

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2004-54721

(P2004-54721A)

(43) 公開日 平成16年2月19日(2004.2.19)

(51) Int. Cl. ⁷	F I	テーマコード (参考)
G06F 12/00	G06F 12/00 514E	5B065
G06F 3/06	G06F 12/00 545A	5B082
	G06F 3/06 301A	

審査請求 未請求 請求項の数 19 O L (全 19 頁)

(21) 出願番号	特願2002-213245 (P2002-213245)	(71) 出願人	000005108 株式会社日立製作所 東京都千代田区神田駿河台四丁目6番地
(22) 出願日	平成14年7月23日 (2002.7.23)	(74) 代理人	100075096 弁理士 作田 康夫
		(72) 発明者	上村 哲也 東京都国分寺市東恋ヶ窪一丁目280番地 株式会社日立製作所中央研究所内
		(72) 発明者	藤原 真二 東京都国分寺市東恋ヶ窪一丁目280番地 株式会社日立製作所中央研究所内
		Fターム(参考)	5B065 BA01 CA11 CA13 CC00 5B082 HA05 HA08

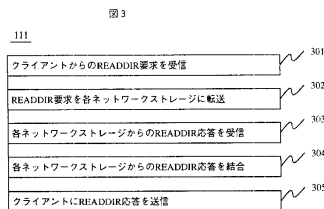
(54) 【発明の名称】 ネットワークストレージ仮想化方法

(57) 【要約】

【課題】 ネットワークストレージの台数増加にともない集中管理型サーバがボトルネックになりシステムのスケラビリティを阻害する。専用のネットワークストレージを用いる場合には、既存のネットワークストレージを仮想化できない。

【解決手段】 301によりクライアントからのREADDIR要求を受信した後、302により受信したREADDIR要求を各ネットワークストレージに転送し、303により各ネットワークストレージからのREADDIR応答を受信し、304により受信した各ネットワークストレージからのREADDIR応答を結合し、305により結合したREADDIR応答をクライアントに送信することにより、専用のネットワークストレージ、集中管理型サーバ、分散ディレクトリを用いずに、汎用のネットワークストレージとネットワークストレージアクセス用プロトコルの処理によりネットワークストレージ仮想化が実現できる。

【選択図】 図3



【特許請求の範囲】**【請求項 1】**

複数のネットワークストレージを仮想的な一台のネットワークストレージとしてアクセス可能にするネットワークストレージ仮想化方法において、クライアントからの要求を受信するステップと、第一のネットワークストレージおよび第二のネットワークストレージに前記受信した要求を転送するステップと、前記転送した要求に対する前記第一のネットワークストレージからの応答を受信するステップと、前記転送した要求に対する前記第二のネットワークストレージからの応答を受信するステップと、前記受信した第一のネットワークストレージの応答と前記受信した第二のネットワークストレージの応答とを結合するステップと、前記結合した応答を前記クライアントに送信するステップと、を含むことを特徴とするネットワークストレージ仮想化方法。

10

【請求項 2】

複数のネットワークストレージを仮想的な一台のネットワークストレージとしてアクセス可能にするネットワークストレージ仮想化方法において、クライアントからの要求を受信するステップと、前記受信した要求を複数のネットワークストレージに転送するステップと、前記転送した要求に対する前記複数のネットワークストレージからの応答を受信するステップと、前記受信した応答が正常かエラーかを判定するステップと、前記判定した応答の中で正常な応答のみを結合するステップと、前記結合した応答を前記クライアントに送信するステップと、を含むことを特徴とするネットワークストレージ仮想化方法。

20

【請求項 3】

複数のネットワークストレージを仮想的な一台のネットワークストレージとしてアクセス可能にするネットワークストレージ仮想化方法において、クライアントからの要求を受信するステップと、前記受信した要求を複数のネットワークストレージに転送するステップと、前記転送した要求に対する前記複数のネットワークストレージからの応答を受信するステップと、前記受信した応答が正常かエラーかを判定するステップと、前記判定した判定した応答の中で正常な応答の一つを前記クライアントへ送信するステップとを含むことを特徴とするネットワークストレージ仮想化方法。

30

【請求項 4】

請求項 1 または請求項 2 または請求項 3 のネットワークストレージ仮想化方法において、前記クライアントに送信した前記応答を基に前記クライアントが生成した要求を受信するステップと、前記ネットワークストレージから受信した応答を基に前記受信した要求を転送するネットワークストレージを判定するステップと、前記判定したネットワークストレージに前記要求を転送するステップと、前記転送した要求に対する前記ネットワークストレージからの応答を受信するステップと、前記受信した応答を前記クライアントに転送するステップとを含むことを特徴とするネットワークストレージ仮想化方法。

40

【請求項 5】

請求項 1 または請求項 2 または請求項 3 のネットワークストレージ仮想化方法において、前記ネットワークストレージの応答の中のアクセス識別子を各ネットワークストレージ固有の情報を用いて符号化するステップと、を含むことを特徴とするネットワークストレージ仮想化方法。

【請求項 6】

請求項 1 または請求項 2 または請求項 3 のネットワークストレージ仮想化方法において、少なくとも前記ネットワークストレージ識別情報と前記ネットワークストレージの応答の

