(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第6260384号
(P6260384)
(45)発行日　平成30年1月17日(2018.1.17)　(24)登録日　平成29年12月22日(2017.12.22)

(51)Int.Cl.　F I
G06F 3/06 (2006.01)　G06F 3/06 301N
G06F 3/06 304N

請求項の数 8　(全 33 頁)

(21)出願番号　特願2014-56799(P2014-56799)
(22)出願日　平成26年3月19日(2014.3.19)
(65)公開番号　特開2015-179425(P2015-179425A)
(43)公開日　平成27年10月8日(2015.10.8)
審査請求日　平成28年12月6日(2016.12.6)

(73)特許権者　000005223
富士通株式会社
神奈川県川崎市中原区上小田中４丁目１番１号
(74)代理人　100092978
弁理士　真田　有
(74)代理人　100112678
弁理士　山本　雅久
(72)発明者　大江　和一
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内
(72)発明者　岩田　聡
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内

最終頁に続く

(54)【発明の名称】ストレージ制御装置，制御プログラム，及び制御方法

(57)【特許請求の範囲】
【請求項１】
第１の記憶装置の記憶領域を所定の大きさで分割した複数の単位領域について、入力された要求に対する応答性能を監視する監視部と、
前記第１の記憶装置の移動対象の単位領域に記憶されたデータを前記第１の記憶装置とは異なる性能の第２の記憶装置に移動する移動処理において、前記移動対象の単位領域を所定の分割数により複数の分割領域に分割し、前記データを前記分割領域の単位で前記第２の記憶装置に移動する分割部と、
前記監視部が監視した前記移動処理実行中の第１の応答性能に基づいて、前記所定の分割数を変更する変更部と、
をそなえることを特徴とする、ストレージ制御装置。
【請求項２】
前記変更部は、前記第１の応答性能と、前記移動処理実行前の第２の応答性能とに基づいて、前記所定の分割数を変更する
ことを特徴とする、請求項１記載のストレージ制御装置。
【請求項３】
前記所定の分割数の初期値は、前記移動処理実行前の第２の応答性能を基に求められることを特徴とする、請求項２記載のストレージ制御装置。
【請求項４】
前記変更部は、前記第１の応答性能と前記移動処理実行前の第２の応答性能とを比較し

、前記第１の応答性能が前記第２の応答性能よりも劣化していると判断した場合に前記所定の分割数を増加させる一方、前記第１の応答性能が前記第２の応答性能よりも優れていると判断した場合に前記所定の分割数を減少させる、
ことを特徴とする、請求項１～３のいずれか１項記載のストレージ制御装置。
【請求項５】
前記変更部は、前記監視部が前記移動処理実行中の複数の時点で監視した複数の応答性能を取得し、前記第２の応答性能と前記複数の応答性能の平均値とを比較して、前記平均値が前記第２の応答性能よりも劣化しているか否かに応じて前記所定の分割数を増減させる、
ことを特徴とする、請求項４記載のストレージ制御装置。
【請求項６】
前記監視部は、前記複数の単位領域について単位領域ごとに入出力数を集計し、
前記ストレージ制御装置は、
前記監視部により集計された入出力数が第１の閾値より大きな単位領域と所定の距離内にある単位領域を繋ぎ合わせた拡張領域と、該拡張領域と繋がる他の拡張領域と、を合わせた移動領域を特定する特定部、をさらにそなえ、
前記分割部は、前記移動領域に記憶されたデータを前記第２の記憶装置に移動する移動処理において、前記移動領域に含まれる複数の単位領域の各々を所定の分割数により複数の分割領域に分割し、前記移動領域に記憶されたデータを前記分割領域の単位で前記第２の記憶装置に移動する、
ことを特徴とする、請求項１～５のいずれか１項記載のストレージ制御装置。
【請求項７】
第１の記憶装置及び第２の記憶装置の制御を行なうコンピュータに、
前記第１の記憶装置の記憶領域を所定の大きさで分割した複数の単位領域について、入力された要求に対する応答性能を監視し、
前記第１の記憶装置の移動対象の単位領域に記憶されたデータを前記第１の記憶装置とは異なる性能の第２の記憶装置に移動する移動処理を行ない、
前記移動処理において、前記移動対象の単位領域を所定の分割数により複数の分割領域に分割し、前記データを前記分割領域の単位で前記第２の記憶装置に移動し、
前記監視により監視した前記移動処理実行中の第１の応答性能に基づいて、前記所定の分割数を変更する、
処理を実行させることを特徴とする、制御プログラム。
【請求項８】
第１の記憶装置及び第２の記憶装置の制御を行なうストレージ制御装置における制御方法であって、
前記第１の記憶装置の記憶領域を所定の大きさで分割した複数の単位領域について、入力された要求に対する応答性能を監視し、
前記第１の記憶装置の移動対象の単位領域に記憶されたデータを前記第１の記憶装置とは異なる性能の第２の記憶装置に移動する移動処理を行ない、
前記移動処理において、前記移動対象の単位領域を所定の分割数により複数の分割領域に分割し、前記データを前記分割領域の単位で前記第２の記憶装置に移動し、
前記監視により監視した前記移動処理実行中の第１の応答性能に基づいて、前記所定の分割数を変更する、
ことを特徴とする、制御方法。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、ストレージ制御装置，制御プログラム，及び制御方法に関する。
【背景技術】
【０００２】

データを格納するストレージシステムとして、複数の記憶媒体（記憶装置）を組み合わせた階層ストレージシステムが用いられることがある。階層ストレージシステムは、例えば、高速アクセスが可能であるが比較的低容量，高価格なＳＳＤ（Solid State Drive）と、大容量で低価格であるが比較的低速なＨＤＤ（Hard Disk Drive）とを含む。

階層ストレージシステムでは、アクセス頻度が低い領域をＨＤＤに配置する一方、アクセス頻度が高い領域をＳＳＤに配置することで、ＳＳＤの使用効率を高め、システム全体の性能を高めることができる。つまり、階層ストレージシステムの性能を向上させるには、アクセス頻度が高い領域を効率的にＳＳＤに配置することが望ましい。

【０００３】

アクセス頻度が高い領域をＳＳＤに配置する手法としては、例えば、前日のアクセス頻度に応じて、１日単位でアクセス頻度が高い領域をＳＳＤに配置する手法がある。具体的には、階層ストレージシステムは、ユーザのアクセス頻度が低い深夜時間帯に２４時間分のアクセス頻度を集計し、アクセス頻度が高い領域から順にＳＳＤへ配置する。毎日ほぼ同じ領域へアクセス集中が発生するワークロードにおいてはこの手法で十分である。

【０００４】

しかし、数分から数十分程度の比較的短時間にアクセス集中（負荷）が移動するワークロードにおいては、1日単位のアクセス頻度の集計では追従できない場合が多い。なお、ワークロードとは、記憶装置へのアクセス分布をいい、時間の経過と記憶装置のオフセット位置（領域）とに応じて変化する。短時間に負荷が移動するワークロードに対応するためには、アクセス頻度が高くなった領域をリアルタイムで把握して、当該領域をＳＳＤに移動することが好ましい。

【０００５】

また、ＨＤＤからＳＳＤへデータを移動する際にも、当該データに対するユーザからのＩＯ（Input Output）（以下、ユーザＩＯという）が発生し得る。このユーザＩＯへの対処としては、例えば、ストレージシステムが、第１のボリュームから第２のボリュームへのデータの移行中に、共有メモリ等に対象データを転送してアクセス応答可能な状態にする技術が知られている（例えば、特許文献１参照）。これにより、移行中における対象データに対するアクセス性能が確保される。

【０００６】

また、記憶制御装置が、データ処理装置から論理的記憶装置にアクセスがあったとき、アクセス位置が再配置完了領域か再配置未完了領域かに応じて、再配置先又は当該論理的記憶装置にアクセスさせる技術も知られている（例えば、特許文献２参照）。

さらに、ストレージ管理装置が、アクセス対象範囲の論理セグメント（segment）と物理セグメントとを、それぞれサブ論理セグメントとサブ物理セグメントとに分割する技術も知られている（例えば、特許文献３参照）。この技術では、ストレージ管理装置は、サブセグメント（sub-segment）単位で対象データを再配置することで、ストレージ装置にかかる負荷を分散し、アクセス性能を向上させることができる。

【先行技術文献】

【特許文献】

【０００７】

【特許文献１】特開２００８－２９９５５９号公報

【特許文献２】特開２００３－２７１４２５号公報

【特許文献３】国際公開第２００８／１２６２０２号パンフレット

【発明の概要】

【発明が解決しようとする課題】

【０００８】

データの移行中に、対象データを一時バッファに転送してユーザＩＯを処理する手法では、移動完了後に一時バッファと移動先のＳＳＤとの間でｓｙｎｃ（同期）等の処理が発生するため、移動時間が伸びてしまう。

最も短時間にデータを移動する手法は、データの移動中は当該データへのユーザＩＯを

ブロックすることが考えられるが、ユーザＩＯがブロックされることで、ユーザＩＯのレスポンスは悪化する。一方、階層ストレージシステムが一度に移動する領域を小さくすれば、ユーザＩＯのレスポンス悪化を小さくできるが、代わりに移動時間が大きくなる。

【０００９】

１つの側面では、本発明は、第１の記憶装置から第２の記憶装置へのデータ移動にかかる処理時間を低減させつつ、入力される要求への応答性能の劣化を抑制することを目的とする。

なお、前記目的に限らず、後述する発明を実施するための形態に示す各構成により導かれる作用効果であって、従来の技術によっては得られない作用効果を奏することも本発明の他の目的の１つとして位置付けることができる。

【課題を解決するための手段】

【００１０】

本件のストレージ制御装置は、第１の記憶装置の記憶領域を所定の大きさで分割した複数の単位領域について、入力された要求に対する応答性能を監視する監視部をそなえる。また、このストレージ制御装置は、前記第１の記憶装置の移動対象の単位領域に記憶されたデータを前記第１の記憶装置とは異なる性能の第２の記憶装置に移動する移動処理において、以下の処理を行なう分割部をそなえる。ここで、分割部は、前記移動対象の単位領域を所定の分割数により複数の分割領域に分割し、前記データを前記分割領域の単位で前記第２の記憶装置に移動する。さらに、このストレージ制御装置は、前記監視部が監視した前記移動処理実行中の第１の応答性能に基づいて、前記所定の分割数を変更する変更部をそなえる。

【発明の効果】

【００１１】

一実施形態によれば、第１の記憶装置から第２の記憶装置へのデータ移動にかかる処理時間を低減させつつ、入力される要求への応答性能の劣化を抑制することができる。

【図面の簡単な説明】

【００１２】

【図１】階層移動中に移動領域に発生したユーザＩＯをブロックする手法の一例を示す図である。

【図２】移動領域をサブセグメント単位で移動する例を示す図である。

【図３】サブセグメントの分割数と領域移動時間との関係の一例を示す図である。

【図４】サブセグメントの分割数とＩＯ応答時間との関係の一例を示す図である。

【図５】サブセグメントの分割数を２５６にした場合のＩＯ応答時間の一例を示す図である。

【図６】サブセグメントの分割数を２０４８にした場合のＩＯ応答時間の一例を示す図である。

【図７】一実施形態に係る階層ストレージシステムの構成例を示す図である。

【図８】図７に示すデータベースの一例を示す図である。

【図９】図７に示す階層テーブルの一例を示す図である。

【図１０】データ収集部によるデータ収集処理の動作例を示すフローチャートである。

【図１１】ワークロード分析部による移動判定処理の動作例を示すフローチャートである。

【図１２】移動指示部による移動指示通知処理の動作例を示すフローチャートである。

【図１３】分割数判定部による分割数判定処理の動作例を示すフローチャートである。

【図１４】階層ドライバによる転送指示通知処理の動作例を示すフローチャートである。

【図１５】階層ドライバによる転送完了受信処理の動作例を示すフローチャートである。

【図１６】分割部による転送指示受信処理の動作例を示すフローチャートである。

【図１７】分割部による分割数更新処理の動作例を示すフローチャートである。

【図１８】ＩＯマップ部によるＩＯ受信処理の動作例を示すフローチャートである。

【図１９】図７に示す階層ストレージ制御装置のハードウェア構成例を示す図である。

【図２０】適用例に係る階層ストレージ制御装置による動的階層制御を説明するための図である。
【図２１】適用例に係る階層ストレージ制御装置による動的階層制御を説明するための図である。
【図２２】適用例に係る階層ストレージシステムの構成例を示す図である。
【図２３】適用例に係るデータ収集部によるデータ収集処理の動作例を示すフローチャートである。
【図２４】図２２に示すデータベースの一例を示す図である。
【図２５】適用例に係るワークロード分析部による移動判定処理の動作例を示すフローチャートである。
【図２６】図２２に示す候補テーブルの一例を示す図である。
【図２７】図２２に示す管理テーブルの一例を示す図である。
【図２８】適用例に係る移動指示部による移動指示通知処理の動作例を示すフローチャートである。
【発明を実施するための形態】
【００１３】
以下、図面を参照して実施の形態を説明する。
〔１〕一実施形態
〔１－１〕対比例
はじめに、図１及び図２に示す対比例を説明する。図１は、階層移動中に移動領域に発生したユーザＩＯをブロックする手法の一例を示す図であり、図２は、移動領域をサブセグメント単位で移動する例を示す図である。なお、図１及び図２では、階層ストレージ制御装置１００がＬｉｎｕｘ（登録商標） ｄｅｖｉｃｅ－ｍａｐｐｅｒの機能を用いるものとする。この例では、ｄｅｖｉｃｅ－ｍａｐｐｅｒがストレージボリュームをセグメント単位で監視し、高負荷となったセグメントのデータをＨＤＤ３００からＳＳＤ２００へ移動することで高負荷セグメントへのＩＯを処理する。
【００１４】
まず、図１において、階層ストレージ制御装置１００のユーザ空間で実行されるアプリケーションがデータ格納先の変更要求としてコピー指示を発行する（図１の（１）参照）。ＯＳ（Operating System）空間で実行される階層ドライバ１１０は、コピー指示を受け取ると、格納先を変更するため、デバイス間のデータコピーを非同期で実行するｋｃｏｐｙｄに、ＳＳＤ２００－ＨＤＤ３００間のコピー（移動）を指示する。ｋｃｏｐｙｄによる移動中、ユーザからＩＯ要求が発行されると（図１の（２）参照）、階層ドライバ１１０は、メモリ等のペンディングキューにＩＯ要求を格納し、移動完了まで待ち合わせを行なう（図１の（３）参照）。なお、ｄｅｖｉｃｅ－ｍａｐｐｅｒおよびｋｃｏｐｙｄはコンピュータプログラムとして実装されている。
【００１５】
移動が完了すると（図１の（４）参照）、階層ドライバ１１０は、移動先のＳＳＤ２００又はＨＤＤ３００を選択し、ＳＳＤドライバ１２０又はＨＤＤドライバ１３０を介してペンディングキューで保留していたＩＯ要求を発行する（図１の（５）参照）。そして、ＩＯ要求を受けた移動先のＳＳＤ２００又はＨＤＤ３００は、ユーザへＩＯ応答を返す（図１の（６）参照）。
【００１６】
図１に示す例では、ＩＯ要求がペンディングキューに保留されていた時間は、そのままユーザにレスポンスの悪化として見えてしまう。例えば、セグメント＝１ＧＢ（Byte），ＨＤＤ３００のスループット性能＝１００ＭＢ／ｓｅｃ，ＳＳＤ２００のスループット性能＝１０００ＭＢ／ｓｅｃであると仮定すると、移動時間は、１［ＧＢ］／１００［ＭＢ／ｓｅｃ］＝１０秒となる。すなわち、ユーザＩＯが最大１０秒待たされる可能性があることが分かる。一時的であっても、階層ストレージシステムにおいて、ユーザＩＯが最大１０秒待たされることは許容されない場合が多い。

