにのみI/Oが集中することを回避することができる。

【0186】

<ダイナミックプロビジョニング（シンプロビジョニング）>。

【0187】

図15は、ダイナミックプロビジョニング（シンプロビジョニング）が適用されたHDKC（#H0）を示す。

10

【0188】

HDKC#H0は、図2にも示したように、CPU#H00とCPU#H01を有する。CPU#H00に、LM#H00が接続されており、CPU#H01に、LM#H01が接続されている。また、CPU#H00及び#H01のいずれも、CM#H0にアクセスできる。CM#H0は、図2に示すCM#H00及び／又は#H01である。

【0189】

プール1500（#H00）は、複数の内部ボリュームで構成されている。ここで言う「内部ボリューム」とは、物理的な記憶容量が存在しているボリュームのことであり、例えば、通常のLU、第2の接続形態に従う仮想LU、或いは、RGの一部又は全部で良い。また、内部ボリュームは、サーバからアクセスされない（サーバに認識されない）。この説明では、内部ボリュームがRGであるとする。

20

【0190】

DP-LUは、通常のLUと同様、サーバからアクセスされるLUである。DP-LUについて、担当のCPUがいる。図15に示す例によれば、DP-LU#60の担当は、CPU#H00であり、DP-LU#61の担当は、CPU#H01である。

【0191】

LM#H00は、仮想ページ管理テーブル325（#H00）を記憶し、同様に、LM#H01は、仮想ページ管理テーブル325（#H01）を記憶する。仮想ページ管理テーブル#H00（#H01）は、頻繁にアクセスされるため、担当CPU#H00（#H01）のみが参照及び更新し、テーブルへのアクセスについての排他制御が不要であることが好ましい。仮想ページ管理テーブル#H00は、CPU#H00が担当するDP-LUに関する情報のみを有し、仮想ページ管理テーブル#H01は、CPU#H01が担当するDP-LUに関する情報のみを有する。

30

【0192】

CM#H0は、実ページ管理テーブル326（#H0）及びキュー先頭管理テーブル327（#H0）を記憶する。この例では、CPU#H00が担当するDP-LU#60とCPU#H01が担当するDP-LU#61の両方にプール#H00が関連付けられているので、実ページ管理テーブル#H0は、全てのCPU（この例では、CPU#H00及び#H01）から参照及び更新可能である必要がある。そのため、LM#H00及びH01ではなく、CM#H0が、実ページ管理テーブル#H0を記憶する。なお実ページ管理テーブル#H0については、排他制御が行われる。

40

【0193】

<実施例1で行われる処理>。

【0194】

以下、実施例1で行われる種々の処理を説明する。

【0195】

<<DKC拡張処理>>。

【0196】

図6に示したように、HDKC#H0に、第1の接続媒体（内部バス）227を介してHDKC#H01を接続し、第3の接続媒体（外部バス）601を介してHDKC#HA

50

を接続し、且つ、第2の接続媒体（外部バス）501を介してLDKC#L0を接続することができる。HDKC#H0に新たにDKCが接続された場合、HDKC#H0が有する管理情報が更新される。管理情報は、管理者によって更新されても良いが、本実施例では、図16に示すように、HDKC#H0によって更新される。

【0197】

図16は、HDKC#H0が行うDKC拡張処理の流れを示す。

【0198】

DKC拡張処理は、HDKC#H0に第1乃至第3の接続形態のいずれかに従って新たにDKCが接続されたことを契機に開始されて良い。また、DKC拡張処理は、内部コントローラ#H0（具体的には、例えば、HDKC#H0内の複数のCPUのうちの1以上のCPU）によって行われて良い。図16の説明において、新たに接続されたDKCを、「増設DKC」と言う。

10

【0199】

S1601で、内部コントローラ#H0が、HDKC#H0と増設DKCとを接続する媒体が、外部バスであるか否かを判断する。その接続媒体の種類は、管理者から入力される情報を基に特定されても良いし、その接続媒体が接続されたポートからの電気信号を基に検出されても良い。なお、S1601で、内部コントローラ#H0は、増設DKCの性能及び／又は機能を表す情報を取得して良い。その情報は、管理者から入力されても良いし、増設DKCから受信しても良い。

20

【0200】

S1601の判断の結果が否定的の場合（つまり接続媒体が内部バスの場合）（S1601：No）、ステップ1602で、内部コントローラ#H0は、接続管理テーブル#H0を更新する。具体的には、例えば、内部コントローラ#H0は、接続管理テーブル#H0に、増設DKCのID801と、接続媒体802「内部バス」と、DKC種別803「HDKC」と、接続形態804「第1接続」とを登録する。

【0201】

次に、S1603で、内部コントローラ#H0は、増設DKC（HDKC）内のLUを認識する。例えば、内部コントローラ#H0は、増設DKC内のメモリ（例えばLM又はCM）が記憶するLU管理テーブル（図示せず）を基に認識しても良い。また、増設DKCがLUを有していなければ、増設DKCに新たにLUが作成され、LUが作成された後に、増設DKCが有するLUが認識されて良い。

30

【0202】

次に、S1604で、内部コントローラ#H0は、担当CPUテーブル#H0を更新する。具体的には、例えば、内部コントローラ#H0は、S1603で認識されたLUのIDを、そのLUを担当するCPUのIDに関連付ける。認識されたLUの担当CPUは、HDKC#H0内のCPUであっても良いし、増設DKC内のCPUであっても良い。また、認識されたLUを担当するCPUは、管理者によって指定されても良いし、CPU#H0が任意に決定したCPUでも良い。

【0203】

さて、S1601の判断の結果が肯定的の場合（つまり接続媒体が外部バスの場合）（S1601：Yes）、ステップ1605で、内部コントローラ#H0は、増設DKCがHDKCか否かを判断する。その判断は、例えば、増設DKCの性能及び／又は機能を表す情報を基に行われて良い。その情報は、管理者から入力されても良いし、増設DKCから受信しても良い。

40

【0204】

S1605の判断の結果が否定的の場合（つまり増設DKCがLDKCの場合）（S1605：No）、ステップ1606で、内部コントローラ#H0は、接続管理テーブル#H0を更新する。具体的には、例えば、内部コントローラ#H0は、接続管理テーブル#H0に、増設DKCのID801と、接続媒体802「外部バス」と、DKC種別803「LDKC」と、接続形態804「第2接続」とを登録する。

50

【0205】

次に、S1607で、内部コントローラ#H0は、増設DKC（LDKC）内のLUを認識する。例えば、内部コントローラ#H0は、増設DKCに1以上の照会コマンド（例えば、レポートLUNコマンド、inquiryコマンド）を送信することで、増設DKCから、増設DKCが有するLUに関する情報を受信して良い。増設DKCがLUを有していなければ、増設DKCに新たにLUが作成され、LUが作成された後に、増設DKCが有するLUが認識されて良い。

【0206】

次に、S1608Aで、内部コントローラ#H0は、認識されたLUがマッピングされた仮想LUに関する情報（仮想LUのID、増設DKCのID、及び、認識されたLUのIDの組合せ）を、LUマッピングテーブル#H0に登録することができる。

10

【0207】

次に、S1608Bで、内部コントローラ#H0は、担当CPUテーブル#H0を更新する。具体的には、例えば、内部コントローラ#H0は、S1607で認識されたLUのIDを、そのLUを担当するCPUのIDに関連付ける。認識されたLUの担当CPUは、HDKC#H0内のCPUであっても良いし、HDKC#H0を含んだHDKCユニット内内の別のCPUであっても良い。また、認識されたLUを担当するCPUは、管理者によって指定されても良いし、CPU#H0が任意に決定したCPUでも良い。

【0208】

S1605の判断の結果が肯定的の場合（つまり増設DKCがHDKCの場合）（S1605：Yes）、ステップ1609で、内部コントローラ#H0は、接続管理テーブル#H0を更新する。具体的には、例えば、内部コントローラ#H0は、接続管理テーブル#H0に、増設DKCのID801と、接続媒体802「外部バス」と、DKC種別803「HDKC」と、接続形態804「第3接続」とに登録する。

20

【0209】

次に、S1610で、内部コントローラ#H0は、増設DKC（HDKC）内のLUを認識する。例えば、内部コントローラ#H0は、増設DKCに1以上の照会コマンド（例えば、レポートLUNコマンド、inquiryコマンド）を送信することで、増設DKCから、増設DKCが有するLUに関する情報を受信して良い。増設DKCがLUを有していなければ、増設DKCに新たにLUが作成され、LUが作成された後に、増設DKCが有するLUが認識されて良い。

30

【0210】

次に、S1611Aで、内部コントローラ#H0は、認識されたLUがマッピングされた仮想LUに関する情報（仮想LUのID、増設DKCのID、及び、認識されたLUのIDの組合せ）を、LUマッピングテーブル#H0に登録することができる。

【0211】

次に、S1611Bで、内部コントローラ#H0は、担当CPUテーブル#H0を更新する。具体的には、例えば、内部コントローラ#H0は、S1610で認識されたLUのIDを、そのLUを担当するCPUのIDに関連付ける。認識されたLUの担当CPUは、HDKC#H0内のCPUであっても良いし、HDKC#H0を含んだHDKCユニット内内の別のCPUであっても良い。また、認識されたLUを担当するCPUは、管理者によって指定されても良いし、CPU#H0が任意に決定したCPUでも良い。

40

【0212】

S1604、S1608B又はS1611Bの後、S1612が行われる。S1612で、内部コントローラ#H0は、ティア管理テーブル#H0を更新する。具体的には、例えば、内部コントローラ#H0は、ティア管理テーブル#H0（例えば、図9A又は図10に示したテーブル321）に、増設DKCのIDと、ステップ1603、S1607又はS1610で認識されたLUのIDとを登録する。また、内部コントローラ#H0は、ティア管理テーブル#H0が有する各種の性能ファクター項目について、認識されたLU毎に、下記、

50

(*) 各種の性能ファクターを特定する、
 (*) 特定した性能ファクターを基に、ポイント決定ルール情報（例えば、図 9 B 又は図 1 1 に示したテーブル）に従い、ポイントを算出する、
 (*) 算出したポイントの合計（ティアレベル）を算出する、
 (*) 算出されたポイントと、ティアレベルとを、ティア管理テーブル # H 0 に登録する、
 を行う。つまり、認識された L U 毎に、ティアレベルが算出される。

【0 2 1 3】

なお、本実施例においては、図 1 6 に示す処理を H D K C # H 0 が実施することを記載したが、本実施例は、この場合に限られるものでない。本実施例におけるいずれの H D K C ユニット 1 4 1 及び／又は H D K C も他の H D K C ユニット 1 4 1、H D K C 及び／又は L D K C 1 5 1 の通常の L U、D P - L U 及び／又は仮想 L U をマッピングした D P - L U 及び／又は仮想 L U を提供することができるので、いずれの H D K C ユニット 1 4 1 及び／又は H D K C においても、図 1 6 に示す処理が実施されることが良い。それに伴い、図 9 及び／又は図 1 0 の Tier 管理テーブルも各々の H D K C ユニット 1 4 1 及び／又は H D K C によって管理されることになる。

【0 2 1 4】

< I / O コマンド処理 >。

【0 2 1 5】

図 1 7 乃至図 3 0 を参照して、I / O コマンド処理を説明する。

【0 2 1 6】

図 1 7 に示すように、S 1 7 0 1 で、H D K C # H 0 の内部コントローラ # H 0 （例えば、F - I / F # H 0 （例えば、F - I / F # H 0 0、# H 0 1、# H 1 0 又は # H 1 1））が、サーバから I / O コマンドを受信する。I / O コマンドは、I / O 先の L U に関する情報、例えば、L U N と、I / O 先情報を有する。I / O 先情報は、例えば、I / O 先の領域のアドレス（例えば、L B A (Logical Block Address)）を有する。I / O コマンドがライトコマンドの場合、I / O 先情報は、ライト対象のデータ長を表す情報を含んで良い。内部コントローラ # H 0 （例えば、F - I / F # H 0、及び／又は、C P U # H 0）は、I / O コマンドから、I / O 先の L U と、I / O 先の領域とを特定することができる。

【0 2 1 7】

次に、S 1 7 0 2 で、内部コントローラ # H 0 （例えば、S 1 7 0 1 で I / O コマンドを受信した F - I / F # H 0））が、ティア管理テーブル # H 0 を参照する。内部コントローラ # H 0 は、I / O 先 L U に対応した接続形態 9 0 3 （又は、第 3 の接続形態 1 0 0 3 及び第 2 の接続形態 1 0 0 4）を基に、I / O 先 L U について、接続形態を特定する。言い換えれば、I / O 先 L U が、いずれの接続形態に従う L U であるかが特定される。具体的には、例えば、下記のように特定されて良い。

(*) 接続形態 9 0 3 （又は、第 3 の接続形態 1 0 0 3 及び第 2 の接続形態 1 0 0 4）が、自 D K C 又は第 1 接続に対応したポイントである場合、接続形態が、「自 D K C」又は「第 1 の接続形態」であると特定されて良い。具体的には、例えば、接続形態 9 0 3 が「0」、又は、第 3 の接続形態 1 0 0 3 が「0」であり且つ第 2 の接続形態 1 0 0 4 が「0」である場合、接続形態が、「自 D K C」又は「第 1 の接続形態」であると特定されて良い。

(*) 接続形態 9 0 3 （又は、第 3 の接続形態 1 0 0 3 及び第 2 の接続形態 1 0 0 4）が、第 2 接続に対応したポイントである場合、接続形態が、「第 2 の接続形態」であると特定されて良い。具体的には、例えば、接続形態 9 0 3 が「5」、又は、第 3 の接続形態 1 0 0 3 が「0」であり且つ第 2 の接続形態 1 0 0 4 が「5」である場合、接続形態が、「第 2 の接続形態」であると特定されて良い。

(*) 接続形態 9 0 3 （又は、第 3 の接続形態 1 0 0 3 及び第 2 の接続形態 1 0 0 4）が、第 3 接続に対応したポイントである場合、接続形態が、「第 3 の接続形態」であると特

10

20

30

40

50

定されて良い。具体的には、例えば、接続形態 903 が「1」である場合、接続形態が、「第3の接続形態」であると特定されて良い。また、例えば、接続形態 903 が「1」、又は、第3の接続形態 1003 が「1」であり且つ第2の接続形態 1004 が「0」である場合、接続形態が、「第3の接続形態」であると特定されて良い。

【0218】

接続媒体が「自DKC」又は「第1の接続媒体」の場合（S1703：Yes）、図19に示す処理が行われる。

【0219】

接続媒体が「第2の接続媒体」の場合（S1703：No、S1704：Yes）、図22に示す処理が行われる。

【0220】

接続媒体が「第3の接続媒体」の場合（S1703：No、S1704：No）、図24に示す処理が行われる。

【0221】

＜＜第1の接続形態に従うLUに対するI/O＞＞。

【0222】

これについて、図18～図21を参照して説明する。

【0223】

図18は、第1の接続形態に従うLUに対するI/Oの処理の流れを示す。図19は、第1の接続形態に従う構成の概要の一例を示す。図18の説明では、適宜、図19を参照する。なお、図19の一点鎖線は、データの流れの一例を示す。

【0224】

前述したように、HDKCユニット内の1つのHDKCは、HDKCユニット内の他のいずれのHDKC内の資源を利用することができる。

【0225】

例えば、図19に示すように、HDKC#H0とHDKC#H1を接続する媒体は、内部バス227である。図19の例によれば、サーバ#0とLU#40間において、I/O対象のデータは、一点鎖線で示すように、CM#H1を経由する。LU#40の担当CPUは、CPU#H1である。すなわち、HDKC#H0が受けた、LU#40に対するデータが、F-I/F#H0からCM#H1に転送され、CPU#H1が、そのデータを、CM#H1からLU#40に書き込む。しかし、I/O対象のデータは、CM#H0を経由しても良いし、LU#40の担当CPUは、CPU#H0でも良い（この場合、CPU#H0が、LU#40に対するデータを、内部バス227を介して、LU#40に書き込むことができる）。また、CMとサーバ間でのデータのI/Oを行うCPUと、CMとLU間でのデータのI/Oを行うCPUは、同じでも異なっても良い。

【0226】

LM#H0（#1）、及び／又は、CM#H0（#H1）の一部が、ティア管理テーブル321、LUマッピングテーブル322、LU/PDEV変換テーブル323、担当CPUテーブル324を記憶している。CPU#H0及び#H1は、これらのテーブルにアクセス可能である。

【0227】

図18に示すように、S1801で、F-I/F#H0（図17のS1701でI/Oコマンドを受信したF-I/F）が、I/O先LUの担当CPU#H1（第1の接続形態で接続されているHDKC内のCPU）を、担当CPUテーブル#H0（図31参照）から特定する。F-I/F#H0は、特定した担当CPU#H1に、I/Oコマンドを受信したことを通知する。ここで、そのI/Oコマンドが、内部バス227を介して担当CPU#H1に転送されて良い。

【0228】

S1802で、担当CPU#H1が、そのI/Oコマンドがライトコマンドか否かを判断する。

10

20

30

40

50

【0229】

S1802の判断の結果が否定的の場合（I/Oコマンドがリードコマンドの場合）（S1802：No）、S1803で、担当CPU#H1が、I/O先LUの種類に応じたリード処理を行う。具体的には、例えば、I/O先LUが、通常のLUの場合、担当CPU#H1は、下記、

（＊）リード元論理領域（I/O先LU#40における、リードコマンドが指定するアドレスが属する領域）に対応する物理領域（RGにおける領域）を、LU/PDEV変換テーブル#H1を基に特定する、

（＊）特定した物理領域（その物理領域を有するPDEV）から、リード対象のデータを読み出し、リード対象データを、CM#H1に書き込む、

（＊）CM#H1内のリード対象データを、内部バス227及びF-I/F#H0を通じて、サーバ#0に送信する、

を行う。I/O先LUが、DP-LUの場合、S1803で、担当CPU#H1は、図20に示すリード処理を行う。

【0230】

S1802の判断の結果が肯定的の場合（I/Oコマンドがライトコマンドの場合）（S1802：Yes）、S1811で、担当CPU#H1がF-I/Fに指示して、ライト対象データを、CM#H1に書き込ませる。

【0231】

次に、S1812で、担当CPU#H1が、応答（例えば、ライト完了通知）を、内部バス227及びF-I/F#H0を介して、サーバ#0に返す。

【0232】

次に、S1813で、担当CPU#H1が、I/O先LUの種類に応じたライト処理を行う。具体的には、例えば、I/O先LUが、通常のLUの場合、担当CPU#H1は、下記、

（＊）ライト先論理領域（I/O先LU#40における、ライトコマンドが指定するアドレスが属する領域）に対応する物理領域（RGにおける領域）を、LU/PDEV変換テーブル#H1を基に特定する、

（＊）特定した物理領域（その物理領域を有するPDEV）に、CM#H1内のライト対象データを書き込む、

を行う。I/O先LUが、DP-LUの場合、S1813で、担当CPU#H1は、図21に示すライト処理を行う。

【0233】

なお、自DKCに従うLU（HDKC#H0以外のDKC内のLUからマッピングされていないLU）に対するI/Oの処理（その処理を、この段落において、「処理A」と言う）は、第1の接続形態に従うLUに対するI/Oの処理（その処理を、この段落において、「処理B」と言う）とほとんど同じで良い。処理Aと処理Bでは、担当CPUと、使用されるCMが異なることがあり得る。例えば、処理Aでは、担当CPUは、CPU#H0で、使用されるCMは、CM#H0で良い。しかし、処理Aにおいて、担当CPUがCPU#H1であり、及び／又は、使用されるCMがCM#H1でも良い。逆に、処理Bにおいて、担当CPUがCPU#H0であり、及び／又は、使用されるCMがCM#H0で良い。

【0234】

また、LDKCは、I/OコマンドをHDKCから受けた場合、図19に示す処理と同様の処理を行うことで、外部LUに対するI/Oを行って良い。

【0235】

第1の接続形態（及び／又は、1つのHDKC）によれば、LM#H00（#H10）は、それを有するCPU#H00（#H10）のみが必要とする情報を有した管理情報を記憶して良い。CPU#H00（#H10）は、LM#H00（#H10）内の管理情報を更新しても、他のLM#H10（#H00）内の管理情報を更新しないで良い。CPU

10

20

30

40

50

#H00 (#H10)は、他のLM#H10 (#H00)内の管理情報が必要な場合、内部バス227を介して、他のLM#H10 (#H00)内の管理情報にアクセスして良い。

【0236】

或いは、第1の接続形態（及び／又は、1つのHDKC）によれば、1つのHDKCユニット（及び／又は、1つのHDKC）に存在する全てのLMが、同じ内容の管理情報を記憶して良い。この場合、いずれかのLMの管理情報がいずれかのCPUによって更新される度に、残り全てのLM内の管理情報が少なくとも1以上のCPUによって更新されて良い。

【0237】

米国においては、第1の接続形態に関して、米国特許第7587551号明細書（Fuji bayashi et al.）及び／又は、米国公開特許第2008/0263190（Serizawa et al.）の明細書及び図面を援用する。これらのうちの少なくとも1つの公報に記載のストレージ装置が、HDKC、及び／又は、HDKCユニットに適用されて良い。但し、本実施例に従う「ティア」という概念は、本願に特有の概念である。

【0238】

<<<DP-LUに対するI/O>>>。

【0239】

I/O先LUが第1の接続形態（又は自DKC）に従うLUの場合、DP-LUは、I/O先LUで良い。

【0240】

I/O先LUが第2の接続形態に従う仮想LUの場合、DP-LUは、I/O先LUにマッピングされている外部LU（例えば、図22（後述）のLU#1）で良い。

【0241】

I/O先LUが第3の接続形態に従う仮想LUの場合、DP-LUは、I/O先LUに直接的に又は間接的にマッピングされている外部LUで良い。I/O先LUに直接的にマッピングされているDP-LUは、例えば、1つの第3の接続媒体を介して接続されているHDKC（又はそのHDKCを含んだHDKCユニット）内のLU、例えば、図26（後述）のLU#45で良い。I/O先LUに間接的にマッピングされているDP-LUは、例えば、1つの第3の接続媒体と1以上の第2又は第3の接続媒体とを介してマッピングされているLU、例えば、図28（後述）のLU#56、又は、図30（後述）のLU#57で良い。

【0242】

図20は、DP-LUに関するリード処理の流れを示す。以下、適宜、図19を例に採り、説明する。その際、DP-LUのIDが、「40」であり、そのDP-LUを担当するCPUのIDが、「H11」とする。

【0243】

S2001で、DP-LU（#40）の担当CPU（#H11）が、下記、
（*）リードコマンドが有するアドレスに属する仮想ページ（リード元仮想ページ）を特定する、
（*）LM（#H11）内の仮想ページ管理テーブル（#H11）を基に、特定したリード元仮想ページについての割当状態1303（図13参照）が、「割当」か否かを判断する、
を行う。

【0244】

S2001の判断の結果が肯定的の場合（S2001：Yes）、S2002で、担当CPU（#H11）が、仮想ページ管理テーブル（#H11）を基に、リード元仮想ページに割り当てられている実ページ（リード元実ページ）を特定する。例えば、担当CPU（#H11）は、リード元仮想ページに対応したページアドレス1305に実ページのサイズを乗算することで、プール内部ボリューム（RG）におけるアドレス（例えば、LB

10

20

30

40

50

A) を算出して良い。

【0245】

次に、S2003で、担当CPU（#H11）が、特定したリード元実ページ（具体的には、例えば、算出されたアドレスに属する領域）から、リード対象のデータを読み出し、読み出したデータをCM（#H1）に書き込む。

【0246】

次に、S2004で、担当CPU（#H11）が、CM（#H1）内のリード対象データを、I/Oコマンドの送信元（サーバ#0）に送信する。

【0247】

なお、S2001の判断の結果が否定的の場合（S2001：No）、S2012で、担当CPU（#H11）が、CM（#H1）上にゼロデータ（例えば、未割当の仮想ページに関連付けられている記憶領域内のデータ）を生成し（S2011）、ゼロデータを、I/Oコマンドの送信元（サーバ#0）に送信する（S2012）。なお、S2001の判断の結果が否定的の場合に行われる処理は、これに限らないで良い。例えば、予めゼロデータ（他種の初期値データでも良い）を記憶している特定の領域（CM以外の領域）から、ゼロデータが、I/Oコマンドの送信元に送信されても良い。

【0248】

図21は、DP-LUに関するライト処理の流れを示す。以下、適宜、図19を例に採り、説明する。その際、DP-LUのIDが、「40」であり、そのDP-LUを担当するCPUのIDが、「H11」とする。

【0249】

S2101で、DP-LU（#40）の担当CPU（#H11）が、下記、
（*）ライトコマンドが有するアドレスに属する仮想ページ（ライト先仮想ページ）を特定する、

（*）LM（#H11）内の仮想ページ管理テーブル（#H11）を基に、特定したライト先仮想ページについての割当状態1303（図13参照）が、「割当」か否かを判断する、
を行う。

【0250】

S2101の判断の結果が肯定的の場合（S2101：Yes）、S2107で、担当CPU（#H11）が、仮想ページ管理テーブル（#H11）を基に、ライト先仮想ページに割り当てられている実ページ（ライト先実ページ）を特定する。例えば、担当CPU（#H11）は、ライト先仮想ページに対応したページアドレス1305に実ページのサイズを乗算することで、プール内部ボリューム（RG）におけるアドレス（例えば、LBA）を算出して良い。

【0251】

次に、S2108で、担当CPU（#H11）が、CM（#H1）内のライト対象データ（例えば、図18のS1811で格納されたデータ）を読み出し、読み出したデータを、特定したライト先実ページ（具体的には、例えば、算出されたアドレスに属する領域）に書き込む。

【0252】

S2101の判断の結果が否定的の場合（S2101：No）、つまり、ライト先仮想ページに実ページが割り当てられていない場合、ライト先仮想ページに実ページを割り当てる必要がある。そのため、まず、実ページ管理テーブル（#H1）にアクセスする必要がある。実ページ管理テーブル（#H1）については、アクセスの排他制御が必要である。このため、S2102で、担当CPU（#H11）は、実ページ管理テーブル（#H1）にアクセスするためのアクセス権を取得する。

【0253】

その後、S2103で、担当CPU（#H11）は、キュー先頭管理テーブル（#H1）が有するポインタと一致するテーブルポインタ1401に対応した未割当実ページを、

10

20

30

40

50

実ページ管理テーブル（＃H1）から特定する。

【0254】

次に、S2104で、担当CPU（＃H11）は、実ページ管理テーブル（＃H1）及びキュー先頭管理テーブル（＃H1）を更新する。具体的には、例えば、担当CPU（＃H11）は、下記、

（＊）S2103で特定した実ページに対応する割当状態1404を、「割当」に更新する、

（＊）S2103で特定した実ページに対応するDP-LU ID1405を、「40」（DP-LU（＃40）のID）に更新する、

（＊）S2103で特定した実ページに対応する未割当キューポインタ1406を、キュー先頭管理テーブル（＃H1）にコピーし、且つ、そのポインタ1406を、無効値「-」に更新する、

を行う。

【0255】

次に、S2105で、担当CPU（＃H11）は、S2102で取得したアクセス権を解放する。

【0256】

次に、S2106で、担当CPU（＃H11）は、仮想ページ管理テーブル（＃H11）を更新する。具体的には、例えば、担当CPU（＃H11）は、下記、

（＊）ライト先仮想ページに対応する割当状態1303を、「割当」に更新する、

（＊）ライト先仮想ページに対応するプールRG ID1304を、S2103で特定した実ページに対応するプールRG ID1402と一致するIDに更新する、

（＊）ライト先仮想ページに対応する実ページアドレス1305を、S2103で特定した実ページに対応する実ページアドレス1403と一致するアドレスに更新する、
を行う。

【0257】

その後、担当CPU（＃H11）は、前述したS2107及びS2108を行う。なお、ここでは、前述したライト先実ページは、S2103で特定された実ページ（S2106で、ライト先仮想ページに割り当てられた実ページ）である。

【0258】

以上が、DP-LUに対するI/Oの説明である。

【0259】

なお、例えば、プールは、（V）同一のティア（例えば、同一のティアレベル又は同一のティアレベル範囲）に属する内部ボリュームで構成されていても良いし、（W）複数の異なるティアに属する内部ボリュームで構成されていても良い。（V）の場合、例えば、同一の実ページ管理テーブルが、内部ボリューム毎に、内部ボリュームが属するティア（例えば、ティアレベル又はティアレベル範囲）を表す情報を有しても良い。（W）の場合、ティア毎にプールが管理されて良い（例えば、ティアに対応するプールを表す管理情報がティア毎にあって良い）。

【0260】

また、LDKC内の外部LUがDP-LUの場合、LDKC内の内部コントローラ（CPU）が、図20に従うリード処理、及び、図21に従うライト処理を行うことができる。

【0261】

また、例えば、図19において、DP-LUが、I/Oコマンドをサーバ#0から受けたHDKC#H0内のI/O先LUの場合、HDKC#H0内でのみ処理が行われ、HDKC#H0を含むHDKCユニット内の他のHDKCにI/O対象のデータ及び／又はI/Oコマンドは転送されないで良い。或いは、例えば、図19において、DP-LUが、前述した例の通り、HDKC#H1内のLU（＃40）の場合、HDKC#H0からHDKC#H1にI/Oコマンド（サーバ#0から受信したI/Oコマンド）が転送されて良

10

20

30

40

50

いし、或いは、I/Oコマンドが転送されることなく、CPU#H0が、内部バス227を介してDP-LU#40に対するI/Oを行っても良い。或いは、DP-LU#40に対するI/Oの処理が、CPU#H0及び#H1に分担されても良い。

【0262】

米国においては、DP-LUに対するI/Oに関して、米国特許第7555601号明細書（Yamagami）の明細書及び図面を援用する。この場合、Storage Control Systemは、本願のHDKCユニット141やLDKC151に対応しており、HDKCユニット141によるDP-LUに対するI/O処理、LDKCユニット151によるDP-LUに対するI/O処理に適用される。なお、この援用は、第1の接続形態のみに限らず、以下に具体的な説明のある第2及び第3の接続形態、及び、実施例2及び実施例3のDP-LU 10に対するI/Oに関しても援用する。但し、本実施例に従う「ティア」という概念等は、本願に特有の概念である

【0263】

<<第2の接続形態に従うLUに対するI/O>>。

【0264】

これについて、図22及び23を参照して説明する。

【0265】

図22は、第2の接続形態に従うLUに対するI/Oの処理の流れを示す。図23は、第2の接続形態に従う構成の概要の一例を示す。図22の説明では、適宜、図23を参照する。なお、図23の一点鎖線は、データの流れの一例を示す。 20

【0266】

図23に示すように、HDKC#H0とLDKC#L0が第2の接続媒体（外部バス）501を介して接続されている。HDKC#H0が有する内部コントローラ#H0内の記憶資源（例えば、LM#H0、又は、CM#H0の一部）が、前述したテーブル321～328を記憶することができる。これらのテーブルには、CPU#H0がアクセス可能であるものの、LDKC#L0内のCPU#L0はアクセスできないで良い。

【0267】

図22に示すように、S2201で、F-I/F#H0（図17のS1701でI/Oコマンドを受信したF-I/F）が、I/O先LU（#31）の担当CPU#H0を、担当CPUテーブル#H0（図31参照）から特定する。F-I/F#H0は、特定した担 30当CPU#H0に、I/Oコマンドを受信したことを通知する。

【0268】

次に、S2202で、担当CPU#H0が、そのI/Oコマンドがライトコマンドか否かを判断する。

【0269】

S2202の判断の結果が否定的の場合（I/Oコマンドがリードコマンドの場合）（S2202：No）、S2203で、内部コントローラ#H0（担当CPU#H0）が、リード元LU（#31）に対応した外部LU（#41）及びそれを有するLDKC（#L0）を、LUマッピングテーブル#H0から特定する。