50

中のアクセス識別子とを一時記憶用の領域に保存するステップとを含むことを特徴とするネットワークストレージ仮想化方法。

【請求項 7】

複数のネットワークストレージを仮想的な一台のネットワークストレージとしてアクセス可能にするネットワークストレージの仮想化方法において、

複数のネットワークストレージの動作状態を検出するステップと、

障害検出時に障害情報を含んだエントリを生成するステップと、

クライアントからの要求を受信するステップと、前記複数のネットワークストレージのうち正常に動作しているネットワークストレージのエントリ情報と前記生成した障害情報を含んだエントリ情報とを結合して前記アクセス要求に対する応答メッセージを生成するステップと、

10

前記生成した応答メッセージを前記クライアントに送信するステップとを含むことを特徴とするネットワークストレージ仮想化方法。

【請求項 8】

複数のネットワークストレージを仮想的な一台のネットワークストレージとしてアクセス可能にするネットワークストレージの仮想化方法において、前記複数のネットワークストレージのエントリ情報の整合性管理を行うことを特徴とするネットワークストレージ仮想化方法。

【請求項 9】

請求項 8 に記載のネットワークストレージの仮想化方法において、障害が発生していたネットワークストレージが正常動作に復帰したかどうかを検出するステップを有し、

20

前記エントリ情報の整合性管理には、前記障害が発生していたネットワークストレージのエントリ情報と、正常動作していた任意のネットワークストレージ上のエントリ情報との間の整合性を調査するステップと、

不整合が発生した場合に前記障害が発生していたネットワークストレージ上のエントリ情報の識別子を変更するステップとを含むことを特徴とするネットワークストレージ仮想化方法。

【請求項 10】

請求項 8 に記載のネットワークストレージの仮想化方法において、障害が発生していたネットワークストレージが正常動作に復帰したかどうかを検出するステップを有し、

30

前記エントリ情報の整合性管理には、前記障害が発生していたネットワークストレージのエントリ情報と、正常動作していた任意のネットワークストレージ上のエントリ情報との間の整合性を調査するステップと、

不整合が発生した場合に前記障害が発生していたネットワークストレージ上に不整合管理用エントリを作成するステップと、

前記不整合の発生したエントリを前記不整合管理用エントリに移動するステップとを含むことを特徴とするネットワークストレージ仮想化方法。

【請求項 11】

請求項 8 に記載のネットワークストレージの仮想化方法において、新しいネットワークストレージの情報を登録するステップを有し、

40

前記エントリ情報の整合性管理には、前記仮想化している複数のネットワークストレージのエントリ情報と前記新しいネットワークストレージのエントリ情報との間の整合性を調査するステップと、

不整合がある場合に前記新しいネットワークストレージ上の前記不整合のあるエントリ情報の識別子を変更するステップとを含むことを特徴とするネットワークストレージ仮想化方法。

【請求項 12】

請求項 8 に記載のネットワークストレージの仮想化方法において、新しいネットワークストレージの情報を登録するステップを有し、

前記エントリ情報の整合性管理には、

50

前記仮想化している複数のネットワークストレージのエントリ情報と前記新しいネットワークストレージのエントリ情報との間の整合性を調査するステップと、不整合がある場合に前記新しいネットワークストレージ上に不整合管理用エントリを作成するステップと、前記不整合のあるエントリを前記不整合管理用エントリに移動するステップとを含むことを特徴としたネットワークストレージ仮想化方法。

【請求項 13】

複数のネットワークストレージを仮想的な一台のネットワークストレージとしてアクセス可能にするネットワークストレージ仮想化方法において、
仮想化している複数のネットワークストレージ中の特定の一台のネットワークストレージを表す仮想的なエントリを生成するステップと、
クライアントからの要求が前記仮想的なエントリへの要求かを判定するステップと、
前記クライアントの要求が前記仮想的なエントリへの要求の場合に前記クライアントの要求を前記特定のネットワークストレージへ転送するステップと、を含むことを特徴とするネットワークストレージ仮想化方法。

10

【請求項 14】

複数のネットワークストレージを仮想的な一台のネットワークストレージとしてアクセス可能にするネットワークストレージ仮想化方法において、
前記複数のネットワークストレージへのアクセスの負荷分散を行うことを特徴とするネットワークストレージ仮想化方法。

20

【請求項 15】

請求項 14 に記載のネットワークストレージ仮想化方法において、
前記負荷分散には、
前記複数のネットワークストレージの負荷をモニタするステップと、前記複数のネットワークストレージの中で最も負荷の低いネットワークストレージを選択するステップと、
クライアントからの新規データ書き込みの要求を前記負荷の最も低いネットワークストレージに転送するステップと、を含むことを特徴とするネットワークストレージ仮想化方法。

【請求項 16】

請求項 14 に記載のネットワークストレージ仮想化方法において、
前記負荷分散には、
前記複数のネットワークストレージへのアクセス集中をモニタするステップと、
前記複数のネットワークストレージの特定のエントリへのアクセス集中をモニタするステップと、
特定のネットワークストレージへのアクセス集中の許容度を指定するステップと、
前記ネットワークストレージへのアクセス集中モニタの結果から前記許容度を越えているネットワークストレージを特定するステップと、
前記許容度を越えたネットワークストレージの中でアクセスが集中しているエントリを特定するステップと、
前記アクセスが集中しているエントリを前記複数のネットワークストレージの中で前記許容度を越えていないネットワークストレージへと移動するステップと、を含むことを特徴とするネットワークストレージ仮想化方法。

30

40

【請求項 17】

請求項 14 に記載のネットワークストレージ仮想化方法において、
前記負荷分散には、
前記複数のネットワークストレージの負荷をモニタするステップと、
前記複数のネットワークストレージの負荷の偏りの許容度を指定するステップと、
前記複数のネットワークストレージの中で前記許容度を越えているネットワークストレージを特定するステップと、
前記許容度を越えているネットワークストレージの中で前記許容度を越える原因となるエントリを特定するステップと、