【００１７】
一方、図２では、セグメントをさらにサブセグメントというより小さい単位に分割し、階層ストレージ制御装置１００がサブセグメント単位で領域移動を行なう。これにより、ユーザＩＯの待ち時間を、セグメント全体の移動時間よりも小さいサブセグメントの移動時間に抑えることができる。
しかし、図２に示すように、サブセグメント単位で領域移動を行なう場合、セグメント単位での領域移動よりも全体での移動コストがかかってしまう。これは、移動時間以外のオーバーヘッドがサブセグメントの分割数分だけ増えるためである。
【００１８】
そこで、セグメント移動時のユーザＩＯ保留時間をできるだけ短くしつつ、可能な限り短時間にセグメント移動を完了させることが望ましい。上述のように、セグメントをサブセグメントに分割し、さらにその分割数を増やすことでユーザＩＯの保留時間を小さくすることは可能である。しかし、セグメント全体の移動時間は大きく増加してしまう。
図３～図６は、サブセグメントの分割数を増やした場合に移動時間がどの程度増加するかを評価した評価結果である。図３はサブセグメントの分割数と領域移動時間との関係の一例を示す図であり、図４はサブセグメントの分割数とＩＯ応答時間との関係の一例を示す図である。図５及び図６は、それぞれ、サブセグメントの分割数を２５６及び２０４８にした場合のＩＯ応答時間の一例を示す図である。なお、図３～図６は、セグメント＝１ＧＢ，ＨＤＤ３００のスループット性能＝１００ＭＢ／ｓｅｃ，ＳＳＤ２００のスループット性能＝１０００ＭＢ／ｓｅｃの環境での評価結果である。
【００１９】
例えば、図３に示すように、ＳＳＤ２００からＨＤＤ３００への領域移動は、セグメントが未分割の場合、１０秒程度で完了する。しかし、分割数を増やしていくことで、領域移動時間は最大１２倍（１２０秒）程度にまで増加することが分かる。
一方、ユーザＩＯへの応答時間に関しては、図４に示すように、セグメントが未分割の場合にはコピー中のＩＯ応答時間は１０秒以上であるが、２５６分割以上の場合には平均０．４秒未満となる。これは、コピーなしの場合の平均レスポンスと同等の応答時間である。但し、図５に示すように、分割数が２５６の場合でも、応答時間が１秒を超えるケースが一部に発生し得るため、分割数をある程度増加させてもセグメントの移動の影響を完全に隠ぺいできるとは限らない。
【００２０】
なお、応答時間が１秒を超えてしまうリクエストに関しても、分割数をより増やすことで応答時間を減らすことは可能であるが、その分コピー時間が増加してしまう。例えば、図６に示すように、２０４８分割にすると応答時間が１秒以上となるリクエストは発生しなくなる。しかし、ＳＳＤ２００からＨＤＤ３００へ移動する場合の領域移動時間は、１４秒（２５６分割）から４５秒（２０４８分割）へと大幅に増加してしまう（図３参照）。
【００２１】
ユーザＩＯの応答時間を１秒未満にする場合、図５及び図６の例では、２０４８分割以上とすることが望ましいが、階層ストレージシステムにおいてそこまでの応答時間保証が要求されていない場合には、２５６分割で十分といえる。このように、保証したいユーザＩＯの応答時間を予め定めておき、これを上回らない程度の移動単位を決定することで、移動時間を抑えることができる。
【００２２】
なお、図３～図６で示した評価（実験）結果は、階層ストレージシステムで使用される機器（例えばＳＳＤ２００やＨＤＤ３００，バス等）やワークロードによって変動することが容易に想像できる。この評価結果では、図４より２５６分割が最適値であるが、使用する機器等の条件が変化すればこの分割数も増減する可能性がある。
ところで、移動単位とユーザＩＯに加わるオーバーヘッド、つまりレスポンス悪化との関係は、階層ストレージシステムで使用される機器やワークロードによって変化する。す

なわち、閾値制御のような単純な手法では、機器の性能やワークロードの変化等に応じて、階層ストレージシステムに最適な移動単位で領域移動を行なうことは困難である。

【００２３】

〔１－２〕階層ストレージシステムの説明

上述した点に鑑み、本実施形態に係る階層ストレージシステム１（図７参照）は、以下に詳述するように、ユーザＩＯの平均レスポンスを監視することにより、ユーザＩＯのレスポンスの変化に応じて移動単位を動的に変更することができる。これにより、階層ストレージシステム１は、直前の平均レスポンスと同等のレスポンスを保ちつつ、分割数を自律的に最小にすることが可能となる。すなわち、第１の記憶装置から第２の記憶装置へのデータ移動にかかる処理時間を低減させつつ、入力される要求への応答性能の劣化を抑制することができる。

【００２４】

図７は、一実施形態に係る階層ストレージシステム１の構成例を示す図である。図７に示すように、階層ストレージシステム（ストレージ装置）１は、階層ストレージ制御装置１０，ＳＳＤ２０，及びＨＤＤ３０をそなえる。

階層ストレージ制御装置１０は、図示しない入力装置や、ネットワークを介したホスト装置からのユーザＩＯに応じて、ＳＳＤ２０及びＨＤＤ３０への種々のアクセスを行なうことができる。例えば、階層ストレージ制御装置１０は、ＳＳＤ２０及びＨＤＤ３０へのリード又はライト等のアクセスを行なうことができる。階層ストレージ制御装置１０としては、ＰＣ（Personal Computer）やサーバ，又はコントローラモジュール（ＣＭ；Controller Module）等の情報処理装置が挙げられる。

【００２５】

また、本実施形態に係る階層ストレージ制御装置１０は、ユーザＩＯのアクセス頻度に応じて、アクセス頻度が低い領域をＨＤＤ３０に配置する一方、アクセス頻度が高い領域をＳＳＤ２０に配置する、動的階層制御を行なうことができる。

ＨＤＤ３０は、種々のデータやプログラム等を格納する記憶装置の一例であり、ＳＳＤ２０は、ＨＤＤ３０とは異なる性能の（例えばより高速な）記憶装置の一例である。本実施形態において、互いに異なる記憶装置（以下、便宜上、第１及び第２の記憶装置と表記する場合がある）として、ＨＤＤ３０等の磁気ディスク装置，ＳＳＤ２０等の半導体ドライブ装置をそれぞれ例に挙げているが、これに限定されるものではない。第１及び第２の記憶装置として、互いに性能差（例えばリード／ライトの速度差）のある種々の記憶装置が用いられればよい。

【００２６】

ＳＳＤ２０及びＨＤＤ３０は、階層ストレージシステム１におけるストレージボリュームを構成する。ＳＳＤ２０及びＨＤＤ３０の各々は、ストレージボリューム上のセグメント（単位領域）のデータを格納可能な記憶領域を含む。セグメントは、階層ストレージ制御装置１０による階層移動の最小単位であり、図７では、１セグメントが１ＧＢであるものとする。階層ストレージ制御装置１０は、セグメント単位で、ＳＳＤ２０－ＨＤＤ３０間の領域移動を制御する。

【００２７】

なお、図７では、階層ストレージシステム１がそれぞれ１つのＳＳＤ２０及びＨＤＤ３０をそなえるものとしているが、これに限定されるものではなく、それぞれ複数のＳＳＤ２０及びＨＤＤ３０をそなえてもよい。

〔１－３〕階層ストレージ制御装置の説明

次に、階層ストレージ制御装置１０の詳細について説明する。

【００２８】

階層ストレージ制御装置１０は、一例として、図７に示すように、階層管理部１１，階層ドライバ１２，ＳＳＤドライバ１３，及びＨＤＤドライバ１４をそなえる。なお、階層管理部１１は、ユーザ空間で実行されるプログラムとして実現され、階層ドライバ１２，ＳＳＤドライバ１３，及びＨＤＤドライバ１４は、ＯＳ空間で実行されるプログラムとし

て実現される。
【００２９】
階層管理部１１は、ｂｌｋｔｒａｃｅを用いて、ＳＳＤ２０又は／及びＨＤＤ３０についてトレースされたＩＯの情報に基づいて、領域移動を行なうセグメントを判定し、判定したセグメントのデータの移動を階層ドライバ１２に指示する。ここで、ｂｌｋｔｒａｃｅは、ブロックＩＯレベルでのＩＯをトレースするコマンドである。階層管理部１１は、ｂｌｋｔｒａｃｅに代えて、ディスクＩＯの利用状況を確認するコマンドであるｉｏｓｔａｔを用いてもよい。なお、ｂｌｋｔｒａｃｅ及びｉｏｓｔａｔはＯＳ空間で実行される。
【００３０】
階層管理部１１は、データ収集部１１ａ，データベース１１ｂ，ワークロード分析部１１ｃ，移動指示部１１ｄ，及び分割数判定部１１ｅをそなえる。
なお、階層管理部１１の動作を実現するために、階層ストレージシステム１の管理者等は、予め以下の情報を決定しておくことが好ましい。
・全セグメントがＨＤＤ３０にある場合の平均レスポンス（コピーなし），並びに平均レスポンスがコピーなしの場合と同等になる分割数。図４に示す例では、分割数は２５６である。なお、平均レスポンス及び分割数は、階層ストレージシステム１で使用予定の機器を用いて、図４に示すような実験を行なうことで求められる。
【００３１】
・平均レスポンスを求める期間。例えば６０ｓ程度。
・平均レスポンスの誤差範囲。例えば５０ｍｓ程度。
データ収集部１１ａは、ｂｌｃｔｒａｃｅを用いてＳＳＤ２０又は／ＨＤＤ３０についてトレースされたＩＯの情報を所定間隔（例えば１分間隔）で収集する。また、データ収集部１１ａは、収集した情報に基づいて、セグメントごとに、例えば、セグメントを特定する情報，合計ＩＯ数（ｉｏｐｍ；I0 per minute），及び平均レスポンス（応答性能）を集計する。そして、データ収集部１１ａは、集計結果をタイムスタンプとともにデータベース１１ｂに書き込む。なお、セグメントを特定する情報としては、ボリューム上のオフセットに関する情報を用いることができる。
【００３２】
また、データ収集部１１ａは、全セグメントを対象にした（全セグメントの）合計ＩＯ数及び平均レスポンスも集計し、タイムスタンプとともにデータベース１１ｂに書き込むことができる。このとき、データ収集部１１ａは、全セグメントを対象にした情報をデータベース１１ｂに追加したことを分割数判定部１１ｅに通知してもよい。
なお、データ収集部１１ａは、各セグメント又は／及び全セグメントへのＩＯのリードライト比（ｒｗ比）を集計し、上述した情報に含めてもよい。
【００３３】
このように、データ収集部１１ａは、ＳＳＤ２０又はＨＤＤ３０で使用される領域を所定の大きさで分割した複数の単位領域について、入力された要求に対する応答性能を監視する監視部の一例である。
データベース１１ｂは、データ収集部１１ａにより集計されたセグメントに関する情報を記憶するものであり、例えば図示しないメモリ等により実現される。
【００３４】
図８は、図７に示すデータベース１１ｂの一例を示す図である。図８に示すように、データベース１１ｂは、セグメントごとに、セグメントを特定する情報，ＩＯ数，平均レスポンス，及びタイムスタンプを対応させて記憶するテーブルである。例えば、セグメント“１”であるセグメントは、合計ＩＯ数が“1000”，平均レスポンスが“0.6”（秒），タイムスタンプが“1”である。
【００３５】
なお、セグメントを特定する情報として、セグメントの番号を用いているが、これに代えてストレージボリュームの先頭オフセットが用いられてもよい。ここで、ＩＯ数は、セ

グメントに対して１分間に行なわれたＩＯの合計数であり、平均レスポンスは、階層ストレージ制御装置１０がセグメントへのＩＯを受け取ってからレスポンスを送信するまでにかかった時間の平均である。タイムスタンプは、時刻を識別する識別子であり、例えば時刻そのものが設定されてもよい。

【００３６】

また、図８において、セグメントが“all”のエントリは、全セグメントを対象にした集計結果である。セグメントが“all”のエントリについては、後述する分割数判定部１１ｅにより、過去ｎ個のデータが参照されるため、複数の“all”のエントリを追加できるようになっている。なお、“all”のエントリの新旧は、タイムスタンプにより識別可能である。一方、個々のセグメントのエントリについては、同一セグメントのデータを上書きできるようにしてもよいし、“all”と同様に複数のデータを登録できるようにしてもよい。

【００３７】

ワークロード分析部１１ｃは、データベース１１ｂが記憶するセグメントから、ＳＳＤ２０又はＨＤＤ３０にデータを移動するセグメントを選択し、選択したセグメントに関する情報を移動指示部１１ｄに渡す。

一例として、ワークロード分析部１１ｃは、セグメント数が同時に階層移動を行なう最大のセグメント数（所定数）に達するまで、ＩＯ数が多い順にセグメントを抽出することができる。又は、ワークロード分析部１１ｃは、ＳＳＤ２０にデータを移動するセグメントとして、ＩＯ数又はアクセスの集中率（全体に対するＩＯ数の割合）が所定の閾値よりも高いセグメントを抽出してもよい。

【００３８】

また、ワークロード分析部１１ｃは、ＨＤＤ３０にデータを移動するセグメントとして、例えばＩＯ数が上記所定数に入らなかった、又はＩＯ数若しくはアクセスの集中率が所定の閾値以下となった、ＳＳＤ２０上のセグメントを抽出することができる。