【0270】

次に、S2204で、S2203で特定された外部LU（#41）に応じた処理が行われる。具体的には、例えば、次の処理が行われる。すなわち、担当CPU#H0が、リードコマンド（図22の説明において、第2のリードコマンド）を、特定されたLDKC（#L0）に送信する。そのリードコマンドは、例えば、特定された外部LU（#41）のIDと、S1701で受信したI/Oコマンド（図22の説明において、「第1のリードコマンド」）で指定されている第1のアドレスに従う第2のアドレス（例えば、第1のアドレスと同じアドレス）とを有する。第2のリードコマンドは、イニシエータポート及び外部バス501を介して、LDKC#L0に送信される。LDKC#L0の内部コントローラ#L0が、例えば、図18に示した処理を行う（外部LUがDP-LUの場合、更に、図20に示す処理を行う）。すなわち、内部コントローラ#L0が、外部LU（#41 40 50

）からリード対象のデータを読み出し、そのデータをCM#L0に格納し、そのデータを、CM#L0から、ターゲットポート及び外部バス501を介して、HDKC#H0に送信する。担当CPU#H0が、LDKC#L0からのリード対象データをCM#H0に書き込む。担当CPU#H0が、CM#H0内のリード対象データを、サーバ#0に送信する。

【0271】

S2202の判断の結果が肯定的の場合（I/Oコマンドがライトコマンドの場合）（S2202:Yes）、S2211で、担当CPU#H0が、ライト対象データを、CM#H0に書き込む。

【0272】

次に、S2212で、担当CPU#H0、応答（例えば、ライト完了通知）を、サーバ#0に返す。

【0273】

次に、S2213で、担当CPU#H0が、S2204と同様の方法で、ライト先LUに対応した外部LUを特定する。

【0274】

次に、S2214で、S2213で特定された外部LU（#41）に応じた処理が行われる。具体的には、例えば、次の処理が行われる。すなわち、担当CPU#H0が、ライトコマンド（図22の説明において、第2のライトコマンド）を、外部LU（#41）を有するLDKC（#L0）に送信する。そのライトコマンドは、例えば外部LU（#41）のIDと、S1701で受信したライトコマンド（図22の説明において、「第1のライトコマンド」）で指定されている第1のアドレスに従う第2のアドレス（例えば、第1のアドレスと同じアドレス）とを有する。第2のライトコマンドと、CM#H0内のライト対象データが、イニシエータポート及び外部バス501を介して、LDKC#L0に送信される。LDKC#L0の内部コントローラ#L0が、例えば、図18に示した処理を行う（外部LUがDP-LUの場合、更に、図21に示す処理を行う）。すなわち、内部コントローラ#L0が、HDKC#H0からの第2のライトコマンドに従い、HDKC#H0からのライト対象データをCM#L0に書き込み、応答（例えばライト完了通知）をHDKC#H0に返し、CM#L0内のライト対象データを外部LU（#41）に書き込む。

【0275】

なお、HDKC#H0内のI/O先LUがDP-LUの場合、図18と図20又は図21とに従う処理が行われて良い。そして、I/O先の実ページを有する内部ボリュームが、第2の接続形態に従う内部ボリュームの場合、担当CPU#H0は、例えば、その内部ボリューム（例えばLU）に対応する外部LUを、LUマッピングテーブル（#H0）から特定し、特定された外部LUを有するLDKCに、第2のI/Oコマンドを送信して良い。

【0276】

また、例えば、HDKC#H0内のプールが、（V）同一のティア（例えば、同一のティアレベル又は同一のティアレベル範囲）に属する内部ボリュームで構成されていても良いし、（W）複数の異なるティアに属する内部ボリュームで構成されていても良い。（V）の場合、例えば、同一の実ページ管理テーブルが、内部ボリューム毎に、内部ボリュームが属するティア（例えば、ティアレベル又はティアレベル範囲）を表す情報を有しても良い。（W）の場合、ティア毎にプールが管理されて良い（例えば、ティアに対応するプールを表す管理情報がティア毎にあって良い）。

【0277】

さらに、例えば、HDKC#H0内のプールが、同種の内部ボリュームのみ（例えば、HDKC#H0内のRGに基づく内部ボリュームのみ、或いは、第2の接続形態に従う内部ボリュームのみ）で構成されていても良いし、異種の内部ボリュームで構成されていても良い。

10

20

30

40

50

【0278】

米国においては、第2の接続形態に関して、米国特許第7441095号 (Tamura et al.) の明細書及び図面を援用する。この場合、First Storage Controllerは、本願のHDKCユニット141に、Second Storage Controllerは、本願のLDKCユニット151に、各々対応する。但し、本実施例に従う「ティア」という概念等は、本願に特有の概念である。

【0279】

＜＜第3の接続形態に従うLUに対するI/O＞＞。

【0280】

これについて、図24乃至図30を参照して説明する。なお、図24乃至図30の説明において、HDKC(#0)がサーバ(#0)から受けたI/Oコマンドを、「第1のI/Oコマンド」と言い、HDKC(#0)から送信されるI/Oコマンドを、「第2のI/Oコマンド」と言い、HDKC(#5)からLDKC(#L3) (又はHDKC(#7)) に送信されるI/Oコマンドを、「第3のI/Oコマンド」と言うことがある。

10

【0281】

図17に示したように、F-I/F(#H0)は、第1のI/Oコマンドに従うI/O先LU(#35、#36、又は#37)が、第3の接続形態に従う仮想LUであると判断した場合、図24に示す処理が行われる。

【0282】

すなわち、図24に示すように、S2401で、内部コントローラ#H0(F-I/F(#H0))が、I/O先LUに対応した外部LU(#45、#46又は#47)及びそれを有するHDKC(#H5)を、LUマッピングテーブル#H0から特定する。

20

【0283】

次に、S2402で、内部コントローラ#H0(F-I/F(#H0))が、特定したHDKC(#5)に、第2のI/Oコマンドを送信する。第2のI/Oコマンドは、例えば、特定した外部LU(#45、#46又は#47)のIDと、第1のI/Oコマンドで指定された第1のアドレスに従う第2のアドレス (例えば第1のアドレスと同じアドレス) を表す情報とを有する。第2のI/Oコマンドは、HDKC(#H0)のイニシエータ/ターゲットポート、及び、第3の接続媒体 (外部バス) 601を介して送信され、HDKC(#H5)のF-I/F(#H5)が有するイニシエータ/ターゲットポートが受信する。

30

【0284】

その後、HDKC(#H5)では、HDKC(#H0)で行われた処理 (図17以降を参照して説明した処理) と同様の処理が行われる。

【0285】

例えば、S2403で、内部コントローラ(#H5)(F-I/F(#H5))が、ティア管理テーブル(#H5)を参照する。内部コントローラ(#H5)は、第2のI/Oコマンドで指定されているLU(#45、#46又は#47)に対応した接続形態903 (又は、第3の接続形態1003及び第2の接続形態1004)を基に、そのLUについて、接続形態を特定する。

40

【0286】

接続媒体が「自DKC」又は「第1の接続媒体」の場合 (S2404: Yes)、図25に示す処理が行われる。

【0287】

接続媒体が「第2の接続媒体」の場合 (S2404: No、S2405: Yes)、図27に示す処理が行われる。

【0288】

接続媒体が「第3の接続媒体」の場合 (S2403: No、S2405: No)、図29に示す処理が行われる。

【0289】

50

すなわち、I/O先LU (HDKC#H0がサーバ#0から受信したI/Oコマンドに従うLU) が第3の接続形態に従うLUである場合、下記のケースがある。

(ケース3-1 (図25、図26)) : I/Oが1つの第3の接続媒体のみを経由するケース。具体的には、例えば、I/O先LU (#35) と、末端LU (I/O先LUにカスケード状に関連付けられている1以上のLUのうちの末端のLU) (#45) との間に、LUが介在しない。つまり、末端LUが、I/O先LU (#35) に直接的にマッピングされている外部LU (#45) である。

(ケース3-2 (図27、図28)) : I/Oが1つの第3の接続媒体と1つの第2の接続媒体とを経由するケース。具体的には、例えば、末端LUが、LDKC (#L3) 内のLU (#56) であり、I/O先LU (#36) と末端LU (#56) との間に、第3の接続媒体を介して接続されているHDKC (#H5) 内の外部LU (#46) が介在する。つまり、末端LUが、外部LU (#46) を介してI/O先LU (#36) に間接的にマッピングされている、LDKC (#L3) 内の外部LU (#56) である。

(ケース3-3 (図29、図30)) : I/Oが2つの第3の接続媒体を経由するケース。具体的には、例えば、I/O先LU (#37) と末端LU (#57) との間に、第3の接続媒体を介して接続されているHDKC (#H5) 内の外部LU (#47) が介在する。つまり、末端LUが、外部LU (#47) を介してI/O先LU (#37) に間接的にマッピングされている、HDKC (#H7) 内の外部LU (#57) である。

【0290】

すなわち、第3の接続形態に従うLU (I/O先LU) に対するI/Oについて、I/O先LUと末端LUとの間に、1つの第3の接続媒体の他に、1以上の第1の接続媒体、1以上の第2の接続媒体、1以上の第3の接続媒体、或いは、それらのうちの異なる2種類以上の接続媒体が介在するケースが考えられる。

【0291】

以下、上述した(ケース3-1)、(ケース3-2)及び(ケース3-3)を説明する。これらのケースについての説明と、「第1の接続形態に従うLUに対するI/O」の説明と、「第2の接続形態に従うLUに対するI/O」の説明とを基に、様々な構成の記憶システムを実現することが可能である。

【0292】

(ケース3-1 : 1つの第3の接続媒体のみ)。

【0293】

これについて、図25及び図26を参照して説明する。

【0294】

図25は、第3の接続形態(ケース3-1)に従うI/Oの処理の流れを示す。図26は、第3の接続形態(ケース3-1)に従う構成の概要の一例を示す。図26の一点鎖線は、データの流れを示す。

【0295】

HDKC#H0とHDKC#H5が第3の接続媒体(外部バス)601を介して接続されている。LM#H0 (#H5)、及び、CM#H0 (#H5)の一部が、ティア管理テーブル#H0 (#H5)、LUマッピングテーブル#H0 (#H5)、LU/PDEV変換テーブル#H0 (#H5)、及び、担当CPUテーブル#H0 (#H5)を記憶している。HDKC#H5がDP-LUを有していれば、更に、LM#H5が、仮想ページ管理テーブル#H5を記憶し、CM#H5の一部が、実ページ管理テーブル#H5、及び、キュー先頭管理テーブル#H5を記憶する。

【0296】

内部コントローラ#H0 (#H5)は、相手のHDKC#H5 (#H0)内の管理情報にアクセスすることはできない。

【0297】

サーバ#0と第3の接続媒体601は、同じF-I/F#H0に接続されていても良いし、異なるF-I/F#H0に接続されていても良い。

【0298】

HDKC#H5内の外部LU#45が、HDKC#H0内の仮想LU#35に直接的にマッピングされている。

【0299】

図25によれば、図18を参照して説明した処理(S1801、S1802、S1803、S1811、S1812、S1813)と同様の処理(S2501、S2502、S2503、S2511、S2512、S2513)が行われる。

【0300】

その際、例えば、S2503及びS2512では、応答(リード対象のデータ、ライト完了通知)は、HDKC(#H5)からHDKC(#H0)を介してサーバ(#0)に送信される。

10

【0301】

また、例えば、S2503及びS2513で、LU/PDEV変換テーブル#H5が参照される。

【0302】

また、例えば、外部LU#45がDP-LUの場合、内部コントローラ#H5が、S2503で、図20を参照して説明した処理を行い、S2513で、図21を参照して説明した処理を行うことができる。その際、仮想ページ管理テーブル#H5、実ページ管理テーブル#H5、及びキュー先頭管理テーブル#H5が、参照及び更新される。

【0303】

20

なお、第3の接続形態に従うLUに対するI/Oにおいて(ケース3-1乃至ケース3-3-のいずれについても)、HDKC#H0のCM#H0が使用されない。

【0304】

具体的には、例えば、内部コントローラ#H0がリードコマンドをサーバ#0から受けた場合、リードコマンドで指定されているアドレスに属する領域内のデータ(リード対象データ)がCM#H0に存在するとしても、CM#H0内のリード対象データは、サーバ#0に送信されない。なぜなら、末端LU(#45)内のリード対象データが、別のサーバ(#5)から更新されている可能性があり、それ故、CM#H0内のリード対象データと、末端LU(#45)内のリード対象データとが一致していない可能性があるからである。この場合、CM#H0内のデータではなく、末端LUを有するHDKC内のCMが記憶するデータが、サーバ#0に送信される。

30

【0305】

また、例えば、内部コントローラ#H0がライトコマンドをサーバ#0から受けた場合、ライトコマンドに従うデータを、CM#H0に格納せず、CM#H1へ転送しても良い。

【0306】

ちなみに、末端LUが、サーバ#0が使用するデータのみを記憶することになっている場合、CM#H0が使用されて良い。すなわち、内部コントローラ#H0は、サーバ#0からの第1のリードコマンドに従うリード対象データ(第3の接続形態に従うLUに関連付けられているLU内のデータ)がCM#H0にあれば、第3の接続媒体を介して別のHDKCに第2のリードコマンドを転送せず、CM#H0内のリード対象データをサーバ#0に返して良い。また、内部コントローラ#H0は、サーバ#0からライトコマンドを受信した場合、そのライトコマンドに従うデータ(ライト対象データ)を、CM#H0に書き込み、サーバ#0に、ライト処理完了の応答を返しても良い。このような機能が、前述したプレキャッシュ機能である。これにより、ライト性能及び/又はリード性能が向上する。

40

【0307】

(ケース3-2: 1つの第3の接続媒体と1つの第2の接続媒体)。

【0308】

これについて、図27及び図28を参照して説明する。

50

【0309】

図27は、第3の接続形態（ケース3-2）に従うI/Oの処理の流れを示す。図28は、第3の接続形態（ケース3-2）に従う構成の概要の一例を示す。図28の一点鎖線は、データの流れを示す。

【0310】

HDKC#H0とHDKC#H5が第3の接続媒体（外部バス）601を介して接続されている。LM#H0（#H5）、及び、CM#H0（#H5）の一部が、ティア管理テーブル#H0（#H5）、LUマッピングテーブル#H0（#H5）、LU/PDEV変換テーブル#H0（#H5）、及び、担当CPUテーブル#H0（#H5）を記憶している。

10

【0311】

HDKC#H5とLDKC#L3が第2の接続媒体（外部バス）501を介して接続されている。LDKC#L3が有するターゲットポート（図示しない）が、HDKC#H5が有するイニシエータポートに接続されている。そのイニシエータポートは、HDKC#H0に接続されているイニシエータ／ターゲットポートを有するF-I/Fと同じ又は別のF-I/Fにあって良い。

【0312】

LDKC#L3がDP-LUを有していれば（例えばLU#56がDP-LUであれば）、内部コントローラ#L3は、仮想ページ管理テーブル#L3、実ページ管理テーブル#L3、及び、キュー先頭管理テーブル#L3を記憶することができる。

20

【0313】

内部コントローラ#H0（#H5）は、相手のHDKC#H5（#H0）内の管理情報にアクセスすることはできない。同様に、内部コントローラ#H5（#L3）は、相手のLDKC#L3（#H5）内の管理情報にアクセスすることはできない。

【0314】

サーバ#0と第3の接続媒体601は、同じF-I/F#H0に接続されていても良いし、異なるF-I/F#H0に接続されていても良い。同様に、サーバ#5と第3の接続媒体601は、同じF-I/F#H5に接続されていても良いし、異なるF-I/F#H5に接続されていても良い。サーバ#5がHDKC#H5に接続されていなくても良い。

【0315】

LDKC#L3内の外部LU#56が、HDKC#H0内の仮想LU#36に間接的にマッピングされている。具体的には、LDKC#L3内の外部LU#56が、HDKC#H5内の外部LU（仮想LU）#46に直接的にマッピングされている。その外部LU（仮想LU）#46が、HDKC#H0内の仮想LU#36に直接的にマッピングされている。

30

【0316】

図27によれば、図22を参照して説明した処理（S2201、S2202、S2203、S2204、S2211、S2212、S2213、S2214）と同様の処理（S2701、S2702、S2703、S2704、S2711、S2712、S2713、S2714）が行われる。

40

【0317】

その際、例えば、S2704及びS2712では、応答（末端LU#56から読み出されたリード対象のデータ、ライト完了通知）は、HDKC（#H5）からHDKC（#H0）を介してサーバ（#0）に送信される。

【0318】

また、例えば、S2703及びS2713で、LUマッピングテーブル#H5が参照される。

【0319】

また、S2714で、ライト対象のデータは、末端LU#56に書き込まれる。

【0320】

50

また、例えば、S 2 7 0 4 及び S 2 7 1 4 で、第 3 の I / O コマンドが、H D K C # H 5 から L D K C # L 3 に送信され、L D K C # L 3 において、L U / P D E V 変換テーブル # L 3 が参照される。第 3 の I / O コマンドは、例えば、L U # 5 6 の I D と、第 2 の I / O コマンドで指定された第 2 のアドレスに従う第 3 のアドレス（例えば、第 2 のアドレスと同じアドレス）を表す情報とを有する。

【0321】

また、例えば、外部 L U # 5 6 が D P - L U の場合、内部コントローラ # L 3 が、S 2 7 0 4 で、図 2 0 を参照して説明した処理を行い、S 2 7 1 4 で、図 2 1 を参照して説明した処理を行うことができる。その際、仮想ページ管理テーブル # L 3、実ページ管理テーブル # L 3、及びキュー先頭管理テーブル # L 3 が、参照及び更新される。

10

【0322】

（ケース 3 - 3 : 2 つの第 3 の接続媒体）。

【0323】

これについて、図 2 9 及び図 3 0 を参照して説明する。

【0324】

図 2 9 は、第 3 の接続形態（ケース 3 - 3）に従う I / O の処理の流れを示す。図 3 0 は、第 3 の接続形態（ケース 3 - 3）に従う構成の概要の一例を示す。図 3 0 の一点鎖線は、データの流れを示す。

【0325】

H D K C # H 0 と H D K C # H 5 が第 3 の接続媒体（外部バス）6 0 1 A を介して接続されている。L M # H 0 （# H 5）、及び、C M # H 0 （# H 5）の一部が、ティア管理テーブル # H 0 （# H 5）、L U マッピングテーブル # H 0 （# H 5）、L U / P D E V 変換テーブル # H 0 （# H 5）、及び、担当 C P U テーブル # H 0 （# H 5）を記憶している。

20

【0326】

H D K C # H 5 と H D K C # H 7 が第 3 の接続媒体（外部バス）6 0 1 B を介して接続されている。L M # H 7、及び、C M # H 7 の一部が、ティア管理テーブル # H 7、L U マッピングテーブル # H 7、L U / P D E V 変換テーブル # H 7、及び、担当 C P U テーブル # H 7 を記憶している。H D K C # H 7 が D P - L U を有していれば（例えば L U # 5 7 が D P - L U であれば）、L M # H 7 が、仮想ページ管理テーブル # H 7 を記憶し、C M # H 7 の一部が、実ページ管理テーブル # H 7、及び、キュー先頭管理テーブル # H 7 を記憶することができる。

30

【0327】

内部コントローラ # H 0 （# H 5）は、相手の H D K C # H 5 （# H 0）内の管理情報にアクセスすることはできない。同様に、内部コントローラ # H 5 （# H 7）は、相手の H D K C # H 7 （# H 5）内の管理情報にアクセスすることはできない。

【0328】

サーバ # 0 と第 3 の接続媒体 6 0 1 は、同じ F - I / F # H 0 に接続されていても良いし、異なる F - I / F # H 0 に接続されていても良い。同様に、サーバ # 5 と第 3 の接続媒体 6 0 1 A 及び 6 0 1 B は、同じ F - I / F # H 5 に接続されていても良いし、異なる F - I / F # H 5 に接続されていても良い。同様に、サーバ # 6 と第 3 の接続媒体 6 0 1 B は、同じ F - I / F # H 5 に接続されていても良いし、異なる F - I / F # H 5 に接続されていても良い。

40

【0329】

H D K C # H 7 内の外部 L U # 5 7 が、H D K C # H 0 内の仮想 L U # 3 7 に間接的にマッピングされている。具体的には、H D K C # H 7 内の外部 L U # 5 7 が、H D K C # H 5 内の外部 L U （仮想 L U）# 4 7 に直接的にマッピングされている。その外部 L U （仮想 L U）# 4 7 が、H D K C # H 0 内の仮想 L U # 3 7 に直接的にマッピングされている。

【0330】

50

図29において、S2901及びS2902は、図24を参照して説明したS2401及びS2402と同様である。すなわち、内部コントローラ#H5（F-I/F#H5）が、第2のI/Oコマンドで指定されているLU#47に対応したLU#57のIDとそのLU#57を有するHDKC#H7のIDとを特定する（S2901）。そして、内部コントローラ#H5（F-I/F#H5）が、第3のI/Oコマンドを、特定したHDKC#H7に送信する。第3のI/Oコマンドは、例えば、特定したLU#57のIDと、第2のI/Oコマンドで指定された第2のアドレスに従う第3のアドレス（例えば、第2のアドレスと同じアドレス）を表す情報とを有する。

【0331】

S2903で、第3のI/OコマンドがHDKC#H7で処理される。HDKC#H7では、例えば、図24のS2403が行われ、その後、S2404：Yesであるが故に、図25に示した処理と同様の処理が行われて良い。その際、例えば、応答（リード対象のデータ、ライト完了通知）は、HDKC#H7からHDKC#H5に届き、HDKC#5からHDKC#H0を介してサーバ#0に送信されて良い。また、S2513で、ライト対象のデータが末端LU#57に書き込まれる。また、例えば、S2503及びS2513で、LU/PDEV変換テーブル#H7が参照される。また、例えば、末端LU#57がDP-LUの場合、内部コントローラ#H7が、S2503で、図20を参照して説明した処理を行い、S2513で、図21を参照して説明した処理を行うことができる。その際、仮想ページ管理テーブル#H7、実ページ管理テーブル#H7、及びキュー先頭管理テーブル#H7が、参照及び更新される。

【0332】

以上が、ケース3-1～3-3についての説明である。

【0333】

なお、第3の接続形態に従う接続に関して、HDKC#H0が他のHDKCのLUヘデータを中継しても良い。PDEV種別が同一の場合には、ティアレベルは、HDKC#H0が有する、(A)から(C)のLUのうち、(A)のLUが最も高く、(B)のLUが2番目に高く、(C)のLUが最も低くて良い。ただし、そのためには、例えば、(A)のLUを有するDKCと(B)のLUを有するDKCについての距離（コマンド応答時間）に、ティアレベルを違える根拠と成る程の差が存在すること、及び、(B)のLUを有するDKCと(C)のLUを有するDKCについての距離（コマンド応答時間）に、ティアレベルを違える根拠と成る程の差が存在しないこと、等の条件が必要で良い。

(A) HDKC#H0内のRGに基づくLUであり、サーバ#0に提供されるLU。

(B) 他のHDKCを中継してサーバ#0に提供されるLU。

(C) LDKCのLUが仮想化された仮想LU。

【0334】

以上、実施例1によれば、HDKCが、第1の接続媒体に代えて又は加えて、長距離接続が可能な第3の接続媒体（外部バス）で接続され、第3の接続媒体で接続された複数のHDKCが仮想的に1つのDKCとされる。このため、複数のHDKCをそれぞれ離れた場所に設置することが可能であり、同一サイトに大きなスペースを必要としない。また、接続できるDKCの数に特に制限が無いので、スケーラビリティが向上する。

【実施例2】

【0335】

本発明の実施例2を説明する。その際、実施例1との相違点を主に説明し、実施例1との共通点については説明を省略或いは簡略する。

【0336】

前述したように、HDKCとそのHDKCに第3の接続形態で接続されたHDKCとを含んだ記憶システムによれば、複数のHDKCを仮想的に1台のDKCとしてサーバに見せることができる。

【0337】

このような記憶システムにおいて、負荷が高くなる傾向にあるHDKC（例えばCPU

10

20

30

40

50

）は、実際にLUに対するI/Oを実行することになるHDKC（以下、実施例2において「担当HDKC」）、例えば、末端LUを有するHDKC、或いは、第2の接続形態でLDKCが接続されているHDKCである。そのため、担当HDKC内のLUの負荷が高い場合（例えば、そのLUにI/Oが集中している場合）、担当HDKCのI/O性能が低下し、担当HDKCが、記憶システムのI/O性能のネックとなる。

【0338】

そこで、実施例2では、HDKC間にて負荷分散が行われる。これにより、記憶システムのI/O性能の低下を回避することが期待できる。

【0339】

負荷は、HDKCから、そのHDKCに第1の接続形態で接続されているHDKC（以下、第1接続HDKC）に分散されても良いし、HDKCから、そのHDKCに第3の接続形態で接続されているHDKC（以下、第3接続HDKC）に分散されても良い。以下、HDKC#H0及びそれを含んだHDKCユニット（以下、本実施例の説明においてHDKCユニット#0）に着目する。HDKCユニット#0が、HDKCユニット#H0及び#H9で構成されているとする。

10

【0340】

各HDKCが、図34A、図34B、図35及び図36に示す管理情報を記憶して良い。

【0341】

図34Aは、実施例2に係る担当CPUテーブル3211（#H0）の構成を示す。

20

【0342】

担当CPUテーブル3211は、LU毎に、担当CPUの使用率のうちの、当該LUについての使用率を表す情報を有する。具体的には、例えば、担当CPUテーブル3211（#H0）は、LU毎に、情報3101～3103の他に（図31参照）、LU別のCPU使用率3404を有する。CPU使用率3102及び3404は、定期的に又は不定期的に、それに対応するCPU、又は、HDKCユニット#0内の特定のCPU（マスタCPU）によって更新されて良い。

【0343】

図34Aのテーブル#H0によれば、次のことがわかる。すなわち、CPU#H00の使用率が、60%である。そのうち、LU#0についての使用率が、40%であり、LU#4についての使用率が、20%であり、LU#8についての使用率が、0%である。

30

【0344】

このテーブル#H0は、定期的に（又は不定期的に）、HDKCユニット#0（又はHDKC#H0）が有する少なくとも1つのCPUによって更新されて良い。本実施例では、マスタCPUによって更新される。マスタCPUは、LUに対するI/O処理以外の特定の処理（例えば実施例2に係る負荷分散）を担当するCPUである。マスタCPUは、LUに対するI/O処理も行っている。

【0345】

また、LUについての負荷は、そのLUについてのCPU使用率に代えて又は加えて、他種の負荷指標、例えば、単位時間当たりのI/O数（例えば、リード数、ライト数、又はその合計）によって表されても良い。

40

【0346】

図34Bは、負荷閾値管理テーブル3214（#H0）の構成を示す。

【0347】

負荷閾値管理テーブル3214は、高負荷であるか低負荷であるかの基準を表す。テーブル3214によれば、その基準は、CPU及びDKCの両方について用意されている。しかし、その基準は、他種の要素（例えばFEポート）について用意されても良い。また、テーブル3214の内容は、複数のHDKCで同じであっても異なっても良い。

【0348】

図34Bのテーブル#H0によれば、次のことがわかる。例えば、CPU#H0は、C

50

P U使用率が70%以上であれば、高負荷のC P Uであり、C P U使用率が50%以下であれば、低負荷のC P Uである。また、H D K C # H 0は、全てのC P U # H 0のC P U使用率に基づく使用率（例えば、C P U使用率の平均）が40%以下であれば、低負荷のD K Cである。

【0349】

図34Bのテーブル#H0に記載の閾値は、ユーザインタフェース（例えば、G U I（Graphical User Interface））を介して、管理者から変更されても良い。

【0350】

また、H D K Cにおいて高負荷のC P U（及び／又は、低負荷のC P U）は、そのH D K Cにおいて負荷が相対的に高い（又は低い）C P Uで良い。例えば、H D K Cが有する複数のC P Uの負荷（C P U使用率）の分布を基に、そのH D K Cにおける高負荷のC P U（又は低負荷のC P U）が決定されてよい。

10

【0351】

図35は、ポート負荷管理テーブル3212（#H0）の構成を示す。

【0352】

ポート負荷管理テーブル3212は、各F Eポートの負荷に関する情報を有する。具体的には、例えば、ポート負荷管理テーブル#H0は、F Eポートに所属するL U毎に、下記の情報、

（＊）L Uの所属先のF Eポートの識別子であるポートI D 3501、

（＊）L Uの所属先のF Eポートに所属している全L UについてのC P U使用率3504に基づく値（例えば合計）である、ポート別のC P U使用率3502、

20

（＊）F Eポートに所属するL Uの識別子である所属L U I D 3503、

（＊）L U別のC P U使用率3504、

（＊）L Uの担当C P Uの識別子である担当C P U I D 3505、
を有する。

【0353】

各C P U使用率3502及び3504は、定期的に又は不定期的に、それに対応するC P U、又は、H D K Cユニット#0内のマスタC P Uによって更新されて良い。

【0354】

図35のテーブル#H0によれば、次のことがわかる。すなわち、F Eポート#H0Aに、L U # 0及び# 1が所属している。L U # 0についてのC P U使用率は40%であり、L U # 1についてのC P U使用率は30%であり、それ故、F Eポート#H0AについてのC P U使用率は、70%（40%+30%）である。また、L U # 0の担当C P Uは、C P U # H 0 0であり、L U # 1の担当C P Uは、C P U # H 0 1である。

30

【0355】

図36は、テンポラリL U管理テーブル3213（#H0）の構成を示す。

【0356】

テンポラリL U管理テーブル3213は、テンポラリL Uに関する情報を有する。「テンポラリL U」とは、予め（又は負荷分散時に）生成されるL Uであり、負荷分散に利用されるL Uである。具体的には、例えば、テンポラリL Uは、高負荷のC P Uが担当するL U（以下、移動元L U）内のデータの移動先として選択され得るL Uである。テンポラリL Uについてもティアレベルが算出される。或るH D K Cから第3接続H D K Cへの負荷分散の際には、移動元L U（或るH D K C内のL U）が属するティア（ティアレベル又はティアレベル範囲）と同じティア（ティアレベル又はティアレベル範囲）に属するテンポラリL Uが選択される。

40

【0357】

テンポラリL U管理テーブル3213は、具体的には、例えば、テンポラリL U毎に、下記の情報、

（＊）テンポラリL Uの基になっているP D E Vの種別を表すP D E V種別3601、

（＊）テンポラリL Uの基になっているP D E V種別に基づく1以上のテンポラリL Uの

50

記憶容量の合計であるトータル容量 3 6 0 2、

(*) テンポラリ L U の識別子であるテンポラリ L U I D 3 6 0 3、

(*) テンポラリ L U の記憶容量であるテンポラリ L U 容量 3 6 0 4、
を有する。

【0358】

図 3 6 のテーブル # H 0 によれば、次のことがわかる。1 以上の S S D に基づく 1 以上のテンポラリ L U のトータル容量が 3 5 8 4 G B である。テンポラリ L U # 1 0 0 0 は、S S D に基づく 1 つの L U であり、その記憶容量は 1 0 2 4 G B である。

【0359】

図 3 4 A ~ 図 3 6 に示した複数のテーブル # H 0 のうちの全てを、全ての C P U # H 0 が参照／更新できるよう、H D K C # H 0 内の特定のコントローラモジュール (# H 0 0) 内の C M (# H 0 0) に配置されて良い (或いは、共有情報管理装置 3 0 1 に配置されて良い)。特定の C P U # H 0 (例えばマスタ C P U) の L M # H 0 が、他の全ての C P U # H 0 から参照／更新が可能であれば、図 3 4 A ~ 図 3 6 に示した複数のテーブル # H 0 が、特定の C P U (# H 0 0) に接続された L M (# H 0 0) に配置されても良い。また、各コントローラモジュール内の L M 及び／又は C M が、図 3 4 A ~ 図 3 6 に示した複数のテーブル # H 0 における、そのコントローラモジュールに関わるテーブル部分のみを記憶しても良い。