50

前記原因となるエントリを前記複数のネットワークストレージの中で前記許容度を越えていないネットワークストレージへと移動するステップと、を含むことを特徴とするネットワークストレージ仮想化方法。

【請求項18】

複数のネットワークストレージを仮想的な一台のネットワークストレージとしてアクセス可能にするネットワークストレージの仮想化方法において、

複数のネットワークストレージを仮想化して運用するのに必要な情報を前記複数のネットワークストレージ上に保存するステップとを含むことを特徴とするネットワークストレージ仮想化方法。

【請求項19】

複数のネットワークストレージを仮想的な一台のネットワークストレージとしてアクセス可能にするネットワークストレージ仮想化方法において、

複数のネットワークストレージの識別子を登録するステップと、

前記複数のネットワークストレージへアクセスするための情報を登録するステップと、

前記複数のネットワークストレージの管理者情報を登録するステップとを含むことを特徴とするネットワークストレージ仮想化方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、ネットワークストレージ仮想化方法に関し、特にクライアントからの要求を複数のネットワークストレージに転送し、ネットワークストレージからの応答を結合してクライアントに送信する、ストレージ仮想化方法に関する。

【0002】

【従来の技術】

従来の情報処理システムでは、計算機システムに直接接続したストレージに情報を格納していた。ストレージに格納した情報は直接接続した計算機のみからアクセス可能なため、他の計算機システムから情報にアクセスするためには直接接続した計算機を経由する必要があった。しかし近年のネットワーク技術の発達と格納すべき情報量の飛躍的な増大により、情報を処理する計算機システムと情報を格納するストレージシステムの分離の動きが進み、ネットワークに接続し複数の計算機システムで情報を共有できるストレージシステムが出現してきた。このようなネットワークに接続するストレージをネットワークストレージと呼ぶ。

【0003】

このようなネットワークストレージの例として、SAN (Storage Area Network) で接続してブロックアクセスを提供するSANストレージ、IPネットワークやInfiniband等で接続してファイルアクセスを提供するNAS (Network Attached Storage)、WebアクセスのインタフェースであるHTTPプロトコルやそれを拡張したプロトコルでのアクセス形態を提供するWebストレージなどがある。

【0004】

ネットワークストレージの普及につれ、システム管理者はネットワークに接続した複数のネットワークストレージを管理する必要が出て来ている。例えば、格納すべきデータ量が既存のネットワークストレージの最大値を超えるような場合には、新たなネットワークストレージを追加しデータの再分配などのシステム再構築が必要になる。ストレージ容量に対する需要の急激な伸びは度重なるシステム再構築を強いるため、システム管理コストを膨張させてきた。

【0005】

システム管理コストの抑制には、複数のネットワークストレージを計算機システムから仮想的にひとつに見せ、新たな装置の追加を行なってもシステム全体に影響が及ばないようにするストレージの仮想化技術が必須となっている。このようなストレージの仮想化技術

10

20

30

40

50

に関しては種々の方式が開発され、提案されている

例えば、1995年にHartmanらが発表したZebraシステム(The Zebra Striped Network File System, Hartman et. al., ACM Transactions on Computer System, vol. 13, No. 3, 1995, pp. 274-310.)では、ひとつのファイルを複数のネットワークストレージに分散格納する方式を提案している。ファイルを一定の長さごとに分割し、ネットワークストレージにラウンド・ロビン方式で順次格納する。Zebraではファイルがどのストレージにどのような順番で分散格納しているか集中管理するサーバがあり、ファイルアクセス処理時にはこの管理サーバにファイルの格納位置に関する情報を問い合わせしてからデータをアクセスするという特長がある。すなわち、ファイルを複数のサーバに分散格納させて一つのサーバで集中管理することにより複数のネットワークストレージを仮想的に一つの大きいネットワークストレージに見せる効果がある。

10

【0006】

また、Alexander H. FrayらはZebraの集中管理サーバ方式と異なる非集中管理サーバ方式によりファイルのストライピングを行なう方法を提案している(USP 6,029,168、Filed: 1998/01/23)。この方式では、ファイルのストライピング情報が格納されているスタートノード情報をファイル識別子に埋め込むことでZebraで集中管理していたストライピング情報を複数のネットワークストレージで分散管理している。ファイル識別子にスタートノードを格納するため、ファイルアクセスの際にはまずスタートノードへアクセスを行ない、そこで、ファイルの格納位置を決定し、必要があればアクセスすべきファイルの断片(ブロック)を格納しているサーバへアクセス要求を転送して処理することが特徴となる。このため、クライアントからの当該ファイルについてのアクセスは、スタートノードへ要求を発行するだけで良く、ファイルのストライピングによる格納位置を考慮する必要がないという効果がある。

20

【0007】

【発明が解決しようとする課題】

しかし、前記Zebraシステムでは、アクセスするファイルの断片を格納するネットワークストレージを決定する際に集中管理サーバに問い合わせる必要があり、特にファイルの更新ではネットワークストレージ上のデータの他に集中管理サーバ上のデータも更新する必要があり、このため、ネットワークストレージの台数が増加すると集中管理サーバがボトルネックとなり、システムのスケラビリティを阻害する原因となっていた。