なお、ワークロード分析部１１ｃは、上記のＳＳＤ２０又はＨＤＤ３０にデータを移動するセグメントの抽出条件に所定回数以上連続して該当したときに、当該セグメントを、ＳＳＤ２０又はＨＤＤ３０にデータを移動するセグメントとして抽出してもよい。また、ワークロード分析部１１ｃは、上記ＩＯ数等の他に、リードライト比（ｒｗ比）に基づいてセグメントを選択してもよい。

【００３９】

ここで、ワークロード分析部１１ｃは、移動指示部１１ｄに対して、ＨＤＤ３０内のセグメントについてＳＳＤ２０への階層移動を指示した後、ＳＳＤ２０内の他のセグメントについてＨＤＤ３０への階層移動を指示することができる。一方、ワークロード分析部１１ｃは、あるセグメントについてＳＳＤ２０への階層移動を行なっている間に当該セグメントへの負荷が下がると予測される場合には、他のセグメントについてのみＨＤＤ３０への階層移動を指示してもよい。

【００４０】

例えば、ワークロード分析部１１ｃは、スパイクの平均余命時間と階層移動にかかる時間とに基づいて、階層移動中のセグメントへの負荷が下がるか否かを判断できる。なお、スパイクとは、一部のセグメントに負荷が集中することであり、平均余命時間は、負荷が継続する継続時間から既に実行済みの実行時間を減じた時間であり、ワークロードに応じて定まる値である。管理者等は予め平均余命時間を求め、階層ストレージ制御装置１０に設定しておくことができる。

【００４１】

具体的には、ワークロード分析部１１ｃは、ＳＳＤ２０にデータを移動するセグメントを抽出し、抽出したセグメントについてＳＳＤ２０にデータを移動するコスト（時間）を計算する。そして、ワークロード分析部１１ｃは、平均余命時間が移動時間以下になる場合には、ＳＳＤ２０からＨＤＤ３０への階層移動のみを行なうと判断することができる。

移動指示部１１ｄは、ワークロード分析部１１ｃからの指示に基づいて、階層ドライバ

１２に、選択されたセグメントのデータの、ＨＤＤ３０からＳＳＤ２０への移動、又は、ＳＳＤ２０からＨＤＤ３０への移動を指示する。このとき、移動指示部１１ｄは、選択されたセグメントのストレージボリューム上のオフセットをＨＤＤ３０上のオフセットに変換してセグメントごとにデータの移動を指示する。例えば、ＨＤＤ３０のセクターサイズが５１２Ｂである場合に、ボリューム上のオフセットが１ＧＢであれば、ＨＤＤ３０上でのオフセットは１×１０２４×１０２４×１０２４／５１２＝２０９７１５２となる。

【００４２】

また、移動指示部１１ｄは、移動指示を出したセグメント数と、ワークロード分析部１１ｃで移動の判定が行なわれたデータのタイムスタンプ（直近のタイムスタンプ）とを含む移動開始通知を、分割数判定部１１ｅに通知する。

分割数判定部１１ｅは、セグメントの分割数を決定するとともに、セグメントのデータの移動開始前後のＩＯレスポンスの変化に基づいて、動的に分割数を変更する。

【００４３】

具体的には、分割数判定部１１ｅは、移動指示部１１ｄから移動開始通知を受け取ると、直前の分割数判定結果（立ち上げ直後なら初期値，例えば２５６）を階層ドライバ１２（分割部１２ｄ）に通知する。階層ドライバ１２は、通知された分割数に従ってセグメントの分割を行ない、データ移動を進める。

また、分割数判定部１１ｅは、データ収集部１１ａにより求められたセグメント移動直前の平均レスポンスを取得し、予め決められた平均レスポンス誤差範囲値より、セグメント移動中の平均レスポンスの期待値を求めておく。なお、セグメント移動直前の平均レスポンスは、移動開始通知に含まれるタイムスタンプに対応する、データベース１１ｂ内の「全セグメントを対象にした平均レスポンス」を取り出すことで取得できる。例えば、平均レスポンスが４００ｍｓであり、平均レスポンス誤差範囲値が５０ｍｓである場合、３５０～４５０ｍｓの範囲が期待値となる。

【００４４】

さらに、分割数判定部１１ｅは、セグメント移動中に、新たにレスポンスが入力されると、期待値の範囲に平均レスポンスが収まっているか否かを評価する。

具体的には、分割数判定部１１ｅは、セグメント移動中に、データ収集部１１ａにより全セグメントを対象にした平均レスポンスが所定数（例えばｎ個）求められると、これらｎ個のデータの平均値を計算し、セグメント移動中の平均レスポンスとする。そして、分割数判定部１１ｅは、セグメント移動中の平均レスポンスが期待値よりも大きい（応答性能が劣化している）場合には、セグメントの分割数を現在の設定値よりも大きくして、階層ドライバ１２に通知する。一方、分割数判定部１１ｅは、セグメント移動中の平均レスポンスが期待値よりも小さい（応答性能が優れている）場合には、セグメントの分割数を現在の設定値よりも小さくして、階層ドライバ１２に通知する。

【００４５】

例えば、分割数判定部１１ｅは、分割数を現在の設定値よりも大きくする場合、２倍（例えば２５６から５１２），３倍，・・・のように変化させてもよいし、所定値を加算させてもよい。また、分割数判定部１１ｅは、分割数を現在の設定値よりも小さくする場合、１／２倍（例えば２５６から１２８にして），１／３倍，・・・のように変化させてもよいし、所定値を減算させてもよい。

【００４６】

このように、分割数判定部１１ｅは、セグメント移動中の平均レスポンスが期待値から外れる場合には、領域移動時の平均レスポンスが移動直前のレスポンスに基づく期待値内に収まるように、セグメント移動中に分割数を動的に変更するのである。これにより、セグメント移動によって、レスポンスを基準としてのセグメント移動前のレスポンスから大きく劣化させずに済み、システムの安定性を保証することができる。

【００４７】

なお、分割数判定部１１ｅは、データ収集部１１ａが階層移動中の複数の時点で監視した複数（ｎ個）の平均レスポンスの平均値を用いることで、ごく短時間にアクセスが集中

した場合のようなワークロードの突発的な変化の影響を緩和することができる。
なお、ここまで、分割数判定部１１ｅは、データ収集部１１ａが監視した移動処理実行中の第１の応答性能と移動処理実行前の第２の応答性能とに基づいて、分割数を変更するものとして説明したが、これに限定されるものではない。
【００４８】
例えば、分割数判定部１１ｅは、移動処理実行中の応答性能に基づいて、分割数を変化させてもよい。一例として、分割数判定部１１ｅは、セグメント移動中に、階層ドライバ１２に通知した分割数でのレスポンスを取得するとともに、分割数を大きく又は小さくさせ、そのときのレスポンスを取得する。そして、分割数判定部１１ｅは、セグメント移動中に取得した複数のレスポンスを比較し、直近のレスポンスが１つ前のレスポンスよりも大きいか否か（応答性能が劣化しているか否か）に応じて、上述の如く分割数を変化させることができる。なお、分割数判定部１１ｅは、比較したレスポンスの差が平均レスポンスの誤差範囲値を超える場合に、分割数を変化させてもよい。
【００４９】
以上のように、分割数判定部１１ｅは、データ収集部１１ａが監視した移動処理実行中の第１の応答性能に基づいて、分割数を変更する変更部の一例であるといえる。
なお、分割数判定部１１ｅは、セグメントの分割数を決定し、決定した分割数を階層ドライバ１２（分割部１２ｄ）に通知するものとして説明したが、移動対象の単位領域を移動する際の移動単位を決定し、通知してもよい。つまり、分割数判定部１１ｅは、移動対象のセグメントについて一度に転送するデータサイズ（移動単位の大きさ）を、上述した判定により決定して（変化させて）、階層ドライバ１２に通知してもよい。
【００５０】
階層ドライバ１２は、ＩＯマップ部１２ａ，ペンディングキュー１２ｂ，階層テーブル１２ｃ，及び分割部１２ｄをそなえる。
ＩＯマップ部１２ａは、ユーザからのストレージボリュームに対するＩＯ要求を階層テーブル１２ｃを用いてＳＳＤドライバ１３又はＨＤＤドライバ１４に振り分け、ＳＳＤドライバ１３又はＨＤＤドライバ１４からのＩＯレスポンスをユーザに返す。
【００５１】
ペンディングキュー１２ｂは、ＩＯ要求を一時的に格納する保持部であり、図示しないメモリ等により実現される。ＩＯマップ部１２ａは、階層移動中のセグメントに対してＩＯ要求が発行されると、当該セグメントのデータの移動が完了するまで、当該ＩＯ要求をペンディングキュー１２ｂに格納し、ＩＯ要求を保留する。データの移動が完了すると、ＩＯマップ部１２ａは、ペンディングキュー１２ｂから当該ＩＯ要求を読み出して、ＳＳＤドライバ１３又はＨＤＤドライバ１４への振り分けを再開する。
【００５２】
階層テーブル１２ｃは、ＩＯマップ部１２ａによるＩＯ要求の振り分け及び分割部１２ｄによる階層制御に用いられるテーブルであり、例えば図示しないメモリ等により実現される。
図９は、図７に示す階層テーブル１２ｃの一例を示す図である。図９に示すように、階層テーブル１２ｃは、ＳＳＤ２０にデータが移動されたセグメントごとに、ＳＳＤオフセットと、ＨＤＤオフセットと、状態とを対応させて記憶するテーブルである。
【００５３】
ＳＳＤオフセットは、ＳＳＤ２０にデータが移動されたセグメントのＳＳＤ２０におけるオフセットを示す。ＳＳＤオフセットは、ボリューム上のサイズ１ＧＢに対応するオフセット“2097152”を単位とする固定値であり、例えば、“０”，“2097152”，“4194304”，“6291456”，．．．となる。
ＨＤＤオフセットは、ＳＳＤ２０にデータが移動されたセグメントのＨＤＤ３０におけるオフセットを示す。ＨＤＤオフセットの値“NULL”は、ＳＳＤオフセットで指定されるＳＳＤ２０の領域が未使用であることを示す。
【００５４】

状態は、セグメントの状態を示し、“allocated”，“Moving（HDD→SSD）”，“Moving（SSD→HDD）”，又は“free”である。“allocated”はセグメントがＳＳＤ２０に割り当てられていることを示し、“Moving（HDD→SSD）”はセグメントのデータがＨＤＤ３０からＳＳＤ２０に転送中であることを示す。“Moving（SSD→HDD）”はセグメントのデータがＳＳＤ２０からＨＤＤ３０に転送中であることを示し、“free”はＳＳＤオフセットで指定されるＳＳＤ２０の領域が未使用であることを示す。

【００５５】

ＩＯマップ部１２ａは、上述した階層テーブル１２ｃを参照することで、ＩＯ要求をＳＳＤドライバ１３又はＨＤＤドライバ１４のいずれに振り分けるかを判定することができるとともに、ＩＯ要求がセグメント移動中であるか否かを判定することができる。

図７の説明に戻り、階層ドライバ１２は、移動指示部１１ｄからセグメント移動指示を受け取ると、ＨＤＤ３０又はＳＳＤ２０の移動対象の単位領域に記憶されたデータをＳＳＤ２０又はＨＤＤ３０に移動する移動処理を実行する。具体的には、階層ドライバ１２は、階層テーブル１２ｃ及び分割部１２ｄにより、セグメント移動指示で指定されたセグメントのデータをＳＳＤ２０－ＨＤＤ３０間で移動する。

【００５６】

より具体的に、階層ドライバ１２は、セグメント移動指示を受け取ると、階層テーブル１２ｃ内のＨＤＤオフセットより“NULL”となっているエントリを探し、セグメント移動指示で指定されたＨＤＤオフセット情報と、状態とを登録する。なお、このとき登録される状態は、“Moving（HDD→SSD）”又は“Moving（SSD→HDD）”である。そして、階層ドライバ１２は、ＳＳＤ２０－ＨＤＤ３０間のデータの転送指示を分割部１２ｄに送出する。

【００５７】

また、階層ドライバ１２は、データの転送完了を通知されると、階層テーブル１２ｃから転送が完了したエントリを探し、状態が“Moving（HDD→SSD）”の場合、当該状態を“allocated”に変更する。一方、階層テーブル１２ｃは、状態が“Moving（SSD→HDD）”である場合、当該状態を“free”に変更するとともに、対応するＨＤＤオフセットを“NULL”に設定する。

【００５８】

分割部１２ｄは、階層ドライバ１２からのＳＳＤ２０－ＨＤＤ３０間のデータの転送指示に応じて、分割数判定部１１ｅから指示された分割数でセグメントを分割し、セグメントのデータの階層移動を行なう。

具体的には、分割部１２ｄは、階層ドライバ１２から転送指示を受け取ると、転送指示に係る各セグメントを、それぞれ分割数判定部１１ｅから指示された分割数ｍｍで分割し、分割した単位でｋｃｏｐｙｄに転送指示を発行する。そして、分割部１２ｄは、ｋｃｏｐｙｄにより、分割した全ての領域内のデータの転送が完了すると、階層ドライバ１２にデータの転送完了を通知する。

【００５９】

また、分割部１２ｄは、分割数判定部１１ｅから分割数ｍｍの更新要求を受け取ると、当該要求に応じて、分割数ｍｍの更新を行なう。例えば、分割部１２ｄは、分割数を２倍にする指示を受け取ると、ｍｍ＝ｍｍ＊２を計算して、分割数ｍｍを更新する。また、分割部１２ｄは、分割数を１／２倍にする指示を受け取ると、ｍｍ＝ｍｍ／２を計算して、分割数ｍｍを更新する。

【００６０】

なお、分割部１２ｄは、階層ドライバ１２からデータの転送指示を受けたときの分割数（移動単位）として、分割数判定部１１ｅによりセグメント移動前（例えば移動直前）のレスポンスを基に求められた所定の分割数（移動単位）を用いることができる。これにより、分割数（移動単位）はセグメント移動前のレスポンスを考慮して設定されるので、ｋｃｏｐｙｄによるデータの転送が開始したときの急激なレスポンスの低下を抑制することができる。