【0360】

以下、本実施例で行われる負荷分散制御処理を説明する。その処理は、例えば、定期的に又は不定的に開始される。以下の説明では、H D K C # H 0 が負荷分散制御処理を行うことを例に採る。しかし、負荷分散制御処理は、全ての H D K C が行うことができる。また、以下の説明では、主に、マスタ C P U によって処理が行われるが、その処理は、例えば、マスタ C P U が所定のコンピュータプログラムを実行することにより、行われる。

【0361】

本実施例に係る負荷分散制御処理は、下記の 3 つ、

(1) 負荷分散を実行するか否かを判断する処理 (負荷分散判断処理)、

(2) 第 1 接続 H D K C へ負荷を分散する処理 (又は 1 つの H D K C 内での負荷を分散する処理) (以下、第 1 種の負荷分散処理)、

(3) 第 3 接続 H D K C へ負荷を分散する処理 (以下、第 2 種の負荷分散処理)、
で構成される。

【0362】

図 3 7 は、負荷分散判断処理の流れを示す。

【0363】

S 3 7 0 1 で、マスタ C P U が、H D K C ユニット # 0 内の全ての担当 C P U テーブル (# 0 及び # 9) を基に、各 C P U の C P U 使用率 3 1 0 2 を特定する。なお、マスタ C P U に代えて、H D K C ユニット # 0 (又は各 H D K C) 内の各 C P U が、それぞれ、負荷分散制御処理を開始しても良い。但し、各 C P U が負荷分散制御処理を行う場合、動作的な排他が必要になると考えられる。なぜなら、複数の C P U で同時に負荷分散制御処理を行うことで、同一の管理情報を重複して更新してしまうことが考えられるからである。そのため、本実施例では、マスタ C P U が、負荷分散制御処理を行う。

【0364】

S 3 7 0 2 で、マスタ C P U が、H D K C ユニット # 0 内の各 C P U の C P U 使用率 3 1 0 2 と、負荷閾値管理テーブル (# H 0 及び # H 9) とを基に、高負荷の C P U が H D K C ユニット # 0 にあるか否かを判断する。C P U 使用率 3 1 0 2 が C P U の高負荷閾値を超えている C P U (# H 0 又は # H 9) が、高負荷の C P U である。

【0365】

S 3 7 0 2 の判断の結果が否定的の場合 (高負荷の C P U がいない場合) (S 3 7 0 2 : N o)、マスタ C P U が、負荷分散判断の処理を、終了する。

【0366】

S 3 7 0 2 の判断の結果が肯定的の場合（高負荷のCPUがある場合）（S 3 7 0 2 : Yes）、S 3 7 0 3 で、マスタCPUが、HDKCユニット# 0 内の各CPUのCPU使用率3 1 0 2 と、負荷閾値管理テーブル（# H 0 及び# H 9）とを基に、低負荷のCPUがHDKCユニット# 0 にあるか否かを判断する。CPU使用率3 1 0 2 がCPUの低負荷閾値未満であるCPU（# H 0 又は# H 9）が、低負荷のCPUである。

【0 3 6 7】

S 3 7 0 3 の判断の結果が肯定的（低負荷のCPUがある場合）（S 3 7 0 3 : Yes）、図3 8 に示す処理、すなわち、第1種の負荷分散処理が行われる。

【0 3 6 8】

S 3 7 0 3 の判断の結果が否定的の場合（低負荷のCPUがない場合）（S 3 7 0 3 : No）、図4 0 に示す処理、すなわち、第2種の負荷分散処理が行われる。

10

【0 3 6 9】

図3 7 によれば、HDKCユニット# 0 に高負荷のCPUがあっても、そのHDKCユニット# 0 に低負荷のCPUがなければ、そのHDKCユニット# 0 において、第1種の負荷分散処理は行われない。なお、CPUの負荷（CPU使用率）が高負荷であるか低負荷であるかは、例えば、負荷閾値管理テーブル3 2 1 4（図3 4 B参照）に記載の高負荷閾値及び低負荷閾値（いずれもCPU使用率の閾値）を基に決定される。高負荷閾値は、低負荷閾値よりも高い。

【0 3 7 0】

また、図3 7 によれば、低負荷のCPUが、HDKCユニット# 0 に直接的又は間接的に接続されている他のHDKCユニットにあっても、HDKCユニット# 0 にあれば、第3接続HDKCへの負荷分散よりも、第1接続HDKCへの負荷分散（又は1つのHDKC内での負荷分散）が優先的に行われる。つまり、外部バスを介した負荷分散よりも内部バスを介した負荷分散が優先される。これにより、負荷分散にかかる時間を短くすることが期待できる。なお、必ずしも、第1接続HDKCへの負荷分散（又は1つのHDKC内での負荷分散）が優先されなくても良い。

20

【0 3 7 1】

以下、第1種の負荷分散処理と、第2種の負荷分散処理とを説明する。

【0 3 7 2】

<第1種の負荷分散処理>。

30

【0 3 7 3】

図3 8 は、第1種の負荷分散処理の流れを示す。

【0 3 7 4】

S 3 8 0 1 で、マスタCPUが、HDKCユニット# 0 内の1以上の高負荷CPUの中から、CPU使用率3 1 0 2 が最高のCPUを、移動元として選択し、且つ、HDKCユニット# 0 内の1以上の低負荷CPUの中から、CPU使用率3 1 0 2 が最低のCPUを、移動先として選択する。図3 9 Aの例によれば、移動元のCPUとして、CPU# H 0 2 が選択され、移動先のCPUとして、CPU# H 0 1 が選択される。なお、図3 9 Aの例によれば、移動元CPUも移動先CPUもHDKC# H 0 に存在するが、それらのCPUのうちの少なくとも1つが、HDKCユニット# 0 内の別のHDKCに存在するCPU

40

【0 3 7 5】

S 3 8 0 2 で、マスタCPUが、HDKCユニット# 0 内の全ての担当CPUテーブルを基に、移動元CPU（# H 0 2）のCPU使用率3 1 0 2 と移動先CPU（# H 0 1）の使用率3 1 0 2 が移動後（第1種の負荷分散処理の完了後）に平均化するLUを選択する。別の言い方をすれば、マスタCPUは、移動元CPU（# H 0 2）が担当CPUである複数のLU（# 2、# 3、# 6、# 7）から、担当CPUが移動元CPU（# H 0 2）から移動先CPU（# H 0 1）に変わることによって移動元CPU（# H 0 2）のCPU使用率3 1 0 2 と移動先CPU（# H 0 1）のCPU使用率3 1 0 2 が実質的に等しくなるようなLUを選択する。「CPU使用率3 1 0 2 が実質的に等しい」とは、例えば、CPU使

50

用率 3102 の差が $x\%$ 以下（例えば $x = 5$ ）であることを意味する。図 39A の例によれば、CPU 使用率 3404 が「20%」である LU#7 の担当 CPU が CPU#H02 から CPU#H01 に変われば、CPU#H01 の CPU 使用率 3102 が、「40%」から「60%」に増加すると推測され、且つ、CPU#H02 の CPU 使用率 3102 が、「85%」から「65%」に低下すると推測される。つまり、移動元 CPU#H02 の CPU 使用率 3102 と移動先 CPU#H01 の CPU 使用率 3102 が、実質的に等しくなると推測される。従って、図 39A の例によれば、この S3802 で、LU#7 が選択される。

【0376】

S3803 で、マスタ CPU が、移動元 CPU（#H02）及び移動先 CPU（#H01）に、LU 移動指示（移動対象 LU（#7）を移動することの指示）を出す。

10

【0377】

S3804 で、移動先 CPU（#H01）は、その指示を受け、移動元 CPU（#H02）の LM（#H02）に記憶されている管理情報のうち、移動対象 LU（#7）に関する個別情報を、移動元の LM（#H02）から、移動先 CPU（#H01）の LM（#H01）にコピーする。コピーが完了した時点で、移動先 CPU（#H01）が、移動元 CPU（#H02）に、コピー終了を報告する。移動元 CPU（#H02）は、コピー終了の報告を受けた時点で、移動元の LM（#H02）から、上記コピーされた個別情報を削除する。これにより、移動対象 LU（#7）に関する個別情報の移動が完了する。ここで、「移動対象 LU に関する個別情報」とは、例えば、移動対象 LU の担当 CPU のみが参

20

【0378】

S3805 で、マスタ CPU が、移動対象 LU（#7）の担当 CPU を、移動元 CPU（#H02）から移動先 CPU（#H01）に変更する。図 39B の例によれば、マスタ CPU が、下記、

（＊）移動対象 LU（#7）に対応した LU-ID3103 及び CPU 使用率 3404 を、担当 CPU#H02 に代えて、担当 CPU#H01 に関連付ける、

（＊）移動元 CPU#H02 に対応した CPU 使用率 3102 から移動対象 LU（#7）に対応した CPU 使用率 3404 を減算し、且つ、移動先 CPU#H01 に対応した CPU 使用率 3102 に移動対象 LU（#7）に対応した CPU 使用率 3404 を加算する、

30

を行う。

【0379】

このように、第 1 種の負荷分散処理によれば、HDKC ユニット #0 では、各 CPU が、その CPU を有しない他の HDKC 内のハードウェア資源を、その CPU を有する HDKC 内のハードウェア資源と同様に利用することができる。第 1 種の負荷分散処理によれば、管理情報又はその一部の移動だけで、LU の担当 CPU を変更する（負荷分散を行う）ことができる。

【0380】

<第 2 種の負荷分散処理>。

40

【0381】

以下、第 2 種の負荷分散処理を説明する。なお、以下の説明では、HDKC#H0 に第 3 の接続形態で接続されている HDKC を、HDKC#H1 とする。

【0382】

HDKC#H0 内の高負荷 CPU の負荷を分散する方法として、高負荷 CPU が担当する LU を、第 3 接続 HDKC#H1 に移動する方法が考えられる。

【0383】

このような LU 移動が行われても、サーバからこれまでと同じようにしてその LU にアクセスできるようにすることが望ましい。サーバが発行するログインコマンド、I/O コマンド、その他の SCSI (Small Computer System Interface) や FC 等に従ったコ

50

マンドでは、例えば、ポート識別情報（例えばWWN（World Wide Name）、N__P o r t __N a m e、N__P o r t __I D、S__I D）や、LU識別情報（例えばLUN）が指定される。このため、LUの移動前と移動後で、ポート識別情報とLU識別情報が変わらないことが好ましい。

【0384】

また、第3接続の複数のHDKCユニット141やHDKCは、例えば外部バスで接続されており、中距離又は遠距離のサイトに設置されているので、第1接続のようなLU単位の細かな負荷分散を頻繁に繰り返していたのではそれによる処理負荷が大きくなってしまう。したがって、中距離又は遠距離のサイトに設置されている複数のHDKCユニット141やHDKCにおいて、本来的に負荷分散の必要な状態（閾値）に達した場合に、複数のLUを纏めて（複数のLU単位、例えば、ポート単位）負荷分散処理を行うことが良い。以下においては、ポート単位で負荷分散を行う処理について記載するが、本願はより上位の概念である複数のLUを纏めて負荷分散を行うことも良い。

10

【0385】

具体的には、例えば、図65Aに示すように、サーバ#0が、ターゲットポート#H00に属する2つのLU#0及び#1を認識しているとする。

【0386】

ここで、内部コントローラ#H0が、図65Bに示すように、LU#0内のデータを、HDKC#H1内のLU#3に移動することで、LU#0を移動したとする。この場合、LU#0の移動前と移動後で、ポート識別情報とLU識別情報が変わらないようにするためには、LU#0を仮想LU#0とし、仮想LU#0にLU#3をマッピングすることが好ましい。なぜなら、サーバ#0は、ターゲットポート#H00に割り当てられているポート識別情報とLU#0に割り当てられているLU識別情報とを指定したI/Oコマンドを発行すれば、仮想LU#0に対するI/Oが行われ、それ故、仮想LU#0にマッピングされているLU（LU#0に記憶されていたデータを記憶するLU）#3に対するI/Oが行われるからである。

20

【0387】

しかし、この場合、LU#0の移動前と移動後で、LU#0に関する接続形態が変わってしまう。すなわち、LU#0の移動前は、外部バス601を介したI/Oは不要であるが、LU#0の移動後は、外部バス601を介したI/Oが必要になる。つまり、LU#0の移動前と移動後で、LU#0が属するティア（ティアレベル又はティアレベル範囲）が変わってしまう。LU#0が属するティアが変わると、LU#0に対するI/Oの性能が変わってしまう。

30

【0388】

ティアを変えないようにする方法として、例えば、LU#3を、HDKC#H1内のFEポートに割り当て（所属させ）、HDKC#H1が、サーバ#0からLU#3に対するI/Oコマンドを受け付けるようにする方法が考えられる。この方法であれば、外部バス601の経路が不要である。しかし、この方法によれば、サーバ#0は、ターゲットポート#H00以外のポートの識別情報を認識する必要がある。

【0389】

そこで、本実施例に係る第2種の負荷分散処理によれば、LUの移動前後でそのLUが属するティア（ティアレベル又はティアレベル範囲）が変わらず、且つ、LUの移動前後でポート識別情報及びLU識別情報が変わらない。具体的には、例えば、第2種の負荷分散処理では、ポート単位の負荷分散が行われる。

40

【0390】

図32及び図33を参照して、ポート単位の負荷分散を説明する。

【0391】

図32に示すように、各HDKCが、テンポラリLUグループ3203を有する。グループ3203は、複数のテンポラリLU3201で構成されている。テンポラリLU3201は、FEポートに割り当てられていないLUであり、且つ、サーバが入出力するデー

50

タが格納されていないLUである。

【0392】

テンポラリLUグループ3203は、ティア（例えばティアレベル又はティアレベル範囲）が同じ複数のテンポラリLUで構成されていても良いし、ティアが異なる複数のテンポラリLUで構成されていても良い。前者の場合、ティア毎に、テンポラリLUグループ3203が存在しても良い。

【0393】

いろいろな種類のテンポラリLUがあることが好ましい。例えば、PDEV種別が異なる複数のテンポラリLUや、LU種類（例えば、通常のLU、DP-LU、第2の接続形態に従う仮想LU）が異なる複数のテンポラリLUがあっても良い。いろいろな種類のテンポラリLUがあれば、異なるティアに属する複数のテンポラリLUがあるということなので、第2種の負荷分散処理において、移動元LUの移動先とするテンポラリLU（移動元LUが属するティアと同じティアに属するテンポラリLU）が見つかる可能性が高まる。

【0394】

FEポートには、LU（通常のLU、仮想LU、或いは、DP-LU）がマッピングされている。以下、説明を簡単にするため、図32において、FEポートに割り当てられているLUは、自DKCに従うLU、或いは、第1の接続形態に従うLUとする。しかし、FEポートに、第2の接続形態に従う仮想LU、及び／又は、第3の接続形態に従う仮想LUが割り当てられていても良い。

【0395】

HDKCが、空きFEポートを有する。空きFEポートは、第2種の負荷分散処理で使用され得るFEポートであり、LUが1つも割り当てられていないFEポートである。図32の例によれば、HDKC#H1のFEポート#H1Aが、空きFEポートである。なお、HDKC#H0については、空きFEポートが図示されていないが、HDKC#H0も、空きFEポートを有しても良い。

【0396】

図33の例によれば、ポート単位の負荷分散は、例えば、次のように行われる。

（33-1）HDKC#H0が有する複数のFEポートから、対象FEポートが選択される。対象FEポートは、そのFEポートに属する全てのLU内のデータがHDKC#H1に移動することによって、HDKC#H0のDKC負荷（全てのCPU#H0のCPU使用率3102の平均）とHDKC#H1のDKC負荷（全てのCPU#H1のCPU使用率3102の平均）が平均化されるようなFEポートである。図33の例によれば、FEポート#H0Bが選択される。FEポート#H0Bに属する全てのLU#2、#3、#4内のデータがHDKC#H1に移動すれば、HDKC#H0のDKC負荷とHDKC#H1のDKC負荷が平均化される。

（33-2）HDKC#H1から空きFEポート#H1Aが選択される。

（33-3）LU#2の移動先のLUとして、LU#2が属するティアと同じティア（ティアレベル又はティアレベル範囲）に属するテンポラリLU#2002が選択される。テンポラリLU#2002の記憶容量は、LU#2の記憶容量以上である。

（33-4）同様に、LU#3（#4）の移動先のLUとして、LU#3（#4）が属するティアと同じティアに属するテンポラリLU#2001（#2000）が選択される。LU#2001（#2000）の記憶容量は、テンポラリLU#3（#4）の記憶容量以上である。

（33-5）LU#2からテンポラリLU#2002にデータが移動する。同様に、LU#3（#4）からテンポラリLU#2001（#2000）にデータが移動する。

（33-6）テンポラリLU#2000、#2001及び#2002が、空きFEポート#H1Aに割り当てられる。

（33-7）対象FEポート#H0Bに割り当てられていたポート識別情報が、対象FEポート#H0Bに代えて空きFEポート#H1Aに割り当てられる。移動元LU#2に割り当てられていたLU識別情報（例えばLUN）が、LU#2に代えてLU#2002に

10

20

30

40

50

割り当てられる。同様に、移動元LU#3（#4）に割り当てられていたLU識別情報が、LU#3（#4）に代えてLU#2001（#2000）に割り当てられる。

【0397】

このような第2種の負荷分散処理の完了後、サーバ#0は、第2種の負荷分散処理の開始前と同様の方法で、LUに対するI/Oを行うことができる。例えば、サーバ#0は、FEポート#H0Bに割り当てられていたポート識別情報とLU#2に割り当てられていたLU識別情報とを有するI/Oコマンドを発行することにより、FEポート#H1Aに割り当てられたLU#2002に対するI/Oを行うことができる。

【0398】

図40乃至42を参照して、第2種の負荷分散処理の流れを説明する。その際、適宜、図32及び図33を参照する。なお、図45A、45B、46A、46Bは、第2種の負荷分散処理の開始前のテーブルを示し、図47A、47B、48A、48Bは、第2種の負荷分散処理における更新後のテーブルを示す。

10

【0399】

図40に示すように、S4001で、マスタCPU#H0が、CPU負荷情報コマンドをHDKC#H1に送信する。HDKC#H1において、F-I/F#H1が、CPU負荷情報コマンドを受信し、そのコマンドを、マスタCPU#H1に転送する。CPU負荷情報コマンドは、担当CPUテーブルが有する情報を取得するためのコマンドである。

【0400】

すなわち、HDKC#H0とHDKC#H1は第3の接続形態で接続されているので、CPU#H0は、HDKC#H1内の管理情報を、直接参照できない（内部バス227を介してアクセスするかのようにして参照することはできない）。このため、CPU#H0は、HDKC#H1内の管理情報が有する情報を、コマンド（或いは他の方法）により取得する。

20

【0401】

なお、以下の説明では、第2種の負荷分散処理に関わるコマンドは、マスタCPU#H0及び#H1間で行われることとする。しかし、その種のコマンドは、マスタCPU以外のCPUとやり取りされても良い。コマンドは、HDKC#H0のイニシエータ/ターゲットポートと、HDKC#H1のイニシエータ/ターゲットポートとを介して、送受信される。

30

【0402】

さて、S4002で、マスタCPU#H1は、CPU負荷情報コマンドを受信し、担当CPUテーブル#H1が有する情報（HDKC#H1内の全てのCPU#H1の情報）3101、3102、3103及び3404を取得し、それらの情報をHDKC#H0に送信する。

【0403】

S4003で、マスタCPU#H0は、HDKC#H1からS4002で送信された情報を受信し、その情報を基に、HDKC#H1内の全てのCPU#H1のCPU使用率3102の平均（つまり、HDKC#H1のDKC負荷）を算出する。

【0404】

40

S4004で、マスタCPU#H0は、S4003で算出されたDKC負荷を基に、HDKC#H1が低負荷DKCか否かを判断する。S4003で算出されたDKC負荷が、DKC低負荷閾値（図34B参照）未満であれば、HDKC#H1は、低負荷DKCである。

【0405】

S4004の判断の結果が否定的であれば（S4004：No）、第2種の負荷分散処理が終了する。なお、複数の第3接続HDKCがあれば、マスタCPU#H0は、S4001で、複数の第3接続HDKCにCPU負荷情報コマンドを送信して良い。全ての第3接続HDKCが低負荷DKCでない場合に、S4004：Noとなって良い。

【0406】

50

S 4 0 0 4 の判断の結果が肯定的であれば（つまり、H D K C # H 1 が低負荷 D K C であれば）（S 4 0 0 4 : Y e s）、下記の S 4 0 0 5 以降が行われる。なお、複数の低負荷 D K C（第 3 接続 H D K C）があれば、それら複数の低負荷 D K C のうちの、D K C 負荷が最も小さい H D K C が、下記の S 4 0 0 5 以降の対象とされて良い。なお、この例では、H D K C # H 0 の第 3 接続 H D K C は H D K C # H 1 のみであり、且つ、図 4 6 A 及び図 4 6 B に示す例によれば、H D K C # H 1 の D K C 負荷は、1 3 % であり、それは、D K C 低負荷閾値「4 0 %」（図 3 4 B 参照）未満なので、H D K C # H 1 が、S 4 0 0 5 以降の対象となる低負荷 D K C である。

【0 4 0 7】

S 4 0 0 5 で、マスタ C P U # H 0 は、H D K C # H 0 が有する複数の F E ポートから、下記のポート条件（x 1）及び（x 2）のうちの少なくとも 1 つを満たす F E ポート（例えば、少なくとも（x 1）を満たす F E ポート）、

（x 1）割り当てられている全ての L U が H D K C # H 1 に移動すれば、H D K C # H 0 の D K C 負荷と H D K C # H 1 の D K C 負荷とが平均化する、

（x 2）高負荷 C P U # H 0 2（C P U 使用率 3 1 0 2 が C P U 高負荷閾値を超えている C P U）（図 4 6 A 参照）が担当する L U が割り当てられており、割り当てられている全ての L U が H D K C # H 1 に移動すれば、高負荷 C P U # H 0 が高負荷でなくなる、を、移動元として選択する。具体的には、例えば、下記が行われる。

（＊）マスタ C P U # H 0 が、担当 C P U テーブル # 0（図 4 6 A 参照）を基に、H D K C # H 0 の D K C 負荷（H D K C # H 0 内の全ての C P U # H 0 の C P U 使用率 3 1 0 2 の平均）を算出する。図 4 6 A の例によれば、その D K C 負荷は、「7 1 %」である。

（＊）マスタ C P U # H 0 が、ポート負荷管理テーブル # H 0（図 4 5 A 参照）と、H D K C # H 0 の D K C 負荷「7 1 %」と、H D K C # H 1 の D K C 負荷「1 3 %」とを基に、H D K C # H 0 が有する複数の F E ポートから、上記（x 1）及び（x 2）のうちの少なくとも 1 つを満たす F E ポートを、移動元として選択する。そのような F E ポートとして、ここでは、F E ポート # H 0 B（ポート別 C P U 使用率 3 5 0 2 「8 5 %」）（図 4 5 A 参照）が選択される。F E ポート # H 0 B に割り当てられている全ての L U が H D K C # H 1 に移動すれば、H D K C # H 0 の D K C 負荷と、H D K C # H 1 の D K C 負荷が平均化され、且つ、高負荷 C P U # H 0 2 が高負荷でなくなる（図 4 6 A、4 6 B、4 8 A 及び 4 8 B 参照）。具体的には、H D K C # H 0 の D K C 負荷が、「7 1 %」から「4 3 %」に下がり、H D K C # H 1 の D K C 負荷が、「1 3 %」から「4 1 %」に上がり、それ故、D K C 負荷が平均化される（D K C 負荷の差が x % 未満（x は整数（例えば 5）））。また、C P U # H 0 2 の負荷が、「8 5 %」から「2 5 %」（C P U 高負荷閾値未満の C P U 使用率 3 1 0 2（図 3 4 B 参照））に下がる。

【0 4 0 8】

次に、図 4 1 に示すように、S 4 1 0 1 で、マスタ C P U # H 0 が、ポート負荷情報コマンドを H D K C # H 1 に送信する。H D K C # H 1 において、F - I / F # H 1 が、ポート負荷情報コマンドを受信し、そのコマンドを、マスタ C P U # H 1 に転送する。ポート負荷情報コマンドは、ポート負荷管理テーブルが有する情報を取得するためのコマンドである。

【0 4 0 9】

次に、S 4 1 0 2 で、マスタ C P U # H 1 は、ポート負荷情報コマンドを受信し、ポート負荷管理テーブル # H 1 が有する情報（H D K C # H 1 内の全ての F E ポート # H 1 の情報）3 5 0 1 ~ 3 5 0 5 を取得し、それらの情報を H D K C # H 0 に送信する。

【0 4 1 0】

次に、S 4 1 0 3 で、マスタ C P U # H 0 は、H D K C # H 1 から S 4 1 0 2 で送信された情報を受信し、その情報を基に、H D K C # H 1 の空き F E ポート（L U が 1 つも割り当てられていない F E ポート）を特定し、特定した空き F E ポートを、移動先として選択する。図 4 5 B のポート負荷管理テーブル # H 1 によれば、F E ポート # H 1 A が、空き F E ポートである。

【0411】

次に、S4104で、マスタCPU#H0が、ティア情報コマンドをHDKC#H1に送信する。HDKC#H1において、F-I/F#H1が、ティア情報コマンドを受信し、そのコマンドを、マスタCPU#H1に転送する。ティア情報コマンドは、ティア管理テーブルが有する情報を取得するためのコマンドである。

【0412】

次に、S4105で、マスタCPU#H1は、ティア情報コマンドを受信し、ティア管理テーブル#H1が有する情報（例えば、情報901～905、或いは、情報1001～1010）を取得し、それらの情報をHDKC#H0に送信する。マスタCPU#H1は、ティア管理テーブル#H1が有する情報のうち、テンポラリLU（テンポラリLU管理

10

【0413】

次に、S4106で、マスタCPU#H0が、HDKC#H1からS4105で送信された情報を受信する。マスタCPU#H0は、その情報を基に、HDKC#H1におけるテンポラリLUとそれのティアレベルとを特定する。マスタCPU#H0は、S4005で選択した移動元FEポート#H0Bに割り当てられているLU（移動元LU）毎に、移動元LUが属するティアと同じティアに属するテンポラリLUがHDKC#H1にあるかどうかを判断する。そのようなテンポラリLUが、移動元LUとペアにされ、移動元LUからのデータの移動先とされる。なお、「移動元LUが属するティアと同じティアに属するテンポラリLU」とは、例えば、下記のうちのいずれかに該当するテンポラリLUである。

20

（＊）テンポラリLUのティアレベルが、移動元LUのティアレベルと同じである。

（＊）テンポラリLUのティアレベルと移動元LUのティアレベルとの差が、一定値以下である（テンポラリLUのティアレベルが属するティアレベル範囲が、移動元LUのティアレベルが属するティアレベル範囲と同じである）。

【0414】

S4106において、移動元FEポート#H0Bに属する少なくとも1つの移動元LUについて、移動元LUが属するティアと同じティアに属するテンポラリLUが無ければ（S4106：No）、マスタCPU#H0は、第2種の負荷分散処理を終了する。

【0415】

30

なお、S4106：Noの場合、別の低負荷DKC（第3接続HDKC）があれば、マスタCPU#H0は、別の低負荷DKCに対して、S4005以降を行ってもよい。

【0416】

S4106において、移動元FEポート#H0Bに属する全ての移動元LUについて、移動元LUが属するティアと同じティアに属するテンポラリLUがあれば（S4106：Yes）、S4107で、マスタCPU#H0は、各移動元LUについて、移動元LUから移動先LU（テンポラリLU）へのデータ移動を行う。

【0417】

その後、S4108で、マスタCPU#H1が、CPU負荷情報コマンドを、HDKC#H0（移動元のHDKC）に送信する。HDKC#H0において、F-I/F#H0が、CPU負荷情報コマンドを受信し、そのコマンドを、マスタCPU#H0に転送する。

40

【0418】

次に、図42に示すように、S4201で、マスタCPU#H0は、CPU負荷情報コマンドを受信し、担当CPUテーブル#H0が有する情報（HDKC#H0内の全てのCPU#H0の情報）3101、3102、3103及び3404を取得し、それらの情報をHDKC#H1に送信する。

【0419】

次に、S4202で、マスタCPU#H1は、HDKC#H0からS4201で送信された情報を受信し、その情報を基に、移動先LU（テンポラリLU）毎に、下記を行う。

（＊）マスタCPU#H1は、移動先LU（例えば#2002）に、移動元LU（例えば

50

2) に割り当てられていた L U N を割り当てる。

(*) マスタ C P U # H 1 は、移動先 L U (例えば # 2 0 0 2) を、移動先 F E ポート (空き F E ポート) # H 1 A に割り当てる (所属させる)。

【0420】

次に、S 4 2 0 3 で、マスタ C P U # H 1 は、担当 C P U テーブル # H 1 と、担当 C P U テーブル # H 0 が有する情報 (S 4 2 0 2 で受信した情報) とを基に、H D K C # H 1 において全ての C P U # H 1 の C P U 使用率 3 1 0 2 が平均化するように、移動先 L U の担当 C P U # H 1 を決定する。具体的には、例えば、下記が行われる。

(*) マスタ C P U # H 1 が、担当 C P U テーブル # H 0 (図 4 5 A 参照) が有する情報を基に、移動元 L U の C P U 使用率 3 1 0 2 を特定する。

10

(*) マスタ C P U # H 1 が、担当 C P U テーブル # H 1 (図 4 5 B 参照) を基に、各 C P U # H 1 の C P U 使用率 3 1 0 2 を特定する。

(*) マスタ C P U # H 1 が、移動先 L U に対応する移動元 L U の C P U 使用率 3 1 0 2 と、各 C P U # H 1 の C P U 使用率 3 1 0 2 とを基に、全ての C P U # H 1 の C P U 使用率 3 1 0 2 が平均化するように、移動先 L U の担当 C P U を決定する。

(*) マスタ C P U # H 1 が、担当 C P U テーブル # H 1 を更新する (図 4 8 B 参照)。具体的には、例えば、マスタ C P U # H 1 は、移動先 L U の担当 C P U の I D 3 1 0 1 に対応した欄に、移動先 L U の I D 3 1 0 3 と、移動先 L U の C P U 使用率 3 4 0 4 とを登録し、且つ、その担当 C P U の C P U 使用率 3 1 0 2 に、移動先 L U の C P U 使用率 3 4 0 4 を加算する。

20

【0421】

一方、S 4 2 0 4 で、H D K C # H 0 において、マスタ C P U # H 0 が、移動先 L U と移動元 L U とのペアを解除し、且つ、移動元 L U をテンポラリ L U とする。例えば、マスタ C P U # H 0 は、移動元 L U に関する特定の情報 (例えば C P U 使用率 3 4 0 4 等) を破棄し、テンポラリ L U 管理テーブル # H 0 に、移動元 L U に関する情報 (例えば情報 3 6 0 3 及び 3 6 0 4) を追加する。

【0422】

最後に、移動元 F E ポート # H 0 B のポート情報 (例えば、WWN 及び inquiry 情報) と移動先 F E ポート # H 1 A のポート情報が交換される。

【0423】

30

具体的には、S 4 2 0 5 で、マスタ C P U # H 0 が、ポート情報コマンドを H D K C # H 1 に送信する。ポート情報コマンドは、ポート情報を取得するためのコマンドである。そのポート情報コマンドでは、移動先 F E ポート # H 1 A の I D が指定されて良い。

【0424】

S 4 2 0 6 で、マスタ C P U # H 1 が、ポート情報コマンドを受信し、移動先 F E ポート # H 1 A に対応したポート情報 (例えば、WWN 及び inquiry 情報) を取得し、そのポート情報を H D K C # H 0 に送信する。そのポート情報は、例えば、ポート負荷管理テーブル # H 1 又はその他の管理情報が有して良い。