30

【0008】

また、Alexanderらの非集中管理サーバ方式では、管理サーバが複数のサーバに分散されている点ではZebraの問題点を解決している。しかしながら、該システムではファイルアクセスする際にファイルを一意に特定するファイル名あるいはディレクトリ名とスタートノードが埋め込まれているファイル識別子との変換を分散ディレクトリ等のサーバが行なうことが前提となっており、ファイルの格納場所を管理するサーバ(スタートノード)とファイル名との対応を管理する情報が別途必要となっている。USP 6,029,168の実施例ではそのような情報を格納する手段としてとして a "well-known" distributed Directory fileを利用すると述べている。また、ファイル識別子にスタートノードの位置が直接埋め込まれるため、新しいネットワークストレージを追加して既存のファイルのスタートノードを新しいネットワークストレージに移動するには前記分散ディレクトリに格納されているファイル名とファイル識別子との対応情報を書きかえる必要があった。

40

【0009】

さらに、これら従来の技術による方式では専用のネットワークストレージが必要であり、異なる形式のネットワークストレージを混在させて仮想化して運用することができない。つまり、管理コスト低減のため既に運用している複数の型式のネットワークストレージを仮想化して単一のネットワークストレージのイメージとして運用することが出来ず、既存

50

のネットワークストレージの運用コストを低減することが出来ない問題がある。

【0010】

また、管理サーバを用いる場合にはファイルの管理情報とデータ本体とが異なるサーバに格納されるため、どちらか一方のサーバに障害が発生してもファイルへのアクセスができなくなる問題がある。

【0011】

【課題を解決するための手段】

本発明の目的の一つは、専用ではなく汎用のネットワークストレージのみによって仮想化が可能なネットワークストレージ仮想化方法を提供することであり、このような目的を実現する構成要件としては、例えば以下のような構成が考えられる。

10

【0012】

クライアントからの要求を受信するステップと、第一のネットワークストレージおよび第二のネットワークストレージに受信した要求を転送するステップと、転送した要求に対する第一のネットワークストレージからの応答を受信するステップと、転送した要求に対する第二のネットワークストレージからの応答を受信するステップと、受信した第一のネットワークストレージの応答と受信した第二のネットワークストレージの応答とを結合するステップと、結合した応答を前記クライアントに送信するステップとを含む構成。

【0013】

あるいは、受信したクライアントからの要求を複数のネットワークストレージに転送するステップと、転送した要求に対する複数のネットワークストレージからの応答を受信するステップと、受信した応答が正常かエラーかを判定するステップと、判定した応答の中で正常な応答のみを結合するステップとを含む構成であってもよい。

20

【0014】

更にまた、受信したクライアントからの要求を複数のネットワークストレージに転送するステップと、転送した要求に対する複数のネットワークストレージからの応答を受信するステップと、受信した応答が正常かエラーかを判定するステップと、判定した応答の中で正常な応答の一つをクライアントへ送信するステップとを含む構成であってもよい。

【0015】

また、本発明の目的を満たすネットワークストレージ仮想化方法としては、ネットワークストレージ識別情報とネットワークストレージの応答の中のアクセス識別子とを一時記憶用の領域に保存するステップとを含んでもよい。あるいは、複数のネットワークストレージのエントリ情報の整合性管理を行う機能を有していてもよい。

30

【0016】

更にまた、本発明の目的を実現する構成の一つとしては、複数のネットワークストレージの動作状態を検出するステップと、障害検出時に障害情報を含んだエントリを生成するステップと、クライアントからの要求を受信するステップと、複数のネットワークストレージのうち正常に動作しているネットワークストレージのエントリ情報と生成した障害情報を含んだエントリ情報とを結合してアクセス要求に対する応答メッセージを生成するステップと、生成した応答メッセージを前記クライアントに送信するステップとを含んでもよい。

40

【0017】

また、仮想化している複数のネットワークストレージのエントリ情報と新しいネットワークストレージのエントリ情報との間の整合性を調査するステップと、不整合がある場合に新しいネットワークストレージ上の不整合のあるエントリ情報の識別子を変更するステップとを含む構成としてもよい。

【0018】

また、仮想化している複数のネットワークストレージ中の特定の一台のネットワークストレージを表す仮想的なエントリを生成するステップと、クライアントからの要求が仮想的なエントリへの要求かを判定するステップと、クライアントの要求が仮想的なエントリへの要求の場合にクライアントの要求を特定のネ

50

ットワークストレージへ転送するステップとを含む構成であっても良い。

【0019】

更にまた、本発明の目的を実現する構成要件の一つとして、複数のネットワークストレージへのアクセスの負荷分散を行う機能を含んでいても良い。

【0020】

あるいは、複数のネットワークストレージを仮想化して運用するのに必要な情報を複数のネットワークストレージ上に保存するステップとを含む構成であっても良い。

【0021】

更には、複数のネットワークストレージの識別子を登録するステップと、複数のネットワークストレージへアクセスするための情報を登録するステップと、複数のネットワークストレージの管理者情報を登録するステップとを含む構成であっても良い。

10

【0022】

【発明の実施の形態】

発明の実施の形態について図面を参照して説明する。同一の参照番号は同一の構成要素を示す。なお、以下の実施例において、READDIRとは、NFS(Network File System)でディレクトリから任意の数のエントリを順番に取りだし、クライアントがそれ以降のREADDIR要求で他のエントリを要求するための情報とともに、名前とファイルの識別子を返す動作を意味する。また、LOOKUPとは、ある名前のディレクトリを検索し、該当するファイルシステムオブジェクトのファイルハンドルと属性を返す動作を意味する。更に、エントリとはディレクトリに格納されたファイルやディ