【００６１】
このように、分割部１２ｄは、移動対象の単位領域を所定の分割数により複数の分割領域に分割し、移動対象の単位領域に記憶されたデータを分割領域の単位でＳＳＤ２０又はＨＤＤ３０に移動するのである。換言すれば、分割部１２ｄは、分割数判定部１１ｅからの指示に応じて移動対象の単位領域の移動単位を変更し、変更した移動単位でｋｃｏｐｙｄにデータ転送を指示するのである。
【００６２】
ＳＳＤドライバ１３は、階層ドライバ１２の指示に基づいてＳＳＤ２０へのアクセスを制御する。ＨＤＤドライバ１４は、階層ドライバ１２の指示に基づいてＨＤＤ３０へのアクセスを制御する。
以上のように、本実施形態に係る階層ストレージシステム１によれば、ユーザＩＯの平均レスポンスを監視し、ユーザＩＯのレスポンス変化に応じてレスポンス悪化が収束する領域サイズに移動単位を動的に設定（変更）することができる。従って、ユーザＩＯへのレスポンス悪化と階層移動時間のバランスを適切にとることができ、セグメントの階層移動時の平均レスポンスを可能な限り小さくしつつ、できるだけ少ない分割数で（短時間で）セグメントの階層移動を実現することができる。
【００６３】
すなわち、本実施形態に係る階層ストレージシステム１によれば、分割数判定部１１ｅ及び分割部１２ｄにより、使用する機器の性能やワークロードに応じた最適な移動単位で、ＳＳＤ２０－ＨＤＤ３０間のデータの階層移動を行なうことができるのである。
〔１－４〕階層ストレージシステムの動作例
次に、上述の如く構成された階層ストレージシステム１の動作例を、図１０～図１８を参照して説明する。
【００６４】
はじめに、図１０を参照してデータ収集部１１ａの動作を説明する。図１０はデータ収集部１１ａによるデータ収集処理の動作例を示すフローチャートである。なお、データ収集部１１ａは、ｂｌｋｔｒａｃｅコマンドを６０秒間実行して、終了することを条件として起動される。
図１０に示すように、データ収集部１１ａにより、ｂｌｋｔｒａｃｅコマンドの実行により得られたトレース結果が取り出される（ステップＳ１）。次いで、データ収集部１１ａにより、１ＧＢオフセット単位すなわちセグメント単位で各セグメントのＩＯ数及び平均レスポンスが集計され、タイムスタンプとともにデータベース１１ｂに書き込まれる（ステップＳ２）。
【００６５】
そして、データ収集部１１ａにより、全セグメントを対象にした合計ＩＯ数，平均レスポンスが集計され、タイムスタンプとともにデータベース１１ｂに格納される（ステップＳ３）。なお、データ収集部１１ａは、ステップＳ３の処理を実行した旨を分割数判定部１１ｅに通知してもよい。
このように、データ収集部１１ａは、定期的に全てのセグメントの平均レスポンスを監視することで、流動的に変化するワークロードがユーザＩＯに与える影響を分割数判定部１１ｅにフィードバックすることができる。
【００６６】
次に、図１１を参照してワークロード分析部１１ｃの動作を説明する。図１１はワークロード分析部１１ｃによる移動判定処理の動作例を示すフローチャートである。
図１１に示すように、ワークロード分析部１１ｃにより、データベース１１ｂから直近のタイムスタンプのセグメントについてＩＯ数が取り出される（ステップＳ１１）。そして、ワークロード分析部１１ｃにより、セグメント数が所定数に達するまで、ＩＯ数が多い順に候補セグメントが抽出される（ステップＳ１２）。
【００６７】
次いで、ワークロード分析部１１ｃにより、予め求められた平均余命時間が全候補セグ

メントにかかる移動時間よりも大きいか否かが判定される（ステップＳ１３）。平均余命時間が当該移動時間以下の場合（ステップＳ１３のＮｏルート）、処理がステップＳ１５に移行する。一方、平均余命時間が当該移動時間よりも大きい場合（ステップＳ１３のＹｅｓルート）、ワークロード分析部１１ｃにより、候補セグメントの情報が移動指示部１１ｄに通知され、データの移動（ＨＤＤ３０からＳＳＤ２０）が指示される（ステップＳ１４）。

【００６８】

ステップＳ１５では、ワークロード分析部１１ｃにより、ＳＳＤ２０上のセグメントから候補セグメントに含まれないセグメント、つまりＩＯ数が比較的少ないセグメントが抽出される。そして、ワークロード分析部１１ｃにより、抽出したセグメントの情報が移動指示部１１ｄに通知され、データの移動（ＳＳＤ２０からＨＤＤ３０）が指示される（ステップＳ１６）。

【００６９】

そして、ワークロード分析部１１ｃは、所定時間、例えば６０秒スリープし（ステップＳ１７）、処理がステップＳ１１に移行する。

なお、ワークロード分析部１１ｃは、ステップＳ１２において、ＩＯ数又はアクセスの集中率（全体に対するＩＯ数の割合）が所定の閾値よりも高いセグメントを抽出してもよい。また、ワークロード分析部１１ｃは、ステップＳ１５において、ＨＤＤ３０にデータを移動するセグメントとして、例えばＩＯ数又はアクセスの集中率が所定の閾値以下となったＳＳＤ２０上のセグメントを抽出してもよい。さらに、ワークロード分析部１１ｃは、ステップＳ１２及びＳ１５で抽出するセグメントとして、当該抽出条件に所定回数以上連続して該当したセグメントを選択してもよい。

【００７０】

このように、ワークロード分析部１１ｃは、ＩＯの集中度が高いセグメントのデータをＨＤＤ３０からＳＳＤ２０に移動するように移動指示部１１ｄに指示することによって、ユーザはＨＤＤ３０のデータに高速にアクセスすることができる。また、ワークロード分析部１１ｃは、ＩＯの集中度が低くなったセグメントのデータをＳＳＤ２０からＨＤＤ３０に移動するように移動指示部１１ｄに指示することによって、比較的高価格，低容量のＳＳＤ２０を有効利用することができる。

【００７１】

次に、図１２を参照して移動指示部１１ｄの動作を説明する。図１２は移動指示部１１ｄによる移動指示通知処理の動作例を示すフローチャートである。

図１２に示すように、移動指示部１１ｄにより、ワークロード分析部１１ｃからの移動指示が待ち受けられる（ステップＳ２１）。移動指示を受け取ると、移動指示部１１ｄにより、各セグメントのボリューム上のオフセットがＨＤＤ３０上のオフセットに変換される（ステップＳ２２）。

【００７２】

そして、移動指示部１１ｄにより、セグメントごとに、ＨＤＤ３０上のオフセットと、データの移動方向とが階層ドライバ１２に通知される（ステップＳ２３）。ここで、データの移動方向は、ＨＤＤ３０からＳＳＤ２０か、又は、ＳＳＤ２０からＨＤＤ３０である。そして、移動指示部１１ｄにより、移動指示を出したセグメント数と移動判断を行なったデータのタイムスタンプ（直近のタイムスタンプ）とが分割数判定部１１ｅに通知され（ステップＳ２４）、処理がステップＳ２１に移行する。

【００７３】

このように、移動指示部１１ｄが各セグメントのボリューム上のオフセットをＨＤＤ３０上のオフセットに変換することによって、階層ドライバ１２はＳＳＤ２０とＨＤＤ３０との間でデータを移動することができる。

次に、図１３を参照して分割数判定部１１ｅの動作を説明する。図１３は分割数判定部１１ｅによる分割数判定処理の動作例を示すフローチャートである。

【００７４】

図１３に示すように、分割数判定部１１ｅにより、移動指示部１１ｄからのセグメントの移動情報（セグメント数，タイムスタンプ）が待ち受けられる（ステップＳ３１）。移動情報を受け取ると、分割数判定部１１ｅにより、受け取ったタイムスタンプｔｉｍｅｓｔａｍｐ＿ｏｒｇに対応するデータベース１１ｂのデータへのアクセスが行なわれ、全セグメントの平均レスポンスｒｅｓｐ＿ｏｒｇが取り出される（ステップＳ３２）。

【００７５】

次いで、分割数判定部１１ｅは、ｔｉｍｅｓｔａｍｐ＿ｏｒｇよりも新しいデータがｎ個データベース１１ｂに登録されるまで（例えば６０×ｎ＋１０秒）スリープする（ステップＳ３３）。

新しいデータがｎ個データベース１１ｂに登録されると、分割数判定部１１ｅにより、新しいｎ個のデータにアクセスされ、全データの全セグメントの平均レスポンスが取り出される。そして、分割数判定部１１ｅにより、取り出した平均レスポンスの平均値ｒｅｓｐ＿ｎｅｗが求められる（ステップＳ３４）。

【００７６】

次に、分割数判定部１１ｅにより、ｒｅｓｐ＿ｎｅｗ＞ｒｅｓｐ＿ｏｒｇ＋ｍが成立するか否かが判定される（ステップＳ３５）。ｒｅｓｐ＿ｎｅｗ＞ｒｅｓｐ＿ｏｒｇ＋ｍが成立する場合（ステップＳ３５のＹｅｓルート）、分割数判定部１１ｅにより、分割部１２ｄに分割数を増加させる（例えば現在の２倍にする）指示が発行され（ステップＳ３６）、処理がステップＳ３１に移行する。なお、ｍは、平均レスポンスの誤差範囲値であり、例えば５０ｍｓに設定することができる。

【００７７】

一方、ｒｅｓｐ＿ｎｅｗ＞ｒｅｓｐ＿ｏｒｇ＋ｍが成立しない場合（ステップＳ３５のＮｏルート）、分割数判定部１１ｅにより、ｒｅｓｐ＿ｎｅｗ＜ｒｅｓｐ＿ｏｒｇ＋ｍが成立するか否かが判定される（ステップＳ３７）。ｒｅｓｐ＿ｎｅｗ＜ｒｅｓｐ＿ｏｒｇ＋ｍが成立する場合（ステップＳ３７のＹｅｓルート）、分割数判定部１１ｅにより、分割部１２ｄに分割数を減少させる（例えば現在の１／２倍にする）指示が発行され（ステップＳ３８）、処理がステップＳ３１に移行する。なお、ｒｅｓｐ＿ｎｅｗ＜ｒｅｓｐ＿ｏｒｇ＋ｍが成立しない場合（ステップＳ３７のＮｏルート）、ｒｅｓｐ＿ｎｅｗは平均レスポンスの期待値の範囲内にあるため、分割数の更新は行なわれず、処理がステップＳ３１に移行する。

【００７８】

このように、分割数判定部１１ｅが、セグメント移動前及び移動中の平均レスポンスに基づいて、セグメント移動による平均レスポンスの悪化を抑制するように分割数を決定することができる。従って、階層ドライバ１２は、移動中のセグメントを動的に最適な分割数に分割することができるため、移動時間を低減させつつ、対象データに対するユーザＩＯへのレスポンス悪化を抑制することができる。

【００７９】

次に、図１４及び図１５を参照して階層ドライバ１２の動作を説明する。

はじめに、移動指示を受け取ったときの階層ドライバ１２の動作を説明する。図１４は、階層ドライバ１２による転送指示通知処理の動作例を示すフローチャートである。

図１４に示すように、階層ドライバ１２により、移動指示部１１ｄからの移動指示が待ち受けられ（ステップＳ４１）、移動指示を受け取ると、ＨＤＤ３０からＳＳＤ２０へのデータの移動であるか否かが判定される（ステップＳ４２）。

【００８０】

ＨＤＤ３０からＳＳＤ２０へのデータの移動である場合（ステップＳ４２のＹｅｓルート）、階層ドライバ１２により、移動を指示されたセグメントがＳＳＤ２０へ移動済みであるか否かが判定される（ステップＳ４３）。移動済みである場合（ステップＳ４３のＹｅｓルート）、処理がステップＳ４１に移行する。

一方、移動済みでない場合（ステップＳ４３のＮｏルート）、階層ドライバ１２により、階層テーブル１２ｃ内のＨＤＤオフセットより“NULL”となっているエントリが探索さ

れ、ＨＤＤオフセット情報と状態とが登録される。このとき階層ドライバ１２が登録する状態は、“Moving（HDD→SSD）”である。そして、階層ドライバ１２により、ＨＤＤ３０からＳＳＤ２０へのデータの転送指示が分割部１２ｄに発行され（ステップＳ４４）、処理がステップＳ４１に移行する。

【００８１】

また、ＨＤＤ３０からＳＳＤ２０へのデータの移動でない場合（ステップＳ４２のＮｏルート）、階層ドライバ１２により、階層テーブル１２ｃ内のＨＤＤオフセットよりセグメントが探索され、ＨＤＤオフセット情報と状態とが登録される。このとき階層ドライバ１２が登録する状態は、“Moving（SSD→HDD）”である。そして、階層ドライバ１２により、ＳＳＤ２０からＨＤＤ３０へのデータの転送指示が分割部１２ｄに発行され（ステップＳ４５）、処理がステップＳ４１に移行する。

【００８２】

次に、転送指示後に転送完了通知を受け取ったときの階層ドライバ１２の動作を説明する。図１５は、階層ドライバ１２による転送完了受信処理の動作例を示すフローチャートである。

図１５に示すように、階層ドライバ１２により、分割部１２ｄからの転送完了通知が待ち受けられる（ステップＳ５１）。転送完了通知を受け取ると、階層ドライバ１２により、転送が完了した階層テーブル１２ｃのエントリがＨＤＤオフセットを用いて探索され、状態が“Moving（HDD→SSD）”の場合は状態が“allocated”に変更される。一方、階層テーブル１２ｃにより、状態が“Moving（SSD→HDD）”の場合は状態が“free”に変更され、且つ、対応するHDDオフセットが“NULL”に設定され（ステップＳ５２）、処理がステップＳ５１に移行する。

【００８３】

このように、階層ドライバ１２が階層テーブル１２ｃを用いてＳＳＤ２０とＨＤＤ３０との間でデータを転送することにより、ＩＯが集中するセグメントのデータをＳＳＤ２０に置くことができる。

次に、図１６及び図１７を参照して分割部１２ｄの動作を説明する。

はじめに、転送指示を受け取ったときの分割部１２ｄの動作を説明する。図１６は、分割部１２ｄによる転送指示受信処理の動作例を示すフローチャートである。

【００８４】

図１６に示すように、分割部１２ｄにより、階層ドライバ１２からのＳＳＤ２０－ＨＤＤ３０間の転送指示が待ち受けられる（ステップＳ６１）。転送指示を受け取ると、分割部１２ｄにより、転送指示で指定された移動する各セグメントが分割数ｍｍで分割され、分割単位でｋｃｏｐｙｄに転送指示が発行される（ステップＳ６２）。

全データの転送が終わると、分割部１２ｄにより、階層ドライバ１２にデータの転送完了が通知され（ステップＳ６３）、処理がステップＳ６１に移行する。

【００８５】

次に、分割数更新指示を受け取ったときの分割部１２ｄの動作を説明する。図１７は、分割部１２ｄによる分割数更新処理の動作例を示すフローチャートである。

図１７に示すように、分割部１２ｄにより、分割数判定部１１ｅからの分割数更新指示が待ち受けられる（ステップＳ７１）。分割数更新指示を受け取ると、分割部１２ｄにより、当該指示に応じて分割数ｍｍが更新され（ステップＳ７２）、処理がステップＳ７１に移行する。