【0425】

S 4 2 0 7 で、マスタ C P U # H 1 が、ポート情報コマンドを H D K C # H 0 に送信する。そのポート情報コマンドでは、移動元 F E ポート # H 0 B の I D が指定されて良い。

40

【0426】

S 4 2 0 8 で、マスタ C P U # H 0 が、ポート情報コマンドを受信し、移動元 F E ポート # H 0 B に対応したポート情報 (例えば、WWN 及び inquiry 情報) を取得し、そのポート情報を H D K C # H 1 に送信する。そのポート情報は、例えば、ポート負荷管理テーブル # H 0 又はその他の管理情報が有して良い。

【0427】

S 4 2 0 9 で、マスタ C P U # H 0 が、移動元 F E ポート # H 0 B に対応したポート情報を、移動先 F E ポート # H 1 A に対応したポート情報に変更する。

【0428】

50

S 4 2 1 0 で、マスタ C P U # H 1 が、移動先 F E ポート # H 1 A に対応したポート情報を、移動元 F E ポート # H 0 B に対応したポート情報に変更する。

【 0 4 2 9 】

以上、第 2 種の負荷分散処理によれば、サーバが I / O の際に使用するポート識別情報及び L U 識別情報を変えることなく、サーバから I / O コマンドを受ける F E ポートを、移動元 F E ポート # H 0 B から移動先 F E ポート # H 1 A に変えさせることができる。また、移動元 L U の移動先となるテンポラリ L U が属するティアは、移動元 L U が属するティアと同じティアなので、ティアレベルを変えることなく、負荷分散が可能となる。

【 0 4 3 0 】

なお、S 4 0 0 1 のコマンドの送信先は、H D K C # H 0 に最も近い第 3 接続 H D K C から、順次に、遠くに遷移して良い。具体的には、マスタ C P U # H 0 は、第 2 種の負荷分散処理における判断（例えば、S 4 0 0 4 又は S 4 1 0 6、或いは、後述の S 4 3 0 4 又は S 4 3 0 5）の結果が否定的となった場合、S 4 0 0 4 の判断の対象とされていない 1 以上の第 3 接続 H D K C のうち、H D K C # H 0 に最も近い H D K C（介在する第 3 接続 H D K C の数が最も少ない第 3 接続 H D K C）を選択し、選択した第 3 接続 H D K C を相手として、第 2 種の負荷分散処理を開始して良い。

10

【 0 4 3 1 】

図 4 3 及び図 4 4 は、図 4 1 の S 4 1 0 7 の処理の流れを示す。

【 0 4 3 2 】

S 4 3 0 1 で、マスタ C P U # H 0 は、移動開始通知を H D K C # H 1 に送信する。H D K C # H 1 のマスタ C P U # H 1 が、移動開始通知を受信する。

20

【 0 4 3 3 】

S 4 3 0 2 で、マスタ C P U # H 1 が、H D K C # H 1 に、移動元情報コマンドを送信する。移動元情報コマンドは、移動元 L U に関する情報である移動元情報（例えば、移動元 L U の数、各移動元 L U の容量、及び、各移動元 L U の P D E V 種別を表す情報）を取得するためのコマンドである。マスタ C P U # H 0 が、移動元情報コマンドを受信する。

【 0 4 3 4 】

S 4 3 0 3 で、マスタ C P U # H 0 が、移動元情報（例えば、移動元 L U の数、各移動元 L U の容量、及び、各移動元 L U の P D E V 種別を表す情報）を、H D K C # H 1 に送信する。マスタ C P U # H 1 が、移動元情報を受信する。

30

【 0 4 3 5 】

S 4 3 0 4 で、マスタ C P U # H 1 が、移動元 L U 毎に、移動元 L U の P D E V 種別と同じ P D E V 種別のテンポラリ L U があるか否かを判断する。移動元 L U の P D E V 種別は、移動元情報から特定される。テンポラリ L U の P D E V 種別は、テンポラリ L U 管理テーブル # H 1 における P D E V 種別 3 6 0 1 から特定される。

【 0 4 3 6 】

少なくとも 1 つの移動元 L U について、移動元 L U の P D E V 種別と同じ P D E V 種別のテンポラリ L U が存在しない場合（S 4 3 0 4 : N o）、マスタ C P U # H 1 が、第 2 種の負荷分散処理を終了する。マスタ C P U # H 1 は、例えば、第 2 種の負荷分散処理の終了を、H D K C # H 0 に通知して良い。これは、S 4 3 0 5 : N o でも同様で良い。

40

【 0 4 3 7 】

全ての移動元 L U について、移動元 L U の P D E V 種別と同じ P D E V 種別のテンポラリ L U が存在する場合（S 4 3 0 4 : Y e s）、S 4 3 0 5 で、マスタ C P U # H 1 が、テンポラリ L U 管理テーブル # H 1 を基に、P D E V 種別毎に、その P D E V 種別に対応したトータル容量 3 6 0 2 が、その P D E V 種別に対応した 1 以上の移動元 L U のトータル容量以上か否かを判断する。

【 0 4 3 8 】

全ての移動元 L U に関する少なくとも 1 つの P D E V 種別について、トータル容量 3 6 0 2 が 1 以上の移動元 L U のトータル容量未満であれば（S 4 3 0 5 : N o）、マスタ C P U # H 1 が、第 2 種の負荷分散処理を終了する。

50

【0439】

全ての移動元LUに関する全てのPDEV種別について、トータル容量3602が1以上の移動元LUのトータル容量以上であれば(S4305:Yes)、S4306で、マスタCPU#H1が、容量が移動元LUと同じになるテンポラリLUを作成する。ここでは、例えば、(y1)既存のテンポラリLUが、容量が移動元LUと同じになるテンポラリLUと、他のテンポラリLUとに分割されても良いし、(y2)複数のテンポラリLUが組み合わせられることで、容量が移動元LUと同じになるテンポラリLUが作成されても良いし、(y3)(y1)と(y2)の両方が行われることで、容量が移動元LUと同じになるテンポラリLUが作成されても良い。

【0440】

次に、図44に示すように、S4401で、マスタCPU#H1が、S4306の処理結果を基に、テンポラリLU管理テーブル#H1を更新する。図49Aに、更新前のテンポラリLU管理テーブル#H1の一例を示し、図49Bに、更新後のテンポラリLU管理テーブル#H1の一例を示す。図49A及び図49Bの例によれば、PDEV種別「SSD」について、「1024GB」のテンポラリLU#1000に「2048GB」のテンポラリLU#1001がマージされることで、テンポラリLU#1000の容量が「3072GB」に拡張されている。また、PDEV種別「SAS」について、「4096GB」のテンポラリLU#1003が、「2048GB」のテンポラリLU#1003と「2048GB」のテンポラリLU#1100に分割されている。

【0441】

移動先LUの容量は、移動元LUの容量と同じ（又はそれ以上）である必要がある。しかし、必ずしも移動元LUの容量と同じ容量のテンポラリLUが存在しているとは限らない。このため、上記のように、テンポラリLUのマージ／分割により、移動元LUと容量が同じテンポラリLUが作成される。そして、テンポラリLU管理テーブル#H1が更新される。なお、移動元LUの容量と同じ容量のテンポラリLUが既に存在している場合には、テンポラリLUを作成する処理は、スキップされて良い。

【0442】

その後、S4402で、マスタCPU#H1が、作成したテンポラリLUの情報（例えば、更新後のテンポラリLU管理テーブル#H1が有する情報）を、HDKC#H0に送信する。送信される情報は、例えば、少なくとも、移動元LUと容量及びPDEV種別が同じテンポラリLUの情報を含む。マスタCPU#H0が、テンポラリLUの情報を受信する。

【0443】

S4403で、マスタCPU#H0が、テンポラリLUの情報を基に、移動元LU毎に、下記、

(*) 移動元LUと、その移動元LUと容量及びPDEV種別が同じテンポラリLUとのペアを作成する、

(*) 移動元LUから、その移動元LUとペアになっているテンポラリLUに、データを移動し始める、

を行う。なお、ここでは、データの「移動」といっても、移動されたデータが移動元LUから削除されなくて良い。つまり、ここで言う「移動」は、「コピー」と同義で良い。

【0444】

マスタCPU#H0は、移動元LUからテンポラリLUへのデータ移動の最中に、移動元LUにサーバからデータが新たに書き込まれた場合(S4404:Yes)、移動元LUに書き込まれたデータを、移動先のテンポラリLUに移動する(S4405)。マスタCPU#H0は、移動元LUを構成する全領域内のデータを移動し終えるまで、データ移動を継続する(S4406:No、S4407)

【0445】

マスタCPU#H0は、移動元LUを構成する全領域内のデータを移動し終えた場合(S4406:Yes)、移動元LUとテンポラリLUとのペアを解除する(S4408)

。

【0446】

これにより、図41のS4107が完了する。

【0447】

なお、移動元FEポート#H0Bに属する全てのLU(#2、#3、#4)を空きFEポート#H1Aに移動させた場合(ポート単位の負荷分散が行われた場合)、移動元のHDKC#H0(又は、移動先のHDKC#H1)が、上位のデバイス(例えば、サーバ、又は、サーバとHDKC#H0(#H1)との間に介在するスイッチ(例えばFCスイッチ))に、所定の情報を送信して良い。スイッチは、複数のポート(以下、SWポート)と、どの宛先が指定されたコマンドを受けた場合にどのポートから出力するかのルーティング制御情報とを有する。上位のデバイスがスイッチの場合、例えば、次のような処理が行われて良い。例えば、移動元のHDKC#H0(又は、移動先のHDKC#H1)が、スイッチに、交換前のWWNと交換後のWWNとを含んだ所定の情報を送信する。スイッチが、その所定の情報に含まれているWWNを基に、ルーティング制御情報を更新する。具体的には、例えば、スイッチは、移動元FEポート#H0Bに対応した交換前のWWNが、移動先FEポート#H1Aに接続されているSWポートに対応するように、ルーティング制御情報を更新する。これにより、その後、スイッチは、移動元FEポート#H0Bに対応した交換前のWWNを含んだI/Oコマンドを受けた場合、そのI/Oコマンドを、移動先FEポート#H1Aに転送することができる。

【0448】

上述の通り、第3接続の複数のHDKCユニット141やHDKCは、本来的に負荷分散の必要な状態(閾値)に達した場合に、複数のLUを纏めて(複数のLU単位、例えば、ポート単位)負荷分散処理(移行処理を含む)を行うことが良い。ポート単位での負荷分散を適用することなく複数のLUを纏めて負荷分散を行う場合の一例としては、FEポートに接続されている1つ又は複数のサーバ101やスイッチ上にパス管理ソフトウェアを搭載して、第3接続の複数のHDKCユニット141やHDKCの間で複数のLUの各々に関する構成情報及び/又は複数のLUへのアクセスの処理を担当する権利(複数のLUの各々に関するオーナー権)を移行又は通知したとき、又はその後に、移行対象の複数のLUの各々に対するパスを移行元のLUのパスから移行先のLUへ変更させることにより実現できる。この場合、パスの移行前に、移行先の複数のLUの各々に関する構成情報を基にして、移行先の複数のLUの各々に対する1つ又は複数のサーバ101やスイッチからのパスを生成又は設定しておくことが良い。このパスは、移行先のLUの識別情報(例えば、LUN)と移行先LUが対応付けられているFEポートのポート識別情報(例えばWWN(World Wide Name)、N__Port__Name、N__Port__ID及び/又はS__ID)を用いて生成又は設定される。また、移行元の複数のLUから移行先の複数のLUへのデータの移行は、上記構成情報やオーナー権の移行と同期して(同様なタイミングで)実施されても良いし、非同期に(後で)実施されても良い。なお、非同期に実施する場合には、移行先の複数のLUを有するHDKCユニット141やHDKC内のプロセッサ222は、1つ又は複数のサーバ101やスイッチから移行先のLUに対するライトアクセスに対応するライトデータを一時的に保持して、移行先のLUに対して移行元のLUからのデータが移行された後に、当該ライトデータにて上書きする、等のデータの整合性を確保するための処理が必要となる。1つ又は複数のサーバ101やスイッチから移行先のLUに対するリードアクセスに対しても、移行元のLUからデータを読み出して、1つ又は複数のサーバ101やスイッチに対して読み出されたデータを応答することが好ましい。また、移行元のLUのパスから移行先のLUのパスへ変更させるには、次のいずれの方法をとっても良い。まず、第一の方法としては、第3接続の複数のHDKCユニット141、その内部のHDKC及び/又は管理装置301のプロセッサが移行元のLUのパスと移行先のLUのパスを判断及び特定し、1つ又は複数のサーバ101やスイッチのパス管理ソフトウェアに対して、移行対象の複数のLUの各々について、移行元のLUのパスと移行先のLUのパスの情報を通知する。これにより、パス管理ソフトウェアが、

移行対象の複数のLUの各々に対するパスを移行先のLUのパスに変更する。第二の方法としては、第3接続の複数のHDKCユニット141、その内部のHDKC及び／又は管理装置301のプロセッサが移行元のLUのパスと移行先のLUのパスを判断及び特定し、第3接続の複数のHDKCユニット141やHDKCの間で複数のLUに関する構成情報を移行又は通知するとき又はした後に、第3接続の複数のHDKCユニット141又はその内部のHDKCのプロセッサが1つ又は複数のサーバ101やスイッチのパス管理ソフトウェアから発行されるコマンド（例えば、Inquiryコマンドや定期的に発行されるその他の問い合わせコマンド）を待ち、特定された複数の移行先のLUのパスを利用して対応する各々のLUにアクセスするように、そのコマンドに対して、移行対象の複数のLUの各々に関して必要な情報（例えば、移行元のLUのパスの情報及び／又は移行先のLUのパスの情報、及び／又はアクティブ、パッシブ等の状態の情報）を応答する。これにより、パス管理ソフトウェアが、移行対象の複数のLUの各々に対するパスを移行先のLUのパスに変更する。第三の方法としては、第3接続の複数のHDKCユニット141、その内部のHDKC及び／又は管理装置301のプロセッサが複数の移行元のLUのパスと複数の移行先のLUのパスを判断及び特定し、第3接続の複数のHDKCユニット141やHDKCの間で複数のLUに関する構成情報を移行又は通知するとき又はした後に、移行元の複数のLUを有するHDKCユニット141やHDKCが、1つ又は複数のサーバ101やスイッチから移行対象の複数のLUのいずれに対するアクセスに対しても、当該アクセスを拒否する、又は当該アクセスに対して応答しないことによる。これにより、パス管理ソフトウェアが、移行対象の複数のLUの各々に対するパスを移行先のLUのパスに変更する。米国においては、1つ又は複数のサーバ101やスイッチから移行対象のLUに対するアクセスに対してアクセスを拒否する技術について、米国特許公開第2006/0271758号（Innan et al.）の明細書及び図面を援用することができる。この場合、当該明細書及び図面のHost Deviceが本願の1つ又は複数のサーバ101又はスイッチ、当該明細書及び図面のVirtualization Storage Device 1及びVirtualization Storage Device 2の各々が本願の1つのHDKCユニット141やHDKCとそれと別のHDKCユニット141やHDKC、にそれぞれ対応付けて本願技術を適用する。

【0449】

また、複数のLUを纏めて負荷分散を行う場合、上記説明のように1つのHDKCユニット141やHDKCから1つのHDKCユニット141やHDKCへ複数の負荷分散を行う場合に限られず、移行元の複数のLUを有するHDKCユニット141やHDKCから複数のHDKCユニット141やHDKCに分散させて移行対象の複数のLUを移行させることも良い。

【0450】

また、本願の技術は、第3接続の複数のHDKCユニット141やHDKC間で複数のLUを纏めて負荷分散処理（移行処理を含む）を行う場合に限られず、第3接続の複数のHDKCユニット141やHDKC間で1つのLUだけ（1つのLU単位）の負荷分散処理（移行処理を含む）を行うことも良い。本願に従う「ティア」を維持した状態での1つのLU単位の負荷分散処理（移行処理を含む）であれば、既存の技術にない全く新しい技術であるからである。この場合、HDKCユニット141やHDKCや管理装置301のプロセッサが、例えば、本願に従う複数の「ティア」の各々に対応するLU（論理ボリュームや仮想ボリューム）の中からティアレベル（ティアレベルの範囲であっても良い）の変更しない適切な「ティア」を有するLUを選択、判断及び決定すること、を一例とする新たな処理を行う。

【0451】

以上が、実施例2についての説明である。例えば、共通情報管理装置301が1つ又は複数のサーバ、1つ又は複数の仮想サーバ（下記）、1つ又は複数のスイッチ、1つ又は複数のHDKCユニット141と1つ又は複数のLDKCユニット151などの処理を総合的に管理することにより、システム全体として、管理コストの低下や、性能向上を図ることができる。この技術と効果は、実施例1及び3についても共通である。さらに、負荷

分散処理で分散される1つ以上のLUのLUNセキュリティ情報もまた、負荷分散処理に応じて、分散元のHDKCユニットから分散先のHDKCユニットへ移行できる。LUNセキュリティ情報は、サーバから未承認のLUへの未承認アクセスを回避するためのものであり、サーバのポートアドレスとLUのLUNとの間の対応情報である。米国では、LUNセキュリティについて、米国特許公報6684209 (Ito et al.) の明細書及び図面を援用する。

【0452】

なお、例えば、第1種の負荷分散処理は、自DKC内での負荷分散処理も含むので、LDKCも、第1種の負荷分散処理を行っても良い。

【0453】

また、実施例2 (例えば、LU単位の負荷分散 (第1種の負荷分散処理) と、ポート単位の負荷分散 (第2種の負荷分散処理) のどちらを行うかの判断及び決定) は、以下のいずれか1つのケースで、HDKCユニット141、HDKC及び／又は管理装置301のプロセッサによって適用されて良い。

(a) 接続形態 (DKCの性能を含んで良い) 及びPDEV種別という基本項目 (性能ファクター) に加えて又は代えて、「プレキャッシュ」、「カスケードの違い」及び／又は「距離 (コマンド応答時間)」等の1以上の他の項目 (性能ファクター) も基に、ティアレベルが算出されるケース。

(b) 「プレキャッシュ」、「カスケードの違い」及び／又は「距離 (コマンド応答時間)」 (図10参照) のうちの1以上の項目 (性能ファクター) のみを基にティアレベルが算出されるケース。

(c) 「プレキャッシュ」、「カスケードの違い」及び／又は「距離 (コマンド応答時間)」 (図10参照) のうちのいずれか1つ以上の項目 (性能ファクター) と、PDEV種別 (性能ファクター) との組合せを基に、ティアレベルが算出されるケース。

(d) 「プレキャッシュ」、「カスケードの違い」及び／又は「距離 (コマンド応答時間)」 (図10参照) のうちのいずれか1つ以上の項目 (性能ファクター) と、「接続形態」、「内部コントローラの性能」、及び「PDEV種別」のうちのいずれか1つ以上の項目 (性能ファクター) との組み合わせを基に、ティアレベルが算出されるケース。

【0454】

更に、実施例2に係る負荷分散制御処理において、移動元及び／又は移動先のHDKCのプロセッサ (CPU) は、中位ティア (中位のティアレベル又はティアレベル範囲) に属するLU (他の複数のHDKCのLUを仮想化した仮想LU) が割り当てられるFEポートを、移動元FEポートの候補から外しても良い。

【0455】

また、移動元及び／又は移動先のHDKCのプロセッサは、当該中位ティアに属するLUが割り当てられているFEポート (この段落において「対象FEポート」) について、移動元FEポートとすることの優先順位を、自DKC又は第1の接続形態に従うLUのみが割り当てられているFEポートに比べて、低くして良い。この場合、対象FEポートに所属するLUの移動は (つまり、対象FEポートが移動元FEポートとして選択されることは)、自DKC又は第1の接続形態に従うLUのみが所属するFEポートが存在しない場合にのみ、実施されて良い。

【0456】

本実施例は、HDKC内の実LU (自DKC又は第1の接続形態に従うLU、言い換えれば、本実施例で言う仮想LU (第2の接続形態に従う仮想LU、又は、第3の接続形態に従う仮想LU) 以外のLU) が他のHDKCに移動されるケース以外のケースに適用されて良い。例えば、本実施例は、上位ティア (上位ティアレベル又は上位ティアレベル範囲) に属するLUのみが割り当てられているFEポートに関する移動 (そのFEポートに所属する全ての実LUの移動) が行われるケースに代えて又は加えて、他のケース、例えば、以下の何れのケース、

(1) 中位ティア (例えば中位ティアレベル又は中位ティアレベル範囲) に属するLUの

10

20

30

40

50

みが所属する F E ポートの移動（ここでいう「中位ティアに属する L U」とは、例えば、第 3 の接続形態に従う仮想 L U のことである）、

（2）下位ティア（例えば下位ティアレベル又は下位ティアレベル範囲）に属する L U のみが割り当てられている F E ポートの移動（ここでいう「下位ティアに属する L U」とは、例えば、第 2 の接続形態に従う仮想 L U のことである）、

（3）上位ティア（例えば上位ティアレベル又は上位ティアレベル範囲）に属する L U と中位ティアに属する L U のみが割り当てられている F E ポートの移動、

（4）中位ティアに属する L U と下位ティアに属する L U のみが割り当てられている F E ポートの移動、

（5）上位ティアに属する L U と中位ティアに属する L U と下位ティアに属する L U のいずれも割り当てられている F E ポートの移動、 10

にも適用されて良い。ただし、移動元及び／又は移動先の H D K C のプロセッサは、ポート単位で負荷分散（L U 移動）を行う必要があり、さらに、移動元ポートに属するいずれの L U についても、ティア（例えば、ティアに影響する性能ファクター）が移動前後において同様である必要がある。

【0457】

3 つ以上の H D K C が接続している場合には、管理端末又は特定の H D K C のプロセッサが、3 つ以上の H D K C から負荷分散元の H D K C と負荷分散先の H D K C とを選択して良い。

【0458】

20

また、第 3 接続形態で接続されている H D K C が有するテンポラリ L U として、（a）その H D K C が有する L U（自 D K C に従う L U）（D P - L U でも良い）、（b）その H D K C を含んだ H D K C ユニット内の他の H D K C が有する L U（第 1 の接続形態に従う L U）（D P - L U でも良い）、（c）第 2 の接続形態に従う仮想 L U、（d）第 3 の接続形態に従う仮想 L U、等がある。

【0459】

また、移動先として選択されるテンポラリ L U は、移動元 L U と比べて、ティア（例えばティアレベル）が同じであることに加えて、下記の（p）、（q）、及び（r）のうちの少なくとも 1 つで良い。

（p）同種の性能ファクターについてポイントが違っている、 30

（q）同種の性能ファクターについてポイントが同じである、

（r）複数種類の性能ファクターのうち特定の種類の性能ファクターのみポイントが同じである。

【0460】

また、本実施例 2 は、負荷分散について説明したが、本願は負荷分散だけに適用される技術でなく、L U の構成情報及び／又はオーナー権の移行を伴うその他の技術に対しても適用される。L U の構成情報及び／又はオーナー権の移行を伴うその他の技術としては、例えば、データの移行を伴う複数の L U 間のコピー、データの移行を伴わない複数の L U 間のコピー、データの移行を伴う L U の移行、データの移行を伴わない L U の移行、等がある。これらの技術において、H D K C ユニット 1 4 1、H D K C 及び／又は管理装置 3 0 1 のプロセッサは、上記コピーや移行に応じて（コピーや移行のとき又は前に）、L U（例えば、通常の L U、D P - L U、又は、仮想 L U）の構成情報及び／又はオーナー権の移行又は通知を実施することが良い。なお、オーナー権の移行又は通知は、上記コピーや移行の後でも良い。

40

【0461】

また、データの移行を伴わない複数の L U 間のコピーや L U の移行においては、図 6 の説明において既に説明したように、同一の 1 又は複数の L D K C が複数の H D K C ユニット 1 4 1 に接続されている状態が適用される。この状態は、図 6 の説明で記載したようなフェールオーバーや負荷分散の場合に限られず、データの移行を伴わない複数の L U 間のコピーや L U の移行のためのその他の場面で適用されることも良い。この場合、予め同一の 50

1又は複数のLDKCが複数（3つ以上の場合も効果的である）のHDKCユニット141やHDKCに接続されている状態で、1つ又は複数のサーバ101、1つ又は複数のサーバ101内の仮想サーバ、1つ又は複数のスイッチ、1つ又は複数のサーバ101内の仮想スイッチの交換、コピーや移行に応じて、交換後、コピー後や移行後の1つ又は複数のサーバ101、1つ又は複数のサーバ101内の仮想サーバ、1つ又は複数のスイッチ、1つ又は複数のサーバ101内の仮想スイッチのサイトやロケーションに対応する最適なサイトやロケーションの特定のHDKCユニット141やHDKCに対して、データの移行を伴わない複数のLU間のコピーやLUの移行が実施される。この場合、HDKCユニット141、HDKC及び／又は管理装置301のプロセッサは、上記交換、コピーや移行に応じて（コピーや移行のとき又は前に）、交換後、コピー後や移行後の1つ又は複数のサーバ101、1つ又は複数のサーバ101内の仮想サーバ、1つ又は複数のスイッチ、1つ又は複数のサーバ101内の仮想スイッチのサイトやロケーションを判断し、当該サイトやロケーションに対応する最適なサイトやロケーションを判断及び決定し、特定のHDKCユニット141やHDKCを判断及び決定した上で、LU（仮想ボリューム）に関する構成情報、及び／又はLU（仮想ボリューム）に対応するLDKCのLU（仮想ボリュームにマッピングされた通常のLU又はDP-LU）へのアクセスの処理を担当する権利（LDKCが有するLUに関するオーナー権）の移行又は通知を実施することが良い（なお、オーナー権の移行又は通知は、上記コピーや移行の後でも良い）。この場合、1又は複数のLDKCが複数（3つ以上の場合も効果的である）のHDKCユニット141やHDKCのFEポートに接続されている1つ又は複数のサーバ101やスイッチ上にパス管理ソフトウェアを搭載した上で、データの移行を伴わない複数のLU間のコピーやLUの移行元のHDKCユニット141やHDKCと上記判断及び決定によって特定されたHDKCユニット141やHDKCとの間で、複数のLU（仮想ボリューム）の各々に関する構成情報、及び／又は複数のLU（仮想ボリューム）の各々に対応するLDKCのLU（仮想ボリュームにマッピングされた通常のLU又はDP-LU）に関するオーナー権を移行又は通知する。そして、この移行又は通知を実施したとき又はその前に、移行元のHDKCユニット141やHDKCのLU（仮想ボリューム）への1つ又は複数のサーバ101やスイッチからのパスから特定された（移行先の）HDKCユニット141やHDKCのLU（仮想ボリューム）への1つ又は複数のサーバ101やスイッチからのパスへ、パスを変更させる（なお、オーナー権の移行又は通知は、上記コピーや移行の後でも良い）。米国においては、この具体的な処理について、米国特許公開第2006/0271758号（Innan et al.）の明細書及び図面を援用することができる。この場合、当該明細書及び図面のHost Deviceが本願の1つ又は複数のサーバ101又はスイッチ、当該明細書及び図面のVirtualization Storage Device 1及びVirtualization Storage Device 2の各々が本願の1つのHDKCユニット141やHDKCとそれと別のHDKCユニット141やHDKC、当該明細書及び図面のExternal Storage Deviceが本願の同一のLU（第一のLU）を有するLDKC、にそれぞれ対応付けて本願技術を適用する。但し、移行先のHDKCユニット141やHDKCの特定や、複数のLU（仮想ボリューム）を纏めての移行などや、本願に従う「ティア」については、本願特有の内容であり、全く新しい技術である。

【0462】

また、本実施例2は、1つ又は複数のサーバ101やスイッチの交換や移行に伴わないLUの構成情報及び／又はオーナー権の移行又は通知を伴う技術のみに適用されるものでなく、1つ又は複数のサーバ101やスイッチの交換や移行に伴うLUの構成情報及び／又はオーナー権の移行又は通知を伴う技術に適用することも良い。この場合、HDKCユニット141、HDKC及び／又は管理装置301のプロセッサは、1つ又は複数のサーバ101やスイッチの交換や移行に応じて（交換や移行のとき又は前に）、LU（例えば、通常のLU、DP-LU、又は、仮想LU）の構成情報及び／又はオーナー権の移行又は通知を実施することが良い。なお、オーナー権の移行又は通知は、上記交換や移行の後でも良い。また、本実施例2は、1つ又は複数のサーバ101内の1つ又は複数の仮想

サーバや1つ又は複数のスイッチ内の1つ又は複数の仮想スイッチのコピーや移行に伴う技術に適用することも良い。この場合、HDKCユニット141、HDKC及び／又は管理装置301のプロセッサは、1つ又は複数の仮想サーバや仮想スイッチのコピーや移行に応じて（コピーや移行のとき又は前に）、LU（例えば、通常のLU、DP-LU、又は、仮想LU）の構成情報及び／又はオーナー権の移行又は通知を実施することが良い。なお、オーナー権の移行又は通知は、上記コピーや移行の後でも良い。

【0463】

また、本実施例2は、複数のLUの移行などの処理に応じてパスが生成又は設定される場合に限られず、複数（3つ以上の場合も効果的である）のHDKCユニット141やHDKCの複数のLU（例えば、通常のLU、DP-LU、又は、仮想LU）と1つ又は複数のサーバ101、1つ又は複数の仮想サーバ、1つ又は複数のスイッチや1つ又は複数の仮想スイッチとが予めパス生成又は設定された状態で存在しており、フェールオーバー、負荷分散や、1つ又は複数のサーバ101、1つ又は複数の仮想サーバ、1つ又は複数のスイッチや1つ又は複数の仮想スイッチの交換、コピーや移行、等に応じて、最適なサイトやロケーションの特定のHDKCユニット141やHDKCを判断及び決定して、複数のLUに関する構成情報、及び／又は複数のLUの各々へのアクセスの処理を担当する権利（LUに関するオーナー権）の移行又は通知を実施することもよい。なお、HDKCユニット141、HDKCや管理装置301のプロセッサは、フェールオーバーに関しては、パス障害に応じて、負荷分散においては負荷の分析結果に応じて、それぞれ最適なサイトやロケーションの特定のHDKCユニット141やHDKCを判断及び決定する。また、HDKCユニット141、HDKCや管理装置301のプロセッサは、1つ又は複数のサーバ101、1つ又は複数の仮想サーバ、1つ又は複数のスイッチや1つ又は複数の仮想スイッチの交換、コピーや移行に関しては、交換後、コピー後や移行後の1つ又は複数のサーバ101、1つ又は複数の仮想サーバ、1つ又は複数のスイッチや1つ又は複数の仮想スイッチのサイトやロケーションを判断し、そのサイトやロケーションに応じた最適なサイトやロケーションの特定のHDKCユニット141やHDKCを判断及び決定する。

【0464】

また、本実施例の何れの技術（代替案も含む）に対しても、1つのLU単位又は複数のLU単位（例えば、ポート単位）のいずれの単位の技術も適用して良い。さらに、第1の接続形態、第2の接続形態又は第3の接続形態のいずれのHDKCユニット141、HDKCやLDKCに対しても、1つのLU単位又は複数のLU単位（例えば、ポート単位）のいずれの単位の技術も適用して良い。第3の接続形態に限られず、第1の接続形態や第2の接続形態においても、本願に従う「ティア」を維持した状態での処理であれば、既存の技術にない全く新しい技術であるからである。この場合、HDKCユニット141、HDKC、LDKCや管理装置301のプロセッサが、例えば、本願に従う複数の「ティア」の各々に対応するLU（例えば、通常のLU、DP-LU、又は、仮想LU）の中からティアレベル（ティアレベルの範囲であっても良い）の変更しない適切な「ティア」を有するLUを選択、判断及び決定すること、を一例とする新たな処理を行う。