20

【0023】

図1は、本発明によるネットワークストレージ仮想化方法の実施の形態のアーキテクチャのブロック図である。図1において、100は本発明によるネットワークストレージ仮想化方法の実施の形態のアーキテクチャ、101は演算装置、102はメモリ、103はLAN接続インタフェース、111はREADDIR処理ハンドラ、112はLOOKUP処理ハンドラ、113はREAD処理ハンドラ、114はWRITE処理ハンドラ、115はCREATE処理ハンドラ、116は動作モニタ処理ハンドラ、117は障害処理ハンドラ、118は負荷モニタ処理ハンドラ、119は特殊エントリ管理処理ハンドラ、である。

30

【0024】

図2は、本発明によるネットワークストレージ仮想化方法の実施の形態のネットワーク接続図である。図2において、200は本発明によるネットワークストレージ仮想化方法の実施の形態のネットワーク接続図、201はクライアント、202は本発明のネットワークストレージ仮想化方法の実施の形態によるネットワークストレージ仮想化用コントローラ、211は第1のネットワークストレージ、212は第2のネットワークストレージ、213は第3のネットワークストレージ、である。

【0025】

図3は、本発明によるネットワークストレージ仮想化方法の実施の形態のREADDIR処理ハンドラ111のフロー図である。図3において、301はクライアントからのREADDIR要求受信ステップ、302は、READDIR要求の各ネットワークストレージへの転送ステップ、303は各ネットワークストレージからのREADDIR応答受信ステップ、304は各ネットワークストレージからのREADDIR応答結合ステップ、305はクライアントへのREADDIR応答送信ステップ、である。

40

【0026】

図4は、本発明によるネットワークストレージ仮想化方法の実施の形態のLOOKUP処理ハンドラ111のフロー図である。図4において、401はクライアントからのLOOKUP要求受信ステップ、402はLOOKUP要求の各ネットワークストレージへの転送ステップ、403は各ネットワークストレージからのLOOKUP応答受信ステップ、404はクライアントへの正常なLOOKUP応答転送ステップ、である。

50

【0027】

図5は、本発明によるネットワークストレージ仮想化方法の実施の形態のREAD処理ハンドラ113のフロー図である。図5において、501はクライアントからのREAD要求受信ステップ、502はREAD要求の適切なネットワークストレージへの転送ステップ、503はネットワークストレージからのREAD応答受信ステップ、504はクライアントへのREAD応答転送ステップ、である。

【0028】

図6は、本発明によるネットワークストレージ仮想化方法の実施の形態のWRITE処理ハンドラ114のフロー図である。図6において、601はクライアントからのWRITE要求受信ステップ、602はWRITE要求の適切なネットワークストレージへの転送ステップ、603はネットワークストレージからのWRITE応答受信ステップ、604はクライアントへのWRITE応答転送ステップ、である。

10

【0029】

図7は、本発明によるネットワークストレージ仮想化方法の実施の形態のCREATE処理ハンドラ115のフロー図である。図7において、701はクライアントからのCREATE要求受信ステップ、702はCREATE要求の適切なネットワークストレージへの転送ステップ、703はネットワークストレージからのCREATE応答受信ステップ、704はクライアントへのCREATE応答転送ステップ、である。

【0030】

図8は、本発明によるネットワークストレージ仮想化方法の実施の形態の負荷モニタ処理ハンドラ118のフロー図である。801はネットワークストレージのディスク容量調査ステップ、802はネットワークストレージのネットワーク帯域調査ステップ、803はネットワークストレージの負荷評価ステップ、804は特定ファイルへのアクセス状況調査ステップ、805はファイルの再配置ステップ、である。

20

【0031】

本実施の形態では、クライアント201からネットワークストレージへの各種要求をコントローラ202が一旦受信し、コントローラ202がネットワークストレージ211、212、213に要求を転送し、ネットワークストレージ211、212、213からの応答をコントローラ202が結合または取捨選択した後に、クライアント201に送信するところに特徴がある。

30

【0032】

RFC1094、RFC1813、RFC3010などに記載のNFSプロトコルのようなネットワークストレージアクセス用プロトコルでは、クライアントからネットワークストレージ上のファイルのデータにアクセスする際に、ファイル名ではなくファイルハンドルという値を用いる。ファイルハンドルはネットワークストレージ上でファイルと一対一に結び付く値で、ネットワークストレージが生成する。クライアントがファイルハンドルを取得するためには、ネットワークストレージに対してディレクトリからファイル名を取得するREADDIR要求や、ディレクトリを検索しファイル名に対応するファイルハンドルを取得するLOOKUP要求などを発行する必要がある。

【0033】

本実施の形態によるREADDIR要求の処理は以下のようになる。クライアント201はREADDIR処理要求を直接ネットワークストレージ211、212、213に送信するのではなく、コントローラ202に送信する。コントローラ202では、LAN接続インタフェース103でREADDIR要求を受信した後、演算装置101が要求の種別を判定し、メモリ102上のREADDIR処理ハンドラ111を呼び出す。

40

【0034】

READDIR処理ハンドラ111ではクライアントからのREADDIR要求受信ステップ301の後、受信したREADDIR要求をステップ302でLAN接続インタフェース103によりネットワークストレージ211、212、213に転送する。コントローラ202が受信したクライアント201のREADDIR要求では、送信元がクライ

50

ント201、受信先がコントローラ202になっているが、ネットワークストレージ211、212、213へ転送する際に、送信元をコントローラ202に変更し、受信先をそれぞれネットワークストレージ211、212、213に変更する。また、これらの送受信関係以外の付加的な情報についても必要に応じて変更する。これにより、ネットワークストレージ211、212、213は通常のファイルアクセスプロトコルのみでアクセス可能であり、専用のネットワークストレージが不要になる効果がある。また同時に集中管理型サーバや分散ディレクトリが不要になる効果がある。