【００８６】

このように、分割部１２ｄが階層ドライバ１２からのセグメント単位の転送指示を、さらに小さい移動単位に分割することによって、ユーザＩＯのレスポンス悪化を抑制することができる。また、分割部１２ｄが分割数判定部１１ｅからの分割数更新指示に応じて分割数ｍｍを適宜更新することができるため、ワークロードの変化に柔軟に対応することができる。

【００８７】

次に、図１８を参照してＩＯマップ部１２ａの動作を説明する。図１８はＩＯマップ部１２ａによるＩＯ受信処理の動作例を示すフローチャートである。

図１８に示すように、ＩＯマップ部１２ａにより、ユーザＩＯの受信が待ち受けられる（ステップＳ８１）。ユーザＩＯを受け取ると、ＩＯマップ部１２ａにより、ユーザＩＯで指定されるオフセットと階層テーブル１２ｃに登録されている各オフセット＋セグメントサイズとが比較され（ステップＳ８２）。

【００８８】

そして、ＩＯマップ部１２ａにより、比較の結果、階層テーブル１２ｃに一致するオフセットが存在し、且つ状態が“allocated”であるか否かが判定される（ステップＳ８３）。一致するオフセットが存在し、且つ状態が“allocated”である場合（ステップＳ８３のＹｅｓルート）、ＩＯマップ部１２ａにより、ＳＳＤドライバ１３へＩＯ要求が送付され（ステップＳ８４）、処理がステップＳ８１に移行する。

【００８９】

一方、一致するオフセットが存在せず、又は状態が“allocated”ではない場合（ステップＳ８３のＮｏルート）、ＩＯマップ部１２ａにより、状態が“Moving(HDD→SSD)”又は“Moving(SSD→HDD)”であるか否かが判定される（ステップＳ８５）。状態が“Moving(HDD→SSD)”又は“Moving(SSD→HDD)”ではない場合（ステップＳ８５のＮｏルート）、ＩＯマップ部１２ａにより、ＨＤＤドライバ１４へＩＯ要求が送付され（ステップＳ８６）、処理がステップＳ８１に移行する。

【００９０】

また、状態が“Moving(HDD→SSD)”又は“Moving(SSD→HDD)”である場合（ステップＳ８５のＹｅｓルート）、ＩＯマップ部１２ａにより、当該状態が“free”又は“allocated”に変化するまでＩＯ要求がペンディングキュー１２ｂに格納される。すなわち、ＩＯマップ部１２ａにより、ＩＯ要求に係るセグメントの階層移動が完了するまで、ＩＯ要求が保留にされる（ステップＳ８７）。階層移動が完了すると、ＩＯマップ部１２ａによりペンディングキュー１２ｂに格納されたＩＯ要求が取り出され、処理がステップＳ８３に移行する。

【００９１】

〔１－５〕ハードウェア構成例

次に、図１９を参照して、図７に示す階層ストレージ制御装置１０のハードウェア構成について説明する。図１９は、図７に示す階層ストレージ制御装置１０のハードウェア構成例を示す図である。

階層ストレージ制御装置１０は、図１９に示すように、ＣＰＵ（Central Processing Unit）１０ａ，メモリ１０ｂ，記憶部１０ｃ，インタフェース部１０ｄ，入出力部１０ｅ，記録媒体１０ｆ，及び読取部１０ｇをそなえることができる。

【００９２】

ＣＰＵ１０ａは、対応する各ブロック１０ｂ～１０ｇと接続され、種々の制御や演算を行なう演算処理装置（プロセッサ）である。ＣＰＵ１０ａは、メモリ１０ｂ，記憶部１０ｃ，記録媒体１０ｆや１０ｈ，又は図示しないＲＯＭ（Read Only Memory）等に格納されたプログラムを実行することにより、階層ストレージ制御装置１０における種々の機能を実現することができる。

【００９３】

メモリ１０ｂは、種々のデータやプログラムを格納する記憶装置である。ＣＰＵ１０ａは、プログラムを実行する際に、メモリ１０ｂにデータやプログラムを格納し展開する。なお、メモリ１０ｂとしては、例えばＲＡＭ（Random Access Memory）等の揮発性メモリが挙げられる。

記憶部１０ｃは、種々のデータやプログラム等を格納するハードウェアである。記憶部１０ｃとしては、例えばＨＤＤ等の磁気ディスク装置，ＳＳＤ等の半導体ドライブ装置，フラッシュメモリ等の不揮発性メモリ等の各種デバイスが挙げられる。なお、記憶部１０ｃとして複数のデバイスが用いられてもよく、これらのデバイスでＲＡＩＤ（Redundant

Arrays of Inexpensive Disks）が構成されてもよい。また、記憶部１０ｃは、図７に示すＳＳＤ２０及びＨＤＤ３０を含んでもよい。

【００９４】

インタフェース部１０ｄは、有線又は無線による、ネットワーク（図示省略）や他の情報処理装置との間の接続及び通信の制御等を行なうものである。インタフェース部１０ｄとしては、例えば、ＬＡＮ（Local Area Network），ファイバチャネル（Fibre Channel；ＦＣ），インフィニバンド（InfiniBand）等に準拠したアダプタが挙げられる。

入出力部１０ｅは、マウスやキーボード等の入力装置及びディスプレイやプリンタ等の出力装置の少なくとも一方を含むことができる。例えば、入出力部１０ｅは、階層ストレージ制御装置１０の使用者又は管理者等による種々の作業に用いられる。

【００９５】

記録媒体１０ｆは、例えばフラッシュメモリやＲＯＭ等の記憶装置であり、種々のデータやプログラムを記録することができる。読取部１０ｇは、コンピュータ読取可能な記録媒体１０ｈに記録されたデータやプログラムを読み出す装置である。記録媒体１０ｆ及び１０ｈの少なくとも一方には、本実施形態に係る階層ストレージ制御装置１０の各種機能の全部もしくは一部を実現する制御プログラムが格納されてもよい。例えば、ＣＰＵ１０ａは、記録媒体１０ｆから読み出したプログラム、又は、読取部１０ｇを介して記録媒体１０ｈから読み出したプログラムを、メモリ１０ｂ等の記憶装置に展開して実行することができる。これにより、コンピュータ（ＣＰＵ１０ａ，情報処理装置，各種端末を含む）は、上述した階層ストレージ制御装置１０の機能を実現することができる。

【００９６】

なお、記録媒体１０ｈとしては、例えばフレキシブルディスク、ＣＤ（Compact Disc）、ＤＶＤ（Digital Versatile Disc）、ブルーレイディスク等の光ディスクや、ＵＳＢ（Universal Serial Bus）メモリやＳＤカード等のフラッシュメモリが挙げられる。なお、ＣＤとしては、ＣＤ－ＲＯＭ、ＣＤ－Ｒ（CD-Recordable）、ＣＤ－ＲＷ（CD-Rewritable）等が挙げられる。また、ＤＶＤとしては、ＤＶＤ－ＲＯＭ、ＤＶＤ－ＲＡＭ、ＤＶＤ－Ｒ、ＤＶＤ－ＲＷ、ＤＶＤ＋Ｒ、ＤＶＤ＋ＲＷ等が挙げられる。

【００９７】

なお、上述した各ブロック１０ａ～１０ｇ間はバスで相互に通信可能に接続される。例えばＣＰＵ１０ａと記憶部１０ｃとの間は、ディスクインタフェースを介して接続される。また、階層ストレージ制御装置１０の上述したハードウェア構成は例示である。従って、階層ストレージ制御装置１０内でのハードウェアの増減（例えば任意のブロックの追加や省略），分割，任意の組み合わせでの統合，バスの追加又は省略等は適宜行なわれてもよい。

【００９８】

〔１－６〕適用例

上述のように、階層ストレージ制御装置１０は、リアルタイムに測定した負荷に基づいて高負荷領域のデータをＳＳＤ２０に移動する動的階層制御に用いて好適である。

ここで、階層ストレージ制御装置１０は、高負荷領域の近傍のデータをＳＳＤ２０に移動するために、近傍として適切な領域を選択する機能をさらにそなえてもよい。すなわち、階層ストレージ制御装置１０を、以下に詳述する階層ストレージ制御装置１０Ａ（図２２参照）に適用してもよい。

【００９９】

まず、本適用例に係る階層ストレージ制御装置１０Ａによる動的階層制御について説明する。図２０及び図２１は、本適用例に係る階層ストレージ制御装置１０Ａによる動的階層制御を説明するための図である。

図２０は、本適用例に係る階層ストレージシステム１Ａ（図２２参照）のワークロードの分析例を示す図であり、縦軸は下に向かってオフセットを示し、横軸は経過時間を示す。図２０において、網掛けの領域１が高負荷領域を示す。階層ストレージ制御装置１０Ａは、図の矢印２が示すように、高負荷領域からある決めた範囲を拡張領域とする。

【０１００】
そして、階層ストレージ制御装置１０Ａは、拡張領域と、拡張領域とつながる別の拡張領域とを合わせて１つの拡張領域とみなす。そして、階層ストレージ制御装置１０Ａは、拡張領域をＳＳＤ２０にデータを移動する移動領域とする。図２０では、上下の破線の間の領域が移動領域である。
また、階層ストレージ制御装置１０Ａは、ある時点で移動領域を決定すると、一定時間高負荷が発生しなくなるまで、移動領域を維持する。すなわち、階層ストレージ制御装置１０Ａは、高負荷領域が消滅し、一定時間高負荷が発生しないと高負荷は消滅したとみなす。図２０では、タイムアウトの矢印が高負荷が発生しない一定時間を示す。
【０１０１】
図２１は、階層ストレージシステム１Ａのワークロードの他の分析例を示す図であり、縦軸は上に向かってオフセットを示し、横軸は経過時間を示す。また、ボリュームは１ＧＢ単位のセグメントに分割され、経過時間は１分間を単位としている。すなわち、図２１において、網掛けの正方形領域３は、１つのセグメントが１分間高負荷であったことを示す。また、ｓは高負荷領域から拡張領域として拡張するセグメント数を示し、図２１ではｓ＝１である。
【０１０２】
そして、階層ストレージ制御装置１０Ａは、高負荷領域のセグメント間で距離がｓ以内のものを繋ぎ合わせてｎ＿セグメントを作成する。ｎ＿セグメントは、ＳＳＤ２０にデータが移動される移動領域であり、データの移動について一体制御される。ｎ＿セグメントのセグメント数は２ｓ＋１以上となる。図２１では、セグメント数が５の２つのｎ＿セグメントが特定されている。
【０１０３】
このように、階層ストレージ制御装置１０Ａは、ＳＳＤ２０にデータを移動する領域としてｎ＿セグメントを特定することにより、高負荷領域の近傍として適切な領域を選択することができる。
次に、適用例に係る階層ストレージ制御装置１０Ａの機能構成について説明する。図２２は、適用例に係る階層ストレージシステム１Ａの構成例を示す図である。図２２に示すように、階層ストレージ制御装置１０Ａは、階層管理部１１Ａ，階層ドライバ１２，ＳＳＤドライバ１３，及びＨＤＤドライバ１４をそなえることができる。なお、以下の説明において、階層ストレージ制御装置１０と同様の機能については、重複した説明を省略する。例えば、階層ドライバ１２，ＳＳＤドライバ１３，及びＨＤＤドライバ１４は、図７に示す階層ストレージ制御装置１０の構成と略同一である。また、図の簡略化のため、図２２において階層ドライバ１２がそなえる機能ブロックの図示を省略している。
【０１０４】
以下、図２２に示す階層ストレージ制御装置１０Ａのうち、主に階層管理部１１Ａの機能及び動作について、図２３～図２８を参照してフローチャートに沿って説明する。
階層管理部１１Ａは、ＨＤＤ３０についてトレースされたＩＯの情報に基づいて、ＳＳＤ２０にデータを移動するｎ＿セグメントを決定し、決定したｎ＿セグメントのデータの移動を階層ドライバ１２に指示する。図２２に示すように、階層管理部１１Ａは、データ収集部１５ａ，データベース１５ｂ，ワークロード分析部１５ｃ，移動指示部１５ｄ，及び分割数判定部１１ｅをそなえる。なお、分割数判定部１１ｅは、図７に示す階層ストレージ制御装置１０の構成と略同一である。
【０１０５】
はじめに、データ収集部１５ａの処理手順について説明する。図２３は、適用例に係るデータ収集部１５ａによるデータ収集処理の動作例を示すフローチャートであり、図２４は、図２２に示すデータベース１５ｂの一例を示す図である。なお、データ収集部１５ａは、ｂｌｋｔｒａｃｅコマンドを６０秒間実行して、終了したことを条件として起動される。
【０１０６】

図２３に示すように、データ収集部１５ａは、ｂｌｋｔｒａｃｅコマンドの実行により得られたトレース結果を取り出し、１ＧＢオフセット単位すなわちセグメント単位で各セグメントのＩＯ数を抽出する（ステップＳ１０１）。

そして、データ収集部１５ａは、セグメントごとにＩＯ数が閾値ｐを上回るかどうかを判定し、ｐを上回ったセグメントの抽出を行なう（ステップＳ１０２）。ＩＯ数が閾値ｐを上回ったセグメントは高負荷領域である。

【０１０７】

そして、データ収集部１５ａは、抽出したセグメントに関して、隣接間距離がｓ以内となったセグメントを繋ぎ合わせていく（ステップＳ１０３）。そして、データ収集部１５ａは、繋ぎ合わせたセグメントとその外側のｓまでの範囲のセグメントをｎ＿セグメントと定義し、抽出順にｎ＿セグメント番号を採番する（ステップＳ１０４）。

そして、データ収集部１５ａは、ｎ＿セグメントごとにｎ＿セグメント番号，セグメント範囲，ＩＯ数，平均レスポンスをタイムスタンプと共にデータベース１５ｂに書き込む（ステップＳ１０５）。

【０１０８】

ここで、データベース１５ｂは、データ収集部１１１により特定されたｎ＿セグメントに関する情報を記憶する。図２４に示すように、データベース１５ｂは、ｎ＿セグメントごとに、ｎ＿セグメント番号，セグメント範囲，ＩＯ数，平均レスポンス，及びタイムスタンプを対応させて記憶する。例えば、ｎ＿セグメント番号が“１”であるｎ＿セグメントは、先頭セグメントのオフセットが“３”であり、最終セグメントのオフセットが“５”であり、平均レスポンスが“0.6”（秒）であり、ＩＯ数が“1000”であり、タイムスタンプが“１”である。