【0465】

なお、本実施例において、仮想LUが移行される場合には、この仮想LUにマッピングされる外部LUのID等の情報も移行先のHDKCユニット141、HDKCやLDKC151に対して引き継がれるが、この場合、仮想LUにマッピングされる外部LUが1つLDKCが有する1つの外部LUの場合には、この1つの外部LUのID等の情報が引き継がれることになり、仮想LUにマッピングされる外部LUがLDKCの各々から1つ又は複数つつ提供される外部LU（複数のLDKCにまたがって提供される複数の外部LU）の場合には、複数の外部LUのID等の情報が引き継がれることになる。

【実施例3】

【0466】

本発明の実施例3を説明する。その際、実施例1又は実施例2との相違点を主に説明し、実施例1又は実施例2との共通点については説明を省略或いは簡略する。

【0467】

第3の接続形態に従うHDKC群（複数のHDKCユニットの集合）にダイナミックプロビジョニング（シンプロビジョニング）が適用された記憶システムにおいて、全てのHDKC間でプール（実ページの集合）を共有することで、容量効率の向上、及び、より多くのPDEVを使用することによる性能向上が期待できる。複数のHDKCユニットがプールを共有するためには、プールの未割当キューを共有することを意味し、実ページ管理テーブルを全てのHDKCユニットで共有する必要がある。

【0468】

しかし、実ページ管理テーブルが、特定のHDKC内のCM（又は、共有情報管理装置301（図7参照））に配置された場合、DPLUに対するI/Oの際に、外部バスを通じて実ページ管理テーブルを参照する必要があるため、DPLUに対するI/Oの性能が低下してしまう。

10

【0469】

実施例3によれば、HDKCユニット間でプールを共有しても、DPLUに対するI/Oの性能の低下を軽減することができる。さらに、これにより、初期導入費用を削減しつつ、小規模構成から大規模な構成へと柔軟な且つ効率的なスケールアウトを実現できる技術を提供できる。

【0470】

以下、実施例3を詳細に説明する。

【0471】

20

図50は、HDKCユニット間でのプール共有の一例を示す。以下、説明を分かり易くするため、プールを共有するHDKCユニットの数は2とするが、プールを共有するHDKCユニットの数は3以上でも良い。また、HDKCユニットを構成するHDKCの数は1であるとする。具体的には、HDKC#H0に、第3の接続媒体（外部バス）601を介してHDKC#H1が接続されているとする。HDKC#H0及び#H1の少なくとも一方に、第2の接続媒体（外部バス）を介してLDKCが接続されても良い。しかし、プールを共有するHDKCユニットを構成するHDKCの数は、2以上であっても良い。また、プールを共有する2以上のHDKCユニットの少なくとも1つのHDKCユニットにおける1以上のHDKCに、1以上のLDKCが第2の接続形態で接続されていても良い。

30

【0472】

プール1500は、HDKC#H0とHDKC#H1に共有される。プール1500は、HDKC#H0が有する1以上の内部ボリュームと、HDKC#H1が有する1以上の内部ボリュームで構成される。「内部ボリューム」とは、既に説明した通り、物理的な記憶容量が存在しているボリュームのことであり、例えば、1つの通常のLU、通常のLUの一部、複数の通常のLUのグループ、第2の接続形態に従う1つの仮想LU、仮想LUの一部、複数の仮想LUのグループ、1つのRG、RGの一部、或いは、複数のRGのグループで良い。ティアの概念（例えば、実施例1で説明されたティアレンジ又はティアレンジ範囲）は、「内部ボリューム」にも適用される。例えば、「内部ボリューム」が1つのRG単位で管理される場合、ティアの概念（例えば、実施例1で説明されたティアレンジ又はティアレンジ範囲）も1つのRG単位で管理される。

40

【0473】

本実施例では、プール1500を構成する複数の内部ボリュームのうち、HDKC#H0が有する1以上の内部ボリュームは、RG#H00、#H01及び#H02であり、HDKC#H1が有する1以上の内部ボリュームは、RG#H13、#H14及び#H15である（つまり、本実施例では、内部ボリュームはRGである）。この構成によれば、HDKC#H0（内部コントローラ#H0（CPU#H0））が、RG#H13、#H14及び#H15に第3の接続媒体601を介してアクセスすることが可能である。同様に、HDKC#H1（内部コントローラ#H1（CPU#H1））が、RG#H00、#H01及び#H02に第3の接続媒体601を介してアクセスすることが可能である。プール

50

を共有する複数のHDKCユニットが有する複数の内部ボリュームには、それぞれ、それら複数のHDKCユニットにおいてユニークな識別情報が設定されて良い。また、HDKC#H0が、RG#H00～#H02に関する情報に加えて、第3の接続形態で接続されているHDKC#H1内のRG#H13～#H15に関する情報を、例えば、HDKC#H1が第3の接続形態で接続されたときに取得し保持することができる。

【0474】

また、プール1500は、そのプール1500を共有するHDKCユニット141の数に応じた数の記憶領域（以下、プール領域）に分割される。例えば、プール領域の数は、プール1500を共有するHDKCユニットの数と同じである。図50に示す例では、プール1500は、2つのHDKCユニット141（HDKC#H0及び#H1）に共有されるので、プール1500は、2つのプール領域5011及び5012に分割される。プール領域5011が、HDKC#H0に関連付けられ、プール領域5012が、HDKC#H1に関連付けられる。具体的には、例えば、プール領域5011を構成する全ての実ページは、実ページ管理テーブル#H0において、使用不可と管理されず、実ページ管理テーブル#H1において、使用不可と管理される。一方、プール領域5012を構成する全ての実ページは、実ページ管理テーブル#H0において、使用不可と管理され、実ページ管理テーブル#H1において、使用不可と管理されない。

10

【0475】

HDKC#H0が有するDPU（例えば#160）の仮想ページには、プール領域5011から実ページが割り当てられる。一方、HDKC#H1が有するDPU（例えば#161）の仮想ページには、プール領域5012から実ページが割り当てられる。

20

【0476】

このように、プール1500は複数のHDKC#H0及び#H1に共有可能であるが、プール1500を構成する実ページ毎に、その実ページを使用できるHDKCは1つに限られている。このため、1つの実ページに対して複数のHDKCからI/Oが競合することが無い。

【0477】

本実施例の説明においては、各プール領域（5011、5012）は、プール1500を構成する複数又は全てのRGにまたがる。すなわち、いずれのプール領域も、プール1500を構成する各RGの一部の記憶領域を含んでいる。言い換えれば、各プール領域は、プール1500を構成する各RGの一部の記憶領域で構成されている。1つのRGにI/Oが集中することを防ぐために、仮想ページに割り当てられる実ページのソース（以下、割当元RG）は、次のように選択されて良い。

30

（＊）例えば、プールを構成するN個のRGがあり（Nは3以上の整数）、且つ、そのプールをM個のHDKCユニットが共有している場合（Mは2以上の整数、且つ、N>M）、M個のHDKCから初めに選択されるM個の割当元RG（例えば、M個の割当元RGの番号（例えば通し番号））が、均等に離れていて良い。図50の例によれば、N=6であり、M=2である。このため、HDKC#H0（内部コントローラ#H0（CPU#H0））が初めに選択する割当元RGがRG#H00の場合、HDKC#H1（内部コントローラ#H1（CPU#H1））が初めに選択する割当元RGは、RG#13であって良い。

40

（＊）その後、少なくとも1つのHDKCが、ラウンドロビン等の所定の規則に従って、選択する割当元RGを切り替えて良い（例えば、選択する割当元RGを均等に違って良い）。例えば、HDKC#H0は、割当元RGを、RG#H00、#H01、#H02、#H13、#H14及び#H15の順で切り替え、HDKC#H1は、割当元RGを、RG#H13、#H14、#H15、#H00、#H01及び#H02の順で切り替えて良い。

（＊）ラウンドロビン等の所定の規則に従って割当元RGを選択していくと、2以上のHDKCが選択する割当元RGがK回（Kは自然数）同じになることがあり得る（例えば、K回連続して同じになることがあり得る）。この場合、それら2以上のHDKCのうちの

50

少なくとも1つのHDKCが、所定の規則に従って選択することに割当元RGと異なる割当元RGを選択し、その時点以降に、再び、所定の規則に従って割当元RGを選択して良い。

【0478】

なお、これらの選択制御は、共有プールを管理する複数のHDKCユニット141のいずれか1つ以上のプロセッサ222や管理装置301内のプロセッサによって管理されて、割当元RGの割り当てや変更が生じる場合には、これらのいずれかのプロセッサから割当元RGの割り当てや変更を実際に制御するHDKCユニット141やHDKCのプロセッサ222に対して当該制御の指示コマンドが発行されることにより実施される。

【0479】

LM#H0(#H1)は、仮想ページ管理テーブル325(#H0(#H1))を記憶する。CM(SM領域)#H0(#H1)は、実ページ管理テーブル327(#H0(#H1))、キュー先頭管理テーブル326(#H0(#H1))、及び、未割当ページ使用量管理テーブル5001(#H0(#H1))を記憶する。それらの様々なテーブルを含む様々な管理情報は、プール1500にも配置されることもでき、また、プール1500の外部の1以上のストレージ媒体の1以上のストレージ領域に配置されることもできる。さらに、HDKCユニット141の各々は、プール1500の外部の1以上のストレージ媒体の1以上のストレージ領域を有することが出来、プール1500の外部の1以上のストレージ媒体の1以上のストレージ領域に、管理情報だけでなく、サーバ101からプール1500外のあるLUに対して送られたデータを格納することもできる。

【0480】

未割当ページ使用量管理テーブル#H0(#H1)は、HDKC#H0(#H1)に割り当てられたプール領域5011(5012)内の未割当の実ページ(仮想ページに割り当てられていない実ページ)に関する情報を有する。未割当ページ使用量管理テーブル#H0(#H1)は、具体的には、例えば、図51に示すように、下記の情報、

(*) プール領域5011(5012)内の未割当ての実ページの数である未割当ページ数5101、

(*) 直前回のプールメンテナンス処理(後述)で算出された未割当ページ数5101である前回未割当ページ数5102、

(*) 前回未割当ページ数5102から未割当ページ数5101を引いた値である未割当ページ使用量5103、
を有する。

【0481】

なお、本実施例で説明される技術は、1つ又は複数のDP-LUへの割り当てに利用される複数の実ページを管理するプール1500を、複数のプール領域(5011、5012)に論理的に分割して管理及び／又は制御する技術であり、キャッシュメモリ224の記憶領域を論理的に分割してHDKCユニット141、HDKC及び／又はLDKC151の1つ又は複数の各々に割り当ててデータの格納に利用する技術(例えば、プレキャッシュ技術)、及び／又はキャッシュメモリ224の記憶領域を論理的に分割して通常のLU、DP-LU及び／又は仮想LUの1つ又は複数の各々に割り当ててデータの格納に利用する技術(例えば、通常のLU、DP-LU及び／又は仮想LUの1つ又は複数毎にサービスレベルを確保するためのキャッシュメモリの論理パーティション技術)とは全く異なるものである。本実施例は、これらの技術と独立して利用してもよく、これらの技術のいずれか又は双方と一緒に利用しても良い。

【0482】

第3の接続形態におけるシンプロビジョニングでは、プールが多くのRGを有する方がI/Oの性能として有利である。しかし、そのプールにおいて、或るRGが属するティア(ティアレベル又はティアレベル範囲)と別のRGが属するティアの差が大きいと、DP-LUに対するI/Oの性能が、下位のティアに属するRGの性能に引きずられてしまう。特に、性能ファクターとして、「距離(コマンド応答時間)」は、本実施例において、

D P - L U に対する I / O の性能に大きく影響する。

【0483】

そこで、本実施例に係るティア管理テーブルは、例えば、図52A及び52Bに示すように、内部ボリューム（R G）毎に、距離（コマンド応答時間）1008を有する。

【0484】

なお、図52Aは、プール1500をHDKC#H0及び#H1で共有して差し支えない状況に係るティア管理テーブル#H0を示し、図52Bは、プール1500をHDKC#H0及び#H1で共有することが好ましくない状況に係るティア管理テーブル#H0を示す。なお、図11には、距離（コマンド応答時間）1008が「5」となる例は開示されていないが、例えば、HDKC間の距離が400km～500kmの場合に、距離（コマンド応答時間）1008が「5」となって良い。

10

【0485】

図52Aによれば、HDKC#H0内のR G#H00～#H02のティアレベルと、HDKC#H1内のR G#H13～#H15のティアレベルとの差の最大値「1」が、所定の閾値以下である。この場合、HDKC#H0及び#H1でプール1500を使用することが許可される。

【0486】

図52Bによれば、HDKC#H0内のR G#H00～#H02のティアレベルと、HDKC#H1内のR G#H13～#H15のティアレベルとの差の最大値「6」が、所定の閾値を越えている。この場合、HDKC#H0及び#H1でプール1500を使用することが禁止される。

20

【0487】

以下、本実施例で行われる処理を説明する。

【0488】

図53は、実施例3に係るプール作成処理の流れを示す。なお、以下、マスタCPUが処理を行うこととするが、他のCPUが行っても良い。

【0489】

S5301で、マスタCPU#H0が、プール作成指示を受信する。その指示は、管理者に対するGUIを通じて受けても良いし、管理計算機から受けても良い。プール作成指示は、複数のHDKCユニットで共有するプールを作成することの指示である。プール作成指示は、例えば、複数のHDKCユニットとして、HDKC#H0及び#H1のDKCIDを含んで良い。また、プール作成指示は、プールを構成するR Gの識別情報、例えば、R G#H00、#H01、#H02、#H13、#H14及び#H15のR G IDを含んで良い。

30

【0490】

なお、マスタCPU#H0は、プール作成指示を受けたことに代えて、特定の条件が満たされたことを検出したときに、自動的に、S5302以降の処理を行なって良い。

【0491】

S5302で、マスタCPU#H0が、ティア管理テーブル#H0を基に、複数のプールR G（#H00、#H01、#H02、#H13、#H14及び#H15）のティアレベルの差の最大値が所定の閾値以下か否かを判断する。

40

【0492】

ティア管理テーブル#H0は、第3の接続形態に従う仮想LUにマッピングされている外部LUの基になっているR Gのティア（第3の接続形態で接続されているHDKC#H1内のR Gのティア）に関する情報を有することができる。そのティア管理テーブル#H0から、HDKC#H0内のR Gのティアレベルだけでなく、HDKC#H1内のR Gのティアレベルも分かる。

【0493】

なお、マスタCPU#H0は、HDKC#H1から、ティア管理テーブル#H1が有する情報（例えば、HDKC#H0内の仮想LU（第3の接続形態に従う仮想LU）にマッ

50

ピングされている外部LUのティアレベル)を受信し、その情報と、第3の接続形態に従う仮想LUのティアレベルとを基に、HDKC#H1内のプールRG毎に、ティアレベルを算出して良い。

【0494】

上述のS5302の判断の結果が否定的の場合(S5302:No)、つまり、複数のプールRGのティアレベルの差の最大値が所定の閾値を超えている場合、それら複数のプールRGで構成された共有プール(HDKC#H0及びH1が共有するプール)を作成することが好ましくないということである。そこで、S5320で、マスタCPU#H0が、HDKC毎にプールを作成し直すよう管理者に通知して良い。或いは、管理者への通知を行わず(又は管理者への通知を行って)、マスタCPU#H0及び#H1が、自動で、プールを構築して良い。例えば、マスタCPU#H0は、管理者からの指示に従い、或いは、管理者からの指示を受けること無しに自動的に、HDKC#H0内のプールとして、複数のプールRGのうちのHDKC#H0が有するRG#H00～#H02で構成されたプールを作成して良い。また、マスタCPU#H1は、管理者からの指示に従い、或いは、管理者からの指示を受けること無しに自動的に、HDKC#H1内のプールとして、複数のプールRGのうちのHDKC#H1が有するRG#H13～#H15で構成されたプールを作成して良い。この場合であっても、次のような例を適用することも良い。HDKC#H0は、HDKC#H0のDP-LUのアロケーションのために自装置の全ての未割当ページを使用した場合、HDKC#H1の未割当ページを使用できる。HDKC#H1は、HDKC#H1のDP-LUのアロケーションのために自装置の全ての未割当ページを使用した場合、HDKC#H0の未割当ページを使用できる。米国においては、US Patent Application Publication No. 2008/0184000 (Kawaguchi et al.)の明細書及び図面を上記未割当ページの使用に対して援用することが出来る。この場合、HDKC#H0は、Kawaguchi et al.のストレージモジュール11100の処理を制御し、HDKC#H1は、Kawaguchi et al.のストレージモジュール11200の処理を制御する。

【0495】

上述のS5302の判断の結果が肯定的の場合(S5302:Yes)、つまり、複数のプールRGのティアレベルの差の最大値が所定の閾値以下の場合、それら複数のプールRGで構成された共有プールを作成して差し支えないということである。この場合、以下の処理が行われる。

【0496】

すなわち、S5303で、マスタCPU#H0が、実ページ管理テーブル#H0を更新する。具体的には、例えば、マスタCPU#H0が、プールRG(#H00、#H01、#H02、#H13、#H14及び#H15)について、テーブルポインタ1401、プールRG ID1402、及び実ページアドレス1403を設定し、且つ、割当状態1404として「未割当」を設定する(図54参照)。

【0497】

S5304で、マスタCPU#H0が、実ページ管理テーブル#H0が有する情報(プールRGについての情報1401～1404)を、HDKC#H1に送信する。その情報は、第3の接続媒体(外部バス)を介してHDKC#H1に届く。

【0498】

S5311で、マスタCPU#H1が、実ページ管理テーブル#H0が有する情報(プールRGについての情報1401～1404)をHDKC#H0から受信し、受信した情報を、実ページ管理テーブル#H1に書き込む(図55参照)。この時点で、実ページ管理テーブル#H1の内容は、実ページ管理テーブル#H0と同じである(図54及び図55参照)。

【0499】

S5305で、マスタCPU#H0は、実ページ管理テーブル#H0において、偶数RGの奇数実ページに対応した割当状態1404を「使用不可」に更新し、且つ、奇数RGの偶数実ページに対応した割当状態1404を「使用不可」に更新する(図57参照)。

一方、S 5 3 1 2で、マスタCPU # H 1が、実ページ管理テーブル # H 1において、偶数R Gの偶数実ページに対応した割当状態1 4 0 4を「使用不可」に更新し、且つ、奇数R Gの奇数実ページに対応した割当状態1 4 0 4を「使用不可」に更新する（図5 8参照）。「偶数R G」とは、プールR G I D 1 4 0 2の末尾が偶数（0を含む）であるR Gであり、「奇数R G」とは、プールR G I D 1 4 0 2の末尾が奇数であるR Gである。

「偶数実ページ」とは、実ページアドレス1 4 0 3の末尾が偶数（0を含む）である実ページであり、「奇数実ページ」とは、実ページアドレス1 4 0 3の末尾が奇数である実ページである。

【0 5 0 0】

S 5 3 0 5及びS 5 3 1 2によれば、複数のプールR G（# H 0 0、# H 0 1、# H 0 2、# H 1 3、# H 1 4及び# H 1 5）で構成されたプールが、半分に2つのプール領域に分割される（例えば、図5 0、図5 7及び図5 8参照）。一方のプール領域が、H D K C # H 0で使用可能であるがH D K C # H 1で使用不可とされ、他方のプール領域が、H D K C # H 1で使用可能であるがH D K C # H 0で使用不可とされる。つまり、プールが2つのプール領域に均等に分割され、各実ページについて、H D K C # H 0及び# H 1のうちの1つだけが使用可能とされる。

【0 5 0 1】

なお、S 5 3 0 5及びS 5 3 1 2によれば、2以上の不連続の実ページの割当状態1 4 0 4が「使用不可」とされるが、2以上の連続した実ページの割当状態1 4 0 4が「使用不可」とされても良い。例えば、プールの前半の実ページ群（アドレスが連続した実ページ群（プール領域））が、H D K C # H 0に割り当てられ、プールの後半の実ページ群（アドレスが連続した実ページ群（プール領域））が、H D K C # H 1に割り当てられて良い。すなわち、プールを構成する全てのR Gにまたがるようにプール領域が得られさえすれば、プールはどのように分割されても良い。

【0 5 0 2】

また、複数のH D K Cユニットにそれぞれ割り当てられる複数のプール領域の容量は、均等でなくても良い。例えば、複数のH D K Cユニットにおいて、第1のH D K Cユニットに、第1のプール領域が割り当てられ、第1のH D K CユニットよりもI / Oコマンドを受ける頻度の低い第2のH D K Cユニットに、第1のプール領域よりも容量の小さい第2のプール領域が割り当てられて良い。この例は、第2のH D K Cユニットが第1のH D K CユニットよりもI / Oコマンドを受ける頻度の低いという条件に代えて又は加えて、図1 0のティア管理テーブルのコントローラ性能の値に基づいて第1のH D K Cユニットのコントローラ性能が第2のH D K Cユニットのコントローラ性能に比べて高いという条件でも適する。また、H D K Cユニット141の各々に含まれる複数の実ストレージエリアの容量は、図8の接続管理テーブルの1以上のエレメント及び／又は図9及び／又は図1 0のティア管理テーブルの1以上のエレメントに基づいて決定される重み付けパラメータに応じて、複数のH D K Cユニット1 4 1に分割及び分配される。例えば、図1 0のティア管理テーブルの距離（コマンド応答時間）のちに基づいて複数のH D K Cユニット1 4 1の第1のH D K Cユニットと第2のH D K Cユニットとの距離が遠い場合（コマンド応答性能に時間がかかる場合）、第1のH D K Cユニットに含まれる複数の実ストレージエリアの容量のうちの第1の容量は、第1のH D K Cユニットに割り当てられ、第1のH D K Cユニットに含まれる複数の実ストレージエリアの容量のうちの第2の容量（第1の容量よりも容量の小さい容量）は、第2のH D K Cユニットに割り当てられる。この例の場合、複数のH D K Cユニット1 4 1の第2のH D K Cユニットに含まれる複数の実ストレージエリアの容量のうちの第3の容量は、第2のH D K Cユニットに割り当てられ、第2のH D K Cユニットに含まれる複数の実ストレージエリアの容量のうちの第4の容量（第3の容量よりも容量の小さい容量）は、第1のH D K Cユニットに割り当てられる。これらの技術は、H D K Cユニット1 4 1、H D K C及び／又は管理装置3 0 1のプロセッサによって制御される。また、これらの技術は、1つの統合プールを共有する3台以上の複数のH D K Cユニット1 4 1にも適用でき、3台以上の複数のH D K Cユニット1 4 1

10

20

30

40

50

の各々に異なるサイズの容量を割り当てることができる。

【0503】

その後、HDKC#H0及び#H1は、それぞれ、割当状態1404が「未割当」の実ページについて、未割当キューを作成する。未割当キューの実ページの並びは、例えば、実ページ毎にRGがラウンドロビンする並び（つまり、実ページを有するRGが順次に異なっている並び）とされる。

【0504】

具体的には、例えば、S5306で、マスタCPU#H0が、初めに割り当てられる実ページのソースをRG#H00とし、且つ、未割当キューの実ページの並びが、実ページ毎にRGがラウンドロビンするように、割当状態1404が「未割当」に対応した未割当キューポインタ1406を更新する。S5307で、マスタCPU#H0が、HDKC#H0の未割当キューの先頭に対応したテーブルポインタ1401を、キュー先頭管理テーブル#H0に登録する。つまり、キュー先頭管理テーブル#H0が有する情報（先頭ポインタ）が、図56Aに示す先頭ポインタから図59Aに示す先頭ポインタに変わる。

10

【0505】

一方、例えば、S5313で、マスタCPU#H1が、初めに割り当てられる実ページのソースをRG#H13とし、且つ、未割当キューの実ページの並びが、実ページ毎にRGがラウンドロビンするように、割当状態1404が「未割当」に対応した未割当キューポインタ1406を更新する。S5314で、マスタCPU#H1が、HDKC#H1の未割当キューの先頭に対応したテーブルポインタ1401を、キュー先頭管理テーブル#H1に登録する。つまり、キュー先頭管理テーブル#H1が有する情報（先頭ポインタ）が、図56Bに示す先頭ポインタから図59Bに示す先頭ポインタに変わる。

20

【0506】

S5306及びS5313によれば、未割当キューの先頭に対応するRGは、HDKC#H0と#H1で別々になる。HDKC#H0及び#H1は、それぞれ、未割当キューに従い、ラウンドロビンで、仮想ページに割り当てる実ページを選ぶことが良い。これにより、プール内のRGが均等に使用され、且つ、HDKC#H0及び#H1から同一のRGから実ページが選択されるといった集中を回避することが期待できる。

【0507】

つまり、プール1500内から分割された各々のプール領域5011（又は5012）においては、仮想ページに割り当てられる実ページは、n回前（nは自然数）に割り当てられた実ページの基になっているRGとは異なるRGが基になっている実ページで良い（例えば、n=1で良い）。具体的には、例えば、ラウンドロビン形式で、割り当てられる実ページの基になっているRGが切り替わって良い。これにより、特定のRGにアロケーションに伴うI/Oが集中してしまうことを回避でき、以って、DKCのI/O性能の向上が期待できる。1つのプール1500に関連付けられるDP-LU又は複数DP-LUのグループ（DP-LUグループ）が、DP-LU又はDP-LUグループ毎に異なるティアに属している場合（例えば、第一のDP-LU又は第一のDP-LUグループは第一のティアに属し、第二のDP-LU又は第二のDP-LUグループは第二のティアに属している場合）、ティア別に、n回前に割り当てられた実ページの基になっているRGと異なるRGから実ページが割り当てられて良い（例えば、第一のDP-LU又は第一のDP-LUグループは第一のティアを構成する複数のRGの範囲内において、ラウンドロビン形式で、割り当てられる実ページの基になっているRGが切り替わり、第二のDP-LU又は第二のDP-LUグループは第二のティアを構成する複数のRGの範囲内において、ラウンドロビン形式で、割り当てられる実ページの基になっているRGが切り替わる）。

30

40

【0508】

以上が、プール作成処理の流れである。HDKC#H0及び#H1が、このプール作成処理で作成された実ページ管理テーブル#H0及び#H1を基にI/O処理（DP-LUに対するI/O）を行うことで、プール1500を構成する全RGから均等に未割当実ページを割り当てることができる。これにより、DP-LUに対するI/Oの性能の低下を

50

軽減することができる。さらに、本実施例の一つの統合プールとしてのプール1500によって、複数のHDKCユニット141は各々のHDKCユニット141の負荷を分散できる。さらに、この実施例と第2の実施例と一緒に使用することによって、複数のHDKCユニット141間の負荷分散処理はより効果的なものとなる。さらに、プール作成処理は、複数のHDKCユニット141の未割当ストレージ領域を複数のHDKCユニット141で共有するため、実ストレージ領域の無駄を効果的に削減できるとともに、複数のHDKCユニット141を含むストレージシステムに費やされる費用を効果的に削減できる。

【0509】

なお、本実施例で言う「未割当実ページ」とは、仮想ページに割り当てられていない実ページであって、仮想ページに割り当て可能な実ページ（割当状態1404が「使用不可」とされていない実ページ）である。

【0510】

以下、プール作成処理後のI/O処理（DP-LUに対するデータのI/Oの処理）の流れを説明する。その際、適宜、図66乃至図69を参照する。なお、以下の説明では、下記の前提があるとする。HDKC#H0及#H1のいずれも、2つ多くのDP-LUを有することができる。

（*）HDKC#H0は、2つのDP-LU#160及び#162を有する。

（*）HDKC#H1は、2つのDP-LU#161及び#163を有する。

（*）DP-LU#160及び#162に、プール領域5011が関連付けられている。

（*）DP-LU#161及び#163に、プール領域5012が関連付けられている。

【0511】

例えば、HDKC#H0が、DP-LU#160を指定したライトコマンドを受信したとする。この場合、HDKC#H0の内部コントローラ#H0が、以下の処理の行うことができる。

（s1）内部コントローラ#H0は、受信したライトコマンドで指定されているアドレスが属する仮想ページ（以下、仮想ページ（0x02））を特定する。

（s2）内部コントローラ#H0は、仮想ページ管理テーブル#H0を基に、仮想ページ（0x02）に実ページが割り当てられているか否かを判断する。

（s3）内部コントローラ#H0は、上記（s2）の判断の結果が否定的の場合に、DP-LU#160が関連付けられているプール領域5011から、キュー先頭管理テーブル#H0の先頭ポインタと一致する未割当てキューポインタに対応した実ページ（RAIDグループ#H15に基づく実ページ（0x01））を、仮想ページ（0x02）に割り当てる。具体的には、例えば、内部コントローラ#H0は、下記の処理、

（*）RAIDグループ#H15に基づく実ページ（0x01）に対応した割当状態1404を「割当」に更新し、その実ページ（0x01）に対応したDP-LU ID1405を「160」に更新する（図67参照）、

（*）DP-LU#160の仮想ページ（0x02）に対応した割当状態1303を「割当」に更新し、その仮想ページ（0x02）に対応したプールRG ID1304を「H15」に更新し、且つ、その仮想ページ（0x02）に対応した実ページアドレス1305を「0x01」に更新する（図66参照）、

（*）キュー先頭管理テーブル#H0の先頭ポインタを、「0x0002」に更新する、を行う。

【0512】

また、例えば、プール領域5012が作成された後、初めて、HDKC#H1が、ライトコマンドを受信したとする。この場合、HDKC#H1の内部コントローラ#H1が、以下の処理の行うことができる。

（s11）内部コントローラ#H1は、受信したライトコマンドで指定されているアドレスが属する仮想ページ（以下、DP-LU#161の仮想ページ（0x02））を特定する。

10

20

30

40

50

(s 1 2) 内部コントローラ#H 1 は、仮想ページ管理テーブル#H 1 を基に、仮想ページ(0 x 0 2)に実ページが割り当てられているか否かを判断する。

(s 1 3) 内部コントローラ#H 1 は、上記(s 1 2)の判断の結果が否定的の場合に、D P-L U#1 6 1 が関連付けられているプール領域5 0 1 2 から、キュー先頭管理テーブル#H 1 の先頭ポインタと一致する未割当てキューポインタに対応した実ページ(R A I Dグループ#H 1 3に基づく実ページ(0 x 0 0))を、D P-L U#1 6 1 の仮想ページ(0 x 0 2)に割り当てる。具体的には、例えば、内部コントローラ#H 1 は、下記の処理、

(*) R A I Dグループ#H 1 3に基づく実ページ(0 x 0 0)に対応した割当状態1 4 0 4を「割当」に更新し、その実ページ(0 x 0 0)に対応したD P-L U I D 1 4 0 5を「1 6 1」に更新する(図6 9参照)、

10

(*) D P-L U#1 6 1 の仮想ページ(0 x 0 2)に対応した割当状態1 3 0 3を「割当」に更新し、その仮想ページ(0 x 0 2)に対応したプールR G I D 1 3 0 4を「H 1 3」に更新し、且つ、その仮想ページ(0 x 0 2)に対応した実ページアドレス1 3 0 5を「0 x 0 0」に更新する(図6 8参照)、

(*) キュー先頭管理テーブル#Hの先頭ポインタを、「0 x 4 0 0 1」に更新する、を行う。

【0 5 1 3】

ところで、本実施例では、共有されているプール内のプール領域別に、未割当実ページが消費される(未割当実ページが仮想ページに割り当てられる)。このため、未割当ページ使用量5 1 0 3(すなわち、消費された未割当実ページの数)(図5 1参照)が、プール領域毎に異なり得る。故に、例えば、プール領域5 0 1 1に未割当実ページが不足しており、プール領域5 0 1 2に未割当実ページが余っていることが発生し得る。

20

【0 5 1 4】

そこで、本実施例では、更なる工夫が採用されている。具体的には、共有されているプール1 5 0 0についてのメンテナンス処理(以下、プールメンテナンス処理)が行われる。この処理は、定期的又は不定期的に行われる。これにより、複数のプール領域の未割当ページ使用量5 1 0 3(残りの実ストレージ領域の使用量)が均等になるよう制御される。プールメンテナンス処理は、未割当実ページが不足している特定のH D K Cユニットの発生を効果的に回避することができるために、高い信頼性を実現できる。さらに、プールメンテナンス処理は、複数のH D K Cユニット1 4 1の未割当ストレージ領域を複数のH D K Cユニット1 4 1で共有することにより、実ストレージ領域の無駄を効果的に削減できるとともに、複数のH D K Cユニット1 4 1を含むストレージシステムに費やされる費用を効果的に削減できる。