【0035】

ネットワークストレージ211、212、213は受信したREADDIR要求を処理した後、コントローラ202にREADDIR応答を返す。コントローラ202はステップ303でLAN接続インタフェース103によりネットワークストレージ211、212、213からのREADDIR応答を受信した後、ステップ304でREADDIR応答を結合する。例えば、ネットワークストレージ211がfile1、file2、file3の情報を返し、ネットワーク212がfileA、fileB、fileCの情報を返し、ネットワークストレージ213がfile、file、fileの情報を返す場合には、コントローラ202ではこれらの応答を結合し、file1、file2、file3、fileA、fileB、fileC、file、file、fileの情報からなる応答を生成する。ステップ304で応答を結合した後、ステップ305でLAN接続インタフェース103により結合した応答をクライアント201に送信する。この時クライアント201からのREADDIR要求に対する応答にするため、送信元
10
20

【0036】

このようなREADDIR要求処理により、複数のネットワークストレージ上に存在するファイルがクライアントからは一つのネットワークストレージ上に存在するよう見え、ネットワークストレージを仮想的に一台のネットワークストレージのイメージで運用できる効果がある。さらに、ネットワークストレージに障害が発生した場合には、そのネットワークストレージ上のファイルへのアクセスができなくなるが、ファイル管理情報とファイル本体が同一のネットワークストレージで閉じており、他のネットワークストレージに関する情報を管理しないため、ネットワークストレージに発生した障害の影響が他のネットワークストレージに伝播せず、サービスを続行できる効果がある。また、どのファイルがどのネットワークストレージ上に存在するかをコントローラが管理しないため、コントローラ202に障害が発生した場合にはコントローラ202を入れ替えるだけで簡単に障害から復帰できる効果がある。
30

【0037】

本実施の形態によるLOOKUP要求の処理は以下ようになる。クライアント201のLOOKUP処理要求は直接ネットワークストレージ211、212、213に送信するのではなく、コントローラ202に送信する。コントローラ202では、LAN接続インタフェース103でLOOKUP要求を受信した後、演算装置101が要求の種別を判定し、メモリ102上のLOOKUP処理ハンドラ112を呼び出す。
40

【0038】

LOOKUP処理ハンドラ112ではクライアントからのLOOKUP要求受信ステップ401の後、ステップ402で受信したLOOKUP要求をLAN接続インタフェース103によりネットワークストレージ211、212、213に転送する。コントローラ202が受信したクライアント201のLOOKUP要求では、送信元がクライアント201、受信先がコントローラ202になっているが、ネットワークストレージ211、212、213へ転送する際には、送信元をコントローラ202に変更し、受信先をそれぞれネットワークストレージ211、212、213に変更する。また、これらの送受信関係
50

以外の付加的な情報についても必要に応じて変更する。これにより、ネットワークストレージ 211、212、213 は通常のファイルアクセスプロトコルのみでアクセス可能であり、専用のネットワークストレージが不要になる効果がある。

【0039】

ネットワークストレージ 211、212、213 は受信した LOOKUP 要求を処理した後、コントローラ 202 に LOOKUP 応答を返す。コントローラ 202 はステップ 403 で LAN 接続インタフェース 103 によりネットワークストレージ 211、212、213 からの LOOKUP 応答を受信する。LOOKUP 要求はファイル名に対するファイルハンドルを検索する要求であるため、ネットワークストレージ 211、212、213 に対して LOOKUP 要求を転送しても、ファイルが存在しない場合にはエラーを返す。本実施の形態では後述する CREATE 要求処理で、複数のネットワークストレージの同一ファイル名が一つしか存在しないよう保証するため、複数のネットワークストレージからの LOOKUP 要求に対する正常応答は高々一つになる。そこで正常な LOOKUP 応答をステップ 404 で LAN 接続インタフェース 103 によりクライアントに転送する。この時クライアント 201 からの LOOKUP 要求に対する応答にするため、送信元をコントローラ 202、受信先をクライアント 201 にし、さらに付加的な情報の値を決定する。ネットワークストレージ 211、212、213 が全てエラー応答を返した場合には、クライアント 201 が要求したファイル名が存在しないか、そのファイル名を保存するネットワークストレージに障害が発生した場合である。その場合には、クライアント 201 への応答としてエラーを返す。

10

20

【0040】

このような LOOKUP 要求処理により、READDIR 要求処理の場合と同様の効果がある。

【0041】

READDIR 要求や LOOKUP 要求によりクライアントがファイルハンドルを取得すると、そのファイルハンドルに基づきクライアントが READ/WRITE 要求を発行する。ファイルハンドルは一つのネットワークストレージ上では一意であるが、複数のネットワークストレージでは一意とは限らず同一のファイルハンドルが異なるネットワークストレージ上に存在する可能性がある。本実施の形態によりネットワークストレージを仮想化し単一のネットワークストレージのイメージで運用するためには、ファイルハンドルに対応するネットワークストレージを特定できる仕組みが必要になる。

30

【0042】

解決する一つの手法として、ネットワークストレージ 211、212、213 が応答として返したファイルハンドルをネットワークストレージのホスト名や IP アドレスなどネットワークストレージに固有の情報を用いて符号化してクライアント 201 に送信する手法がある。ネットワークストレージ 211、212、213 には同一のファイルハンドルが存在する可能性があるが、これによりクライアント 201 が同一のファイルハンドルを受信しなくなる。ただし、通常ファイルハンドルは 32 バイトや 64 バイトなどの有限の値であるため、異なるネットワークストレージの異なるファイルハンドルを符号化した結果、同一の値になる可能性がある。このような場合には次に述べるキャッシュを併用することで解決できる。詳細は後述する。