【０１０９】

図２３の説明に戻り、データ収集部１５ａは、図７に示すデータ収集部１１ａと同様に、全セグメントを対象にした合計ＩＯ数，平均レスポンスを集計し、タイムスタンプとともにデータベース１５ｂに格納する（ステップＳ１０６）。なお、ステップＳ１０６でデータベース１５ｂに格納される情報は、図２４におけるｎ＿セグメント番号“all”のエントリに対応する。

【０１１０】

以上により、データ収集部１５ａの処理が終了する。

このように、データ収集部１５ａは、高負荷のセグメントに関して隣接間距離がｓ以内となったセグメントを繋ぎ合わせてｎ＿セグメントを抽出することにより、高負荷のセグメントの近傍を適切に選択することができる。

次に、ワークロード分析部１５ｃの処理手順について説明する。図２５は、適用例に係るワークロード分析部１５ｃによる移動判定処理の動作例を示すフローチャートであり、図２６及び図２７は、それぞれ図２２に示す候補テーブル１５１及び管理テーブル１５２の一例を示す図である。

【０１１１】

図２５に示すように、ワークロード分析部１５ｃは、データベース１５ｂから直近のタイムスタンプのｎ＿セグメントについてＩＯ数を取り出し（ステップＳ１１１）、ＩＯ数が多い順にｎ＿セグメントを並べ替える（ステップＳ１１２）。

そして、ワークロード分析部１５ｃは、各ｎ＿セグメントのＩＯ数を合計することでｉｏ＿ａｌｌを求める（ステップＳ１１３）。そして、ワークロード分析部１５ｃは、以下の式（１）の計算をｍがｍａｘ＿ｓｅｇ＿ｎｕｍに到達するか、ｉｏ＿ｒａｔｅがｉｏ＿ｒａｔｅ＿ｖａｌｕｅを超えるまで行なう（ステップＳ１１４）。

【０１１２】

【数１】

$$io\_concentration = \sum_{k=1}^{m} seg\_sort(k)$$
$$io\_rate = (io\_concentration * 100)/io\_all \quad \cdots(1)$$

【０１１３】

ここで、ｍａｘ＿ｓｅｇ＿ｎｕｍは、同時にＳＳＤ２０へデータの移動を行なうｎ＿セグメント数である。また、ｓｅｇ＿ｓｏｒｔ（ｋ）は、ｋ番目にアクセス数が多いｎ＿セグメントのＩＯ数である。ｉｏ＿ｃｏｎｃｅｎｔｒａｔｉｏｎは、トップｋ個のｎ＿セグメントのＩＯ数の合計であり、この数が多いほどトップｋ個のｎ＿セグメントにアクセスが集中していることを示す。また、ｉｏ＿ａｌｌは、ＩＯ数を全ｎ＿セグメントについて合計した総数であり、ｉｏ＿ｒａｔｅは、トップｋ個のｎ＿セグメントのＩＯ数の合計の総数に対する割合を％表示したものである。従って、ｉｏ＿ｒａｔｅの値が大きいほどトップｋ個のｎ＿セグメントへのアクセスの集中率が高いことを示す。

【０１１４】

ｉｏ＿ｒａｔｅ＿ｖａｌｕｅは、ＳＳＤ２０へデータを移動する候補としてトップｋ個のｎ＿セグメントを選択するか否かの閾値である。

そして、ｉｏ＿ｒａｔｅがｉｏ＿ｒａｔｅ＿ｖａｌｕｅを超えた場合には、ワークロード分析部１５ｃは、以下のステップＳ１１５～ステップＳ１２２を行ない、ｍがｍａｘ＿ｓｅｇ＿ｎｕｍに到達した場合には、ステップＳ１２３に移動する。すなわち、ｉｏ＿ｒａｔｅがｉｏ＿ｒａｔｅ＿ｖａｌｕｅを超えた場合には、ワークロード分析部１５ｃは、対応するｎ＿セグメント番号が連続して何回このトップｋに入ったのかを候補テーブル１５１に記録する（ステップＳ１１５）。

【０１１５】

ここで、候補テーブル１５１は、ワークロード分析部１５ｃがそなえるテーブルであり、ＳＳＤ２０へデータを移動する候補を記憶する。図２６に示すように、候補テーブル１５１は、ｎ＿セグメントごとに、ｎ＿セグメント番号，先頭セグメント番号，セグメント数，及び連続数を対応させて記憶する。ここで、先頭セグメント番号は、ｎ＿セグメントの先頭セグメントのオフセットである。セグメント数は、ｎ＿セグメントに含まれるセグメントの個数である。連続数は、候補として連続して候補テーブル１５１に登録された回数を示す。

【０１１６】

図２５の説明に戻り、ワークロード分析部１５ｃは、前タイムスライスでトップｋに入ったｎ＿セグメントで今回トップｋから外れたｎ＿セグメントは、連続数をリセットする（ステップＳ１１６）。

そして、ワークロード分析部１５ｃは、連続数が所定の閾値t_1を超えたｎ＿セグメントを移動候補として抽出し、抽出したｎ＿セグメントに含まれるセグメント数をｎとし、ｎ＿セグメントのデータの移動時間Ｔｉｅｒｉｎｇ＿ｔｉｍｅを計算する（ステップＳ１１７）。

【０１１７】

ここで、Ｔｉｅｒｉｎｇ＿ｔｉｍｅ＝ｓｅｇ＿ｍｏｖｅ＿ｔｉｍｅ×ｎ＋検出遅延であり、ｓｅｇ＿ｍｏｖｅ＿ｔｉｍｅは、１セグメントのデータをＨＤＤ３０からＳＳＤ２０へ移動するのにかかる時間である。また、検出遅延は、移動候補の検出にかかる時間であり、ここではデータの収集間隔の６０秒とする。

そして、ワークロード分析部１５ｃは、Ｔｉｅｒｉｎｇ＿ｔｉｍｅとＩＯの集中率が高い状態が続くと期待される時間Ｌｉｆｅ＿ｅｘ＿ｔｉｍｅ（平均余命時間）とを比較する（ステップＳ１１８）。Ｔｉｅｒｉｎｇ＿ｔｉｍｅがＬｉｆｅ＿ｅｘ＿ｔｉｍｅ以上の場合（ステップＳ１１８のＮｏルート）、処理がステップＳ１２１に移行する。一方、Ｔｉｅｒｉｎｇ＿ｔｉｍｅがＬｉｆｅ＿ｅｘ＿ｔｉｍｅより小さい場合（ステップＳ１１８のＹｅｓルート）、ワークロード分析部１５ｃは、移動候補ｎ＿セグメントの情報を移動指

示部１５ｄへ通知し、移動候補ｎ＿セグメントのデータのＨＤＤ３０からＳＳＤ２０への移動を指示する（ステップＳ１１９）。また、ワークロード分析部１５ｃは、ＳＳＤ２０へのデータの移動を指示したｎ＿セグメントの情報を管理テーブル１５２に記録する（ステップＳ１２０）。

【０１１８】

ここで、管理テーブル１５２は、ワークロード分析部１５ｃがそなえるテーブルであり、ＳＳＤ２０へデータを移動する対象として選択したｎ＿セグメントを記憶する。図２７に示すように、管理テーブル１５２は、ｎ＿セグメントごとに、ｎ＿セグメント番号，先頭セグメント番号，セグメント数，及び連続数を対応させて記憶する。ここで、連続数は、トップｋ個の候補が選択された場合に、連続して候補として選択されなかった回数を示す。

【０１１９】

図２５の説明に戻り、ワークロード分析部１５ｃは、トップｋに入ったｎ＿セグメント番号と管理テーブル１５２に登録されているｎ＿セグメント番号の突合せを行なう。また、ワークロード分析部１５ｃは、管理テーブル１５２に登録されているｎ＿セグメントごとにトップｋに入らなかったｎ＿セグメント番号の連続数を“+1”し、トップｋに入っていたら連続数を“0”にリセットする（ステップＳ１２１）。

【０１２０】

次いで、ワークロード分析部１５ｃは、管理テーブル１５２に登録されているｎ＿セグメント番号ごとに連続数が所定の閾値t_2を超えているか否かの判断を行なう。連続数が所定の閾値t_2を超えている場合、ワークロード分析部１５ｃは、ｎ＿セグメント番号を移動指示部１５ｄに通知してＳＳＤ２０からＨＤＤ３０へのデータの移動を指示する。また、ワークロード分析部１５ｃは、管理テーブル１５２に登録されているｎ＿セグメントの情報を削除する（ステップＳ１２２）。そして、ワークロード分析部１５ｃは、６０秒スリープし（ステップＳ１２３）、処理がステップＳ１１１に移行する。

【０１２１】

このように、ワークロード分析部１５ｃがＩＯの集中度が高いｎ＿セグメントのデータをＨＤＤ３０からＳＳＤ２０に移動するように移動指示部１５ｄに指示することによって、ユーザはＨＤＤ３０のデータに高速にアクセスすることができる。

以上のように、ワークロード分析部１５ｃは、データ収集部１５ａにより集計された入出力数が第１の閾値より大きな単位領域と所定の距離内にある単位領域を繋ぎ合わせた拡張領域と、当該拡張領域と繋がる他の拡張領域と、を合わせた移動領域を特定する特定部の一例であるといえる。

【０１２２】

次に、移動指示部１５ｄの処理手順について説明する。図２８は、適用例に係る移動指示部１５ｄによる移動指示通知処理の動作例を示すフローチャートである。

図２８に示すように、移動指示部１５ｄは、ワークロード分析部１５ｃからの移動指示を待つ（ステップＳ１３１）。移動指示があると、移動指示部１５ｄは、ｎ＿セグメント番号に属する各セグメントのボリューム上のオフセットをＨＤＤ３０上のオフセットに変換する（ステップＳ１３２）。

【０１２３】

そして、移動指示部１５ｄは、セグメントごとに、セグメント番号に対応するＨＤＤ３０上のオフセットと、データの移動方向を階層ドライバ１２に通知する（ステップＳ１３３）。ここで、データの移動方向は、ＨＤＤ３０からＳＳＤ２０か、又は、ＳＳＤ２０からＨＤＤ３０である。

また、移動指示部１５ｄは、移動指示を出したセグメント数と移動判断を行なったデータのタイムスタンプ（直近のタイムスタンプ）とを分割数判定部１１ｅに通知し（ステップＳ１３４）、処理がステップＳ１３１に移行する。

【０１２４】

このように、移動指示部１５ｄが各セグメントのボリューム上のオフセットをＨＤＤ３

０上のオフセットに変換することによって、階層ドライバ１２はＳＳＤ２０とＨＤＤ３０との間でデータを移動することができる。

以上のように、本適用例に係る階層ストレージ制御装置１０Ａによれば、データベース１５ｂが、ＩＯ数が閾値ｐを上回るセグメントに関して、隣接間距離ｓ以内のセグメントを繋ぎ合わせる。そして、データ収集部１５ａは、繋ぎ合わせたセグメントとその外側のｓまでの範囲をｎ＿セグメントとして抽出する。また、ワークロード分析部１５ｃが、ｎ＿セグメントを単位として、ＨＤＤ３０からＳＳＤ２０にデータを移動する対象を決定する。

【０１２５】

このとき、本適用例に係る分割数判定部１１ｅ及び分割部１２ｄは、移動単位のｎ＿セグメントに属する複数のセグメントの各々について、動的に決定した分割数に分割して階層移動を行なうことができる。例えば、分割部１２ｄは、ｎ＿セグメントの移動領域に記憶されたデータをＳＳＤ２０に移動する移動処理において、移動領域に含まれる複数の単位領域の各々を所定の分割数により複数の分割領域に分割する。そして、分割部１２ｄは、移動領域に記憶されたデータを分割領域の単位でＳＳＤ２０に移動する。

【０１２６】

従って、階層ストレージシステム１Ａは、高負荷領域の近傍を適切に選択した上で、使用する機器の性能やワークロードに応じた最適な移動単位で、ＨＤＤ３０からＳＳＤ２０にデータを移動することができ、ＨＤＤ３０へのアクセスを高速化することができる。

〔２〕その他

以上、本発明の好ましい実施形態について詳述したが、本発明は、係る特定の実施形態に限定されるものではなく、本発明の趣旨を逸脱しない範囲内において、種々の変形、変更して実施することができる。

【０１２７】

例えば、一実施形態において、ＳＳＤ２０及びＨＤＤ３０を用いた階層ストレージシステム１及び１Ａについて説明したが、これに限定されるものではなく、例えばキャッシュメモリと主記憶装置とを用いた階層記憶システムにも同様に適用することができる。すなわち、本発明は、不揮発性記憶装置の階層記憶システムだけでなく、揮発性記憶装置を含む階層記憶システムにも同様に適用することができる。

【０１２８】

また、一実施形態に係る階層ストレージシステム１及び１Ａは、ＳＳＤ２０及びＨＤＤ３０の他に、速度差のある記憶装置にも適用することが可能である。例えばＨＤＤと、ＨＤＤよりも大容量だが低速なテープドライブ等の磁気記録装置とを用いた階層ストレージシステム等にも適用することが可能である。

さらに、一実施形態において、階層ストレージ制御装置１０及び１０Ａの動作を１つのＳＳＤ２０及び１つのＨＤＤ３０に着目して説明したが、複数のＳＳＤ２０及び複数のＨＤＤ３０が階層ストレージシステム１及び１Ａにそなえられる場合も同様である。

【０１２９】

〔３〕付記

以上の実施形態に関し、更に以下の付記を開示する。

（付記１）

第１の記憶装置の記憶領域を所定の大きさで分割した複数の単位領域について、入力された要求に対する応答性能を監視する監視部と、

前記第１の記憶装置の移動対象の単位領域に記憶されたデータを前記第１の記憶装置とは異なる性能の第２の記憶装置に移動する移動処理において、前記移動対象の単位領域を所定の分割数により複数の分割領域に分割し、前記データを前記分割領域の単位で前記第２の記憶装置に移動する分割部と、