30

【0 5 1 5】

なお、未割当実ページの数、一般に、数時間のうちに大きく変動することは少ない。そのため、プールメンテナンス処理の実施頻度は、例えば、数日～数週間に1回で良いと考えられる。この程度の頻度であれば、プールメンテナンス処理が、D P-L Uに対するI/Oの性能に大きく影響することは無いと考えられる。プールメンテナンス処理は、管理者任意の頻度(管理者が設定したパラメータ)に従って開始されて良いし、管理者から手動で明示的な指示がある都度に開始されても良い。

40

【0 5 1 6】

図6 0乃至図6 1は、プールメンテナンス処理の流れを示す。なお、以下、マスタC P Uが処理を行うこととするが、他のC P Uが行っても良い。つまり、複数のH D K Cユニット1 4 1、H D K C及び/又は管理装置3 0 1のプロセッサが行っても良い。この点は、本命最初の何れの実施例においても同様である。

【0 5 1 7】

例えば、プールメンテナンス処理の開始の時点で、マスタC P U#H 0 が、S 6 0 0 1を開始し、マスタC P U#H 0 1が、S 6 0 5 1を開始する。

【0 5 1 8】

50

S 6 0 0 1で、マスタCPU#H0が、下記、
(6 0 0 1 a) 実ページ管理テーブル#H0を基に、HDKC#H0の未割当キューに属する実ページの数（つまり、プール領域5 0 1 1内の未割当実ページの数）をカウントする、

(6 0 0 1 b) 未割当ページ使用量管理テーブル#H0内の未割当ページ数5 1 0 1を、前回未割当ページ数5 1 0 2として、未割当ページ使用量管理テーブル#H0に設定する、

(6 0 0 1 c) 上記(6 0 0 1 a)でカウントされた数を表す情報を、未割当ページ数5 1 0 1として、未割当ページ使用量管理テーブル#H0に設定する、

を行う。そして、S 6 0 0 2で、マスタCPU#H0が、前回未割当ページ数5 1 0 2（上記(6 0 0 1 b)で設定した値）から未割当ページ数5 1 0 1（上記(6 0 0 1 c)で設定した値）を引いた値である未割当ページ使用量5 1 0 3を、未割当ページ使用量管理テーブル#H0に設定する。

10

【0 5 1 9】

同様に、S 6 0 5 1で、マスタCPU#H1が、下記、

(S 6 0 5 1 a) 実ページ管理テーブル#H1を基に、HDKC#H1の未割当キューに属する実ページの数（つまり、プール領域5 0 1 2内の未割当実ページの数）をカウントする、

(S 6 0 5 1 b) 未割当ページ使用量管理テーブル#H1内の未割当ページ数5 1 0 1を、前回未割当ページ数5 1 0 2として、未割当ページ使用量管理テーブル#H1に設定する、

20

(S 6 0 5 1 c) 上記(6 0 5 1 a)でカウントされた数を表す情報を、未割当ページ数5 1 0 1として、未割当ページ使用量管理テーブル#H1に設定する、

を行う。そして、S 6 0 5 2で、マスタCPU#H1が、前回未割当ページ数5 1 0 2（上記(6 0 5 1 b)で設定した値）から未割当ページ数5 1 0 1（上記(6 0 5 1 c)で設定した値）を引いた値である未割当ページ使用量5 1 0 3を、未割当ページ使用量管理テーブル#H1に設定する。

【0 5 2 0】

S 6 0 0 3で、マスタCPU#H0は、HDKC#H1に、使用量情報コマンドを送信する。使用量情報コマンドは、未割当ページ使用量管理テーブル#H1が有する情報を取得するためのコマンドである。

30

【0 5 2 1】

S 6 0 5 3で、マスタCPU#H1が、HDKC#H0からの使用量情報コマンドを受信し、未割当ページ使用量管理テーブル#H1が有する情報（例えば、情報5 1 0 1～5 1 0 3）を、HDKC#H0に送信する。

【0 5 2 2】

マスタCPU#H0が、未割当ページ使用量管理テーブル#H1が有する情報をHDKC#H1から受信する。マスタCPU#H0は、その受信した情報（未割当ページ使用量管理テーブル#H1が有する情報）と、未割当ページ使用量管理テーブル#H0とを基に、理想的な未割当実ページ数を算出する。「理想的な未割当実ページ数」は、例えば、下記の（観点A）及び（観点B）のいずれに従う数でも良いが、プール領域毎に未割当ページ使用量5 1 0 3が異なるため、（観点A）よりも（観点B）の方が好ましいと考えられる。なぜなら、各プール領域に、そのプール領域の未使用実ページの消費頻度に応じた数の未使用実ページを設置することができるからである。

40

（観点A）全てのプール領域の未割当実ページ数が同じになるようにする。

（観点B）プール領域のページ比率と同じ比率でそのプール領域に未割当実ページが存在するように、プール全体についての未割当実ページの数が複数のプール領域に分配される。

【0 5 2 3】

S 6 0 0 4で、マスタCPU#H0が、未割当ページ使用量管理テーブル#H1が有す

50

る情報と、未割当ページ使用量管理テーブル#H0とを基に、プール領域5011（すなわち、HDKC#H0が使用するプール領域）のページ比率を算出する。プール領域5011のページ比率は、全てのプール領域の未割当ページ使用量5103の合計に対する、プール領域5011の未割当ページ使用量5103の割合である。従って、例えば、プール領域5011のページ比率は、下記の式、

（プール領域5011のページ比率）＝

（プール領域5011の未割当ページ使用量5103）／{（プール領域5011の未割当ページ使用量5103）＋（プール領域5012の未割当ページ使用量5103）}
で算出される。

【0524】

10

そして、S6005で、マスタCPU#H0が、プール領域5011について、算出されたページ比率を基に理想未割当ページ数を算出する。例えば、プール領域5011の理想未割当ページ数は、プール領域5011のページ比率と、全てのプール領域の未割当ページ数5101の合計との積である。従って、例えば、プール領域5011の理想未割当ページ数は、下記の式、

（プール領域5011の理想未割当ページ数）＝

（プール領域5011のページ比率）×{（プール領域5011の未割当ページ数5101）＋（プール領域5012の未割当ページ数5101）}
で算出される（小数点以下は四捨五入されても切り捨てられても良い）。

【0525】

20

なお、プール領域5011の未割当ページ使用量5103が極端に少ない（又は極端に多い）場合、プール領域5011における未割当実ページを極端に少なく（又は極端に多く）してしまう可能性がある。そのため、ページ比率の閾値として、最低比率（及び／又は最高比率）が設けられて良い。S6004で算出されたページ比率が最低比率未満（又は最高比率を超えている）場合、マスタCPU#H0は、そのページ比率を最低比率（又は最高比率）と同じ値として、S6004で理想未割当ページ数を算出して良い。

【0526】

S6006で、マスタCPU#H0が、プール領域5011の移動未割当ページ数を算出する。プール領域5011の移動未割当ページ数は、プール領域5011から減らすべき未割当実ページの数、又は、プール領域5011に増やすべき未割当実ページの数である。具体的には、例えば、プール領域5011の移動未割当ページ数は、下記の式、

（プール領域5011の移動未割当ページ数）＝

（プール領域5011の未割当ページ数5101）－（プール領域5011の理想未割当ページ数）

で算出される。

【0527】

30

S6004乃至S6006によれば、図63A及び図63Bを例にとると、プール領域5011について、下記の通り、

（S6004）ページ比率＝1000／（1000＋500）＝2／3

（S6005）理想未割当ページ数＝（8500＋6500）×2／3＝10000

（S6006）移動未割当ページ数＝8500－10000＝－1500

に算出される。

【0528】

40

移動未割当ページ数が正の値の場合、図61に示す処理が行われ、移動未割当ページ数が負の値の場合、図62に示す処理が行われる。移動未割当ページ数が0（ゼロ）の場合、図61及び図62のいずれも行われなくて良い。

【0529】

図61によれば（すなわち、移動未割当ページ数が正の値の場合）、HDKC#H0がHDKC#H1に未割当実ページを与える。言い換えれば、プール領域5011からプール領域5012に移動未割当ページ数と同数の未割当実ページが移動する。

50

【0530】

具体的には、S6101で、マスタCPU#H0が、実ページ管理テーブル#H0における、移動未割当ページ数と同数の未割当実ページの割当状態1404を、「未割当」から「使用不可」に更新する。且つ、マスタCPU#H0は、割当状態1404が「使用不可」に更新された1以上の実ページに対応する1以上のテーブルポインタを退避する。ここで言う「退避する」とは、例えば、マスタCPU#H0が、LM#H0又はCM#H0等の記憶領域に、それら1以上のテーブルポインタを書き込むことである。

【0531】

S6102で、マスタCPU#H0が、S6101で退避した1以上のテーブルポインタをHDKC#H1に通知する。

10

【0532】

S6151で、マスタCPU#H1が、HDKC#H0から1以上のテーブルポインタを受信する。それら1以上のテーブルポインタに対応した1以上の割当状態1404は、実ページ管理テーブル#H1において、いずれも、「使用不可」となっている。HDKC#H0が使える実ページはHDKC#H1が使えないように管理されているからである。そこで、マスタCPU#H1は、次のように実ページ管理テーブル#H1を更新する。すなわち、マスタCPU#H1は、受信した1以上のテーブルポインタに対応する1以上の割当状態1404を、「使用不可」から「未割当」に更新する。また、マスタCPU#H1は、「未割当」とされた実ページに対応する未割当キューポインタ1406を更新することで、その実ページをHDKC#H1の未割当キューに接続する。

20

【0533】

以上のS6101、S6102及びS6151で、HDKC#H0からHDKC#H1への、移動未割当ページ数と同数の未割当実ページの移動が完了される。

【0534】

一方、図62によれば（すなわち、移動未割当ページ数が負の値の場合）、HDKC#H0がHDKC#H1から未割当実ページをもらう。言い換えれば、プール領域5012からプール領域5011に移動未割当ページ数と同数の未割当実ページが移動する。

【0535】

具体的には、S6201で、マスタCPU#H0が、移動未割当ページ数を表す情報をHDKC#H1に送信する。

30

【0536】

S6251で、マスタCPU#H1が、移動未割当ページ数を表す情報をHDKC#H0から受信する。そして、マスタCPU#H1が、実ページ管理テーブル#H1における、移動未割当ページ数と同数の未割当実ページの割当状態1404を、「未割当」から「使用不可」に更新する。且つ、マスタCPU#H1は、割当状態1404が「使用不可」に更新された1以上の実ページに対応する1以上のテーブルポインタを退避する。ここで言う「退避する」とは、例えば、マスタCPU#H1が、LM#H1又はCM#H1等の記憶領域に、それら1以上のテーブルポインタを書き込むことである。

【0537】

S6252で、マスタCPU#H1が、S6251で退避した1以上のテーブルポインタをHDKC#H0に通知する。

40

【0538】

S6202で、マスタCPU#H0が、HDKC#H1から1以上のテーブルポインタを受信する。そして、マスタCPU#H0は、次のように実ページ管理テーブル#H0を更新する。すなわち、マスタCPU#H0は、受信した1以上のテーブルポインタに対応する1以上の割当状態1404を、「使用不可」から「未割当」に更新する。また、マスタCPU#H0は、「未割当」とされた実ページに対応する未割当キューポインタ1406を更新することで、その実ページをHDKC#H0の未割当キューに接続する。

【0539】

以上のS6201、S6251、S6252及びS6202で、HDKC#H1からH

50

D K C # H 0 への、移動未割当ページ数と同数の未割当実ページの移動が完了される。

【0540】

以上が、プールメンテナンス処理の説明である。プール1500は一つの統合プールであるので、複数のHDKCユニット141、HDKC及び／又は管理装置301のプロセッサが、プール1500の残り容量（未割り当ての全容量）がある閾値より以下又は同等と判断した場合、そのプロセッサが実ストレージ媒体を追加のために管理装置301に警告を表示させるように制御する。この警告に基づいて、ユーザ及び／又は管理者は、実ストレージ媒体を各々のHDKCユニット141へ追加できる。しかしながら、本実施例においては、ユーザは、各々のHDKCユニット141でなく、少なくとも一つのHDKCユニット141へ実ストレージ媒体を追加することもできる。理由は、プールメンテナンス処理が自動的に実ストレージエリアを複数のHDKCユニット141へ割り当て及び分配することができるからである。さらに、この実施例は、HDKCユニット141の各々が他のHDKCユニットに対して自装置の実ストレージ領域を配分／割当しない、という例を有することも出来る。この例においては、HDKCユニット141の各々は、他のHDKCユニットのPool-RG IDや実ページアドレスに対応する図57及び58の割り当て状態を非割当として管理する。この例においても、プールメンテナンス処理を利用することにより、高い信頼性と非高コストを実現できる。

10

【0541】

なお、上記の説明では、HDKC#H0に第3の接続形態で接続されているHDKC（第3接続HDKC）の数が1であることを例に採ったが、複数の第3接続HDKCがある場合にも、プール作成処理（図53参照）、及び、プールメンテナンス処理（図60乃至図63参照）を行うことができる。なお、複数の第3接続HDKCがある場合、例えば、プールメンテナンス処理では、次のいずれかが採用されて良い。

20

<<移動未割当ページ数が正の値>>

（＊）HDKC#H0が、移動未割当ページ数と同数の未割当実ページを、複数の第3接続HDKCのうちの、未割当ページ数5101が最も少ないHDKCに与える。

（＊）HDKC#H0が、移動未割当ページ数と同数の未割当実ページを、複数の第3接続HDKCに均等に与える。

<<移動未割当ページ数が負の値>>

（＊）HDKC#H0が、移動未割当ページ数と同数の未割当実ページを、複数の第3接続HDKCのうちの、未割当ページ数5101が最も多いHDKCからもらう。

30

（＊）HDKC#H0が、移動未割当ページ数と同数の未割当実ページを、複数の第3接続HDKCから均等にもらう。

【0542】

また、HDKC#0が、HDKC#0と複数の第3接続HDKCとのそれぞれについて、ページ比率及び理想未割当ページ数を算出して良い。それらのHDKCにおける、或るHDKCと別のHDKC間で、未割当実ページが受け渡しされても良い。

【0543】

本実施例においては、プール1500がそのプール1500を共有するHDKCユニット141の数に応じた数の記憶領域（以下、プール領域）に分割されることを説明した。本実施例としては、HDKCユニット141によってプール1500を生成又は形成する場合に、複数のHDKCユニット141、HDKC及び／又は管理装置301のプロセッサが、プール1500を共有する各HDKCユニット141の各々にとって均等な記憶領域のサイズになるようにプール1500の総記憶領域サイズを分割して複数のプール領域を構成したりすることも良い。この技術は、プールメンテナンス処理にも適用できる。

40

【0544】

又は、HDKCユニット141、HDKC及び／又は管理装置301のプロセッサが、プール1500を共有する各HDKCユニット141の各々に対して均等な数の実ページが分配されるようにプール1500の総実ページ数を分割して、複数のプール領域を構成することも良い。この技術は、プールメンテナンス処理にも適用できる。

50

【0545】

又は、複数のHDKCユニット141によってプール1500を生成又は形成する場合に、HDKCユニット141、HDKC及び／又は管理装置301のプロセッサが、プール1500を共有する各HDKCユニット141の性能の違いに応じて、HDKCユニット141の性能に適した記憶領域のサイズ又は実ページ数になるようにプール1500の総記憶領域サイズ又は総実ページ数を分割して複数のプール領域を構成したりすることも良い。この場合、性能が大きいHDKCユニット141が、性能が小さいHDKCユニット141に比べて、より大きな記憶領域のサイズ又は多くの実ページ数になるように、HDKCユニット141、HDKC及び／又は管理装置301のプロセッサが、プール1500の総記憶領域サイズ又は総実ページ数を分割することになる。この技術は、プールメンテナン

10

【0546】

又は、複数のHDKCユニット141によってプール1500を生成又は形成する場合に、HDKCユニット141、HDKC及び／又は管理装置301のプロセッサが、プール1500を共有する各HDKCユニット141とその他の各々のHDKCユニット141との間の距離（コマンド応答時間）に応じて、複数のHDKCユニット141間の距離（コマンド応答時間）に適した記憶領域のサイズ又は実ページ数になるようにプール1500の総記憶領域サイズ又は総実ページ数を分割して複数のプール領域を構成したりすることも良い。この場合、あるHDKCユニットに着目した場合、距離（コマンド応答時間）が近い他のHDKCユニット141に対しては、距離（コマンド応答時間）が遠い他のHDKCユニット141に比べて、より大きな自ユニット内の記憶領域のサイズ又は多くの自ユニット内の実ページ数を提供することになるように、HDKCユニット141、HDKC及び／又は管理装置301のプロセッサが、プール1500の総記憶領域サイズ又は総実ページ数を分割することになる。この技術は、プールメンテナン

20

【0547】

なお、プール1500の総記憶領域サイズ又は総実ページ数の配分は、HDKCユニット141の性能と複数のHDKCユニット141間の距離（コマンド応答時間）との両方（複合的な要因）を考慮して実施されることも良い。この技術は、プールメンテナン

30

【0548】

又は、プール1500の総記憶領域サイズ又は総実ページ数の配分は、HDKCユニット141の性能及び／又は複数のHDKCユニット141間の距離（コマンド応答時間）を考慮する又はしないに関わらず、あるHDKCユニット自身に対しては、他のHDKCユニット141に比べて、より大きな自ユニット内の記憶領域のサイズ又は多くの自ユニット内の実ページ数を割り当てるように、HDKCユニット141、HDKC及び／又は管理装置301のプロセッサが実施することも良い。この技術は、プールメンテナン

【0549】

本実施例のプール1500の生成又は形成は、複数のHDKCユニット141が導入された場合に、複数のHDKCユニット141、HDKC及び／又は管理装置301のプロセッサによって実施されても良く（これらのプロセッサは管理者の指示を受けて実施しても良い）、一部のHDKCユニット141やHDKCがその他のHDKCユニット141やHDKCに対して増設や追加された場合に、複数のHDKCユニット141、HDKC及び／又は管理装置301のプロセッサによって実施されても良く（これらのプロセッサは管理者の指示を受けて実施しても良い）。この技術は、プールメンテナン

40

【0550】

本実施例では、複数のHDKCユニット141が1つのプールを管理して、この1つのプールに対して1又は複数のDPUが関連付けられる場合を説明した。しかしながら

50

、本実施例は、この場合に限られず、複数のHDKCユニット141が複数のプール（各々が上記1つのプールに対応する）を管理して、その各々のプールに対して、共通又は相違する1又は複数のDPUが関連付けられる場合も良い。

【0551】

本実施例では、プール1500を構成する各内部ボリュームの各々がRGである場合を用いて技術的な内容を説明したが、上述のとおり、本実施例は、その場合に限られない。例えば、各内部ボリュームの各々が複数RGを有するグループ（RGグループ）である場合、図54、図55、図57及び図58の各々の実ページ管理テーブルには、Pool-RG IDに代えて、又はPool-RG IDに加えて、各RGグループに関する識別情報が入り、各RGグループ毎に実ページアドレス又は割当状態等が図54、図55、図57及び図58と同様に管理されることになる。この場合、各プール領域（5011、5012）は、プール1500を構成する複数のRGグループにまたがることになり、プール1500を構成する全てのRGにまたがることにならない。また、この場合、プール1500内から分割された各々のプール領域5011（又は5012）において、仮想ページに割り当てられる実ページは、n回前（nは自然数）に割り当てられた実ページの基になっているRGグループとは異なるRGグループが基になっている実ページで良い（例えば、n=1で良い）。具体的には、例えば、ラウンドロビン形式で、割り当てられる実ページの基になっているRGグループが切り替わって良い。HDKCユニット141及び／又はHDKCのプロセッサ222は、あるRGグループから1又は複数の実ページを割り当てた場合には、次に他の複数のRGグループから1又は複数の実ページを割り当てることにより、特定のRGグループにアロケーションに伴うI/Oが集中してしまうことを回避でき、以って、DKCのI/O性能の向上が期待できる。1つのプール1500に関連付けられるDPU又は複数のDPUのグループ（DPUグループ）が、DPU又はDPUグループ毎に異なるティアに属している場合（例えば、第一のDPU又は第一のDPUグループは第一のティアに属し、第二のDPU又は第二のDPUグループは第二のティアに属している場合）、ティア別に、n回前に割り当てられた実ページの基になっているRGグループ又は複数のRGグループと異なるRGグループ又は複数のRGグループから実ページが割り当てられて良い（例えば、第一のDPU又は第一のDPUグループは第一のティアを構成するRGグループ又は複数のRGグループの範囲内において、ラウンドロビン形式で、割り当てられる実ページの基になっているRGグループ又は複数のRGグループが切り替わり、第二のDPU又は第二のDPUグループは第二のティアを構成するRGグループ又は複数のRGグループの範囲内において、ラウンドロビン形式で、割り当てられる実ページの基になっているRGグループ又は複数のRGグループが切り替わる）。

【0552】

また、本実施例では、プール1500を構成する各内部ボリュームの各々が、既に説明した通り、1つの通常のLU、通常のLUの一部、複数の通常のLUのグループ、第2の接続形態に従う1つの仮想LU、仮想LUの一部、複数の仮想LUのグループ、或いは、RGの一部の何れかである場合にも、上記同様に、図54、図55、図57及び図58の各々の実ページ管理テーブルには、Pool-RG IDに代えて、又はPool-RG IDに加えて、タイプの内部ボリュームに関する識別情報が入り、各内部ボリューム毎に実ページアドレス又は割当状態等が図54、図55、図57及び図58と同様に管理されることになる。

【0553】

本実施例においても、1又は複数のDPUの各仮想ページは、各仮想ページのアクセス特性に応じて、このアクセス特性に適する記憶媒体に対応付けられた実ページにマッピングさせることもできる。又は、各仮想ページのアクセス特性に応じて、このアクセス特性に適していない記憶媒体に対応付けられた実ページから、このアクセス特性に適する記憶媒体に対応付けられた実ページにマッピングを変更することもできる。ここでいうアクセス特性とは、例えば、各仮想ページへのアクセスの種類（ランダムアクセスやシーケンスアクセスの別、又はアクセス送信元のサーバ101やアプリケーション111の上述

のような種類)、各仮想ページへの I/O 頻度又はライト頻度、又は各仮想ページに書き込まれたデータがキャッシュメモリ 224 から仮想ページにマッピングされた実ページに対応する記憶媒体へ書き込まれる頻度、又は各仮想ページに書き込まれたり読み出されたりするデータがキャッシュメモリ 224 と仮想ページにマッピングされた実ページに対応する記憶媒体との間で書き込まれたり読み出されたりする頻度である。この場合、本実施例でいうところの複数のティア(例えば、実施例 1 で説明されたティアレベル又はティアレベル範囲)は、1 又は複数の D P-L U の各仮想ページ毎に管理されることになる、上記のように L U のティアという技術ではなく、1 又は複数の D P-L U の各仮想ページ毎のティアという技術と捉えることになる。ただし、各仮想ページのアクセス特性に応じてこのアクセス特性に適する記憶媒体に対応付けられた実ページにマッピングさせる場合には、各仮想ページ毎のティアという技術になるが、各仮想ページのアクセス特性に応じて仮想ページにマッピングされる実ページが変更する場合には、予め L U とティアとの関係が設定されるため、D P-L U の作成時等には L U のティアという技術が適用され、その後、アクセス特性に応じた仮想ページへマッピングされる実ページの変更に従って各仮想ページ毎のティアという技術が適用されることになる。

10

【0554】

実施例 3 (例えば、プール作成処理、及び/又は、プールメンテナンス処理)は、以下のいずれか 1 つのケースで、適用されて良い。

(a) 接続形態(DKC の性能を含んで良い)及び P D E V 種別という基本項目(性能ファクター)に加えて又は代えて、「プレキャッシュ」、「カスケードの違い」及び/又は「距離(コマンド応答時間)」等の 1 以上の他の項目(性能ファクター)も基に、ティアレベルが算出されるケース。

20

(b) 「プレキャッシュ」、「カスケードの違い」及び/又は「距離(コマンド応答時間)」(図 10 参照)のうちの 1 以上の項目(性能ファクター)のみを基にティアレベルが算出されるケース。

(c) 「プレキャッシュ」、「カスケードの違い」及び/又は「距離(コマンド応答時間)」(図 10 参照)のうちのいずれか 1 つ以上の項目(性能ファクター)と、P D E V 種別(性能ファクター)との組合せを基に、ティアレベルが算出されるケース。

(d) 「プレキャッシュ」、「カスケードの違い」及び/又は「距離(コマンド応答時間)」(図 10 参照)のうちのいずれか 1 つ以上の項目(性能ファクター)と、「接続形態」、「内部コントローラの性能」、及び「P D E V 種別」のうちのいずれか 1 つ以上の項目(性能ファクター)との組み合わせを基に、ティアレベルが算出されるケース。

30

【0555】

本実施例に係るプール作成処理及び/又はプールメンテナンス処理は、複数の H D K C 間に限らず、複数の L D K C 間にも適用されて良い。

【0556】

以上、本発明の幾つかの実施例を説明したが、本発明は、これらの実施例に限定されるものでなく、その要旨を逸脱しない範囲で種々変更可能であることはいうまでもない。例えば、本実施例においては、図 53, 54, 55, 57, 58, 67 及び 69 などで説明されたティア管理テーブル #H0 及び #H1 は、Pool-RG ID 1402 と実ページアドレス 1403 の欄内に同一の情報を有する。しかしながら、本実施例はこの場合に限られず、次のような場合でも良い。HDKC ユニット #H0 のティア管理テーブル #H0 は、HDKC ユニット #H0 がアロケーションに利用できる実ページに関する情報を有し、HDKC ユニット #H0 がアロケーションに利用できない実ページに関する情報を有しない。HDKC ユニット #H1 のティア管理テーブル #H1 は、HDKC ユニット #H1 がアロケーションに利用できる実ページに関する情報を有し、HDKC ユニット #H1 がアロケーションに利用できない実ページに関する情報を有しない。

40

【0557】

例えば、上述した実施例 1 乃至 3 の 2 以上が組み合わせられても良い。実施例 1 乃至 3 のいずれか又はそれらの 2 以上の組み合わせのいずれにおいても、複数の HDKC ユニット 141

50

及び／又はLDKC ユニット 151を含むシステムは、同一のLU（通常LU、DP-LU及び／又は仮想LU）を、第1実施例で説明した仮想ボリュームの技術を使うことによって、異なるHDKC ユニット 141の各々を介して複数の異なるアプリケーション、サーバ及び／又は仮想サーバへ提供できる。この場合、例えば、同一のLU（通常LU、DP-LU及び／又は仮想LU）は、上記同一のLUを管理するHDKC ユニット 141からアプリケーション、サーバ及び／又は仮想サーバへ上記同一のLUを提供することと、上記同一のLUにマッピングされた仮想LUを管理するHDKC ユニット 141によって他のアプリケーション、サーバ及び／又は仮想サーバへ上記仮想LUを提供することによって、複数の異なるアプリケーション、サーバ及び／又は仮想サーバへ提供される。この場合、上記同一のLUを管理するマスタHDKC ユニットとしてのHDKC ユニット 141が、実ページのDP-LUの仮想ページへのアロケーション（上記同一のLUがDP-LUの場合）と、キャッシュメモリ 224へのデータのリード/ライトと、を制御することにより、上記同一のLUに関する上記アロケーションとデータキャッシュ管理との各々に関する一貫性の制御を確保することが出来る。

10

【0558】

また、例えば、LUのIDとして、サーバが認識する種類の第1のID（例えばLUN）と、サーバが認識できない種類の第2のIDとがあっても良い。この場合、第1のIDと第2のIDとの組が、例えば、前述したLU管理テーブルに登録されていて良い。例えば、LU（x）からLU（y）にデータが移動された場合、LU（x）及びLU（y）について、第2のIDはそれぞれ変更されず、LU（x）の第1のID（x）とLU（y）の第2のID（y）が、交換されて良い。データ移動後、サーバは、第1のID（x）を指定したI/Oコマンドを発行すれば、LU（y）に対するI/Oを行うことができる。

20

【0559】

また、ライトコマンドに対する応答（例えばライト完了通知）が返されるタイミングは、ライト対象データがCMに書き込まれたときに代えて、LUにライト対象データが書き込まれたときでも良い。

【0560】

また、上記の説明において、CPU#H0（#H1）（マスタCPUであっても良い）が行う処理は、適宜、内部コントローラ#H0（#H1）が行う処理とされて良い。CPU#H0（#H1）或いは内部コントローラ#H0（#H1）が行う処理は、適宜、HDKC#H0（#H1）が行う処理とされて良い。これは、HDKC#H0（#H1）に限らず、他のDKCについても同様であっても良い。

30

【0561】

また、例えば、実施例2において、ポート単位の負荷分散は、HDKCから、そのHDKCに第2の接続形態で接続されているDKC（第2接続DKC）に対して行われても良い。すなわち、第2接続DKCが、空きFEポート及びテンポラリLUを有していても良い。なお、この場合、第3の接続形態に対応したポイント（ティアレベルの要素）と、第2の接続形態に対応したポイントは、同じとされて良い。また、第2接続DKCは、LDKCでも良いし、HDKCでも良い。

【0562】

また、例えば、第1のサーバからI/Oコマンドを受信し得る第1のDKCと、第2のサーバからI/Oコマンドを受信し得る第2のDKCが、第2の接続形態で接続されて良い。第1及び第2のDKCは、いずれも、HDKCでもLDKCでも良い。第1のサーバと第2のサーバは、同一のサーバであっても異なるサーバであっても良い。第2の接続形態によれば、第2のDKC内のLUが第1のDKCによって仮想化されて第1のサーバに提供されること（第2のDKC内のLUが第1のDKC内の仮想LU（第1のサーバに提供されるLU）にマッピングされること）があっても、第1のDKC内のLUが第2のDKCによって仮想化されて第2のサーバに提供されること（第1のDKC内のLUが第2のDKC内の仮想LU（第2のサーバに提供されるLU）にマッピングされること）は無い。一方、例えば、第1のサーバからI/Oコマンドを受信し得る第1のDKCと、第3のサーバからI/Oコマンドを受信し得る第3のDKCが、第3の接続形態で接続されて

40

50

いる場合、第3のDKC内のLUが第1のDKCによって仮想化されて第1のサーバに提供されること（第3のDKC内のLUが第1のDKC内の仮想LU（第1のサーバに提供されるLU）にマッピングされること）もあれば、第1のDKC内のLUが第3のDKCによって仮想化されて第3のサーバに提供されること（第1のDKC内のLUが第3のDKC内の仮想LU（第3のサーバに提供されるLU）にマッピングされること）もある。

【0563】

また、第1の接続媒体（内部バス）227は、通信可能な最大距離が第2及び第3の接続媒体（外部バス）501又は601より短い接続媒体である。第1の接続媒体227は、通信最高速度が第2及び第3の接続媒体501及び601より速くて良い。第1のDKCが第1の接続媒体227で第2のDKCに接続されている場合、第1のDKC内のCPUが、第1のDKC内のハードウェア資源（例えば記憶資源）にアクセスするのと同様に、第2のDKC内のハードウェア資源（例えば記憶資源）にアクセスできる。第1の接続媒体（例えばPCI-Expressバス）227は、例えば、DKC内の複数のハードウェア資源（例えば、図2において、SW226、CPU222、F-I/F221、B-I/F225、CM224）を互いに接続する内部バス（例えばPCI-Expressバス）と同じ種類の接続媒体である。