40

【0043】

また別の解決手法として、ファイルハンドルをキャッシュする手法がある。ネットワークストレージから READDIR 応答や LOOKUP 応答を受信した際に、ファイル名、ファイルハンドル、そのファイルハンドルを返したネットワークストレージに固有な情報をキャッシュしておくことで、クライアントからファイルハンドルを用いた READ/WRITE 要求を受信した時に、キャッシュのファイルハンドルとネットワークストレージの対応を調べることで、どのネットワークストレージのファイルハンドルかを特定できる。

【0044】

また、本実施の形態による LOOKUP 要求処理の際に、複数のネットワークストレージ

50

から同一のファイルハンドルが返る可能性がある。この場合には、上記のファイルハンドルの符号化とファイルハンドルのキャッシュを組み合わせることで解決できる。つまり、ファイルハンドルをネットワークストレージに固有な情報で符号化した値をクライアントに送信するファイルハンドルとしてキャッシュする。前述した符号化後のファイルハンドルが同一の値になる可能性がある問題を解決するためには、キャッシュ全体で符号化したファイルハンドルの一意性を調べ、同一のファイルハンドルが存在する場合には、その部分にのみさらに符号化を行う。こうすればキャッシュ全体でのファイルハンドルの一意性を保証できる。

【0045】

本実施の形態によるファイルのREAD/WRITE要求の処理は以下のようになる。まずREAD要求の処理について説明する。クライアント201のREAD処理要求は直接ネットワークストレージ211、212、213に送信するのではなく、コントローラ202に送信する。コントローラ202では、LAN接続インタフェース103によりREAD要求を受信した後、演算装置101が要求の種別を判定し、メモリ102上のREAD処理ハンドラ113を呼び出す。

10

【0046】

コントローラ202は前述のファイルハンドラの符号化やキャッシュにより、ファイルハンドルに対応するネットワークストレージを特定できる。そこで、ステップ501でクライアントからのREAD要求を受信した後、ステップ502でLAN接続インタフェース103によりREAD要求を適切なネットワークストレージに転送する。コントローラ202が受信したクライアント201のREAD要求では、送信元がクライアント201、受信先がコントローラ202になっているが、適切なネットワークストレージに転送する際には、送信元をコントローラ202に変更し、受信先を適切なネットワークストレージに変更する。また、ファイルハンドルが符号化されている場合には復号化する。さらにまた、これら以外の付加的な情報についても必要に応じて変更する。次に、ステップ503でLAN接続インタフェース103によりネットワークストレージからのREAD応答を受信する。最後にステップ504でLAN接続インタフェース103により受信した応答をクライアント201に転送する。この時、送信元をコントローラ202に変更し、受信先をクライアント201に変更する。さらにこれら以外の付加的な情報についても必要に応じて変更する。

20

30

【0047】

これにより、ネットワークストレージ211、212、213は通常のファイルアクセスプロトコルのみでアクセス可能であり、専用のネットワークストレージが不要になる効果がある。さらにクライアントからのREAD要求の際のコントローラ202の処理はREAD要求と応答の書き換えと転送だけであり、実際のファイルの読み出しは各ネットワークストレージが行うため、コントローラ202への負荷を低減できる効果がある。

【0048】

WRITE要求の処理についてもREAD処理ハンドラ113を呼び出す代わりにWRITE処理ハンドラ114を呼び出す点を除き、READ要求処理の場合と同様である。クライアント201のWRITE処理要求は直接ネットワークストレージ211、212、213に送信するのではなく、コントローラ202に送信する。コントローラ202では、LAN接続インタフェース103によりWRITE要求を受信した後、演算装置101が要求の種別を判定し、メモリ102上のWRITE処理ハンドラ114を呼び出す。

40

【0049】

コントローラ202は前述のファイルハンドラの符号化やキャッシュにより、ファイルハンドルに対応するネットワークストレージを特定できる。そこで、ステップ601でクライアントからのWRITE要求を受信した後、ステップ602でLAN接続インタフェース103によりWRITE要求を適切なネットワークストレージに転送する。コントローラ202が受信したクライアント201のWRITE要求では、送信元がクライアント201、受信先がコントローラ202になっているが、適切なネットワークストレージに転

50

送する際には、送信元をコントローラ 202 に変更し、受信先を適切なネットワークストレージに変更する。また、ファイルハンドルが符号化されている場合には復号化する。さらにまた、これら以外の付加的な情報についても必要に応じて変更する。次に、ステップ 603 で LAN 接続インタフェース 103 によりネットワークストレージからの WRITE 応答を受信する。最後にステップ 604 で LAN 接続インタフェース 103 により受信した応答をクライアント 201 に転送する。この時、送信元をコントローラ 202 に変更し、受信先をクライアント 201 に変更する。さらにこれら以外の付加的な情報についても必要に応じて変更する。

【0050】

このような WRITE 要求処理により、READ 要求処理の場合と同様の効果がある。

10

【0051】

CREATE 要求によりファイルを新規に作成する場合には、ネットワークストレージの負荷を平均化するようにネットワークストレージを選択してファイルを作成するのが望ましい。そこで本実施の形態による CREATE 要求処理は以下ようになる。クライアント 201 の CREATE 処理要求は直接ネットワークストレージ 211、212、213 に送信するのではなく、コントローラ 202 に送信する。コントローラ 202 では、LAN 接続インタフェース 103 で CREATE 要求を受信した後、演算装置 101 が要求の種別を判定し、メモリ 102 上の CREATE 処理ハンドラ 115 を呼び出す。