前記監視部が監視した前記移動処理実行中の第１の応答性能に基づいて、前記所定の分割数を変更する変更部と、

をそなえることを特徴とする、ストレージ制御装置。

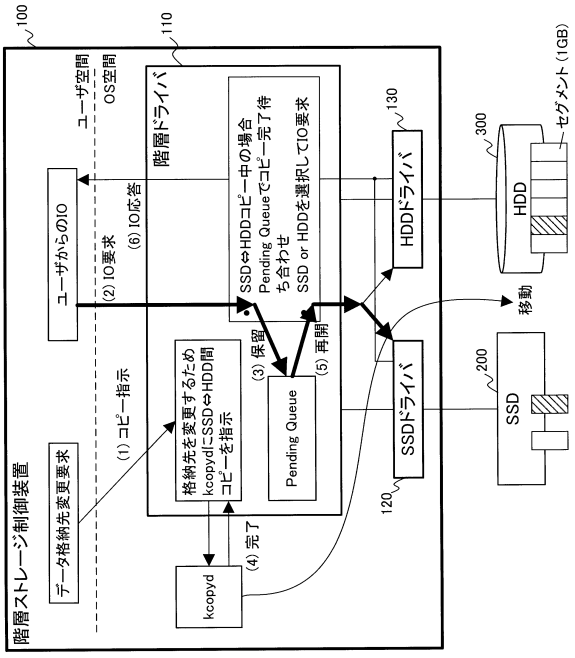
【０１３０】
（付記２）
前記変更部は、前記第１の応答性能と、前記移動処理実行前の第２の応答性能とに基づいて、前記所定の分割数を変更する
ことを特徴とする、付記１記載のストレージ制御装置。
（付記３）
前記所定の分割数は、前記移動処理実行前の第２の応答性能を基に求められる
ことを特徴とする、付記２記載のストレージ制御装置。
【０１３１】
（付記４）
前記変更部は、前記第１の応答性能と前記第２の応答性能とを比較し、前記第１の応答性能が前記第２の応答性能よりも劣化していると判断した場合に前記所定の分割数を増加させる一方、前記第１の応答性能が前記第２の応答性能よりも優れていると判断した場合に前記所定の分割数を減少させる、
ことを特徴とする、付記１～３のいずれか１項記載のストレージ制御装置。
【０１３２】
（付記５）
前記変更部は、前記監視部が前記移動処理実行中の複数の時点で監視した複数の第１の応答性能を取得し、前記第２の応答性能と前記複数の応答性能の平均値とを比較して、前記平均値が前記第２の応答性能よりも劣化しているか否かに応じて前記所定の分割数を増減させる、
ことを特徴とする、付記４記載のストレージ制御装置。
【０１３３】
（付記６）
前記監視部は、前記複数の単位領域について単位領域ごとに入出力数を集計し、
前記ストレージ制御装置は、
前記監視部により集計された入出力数が第１の閾値より大きな単位領域と所定の距離内にある単位領域を繋ぎ合わせた拡張領域と、該拡張領域と繋がる他の拡張領域と、を合わせた移動領域を特定する特定部、をさらにそなえ、
前記分割部は、前記移動領域に記憶されたデータを前記第２の記憶装置に移動する移動処理において、前記移動領域に含まれる複数の単位領域の各々を所定の分割数により複数の分割領域に分割し、前記移動領域に記憶されたデータを前記分割領域の単位で前記第２の記憶装置に移動する、
ことを特徴とする、付記１～５のいずれか１項記載のストレージ制御装置。
【０１３４】
（付記７）
第１の記憶装置及び第２の記憶装置の制御を行なうコンピュータに、
前記第１の記憶装置の記憶領域を所定の大きさで分割した複数の単位領域について、入力された要求に対する応答性能を監視し、
前記第１の記憶装置の移動対象の単位領域に記憶されたデータを前記第１の記憶装置とは異なる性能の第２の記憶装置に移動する移動処理を行ない、
前記移動処理において、前記移動対象の単位領域を所定の分割数により複数の分割領域に分割し、前記データを前記分割領域の単位で前記第２の記憶装置に移動し、
前記監視により監視した前記移動処理実行中の第１の応答性能に基づいて、前記所定の分割数を変更する、
処理を実行させることを特徴とする、制御プログラム。
【０１３５】
（付記８）
前記変更する処理において、前記第１の応答性能と、前記移動処理実行前の第２の応答性能とに基づいて、前記所定の分割数を変更する、

ことを特徴とする、付記７記載の制御プログラム。
（付記９）
前記所定の分割数は、前記移動処理実行前の第２の応答性能を基に求められる、
ことを特徴とする、付記８記載の制御プログラム。
【０１３６】
（付記１０）
前記変更する処理において、前記第１の応答性能と前記第２の応答性能とを比較し、前記第１の応答性能が前記第２の応答性能よりも劣化していると判断した場合に前記所定の分割数を増加させる一方、前記第１の応答性能が前記第２の応答性能よりも優れていると判断した場合に前記所定の分割数を減少させる、
ことを特徴とする、付記７～９のいずれか１項記載の制御プログラム。
【０１３７】
（付記１１）
前記変更する処理において、前記監視により前記移動処理実行中の複数の時点で監視した複数の第１の応答性能を取得し、前記第２の応答性能と前記複数の応答性能の平均値とを比較して、前記平均値が前記第２の応答性能よりも劣化しているか否かに応じて前記所定の分割数を増減させる、
ことを特徴とする、付記１０記載の制御プログラム。
【０１３８】
（付記１２）
前記コンピュータに、
前記複数の単位領域について単位領域ごとに入出力数を集計し、
前記集計された入出力数が第１の閾値より大きな単位領域と所定の距離内にある単位領域を繋ぎ合わせた拡張領域と、該拡張領域と繋がる他の拡張領域と、を合わせた移動領域を特定し、
前記移動領域に記憶されたデータを前記第２の記憶装置に移動する移動処理において、前記移動領域に含まれる複数の単位領域の各々を所定の分割数により複数の分割領域に分割し、前記移動領域に記憶されたデータを前記分割領域の単位で前記第２の記憶装置に移動する、
処理を実行させることを特徴とする、付記７～１１のいずれか１項記載の制御プログラム。
【０１３９】
（付記１３）
第１の記憶装置及び第２の記憶装置の制御を行なうストレージ制御装置における制御方法であって、
前記第１の記憶装置の記憶領域を所定の大きさで分割した複数の単位領域について、入力された要求に対する応答性能を監視し、
前記第１の記憶装置の移動対象の単位領域に記憶されたデータを前記第１の記憶装置とは異なる性能の第２の記憶装置に移動する移動処理を行ない、
前記移動処理において、前記移動対象の単位領域を所定の分割数により複数の分割領域に分割し、前記データを前記分割領域の単位で前記第２の記憶装置に移動し、
前記監視により監視した前記移動処理実行中の第１の応答性能に基づいて、前記所定の分割数を変更する、
ことを特徴とする、制御方法。
【０１４０】
（付記１４）
前記変更する処理において、前記第１の応答性能と、前記移動処理実行前の第２の応答性能とに基づいて、前記所定の分割数を変更する、
ことを特徴とする、付記１３記載の制御方法。
（付記１５）

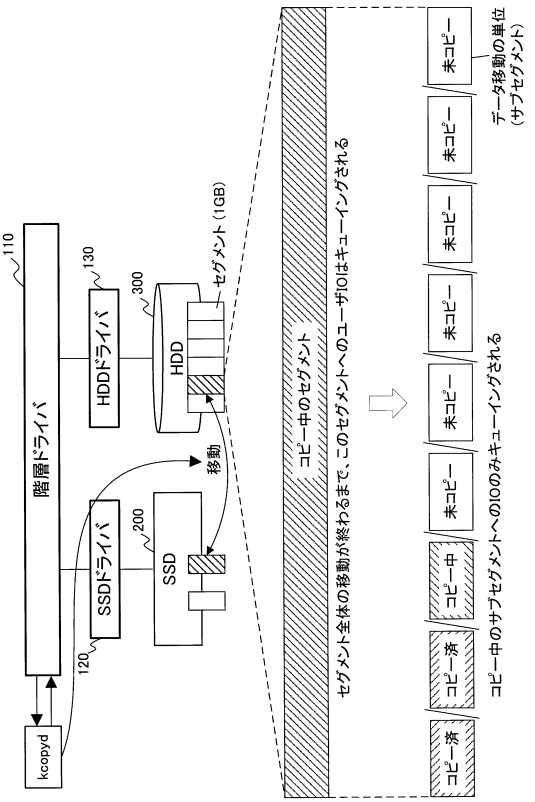
前記所定の分割数は、前記移動処理実行前の第２の応答性能を基に求められる、
ことを特徴とする、付記１４記載の制御方法。
【０１４１】
（付記１６）
前記変更する処理において、前記第１の応答性能と前記第２の応答性能とを比較し、前記第１の応答性能が前記第２の応答性能よりも劣化していると判断した場合に前記所定の分割数を増加させる一方、前記第１の応答性能が前記第２の応答性能よりも優れていると判断した場合に前記所定の分割数を減少させる、
ことを特徴とする、付記１３～１５のいずれか１項記載の制御方法。
【０１４２】
（付記１７）
前記変更する処理において、前記監視により前記移動処理実行中の複数の時点で監視した複数の第１の応答性能を取得し、前記第２の応答性能と前記複数の応答性能の平均値とを比較して、前記平均値が前記第２の応答性能よりも劣化しているか否かに応じて前記所定の分割数を増減させる、
ことを特徴とする、付記１６記載の制御方法。
【０１４３】
（付記１８）
前記複数の単位領域について単位領域ごとに入出力数を集計し、
前記集計された入出力数が第１の閾値より大きな単位領域と所定の距離内にある単位領域を繋ぎ合わせた拡張領域と、該拡張領域と繋がる他の拡張領域と、を合わせた移動領域を特定し、
前記移動領域に記憶されたデータを前記第２の記憶装置に移動する移動処理において、前記移動領域に含まれる複数の単位領域の各々を所定の分割数により複数の分割領域に分割し、前記移動領域に記憶されたデータを前記分割領域の単位で前記第２の記憶装置に移動する、
ことを特徴とする、付記１３～１７のいずれか１項記載の制御方法。
【符号の説明】
【０１４４】
１，１Ａ　階層ストレージシステム（ストレージ装置）
１０，１０Ａ　階層ストレージ制御装置（ストレージ制御装置）
１０ａ　ＣＰＵ
１０ｂ　メモリ
１０ｃ　記憶部
１０ｄ　インタフェース部
１０ｅ　入出力部
１０ｆ，１０ｈ　記録媒体
１０ｇ　読取部
１１，１１Ａ　階層管理部
１１ａ，１５ａ　データ収集部
１１ｂ，１５ｂ　データベース
１１ｃ，１５ｃ　ワークロード分析部
１１ｄ，１５ｄ　移動指示部
１１ｅ　分割数判定部
１２，１１０　階層ドライバ
１２ａ　ＩＯマップ部
１２ｂ　ペンディングキュー
１２ｃ　階層テーブル
１２ｄ　分割部
１３，１２０　ＳＳＤドライバ

１４，１３０　ＨＤＤドライバ
１５１　候補テーブル
１５２　管理テーブル
２０，２００　ＳＳＤ
３０，３００　ＨＤＤ
１００　階層ストレージ制御装置

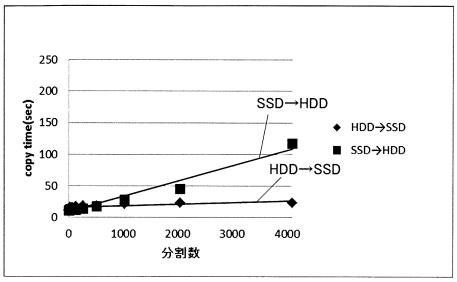
【図１】



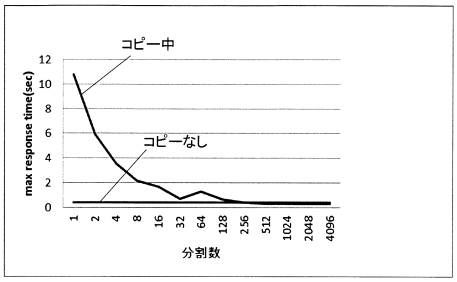
【図２】



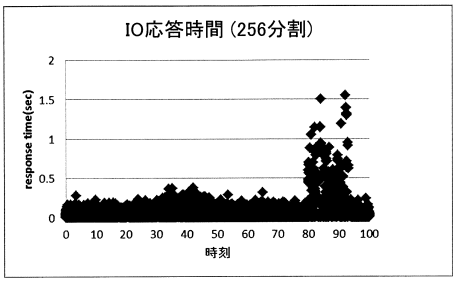
【図３】



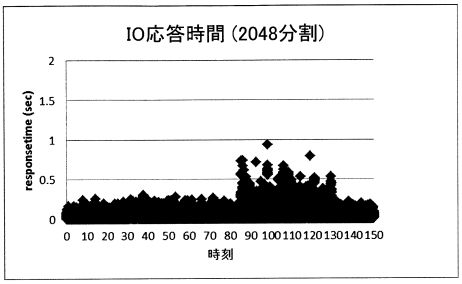
【図４】



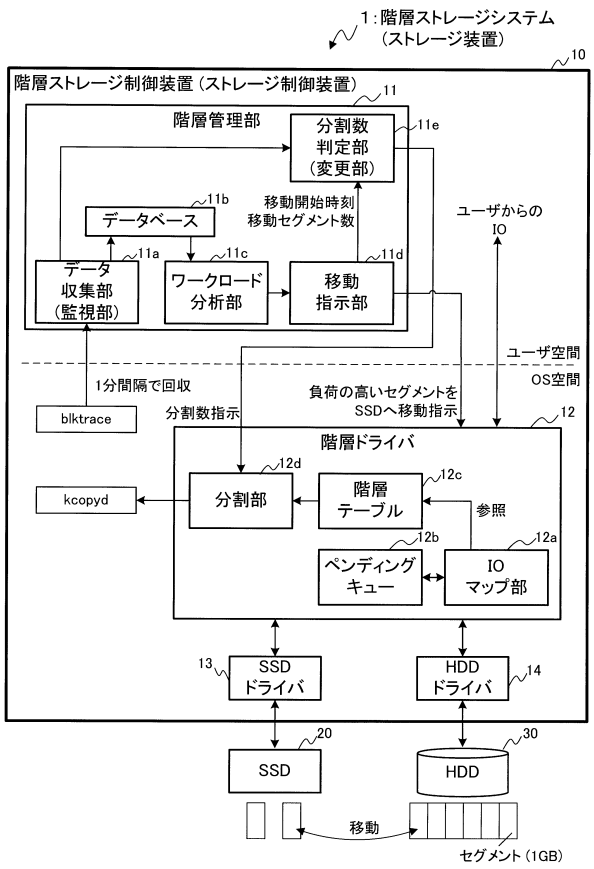
【図５】



【図６】



【図７】



【図８】

データベース 11b

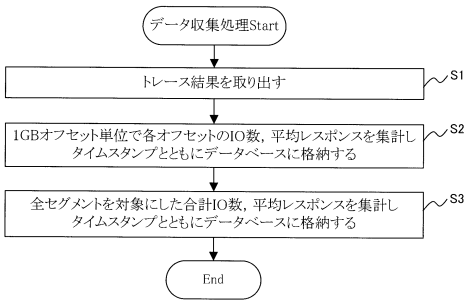
セグメント	IO数	平均レスポンス(s)	タイムスタンプ
0	1000	0.6	1
1	1200	0.4	3
:	:	:	:
all	50000	0.8	10
all	55000	0.9	11