10

【0564】

一方、第2及び第3の接続媒体（外部バス）501及び601は、通信可能な最大距離が第1の接続媒体227より長い接続媒体である。第2及び第3の接続媒体501及び601は、通信最高速度が第1の接続媒体227より遅くて良い。第1のDKCが第3の接続媒体601（又は第2の接続媒体501）で第2のDKCに接続されている場合、第1のDKC内のCPUは、第1のDKC内のハードウェア資源（例えば記憶資源）にアクセスするのと同じようには、第2のDKC内のハードウェア資源（例えば記憶資源）にアクセスすることはできない。例えば、第1のDKC内のCPUは、第2のDKC内の記憶資源が記憶する情報を欲しい場合、第2のDKCにコマンドを発行し、そのコマンドに対する応答として、第2のDKCから、第2のDKC内の記憶資源が記憶する情報を受けることができる。第2及び第3の接続媒体501及び601及（例えばFibre Channelケーブル）は、例えば、DKC内の複数のハードウェア資源（例えば、図2において、SW226、CPU222、F-I/F221、B-I/F225、CM224）を互いに接続する内部バス（例えばPCI-Expressバス）とは違う種類の接続媒体である。

20

30

【0565】

また、実施例1乃至3のうちの少なくとも1つにおいて、第1のHDKCと第2のHDKCが1以上のHDKCユニット（以下、中継HDKCユニット）を介して第3の接続形態で直列に接続されている場合、第1のHDKCと第2のHDKC間のやり取りを、1以上の中継HDKCユニットが中継して良い。例えば、下記が行われて良い。なお、1つの中継HDKCユニットが、第1の接続形態で直列に接続された複数のHDKCで構成されている場合、その1つの中継HDKCユニットを流れるコマンド或いはデータは、それら複数のHDKCのうちの1以上のHDKCを経由して良い。

（＊）第1のHDKCから第2のHDKCへのコマンドが、1以上の中継HDKCユニットを順次に介して第2のHDKCに届いて良い。

40

（＊）第1のHDKC内のLUから第2のHDKC内のLUへのデータが、1以上の中継HDKCユニットを順次に介して第2のHDKCに届いて良い。

【0566】

また、第3の接続形態に対応した仮想LUにマッピングされている外部LUは、その仮想LUを提供するHDKCに第3の接続形態で接続されているHDKCに第1の接続形態で接続されているHDKCが有するLUであっても良い。すなわち、第3の接続形態に対応した仮想LUにマッピングされている外部LUは、第3の接続形態で接続されているHDKCユニットが有するいずれのHDKC内のLUであっても良い。

【0567】

また、第3の接続形態で直列に接続されているDKCユニット群（以下、上位DKCユ

50

ニット群)における1以上の上位DKCユニット内の1以上の上位DKC(例えばHDKC)に、下位DKCユニット(例えば、第1の接続形態で直列に接続された1以上のDKC(例えばLDKC)で構成されたDKC群)が第2の接続形態で接続されて良い。下位DKCユニット内の1つのDKCが1以上の上位DKCユニット内の複数のDKCに第2の接続形態で接続されても良い。

【0568】

また、予め複数のティアが固定で定義されていても良い。例えば、ティアレベルが0以上5未満のティアと、ティアレベルが5以上10未満のティアのように、予めティアが決められても良い。或いは、例えば、ティアが同じとは、ティアレベルの差が所定値以下のティアでも良い。

10

【0569】

また、実施例3において、共有されるプールが、異なる複数のティアに属する複数のボリューム群で構成されていて良い。同一ティアに属するボリューム群において、ティアレベルが最低値のボリュームとティアレベルが最高値のボリュームとの差が、所定値以下であることが好ましい。また、共有されるプールが、異なる複数のティアに属する複数のボリューム群で構成されている場合、例えば、アクセス頻度の高い仮想ページには、高位ティアのボリューム群における実ページが割り当てられ、アクセス頻度の低い仮想ページには、低位ティアのボリューム群における実ページが割り当てられても良い。また、HDKC#H0が、或る仮想ページのアクセス頻度の変化等に応じて、その仮想ページに割り当てられている実ページ内のデータを、その仮想ページのアクセス頻度等に適したティアに属するボリューム群内の実ページに移動されても良い。その実ページは、HDKC#0とは別のHDKC内の実ページでも良い。また、各ティアに属するボリューム群は、同一のHDKCにおける複数のボリュームで構成されても良いし、異なる複数のHDKCにおける複数のボリュームで構成されても良い。

20

【0570】

また、実施例2において、ポート単位の負荷分散等は、以下の(F)及び(G)で構成されて良い。

(F) 1以上のコントローラの少なくとも1つが、上記第1の記憶制御装置ユニットが有する複数の通信ポートから使用ポートを選択する。

(G) 上記1以上のコントローラの少なくとも1つが、上記(F)で選択された使用ポートである選択ポートに割り当てられている論理ボリュームを、上記選択ポートから上記第2の記憶制御装置ユニットの空きポートに移動する。

30

【0571】

上記(G)は、以下の(G1)乃至(G5)で構成されて良い。

(G1) 上記1以上のコントローラの少なくとも1つが、上記選択ポートに割り当てられている論理ボリュームである移動元ボリュームが属するティアを特定する。

(G2) 上記1以上のコントローラの少なくとも1つが、上記第2の記憶制御装置ユニットが有する複数の論理ボリュームから、上記(G1)で特定したティアと同じティアに属する論理ボリュームである移動先ボリュームを選択する。

(G3) 上記1以上のコントローラの少なくとも1つが、上記移動元ボリュームから上記移動先ボリュームにデータをコピーする。

40

(G4) 上記1以上のコントローラの少なくとも1つが、上記空きポートに上記移動先ボリュームを割り当てる。

(G5) 上記1以上のコントローラの少なくとも1つが、上記選択ポートの識別情報を上記空きポートに割り当て、且つ、上記移動元ボリュームの識別情報を上記移動先ボリュームに割り当てる。

【0572】

さらに、実施例3において、プールメンテナンス処理は、下記の(J)乃至(L)で構成されて良い。

(J) 上記1以上のコントローラの少なくとも1つが、プール領域毎に、シンプロビジョ

50

ニングボリュームに割り当てられていないがシンプロビジョニングボリュームに割り当てられ得るページである未割当ページの数进行特定する。

(K) 上記1以上のコントローラの少なくとも1つが、上記(J)で特定された未割当ページ数进行基に、1つの対象ユニットに関連付けられているプール領域である対象プール領域について、移動すべき未割当ページの数である移動ページ数进行決定する。

(L) 上記1以上のコントローラの少なくとも1つが、上記対象プール領域と別のプール領域との間で、上記移動ページ数分の未割当ページ进行移動する。

【0573】

上記(K)において、以下の(K1)乃至(K4)が行われて良い。

(K1) 上記1以上のコントローラの少なくとも1つが、プール領域毎に、前回の未割当ページ数から今回の未割当ページ数进行減算した使用ページ数が算出して良い。

10

(K2) 上記1以上のコントローラの少なくとも1つが、上記対象プール領域について、上記2以上のプール領域に対応した使用ページ数の合計に対する、上記対象プール領域に対応した使用ページ数の割合、进行算出する。

(K3) 上記1以上のコントローラの少なくとも1つが、上記対象プール領域について、上記2以上のプール領域に対応した未割当ページ数の合計と、上記割合との積、に従うページ数である理想未割当ページ数进行算出する。

(K4) 上記1以上のコントローラの少なくとも1つが、上記対象プール領域に対応した今回の未割当ページ数から上記理想未割当ページ数进行減算することで上記移動ページ数进行算出する。

20

【0574】

上記(L)において、下記の(L1)及び(L2)が行われて良い。

(L1) 上記移動ページ数が正の値の場合、上記1以上のコントローラの少なくとも1つが、上記対象プール領域から別のプール領域に上記移動ページ数分の未割当ページが移動する。

(L2) 上記移動ページ数が負の値の場合、上記1以上のコントローラの少なくとも1つが、別のプール領域から上記対象プール領域に上記移動ページ数分の未割当ページが移動する。

【0575】

上記1以上のコントローラは、1以上のDKCユニットにあっても良いし、複数のDKCユニットに分散していても良いし、複数のDKCユニットの外にあっても良いし、少なくとも1つのDKCユニットと複数のDKCユニットの外とに分散していても良い。上記1以上のコントローラの一部又は全部は、DKCの内部コントローラであっても良いし、DKCユニットにおいて複数のDKC进行管理する装置であっても良いし、複数のDKCに接続された管理システム(例えば管理計算機)であっても良い。

30

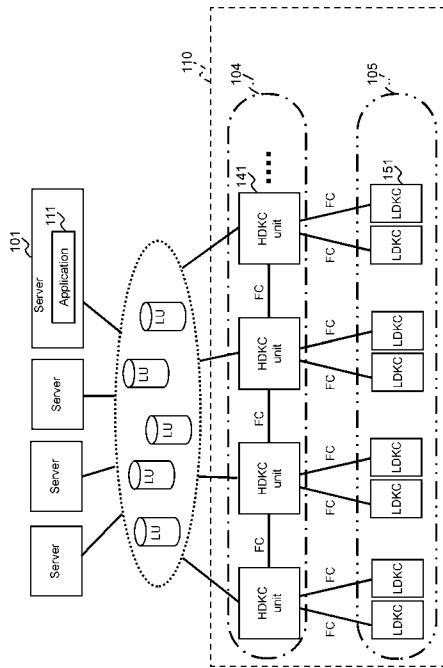
【符号の説明】

【0576】

110…記憶システム

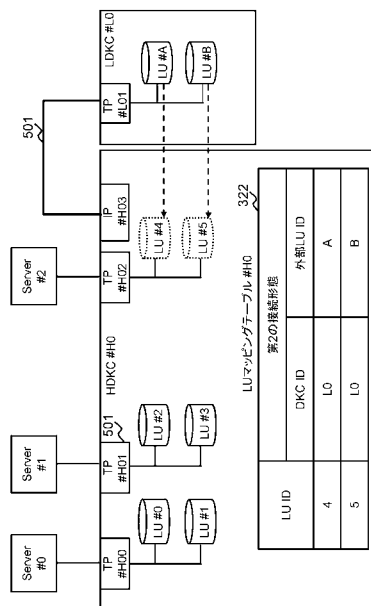
【図 1】

FIG. 1



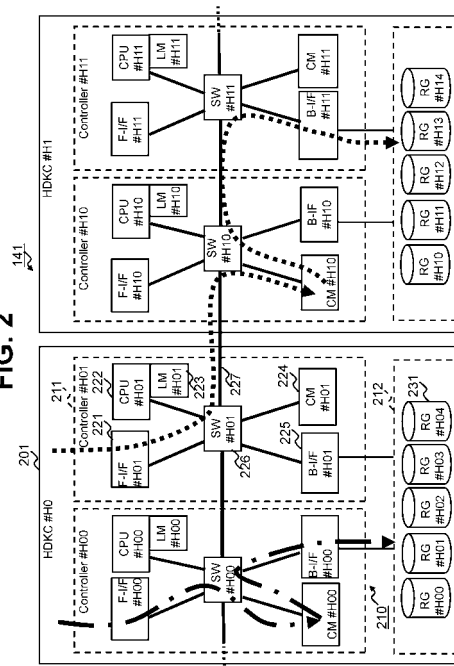
【図 3】

FIG. 3



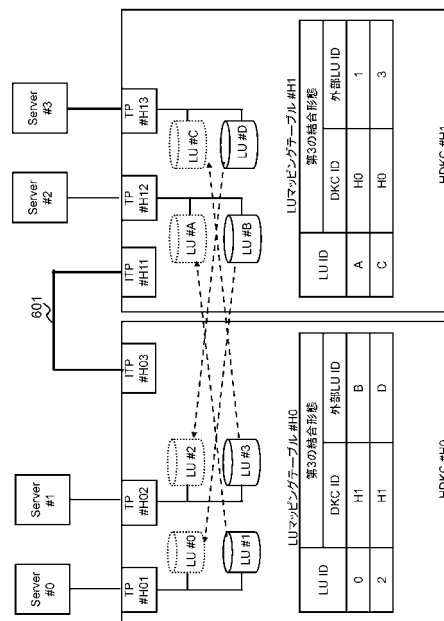
【図 2】

FIG. 2

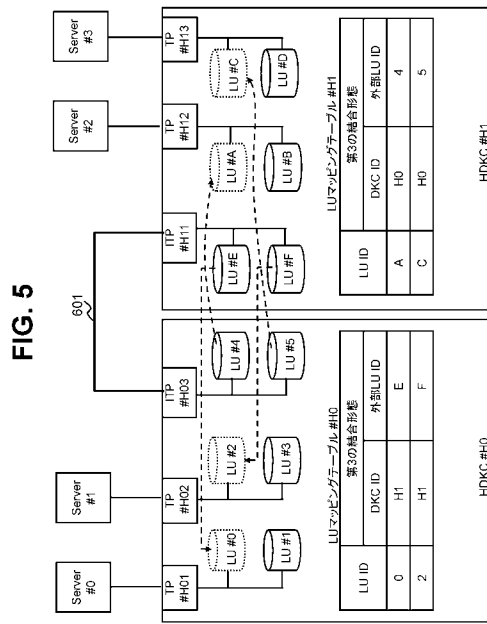


【図 4】

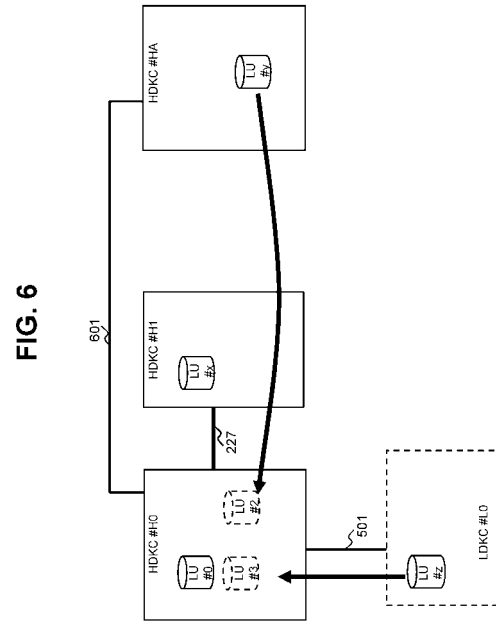
FIG. 4



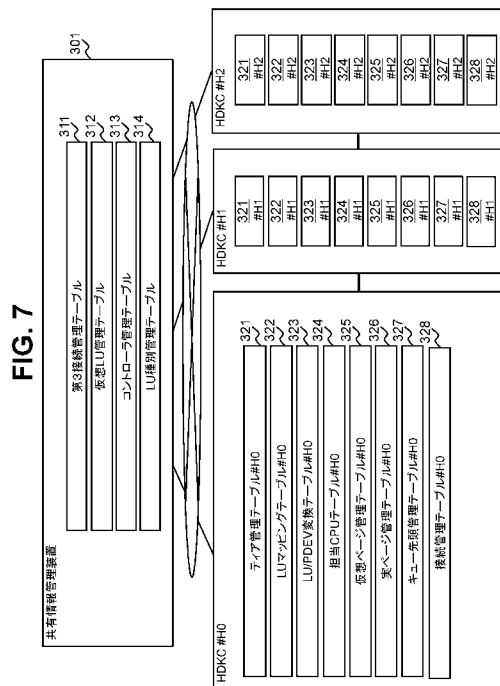
【図 5】



【図 6】



【图 7】



【图 8】



【図 1 1】

FIG. 11

要素	項目毎のポイント	
	自DKC or 第1接続 0	第2接続 1
第3の接続形態	非第2接続 0	第2接続 5
第2の接続形態	有 0	無 2
フレキシビリティ	SSD 1	SAS 3
PDEV種別	0段 0	1段 1
カスケードの違い	1km以内 0	1~100Km 1
距離(コマンド応答時間)	高速 0	中速 2
コントローラ性能の違い		低速 4

【図 9】

FIG. 9A

Tier管理テーブル (#H0)		Tier管理テーブル (#H0)		Tier管理テーブル (#H0)	
901	902	903	904	905	321
RG ID	LU ID	接続形態	PDEV種別	合計 (Tierレベル)	
H00	0	0	1	1	
H01	1	0	1	1	
H01	2	1	1	2	
H02	3	5	1	6	
H04	4	0	3	3	
...	...	...	...	...	

FIG. 9B

接続形態	PDEV種別
自DKC or 第1接続 0	SSD 1
第3接続 1	SAS 3
第2接続 5	SATA 10

(92)

JP 5567147 B2 2014.8.6

【図 1 0】

FIG. 10

Tier管理テーブル (#H0)		Tier管理テーブル (#H0)		Tier管理テーブル (#H0)	
1001	1002	1003	1004	1005	321
RG ID	LU ID	第30の接続形態	第2の接続形態	フレキシビリティ	PDEV種別
H00	0	0	0	0	1
H00	1	1	0	0	3
H01	2	1	5	2	10
...	...	...	...	...	...

Tier管理テーブル (#H0)		Tier管理テーブル (#H0)		Tier管理テーブル (#H0)	
1007	1008	1009	1010	1011	1012
カスケードの違い	距離(コマンド応答時間)	コントローラ性能	合計 (Tierレベル)		
0	0	0	1		
1	1	0	6		
2	5	4	29		
...	...	...	...		

【図 1 2】

FIG. 12

LUPDEV変換テーブル (#H0)		LUPDEV変換テーブル (#H0)		LUPDEV変換テーブル (#H0)	
1201	1202	1203	1204	1205	323
LU ID	RG ID	RAID level (combination)	Capacity	Start address	PDEV種別
0	0	RAID5 (4D+1P)	1024GB	0x00000000	SAS
1	0	RAID5 (4D+1P)	512GB	0x08000000	SAS
2	1	RAID5 (4D+4D)	2048GB	0x00000000	SSD
3	2	RAID5 (8D+2P)	256GB	0x00000000	SATA
...	...	...	...	...	...

【図 1 3】

FIG. 13

仮想ページ管理テーブル (#H0)				325
1301	1302	1303	1304	1305
DP-LU ID	仮想ページ アドレス	割当状態	Pool-RG ID	実ページ アドレス
60	0x00	未割当	—	—
	0x01	割当	H05	0x00
	0x02	未割当	—	—
	0x03	割当	H06	0x01
	0x04	未割当	—	—
	0x05	割当	H07	0x02
...	...	...	...	...

【図 1 4】

FIG. 14A

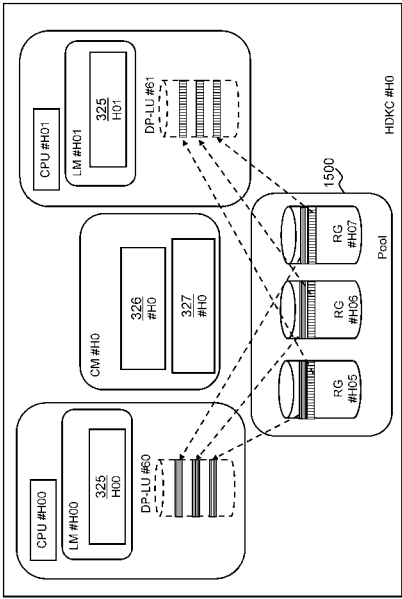
実ページ管理テーブル (#H0)				326
1401	1402	1403	1404	1405
テーブル ポインタ	Pool-RG ID	実ページ アドレス	割当状態	DP-LU ID
0x0000	H050	0x00	割当	60
0x0001		0x01	未割当	—
0x0002		0x02	割当	61
...	...	...	...	...
0x1000	H051	0x00	未割当	—
0x1001		0x01	割当	60
0x1002		0x02	未割当	—
...	...	...	...	...
0x2000	H052	0x00	未割当	—
0x2001		0x01	未割当	—
0x2002		0x02	割当	60
...	...	...	...	...

FIG. 14B

キュー先頭管理テーブル (#H0)		327
先頭ポインタ		0x0001

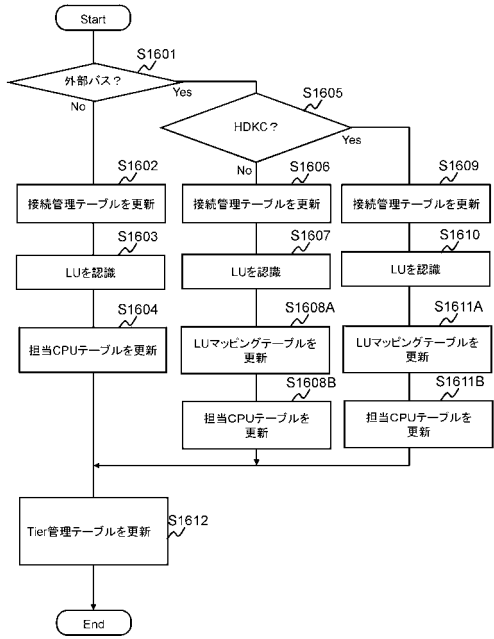
【図 1 5】

FIG. 15

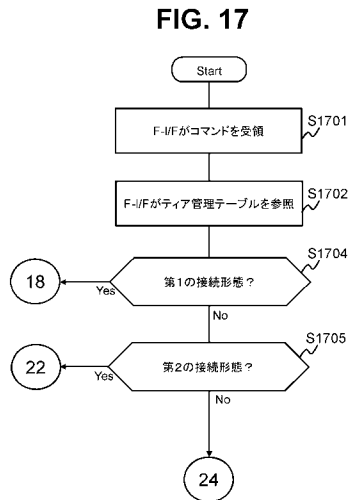


【図 1 6】

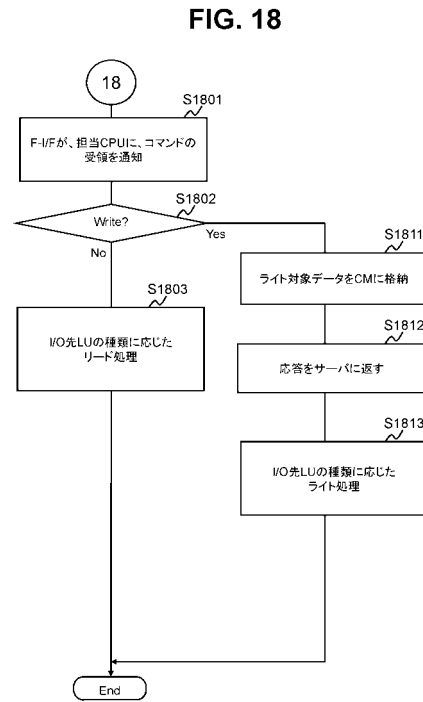
FIG. 16



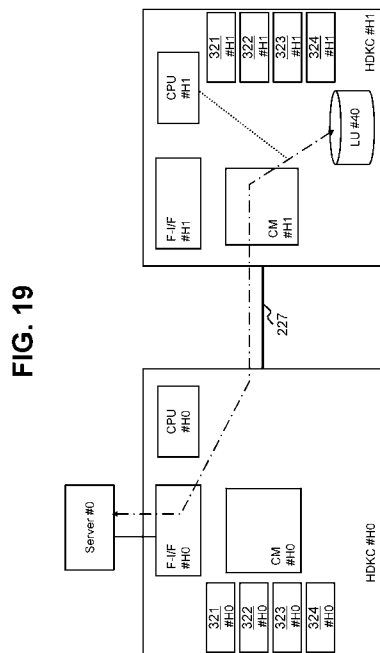
【図 17】



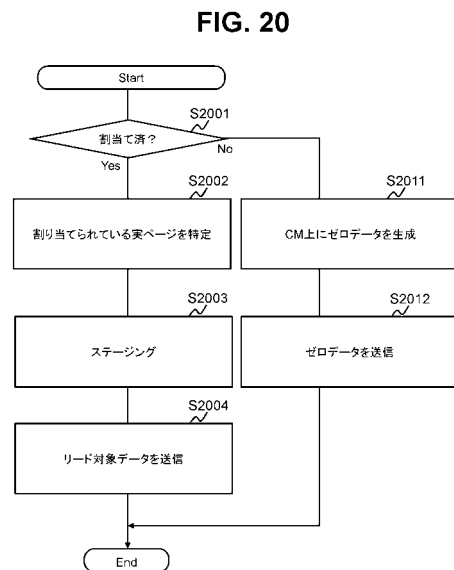
【図 18】



【図 19】

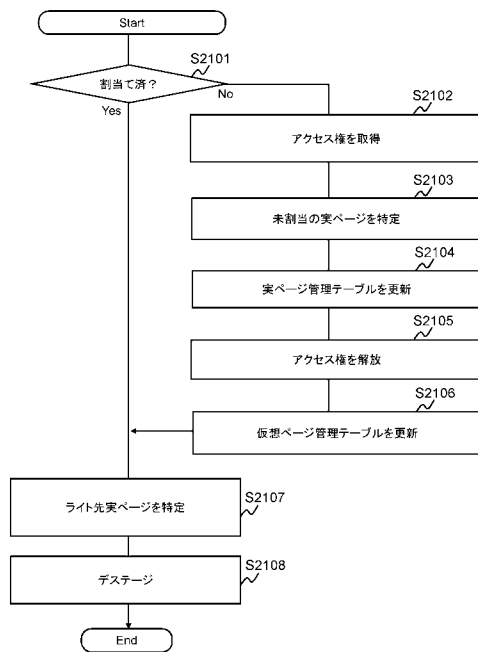


【图 20】



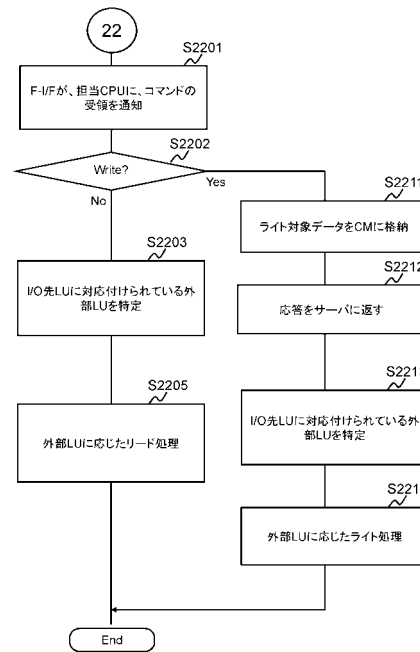
【図 2 1】

FIG. 21



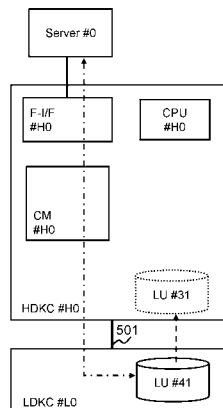
【図 2 2】

FIG. 22



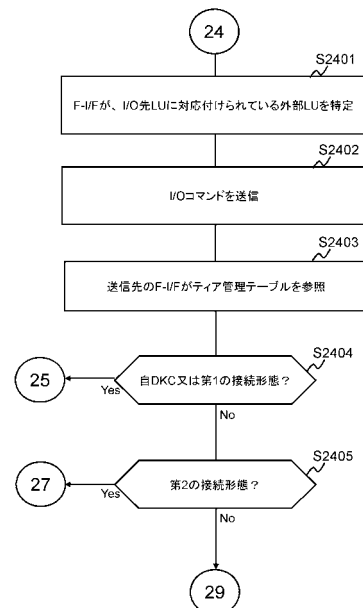
【図 2 3】

FIG. 23



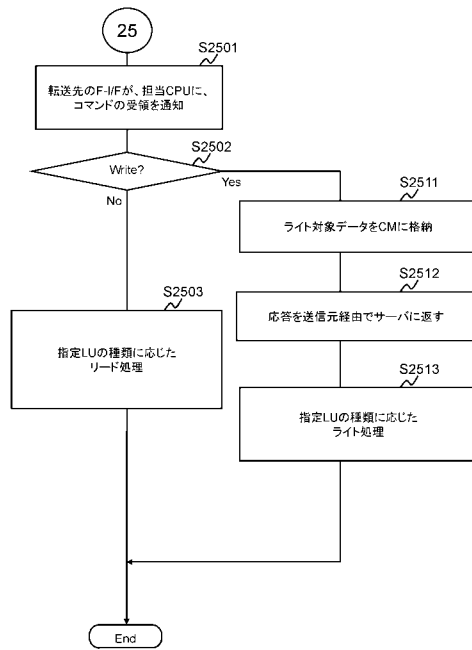
【図 2 4】

FIG. 24



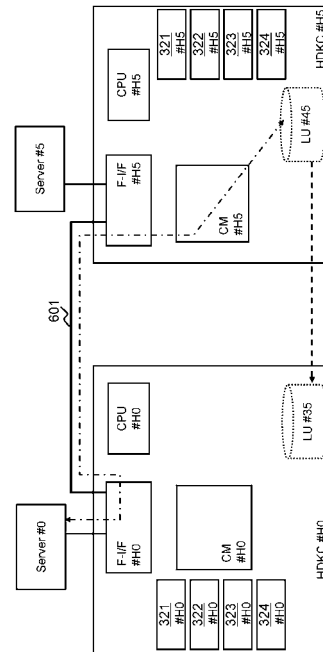
【図 25】

FIG. 25



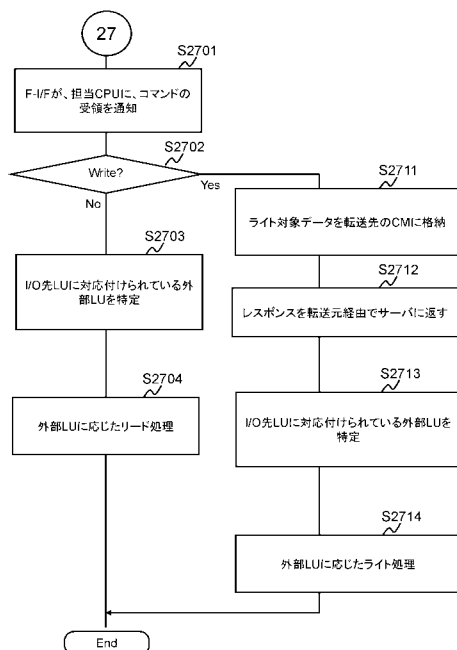
【図 26】

FIG. 26



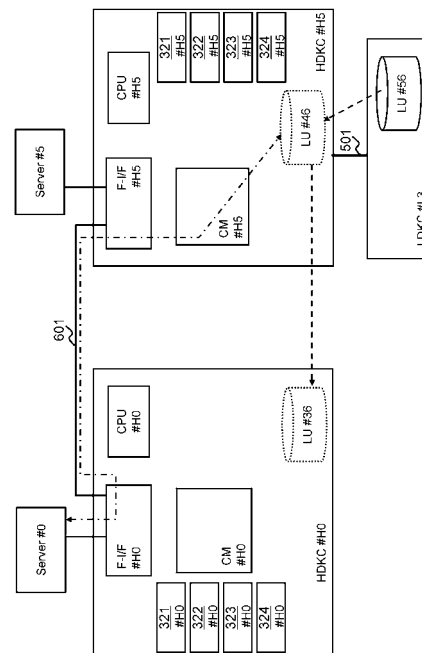
【図 27】

FIG. 27



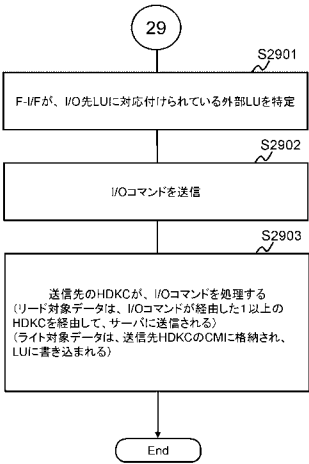
【図 28】

FIG. 28



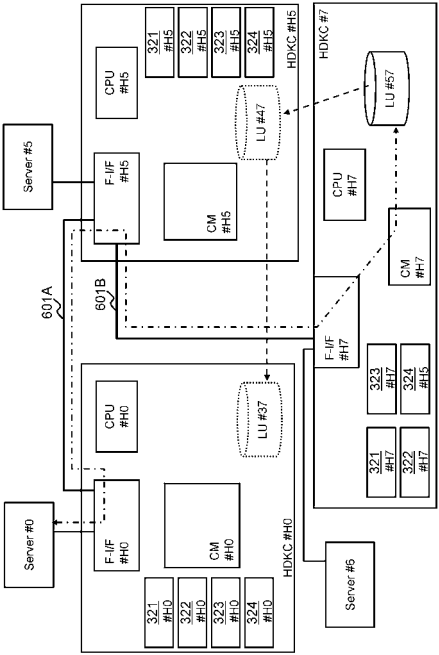
【図 29】

FIG. 29



【図 30】

FIG. 30



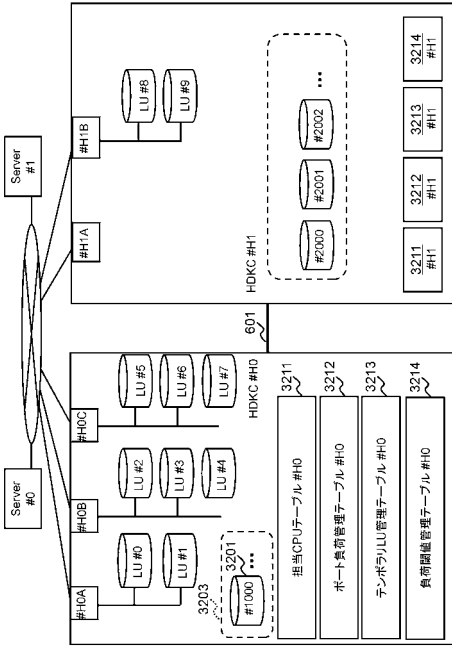
【図 31】

FIG. 31

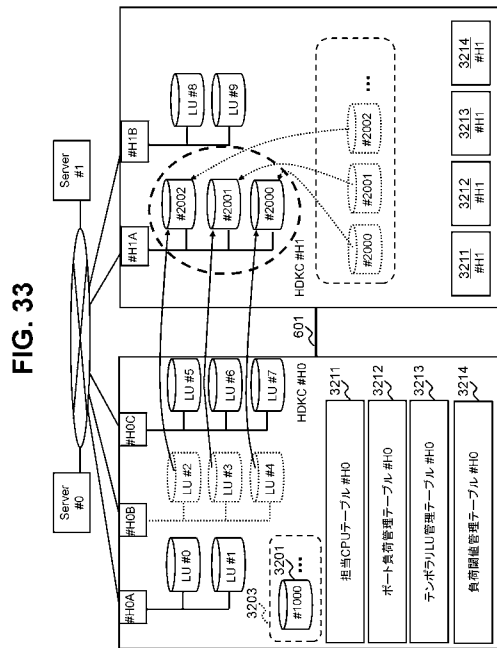
担当CPUテーブル (#H0)			324
3101 担当CPU ID	3102 CPU使用率	3103 LU ID	
H00	60%	0	
		4	
		8	
		...	
H01	40%	1	
		5	
		...	
		...	

【図 32】

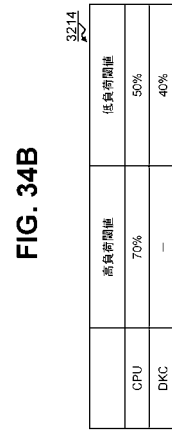
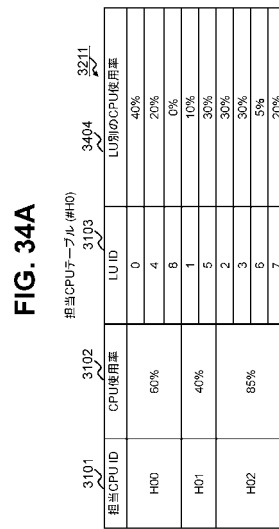
FIG. 32



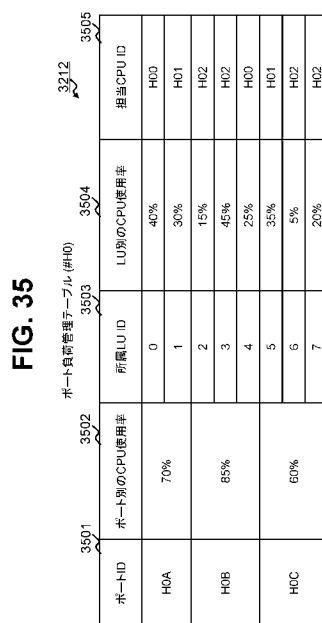
【図 3 3】



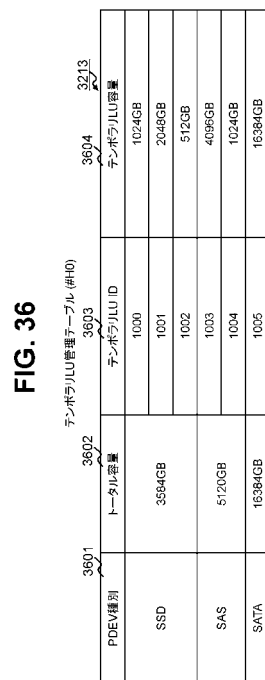
【図 3 4】



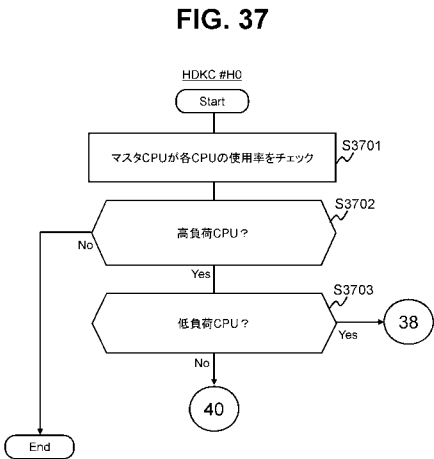
【図 3 5】



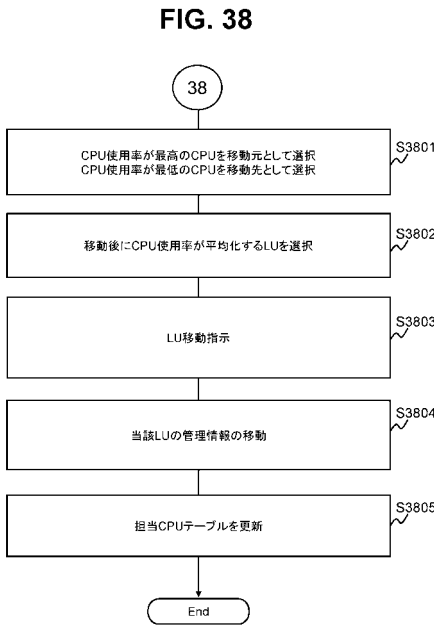
【図 3 6】



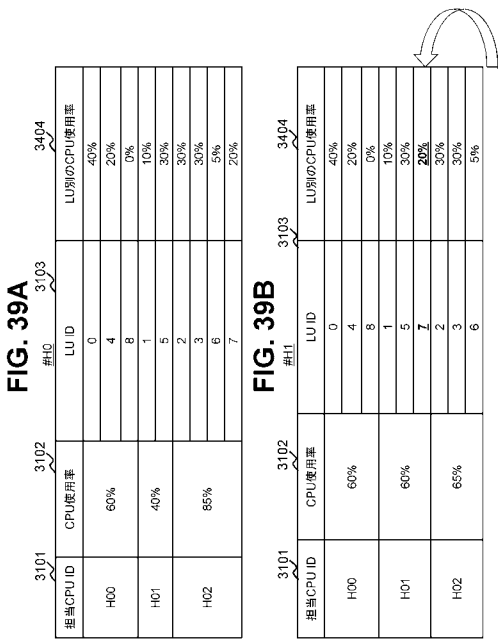
【図 3 7】



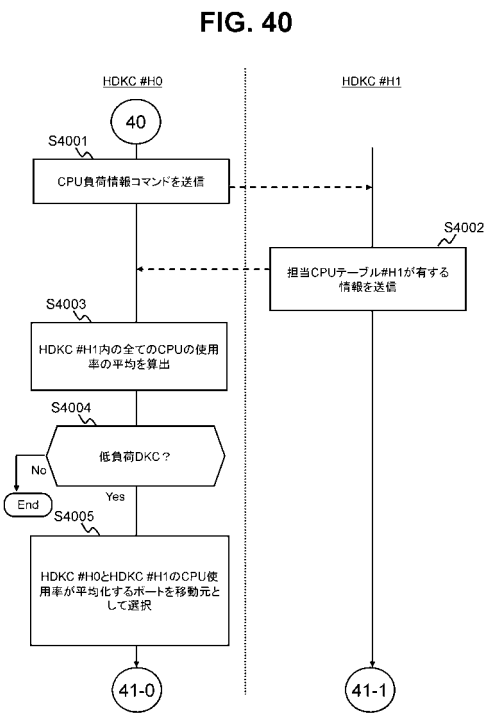
【図 3 8】



【図 3 9】

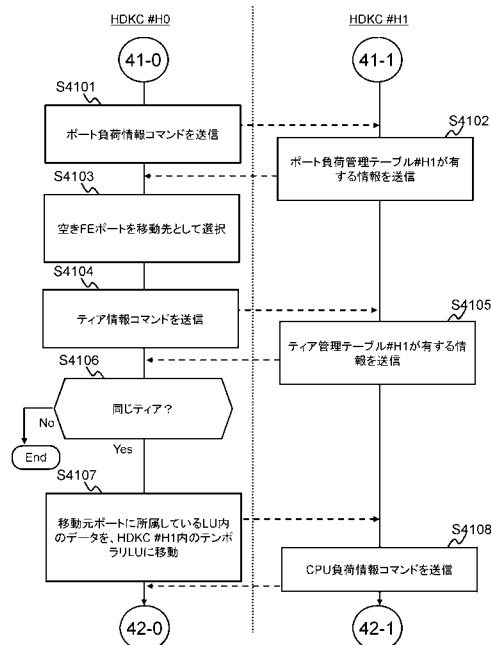


【図 4 0】



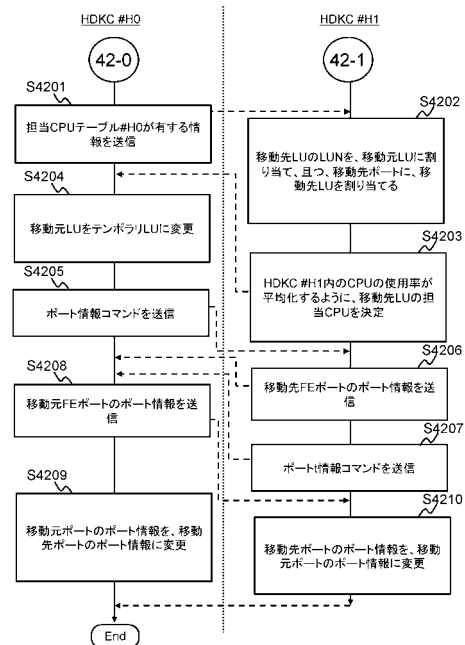
【図 4 1】

FIG. 41



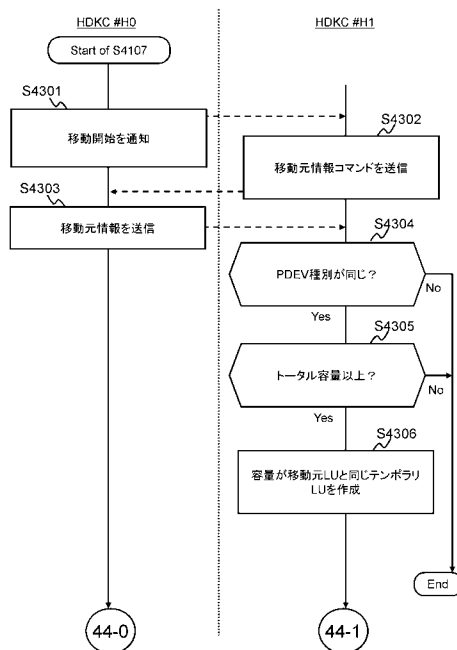
【図 4 2】

FIG. 42



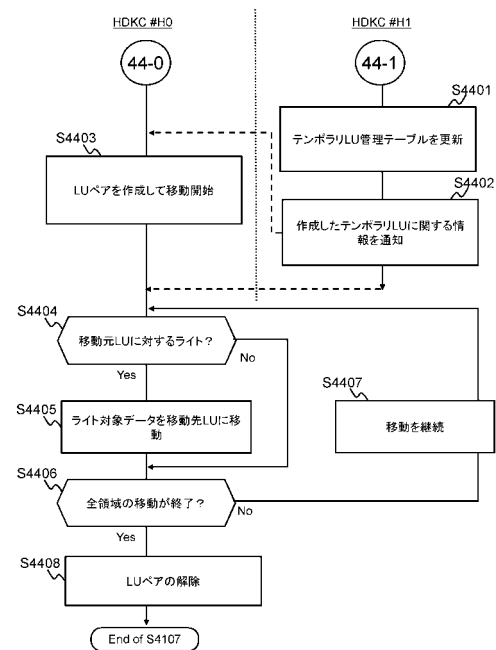
【図 4 3】

FIG. 43



【図 4 4】

FIG. 44



【図 4 5】

FIG. 45A

3501		3502	3503	3504	3505
ポートID	ポート別のCPU使用率	所属LUID	LU別のCPU使用率	担当CPUID	
HOA	70%	0	40%	H00	
		1	30%	H01	
		2	15%	H02	
HOB	85%	3	45%	H02	
		4	25%	H00	
		5	35%	H01	
HOC	60%	6	5%	H01	
		7	20%	H02	

FIG. 45B

3501	3502	3503	3504	3505
ポートID	ポート別のCPU使用率	所属LUID	LU別のCPU使用率	担当CPUID
H1A	0%	—	—	—
H1B	40%	8	10%	H10
		9	30%	H11

【図 4 7】

FIG. 47A

3501		3502	3503	3504	3505
ポートID	ポート別のCPU使用率	所属LU ID	LU別のCPU使用率	担当CPU ID	
HOA	70%	0	40%	H00	
		1	30%	H01	
HOB	0%	—	—	—	
HOC	60%	5	35%	H01	
		6	5%	H02	
		7	20%	H02	

FIG. 47B

3501		3502	3503	3504	3505
ポートID	ポート別のCPU使用率	所属LUID	LU別のCPU使用率	担当CPUID	
H1A	85%	2	15%	H12	
		3	45%	H12	
		4	25%	H10	
		8	10%	H10	
H1B	40%	9	30%	H11	

【図 4 6】

FIG. 46A

3101		3102	#HQ	3103	3404
担当CPU ID	CPU使用率			LU ID	LU別のCPU使用率
H00	65%			0	40%
				4	25%
				1	30%
H01	65%			5	35%
				2	15%
				3	45%
H02	65%			6	5%
平均				7	20%

FIG. 46B

3101		3102	3103	3404
担当CPU ID	CPU使用率	LU ID	LU別のCPU使用率	
H10	10%	8	10%	
H11	30%	9	30%	
H12	0%	—	—	
平均	13%			

【図 4 8】

FIG. 48A

3101		3102	3103	3404
担当CPU ID	CPU使用率	LU ID	LU別のCPU使用率	
H00	40%	0	40%	
H01	65%	1	30%	
		5	35%	
		6	5%	
H02	25%	7	20%	
平均	43%			

FIG. 48B

3101		3102	3103	3404
担当CPU ID	CPU使用率	LU ID	LU別のCPU使用率	
H10	35%	8	10%	
		4	25%	
		9	30%	
H11	45%	2	15%	
H12	45%	3	45%	
平均	41%			

【図 4 9】

FIG. 49A

3601		3602	3603	3604
PDEV種別		テンポラリU ID	テンポラリU ID	テンポラリU容量
SSD		1000	1000	1024GB
		1001	1001	2048GB
		1002	1002	512GB
SAS		1003	1003	4096GB
		1004	1004	1024GB
SATA		1005	1005	16384GB

FIG. 49B

3601		3602	3603	3604
PDEV種別		テンポラリU ID	テンポラリU ID	テンポラリU容量
SSD		1000	1000	3072GB
		1002	1002	512GB
		1003	1003	2048GB
SAS		1100	1100	2048GB
		1004	1004	1024GB
SATA		1005	1005	16384GB

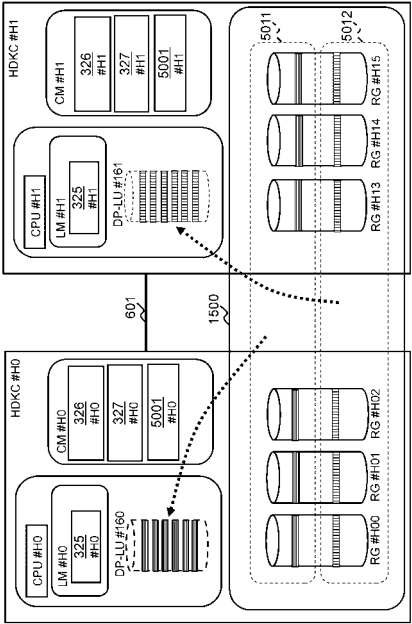
【図 5 1】

FIG. 51

5101			5102	5001	5103
未割当ページ数			前回未割当ページ数	未割当ページ使用量	
8500			9500	1000	

【図 5 0】

FIG. 50



【図 5 2】

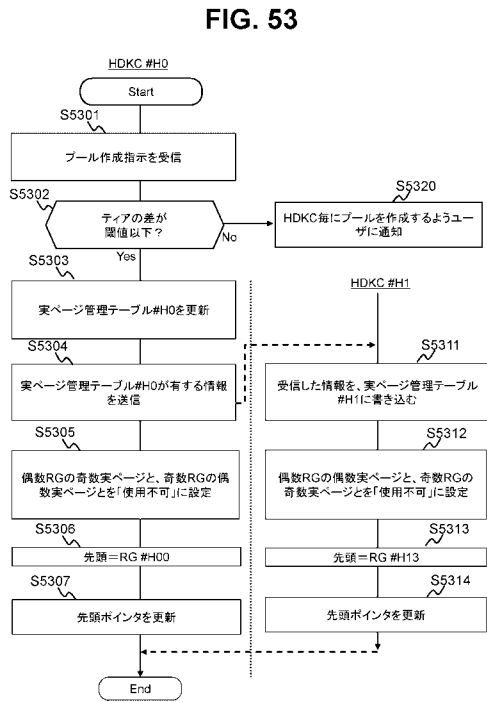
FIG. 52A

901		903	904	1008	5210
Pod/RG ID		接続形態	PDEV種別	距離 (コマンド応答時間)	合計 (Tabel)
H00		0	3	0	3
H01		0	3	0	3
H02		0	3	0	3
H13		1	3	0	4
H14		1	3	0	4
H15		1	3	0	4

FIG. 52B

901		903	904	1008	5210
Pod/RG ID		接続形態	PDEV種別	距離 (コマンド応答時間)	合計 (Tabel)
H00		0	3	0	3
H01		0	3	0	3
H02		0	3	0	3
H13		1	3	5	9
H14		1	3	5	9
H15		1	3	5	9

【図 5 3】



【図 5 4】

FIG. 54

1401	1402	1403	1404	1405	1406
プール ポインタ	Pool-RG ID	実ページアドレス	第1状態	DP-LU ID	未割当キューポインタ
0x0000	H00	0x00	未割当	---	---
0x0001	---	0x01	未割当	---	---
0x0002	---	0x02	未割当	---	---
0x1000	H01	0x00	未割当	---	---
0x1001	---	0x01	未割当	---	---
0x1002	---	0x02	未割当	---	---
0x2000	H02	0x00	未割当	---	---
0x2001	---	0x01	未割当	---	---
0x2002	---	0x02	未割当	---	---
0x3000	H13	0x00	未割当	---	---
0x3001	---	0x01	未割当	---	---
0x3002	---	0x02	未割当	---	---
0x4000	H14	0x00	未割当	---	---
0x4001	---	0x01	未割当	---	---
0x4002	---	0x02	未割当	---	---
0x5000	H15	0x00	未割当	---	---
0x5001	---	0x01	未割当	---	---
0x5002	---	0x02	未割当	---	---
...	...	...	...	...	...

【図 5 5】

FIG. 55

1401	1402	1403	1404	1405	1406
プール ポインタ	Pool-RG ID	実ページアドレス	第1状態	DP-LU ID	未割当キューポインタ
0x0000	H00	0x00	未割当	---	---
0x0001	---	0x01	未割当	---	---
0x0002	---	0x02	未割当	---	---
0x1000	H01	0x00	未割当	---	---
0x1001	---	0x01	未割当	---	---
0x1002	---	0x02	未割当	---	---
0x2000	H02	0x00	未割当	---	---
0x2001	---	0x01	未割当	---	---
0x2002	---	0x02	未割当	---	---
0x3000	H13	0x00	未割当	---	---
0x3001	---	0x01	未割当	---	---
0x3002	---	0x02	未割当	---	---
0x4000	H14	0x00	未割当	---	---
0x4001	---	0x01	未割当	---	---
0x4002	---	0x02	未割当	---	---
0x5000	H15	0x00	未割当	---	---
0x5001	---	0x01	未割当	---	---
0x5002	---	0x02	未割当	---	---
...	...	...	...	---	---

【図 5 6】

FIG. 56A

キュー先頭管理テーブル (#H0)
先頭ポインタ

FIG. 56B

キュー先頭管理テーブル (#H1)
先頭ポインタ

【図 57】

FIG. 57

1401	1402	1403	1404	1405	1406
テーブル ポインタ	Pool-RG ID	実ページアドレス	割当状態	DP-LU ID	未割当キューポインタ
0x0000	H00	0x00	未割当	—	—
0x0001		0x01	使用不可	—	0x1001
0x0002		0x02	未割当	—	0x1003
...	H01	...	...	...	...
0x1000		0x00	使用不可	—	—
0x1001		0x01	未割当	—	0x2000
0x1002		0x02	使用不可	—	—
...	H02	...	...	...	...
0x2000		0x00	未割当	—	0x3001
0x2001		0x01	使用不可	—	—
0x2002		0x02	未割当	—	0x3003
...	H13	...	...	...	...
0x3000		0x00	使用不可	—	—
0x3001		0x01	未割当	—	0x4000
0x3002		0x02	使用不可	—	—
...	H14	...	...	...	...
0x4000		0x00	未割当	—	0x5001
0x4001		0x01	使用不可	—	—
0x4002		0x02	未割当	—	0x5003
...	H15	...	...	...	...
0x5000		0x00	使用不可	—	—
0x5001		0x01	未割当	—	0x6002
0x5002		0x02	使用不可	—	—
...	...	...	...	...	...

【図 58】

FIG. 58

1401	1402	1403	1404	1405	1406
テーブル ポインタ	Pool-RG ID	実ページアドレス	割当状態	DP-LU ID	未割当キューポインタ
0x0000	H00	0x00	使用不可	—	—
0x0001		0x01	未割当	—	0x1000
0x0002		0x02	使用不可	—	—
...	H01	...	...	...	...
0x1000		0x00	未割当	—	0x2001
0x1001		0x01	使用不可	—	—
0x1002		0x02	未割当	—	0x2003
...	H02	...	...	...	...
0x2000		0x00	使用不可	—	—
0x2001		0x01	未割当	—	0x3002
0x2002		0x02	使用不可	—	—
...	H13	...	...	...	...
0x3000		0x00	未割当	—	0x4001
0x3001		0x01	使用不可	—	—
0x3002		0x02	未割当	—	0x4003
...	H14	...	...	...	...
0x4000		0x00	使用不可	—	—
0x4001		0x01	未割当	—	0x5000
0x4002		0x02	使用不可	—	—
...	H15	...	...	...	...
0x5000		0x00	未割当	—	0x6001
0x5001		0x01	使用不可	—	—
0x5002		0x02	未割当	—	0x6003
...	...	...	...	...	...

【図 59】

FIG. 59A

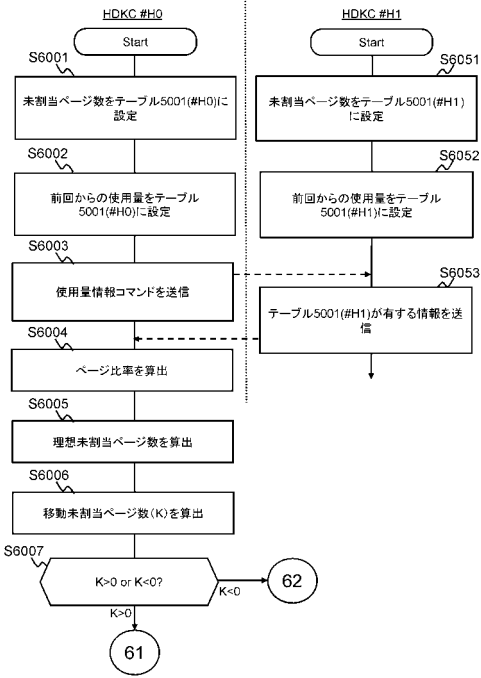
キュー先頭管理テーブル (#H0)	
先頭ポインタ	0x0000

FIG. 59B

キュー先頭管理テーブル (#H1)	
先頭ポインタ	0x3000

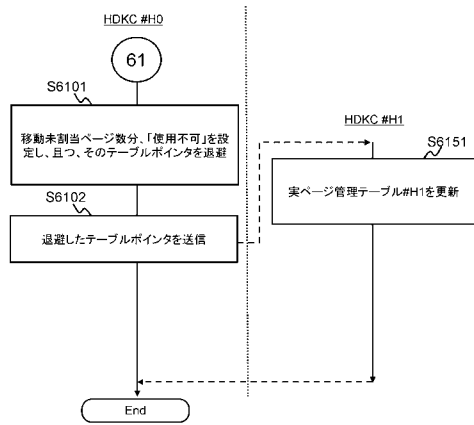
【図 60】

FIG. 60



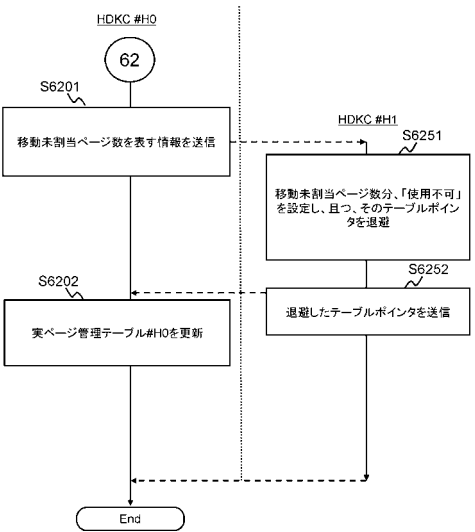
【図 6 1】

FIG. 61



【図 6 2】

FIG. 62



【図 6 3】

FIG. 63A

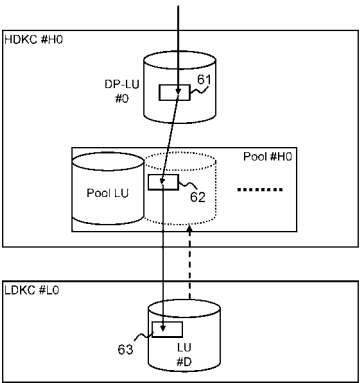
#H0		5101	5102	5103
未割当ページ数	前回未割当ページ数	8500	9500	未割当ページ使用量
				1000

FIG. 63B

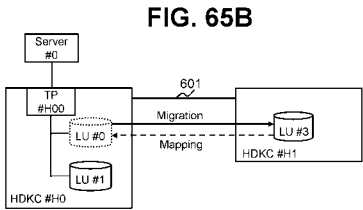
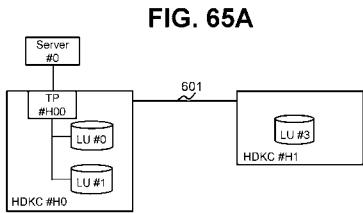
#H1		5101	5102	5103
未割当ページ数	前回未割当ページ数	6500	7000	未割当ページ使用量
				500

【図 6 4】

FIG. 64



【図 6 5】



【図 6 6】

FIG. 66

仮想ページ管理テーブル (#H0)					1304	1305
1301		1302	1303			225
DP-LU ID	仮想ページ アドレス	割当状態	Prod-RG ID	実ページ アドレス		
160	0x00	割当	H00	0x00		
	0x01	割当	H13	0x01		
	0x02	割当	H15	0x01		
162	0x00	割当	H01	0x01		
	0x01	割当	H02	0x00		
	0x02	割当	H14	0x00		

【図 6 7】

FIG. 67

1401	1402	1403	1404	1405	1406
テーブル ポインタ	Prod-RG ID	実ページアドレス	割当状態	DP-LU ID	未割当キューポインタ
0x0000	H00	0x00	割当	160	—
0x0001		0x01	使用不可	—	—
0x0002		0x02	未割当	—	0x1003
0x1000	H01	0x00	使用不可	—	—
0x1001		0x01	割当	162	—
0x1002		0x02	使用不可	—	—
0x2000	H02	0x00	割当	162	—
0x2001		0x01	使用不可	—	—
0x2002		0x02	未割当	—	0x3003
0x3000	H03	0x00	使用不可	—	—
0x3001		0x01	割当	160	—
0x3002		0x02	使用不可	—	—
0x4000	H14	0x00	未割当	162	—
0x4001		0x01	未割当	—	0x5003
0x4002		0x02	未割当	—	—
0x5000	H15	0x00	使用不可	—	—
0x5001		0x01	使用不可	160	—
0x5002		0x02	使用不可	—	—
...	...	...	...	...	...

【図 6 8】

FIG. 68

1301	1302	1303	1304	1305
DP-LU ID	仮想ページアドレス	割当状態	Prod-RG ID	実ページアドレス
161	0x00	未割当	—	—
	0x01	割当	H13	0x00
	0x02	未割当	—	—
163	0x00	未割当	—	—
	0x01	未割当	—	—
	0x02	未割当	—	—
...		...	...	...

FIG. 69

1401	1402	1403	1404	1405	1406
テーブル ポイント	Pool-RG ID	実ページアドレス	割当状態	DP-LU ID	未割当キューポイント
0x0000	H00	0x00	使用不可	—	—
0x0001		0x01	未割当	—	0x1000
0x0002		0x02	使用不可	—	—
⋮	H01	⋮	⋮	⋮	⋮
0x1000		0x00	未割当	—	0x2001
0x1001		0x01	使用不可	—	—
0x1002		0x02	未割当	—	0x2003
⋮	H02	⋮	⋮	⋮	⋮
0x2000		0x00	使用不可	—	—
0x2001		0x01	未割当	—	0x3002
0x2002		0x02	使用不可	—	—
⋮	H13	⋮	⋮	⋮	⋮
0x3000		0x00	割当	161	—
0x3001		0x01	使用不可	—	—
0x3002		0x02	未割当	—	0x4003
⋮	H14	⋮	⋮	⋮	⋮
0x4000		0x00	使用不可	—	—
0x4001		0x01	未割当	—	0x5000
0x4002		0x02	使用不可	—	—
⋮	H15	⋮	⋮	⋮	⋮
0x5000		0x00	未割当	—	0x0001
0x5001		0x01	使用不可	—	—
0x5002		0x02	未割当	—	0x0003
⋮	⋮	⋮	⋮	⋮	⋮

フロントページの続き

(72)発明者 本間 繁雄

神奈川県小田原市中里322番2号 株式会社日立製作所 ITプラットフォーム事業本部内

(72)発明者 山本 政行

神奈川県横浜市戸塚区吉田町292番地 株式会社日立製作所 横浜研究所内

審査官 稲葉 崇

(56)参考文献 特開2009-163703 (JP, A)

特開2006-260292 (JP, A)

特開2008-299387 (JP, A)

特開2010-122814 (JP, A)

特開2009-277211 (JP, A)

特開2003-150318 (JP, A)

特開2010-033466 (JP, A)

(58)調査した分野(Int.Cl., DB名)

G06F 3/06-3/08

G06F 13/10-13/14

G06F 12/00