【図９】

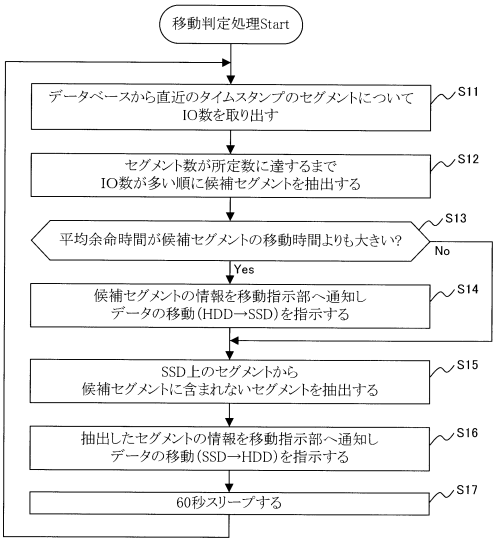
階層テーブル 12c

SSDオフセット	HDDオフセット	状態
0	268435456	allocated
2097152	306184192	Moving(HDD→SSD)
4194304	505413632	Moving(SSD→HDD)
6291456	NULL	free
:	:	:

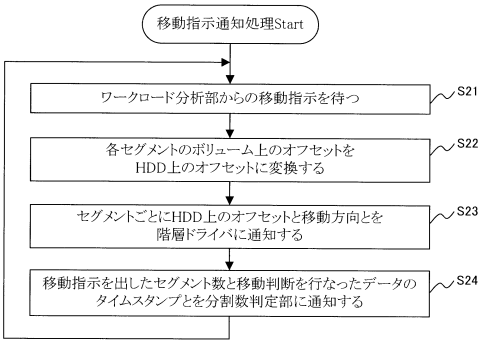
【図１０】



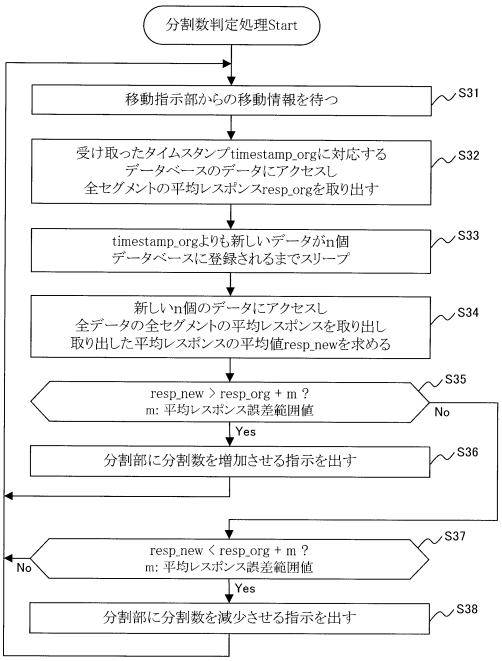
【図１１】



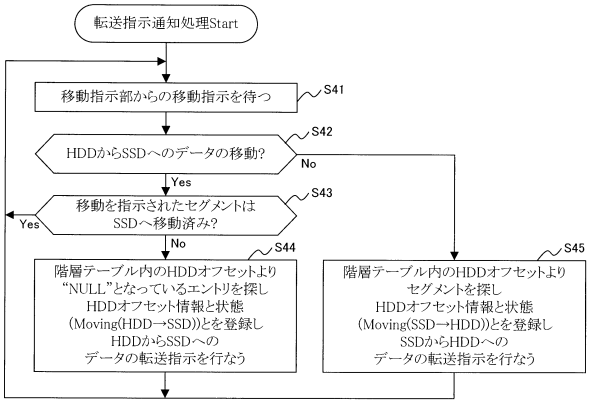
【図１２】



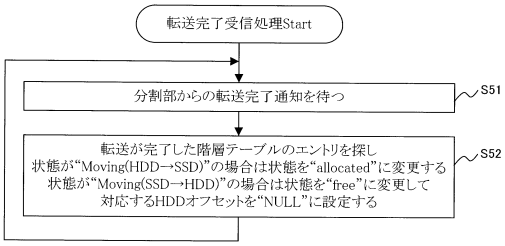
【図１３】



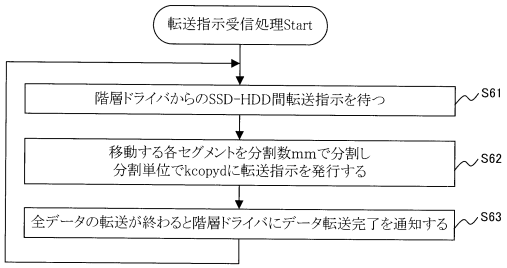
【図１４】



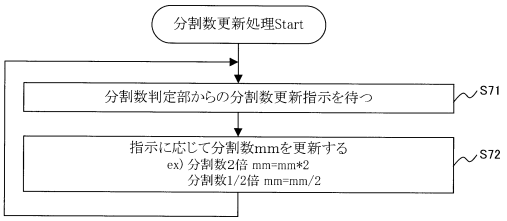
【図１５】



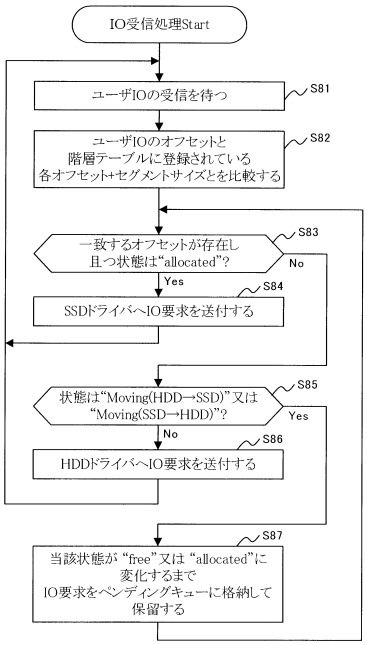
【図１６】



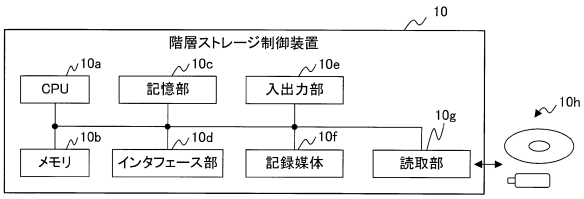
【図１７】



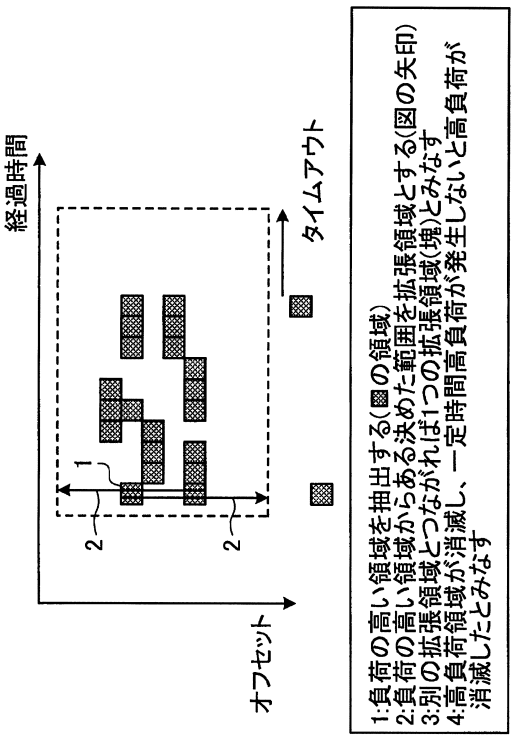
【図１８】



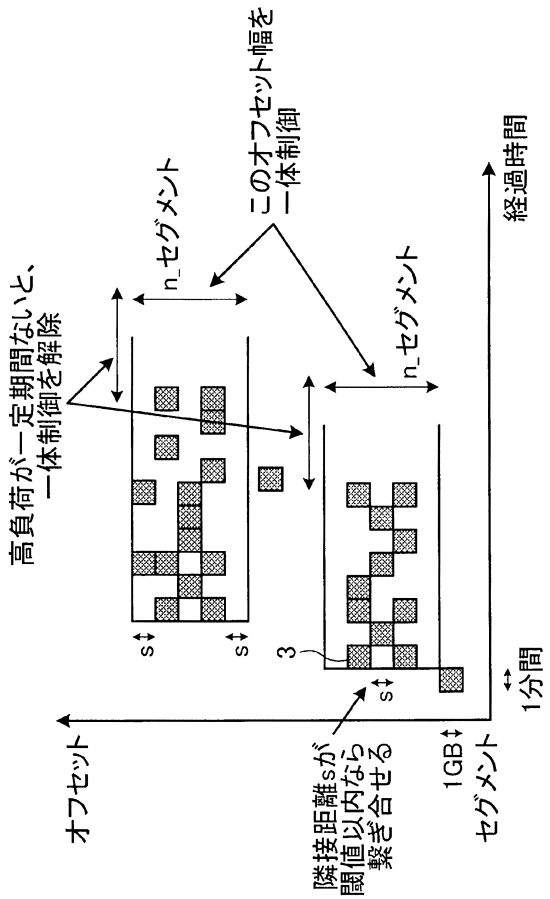
【図１９】



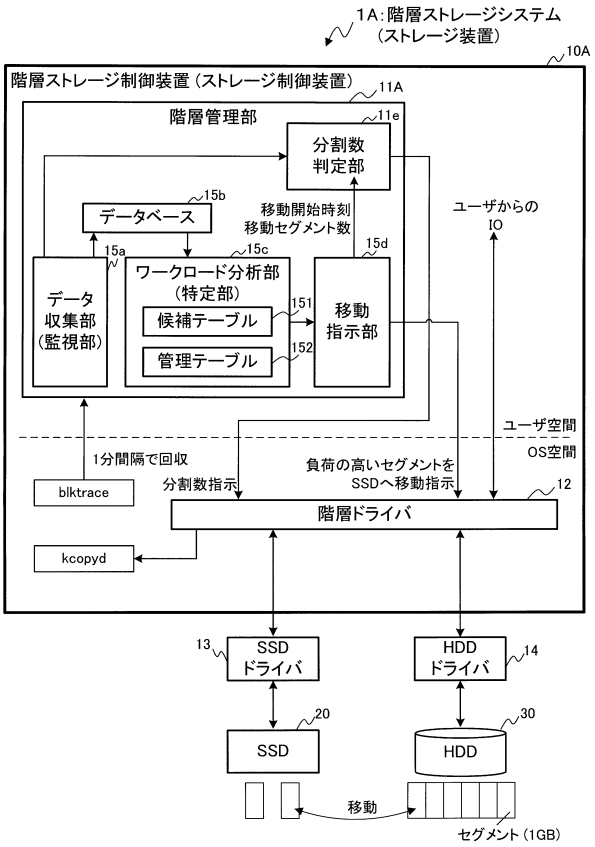
【図２０】



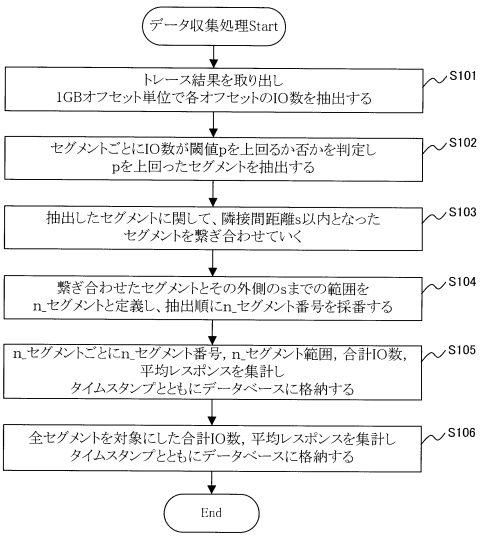
【図２１】



【図２２】



【図２３】

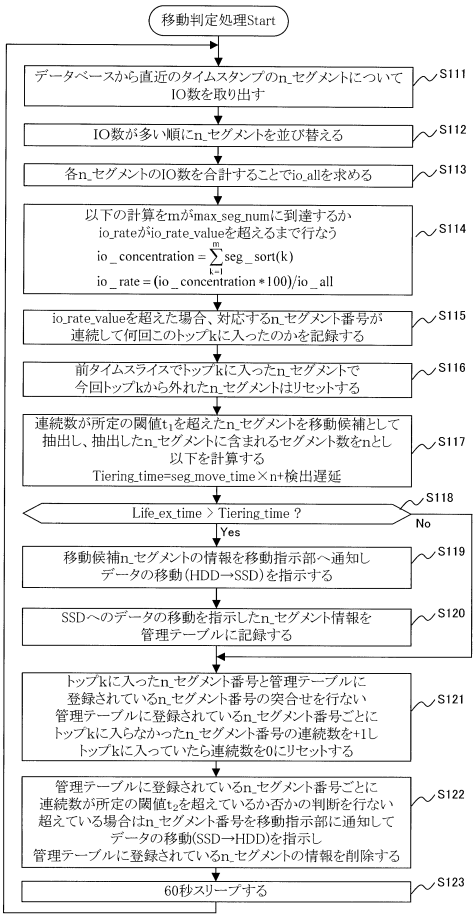


【図２４】

15b
データベース

n_セグメント番号	セグメント範囲	IO数	平均レスポンス(s)	タイムスタンプ
0	3	1000	0.6	1
：	：	：	：	：
all	1	50000	0.8	10
all	1	55000	0.9	11

【図２５】



【図２６】

151

候補テーブル

n_セグメント番号	先頭セグメント番号	セグメント数	連続数
0	10	5	1
:	:	:	:

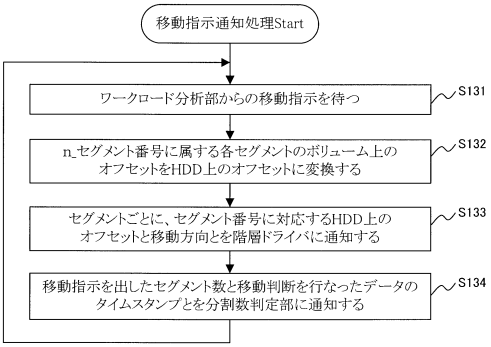
【図２７】

152

管理テーブル

n_セグメント番号	先頭セグメント番号	セグメント数	連続数
0	10	5	2
:	:	:	:

【図２８】



フロントページの続き

(72)発明者　河場　基行
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内

審査官　田名網　忠雄

(56)参考文献　特開２００６－１１３８８２（ＪＰ，Ａ）
特開２０１０－１４６４５０（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　　３／０６－３／０８
Ｇ０６Ｆ　１２／００
Ｇ０６Ｆ　１３／１０－１３／１４