【0052】

コントローラ 202 は後述する負荷モニタ処理ハンドラ 118 によって、最も負荷の小さなネットワークストレージを特定できる。そこで、ステップ 701 でクライアントからの CREATE 要求を受信した後、ステップ 702 で LAN 接続インタフェース 103 により CREATE 要求を負荷の小さい適切なネットワークストレージに転送する。コントローラ 202 が受信したクライアント 201 の CREATE 要求では、送信元がクライアント 201、受信先がコントローラ 202 になっているが、適切なネットワークストレージに転送する際には、送信元をコントローラ 202 に変更し、受信先を適切なネットワークストレージに変更する。また、これら以外の付加的な情報についても必要に応じて変更する。次に、ステップ 703 で LAN 接続インタフェース 103 によりネットワークストレージからの CREATE 応答を受信する。最後にステップ 704 で LAN 接続インタフェース 103 により受信した応答をクライアント 201 に転送する。この時、送信元をコントローラ 202 に変更し、受信先をクライアント 201 に変更する。また必要に応じてネットワークストレージが返したファイルハンドルを前述の手法により符号化し、さらにこれら以外の付加的な情報についても必要に応じて変更する。

20

30

【0053】

このような CREATE 要求処理により、READ / WRITE 要求処理の場合と同様の効果に加え、ネットワークストレージの負荷を均等化する効果がある。

【0054】

ネットワークストレージ 211、212、213 の負荷は負荷モニタ処理ハンドラ 118 によって調査する。このハンドラはコントローラ 202 が定期的に行い、内容を常に更新する。負荷モニタ処理ハンドラ 118 が起動すると、まずステップ 801 によりネットワークストレージ 211、212、213 のディスクの使用状況調査を行う。これにより、各ネットワークストレージのディスク使用率 $U\%$ を求める。次にステップ 802 によりネットワークストレージ 211、212、213 のネットワーク帯域使用状況調査を行う。これによって、各ネットワークストレージのネットワーク帯域使用率 $B\%$ を求める。次に、ステップ 803 で別途定めるディスク容量とネットワーク帯域のどちらを優先して評価するかを定めるパラメータ α を用いて、評価式 $\alpha \cdot U + (1 - \alpha) \cdot B$ により各ネットワークストレージの負荷を評価する。評価式の値が大きいほどネットワークストレージの負荷が大きいことを意味する。ただし、ここで α は正数とする。ここで評価した各ネットワークストレージの負荷は CREATE 要求処理で新たにファイルを作成する場合のネットワークストレージ選択基準として使用する。ネットワークストレージ 211、212

40

50

、213の負荷の最大値を $\max(*U + *B)$ 、最小値を $\min(*U + *B)$ とする。負荷の偏りの許容度パラメータを用いて、評価式 $\max(*U + *B) - \min(*U + *B) <$ を満たすように、CREATE要求でファイルを作成するネットワークストレージを選択する。

【0055】

ここで、「負荷の偏りの許容度」とは、システム内の構成要素にかかる負荷の最大と最小でどの程度の差を許容するかの度合いをいう。システム内の構成要素としては、例えばディスク容量などが考えられ、ディスク容量の使用状況の最大と最小でどの程度の差を許容するかを「負荷の偏りの許容度」と表現する。なお、負荷は容量に限らず、プロセッサ使用状況、ネットワーク帯域の使用状況など色々な負荷について考えることができる。許容度は種々の方法で表現可能だが、本実施例では、パラメータを用いて表現している。

10

【0056】

次にステップ804により特定ファイルへのアクセス状況を調査する。各ネットワークストレージについて、ネットワークストレージへのアクセス頻度FNを調べる。別途定めるネットワークストレージにどの程度のアクセス集中の許容度パラメータを用いて、各ネットワークストレージについて評価式 $FN >$ を評価し、この式が成り立つ場合には、そのネットワークストレージの個々のファイルのアクセス頻度FFを調査する。

【0057】

ここで、「アクセス集中の許容度」とは、システムが、どの程度のアクセス集中を許容するかの度合いを言う。ファイルサーバのプロセッサ使用状況、ネットワーク帯域の使用状況、ディスク容量、ディスクアクセスの頻度などが高くなる状態を「アクセス集中」と呼ぶが、このような状態ではシステムの性能が急激に悪化するため、ある程度以上アクセスが集中しないように制御する必要がある。許容度は種々の方法で表現可能だが、本実施例では、パラメータを用いて表現している。

20

【0058】

最後にステップ804でFFの大きなファイルからFNの小さなストレージに移動し、FNを再評価しFNを均等化する。

【0059】

これらにより、ファイル新規作成時の以外の場合、特にアクセス頻度の高い複数のファイルが同一のネットワークストレージに保存されていてホットスポットが発生した場合であっても、ホットスポットの原因となるファイルを移動することで、各ネットワークストレージ負荷を均等化する効果がある。

30

【0060】

動作モニターハンドラ116はネットワークストレージ211、212、213が正常に動作しているかどうかを監視する。あるネットワークストレージが動作していない場合には、特殊エントリ管理ハンドラ119を呼び出し、ネットワークストレージが動作していないことを伝えるための仮想的なファイルエントリを作成する。この特殊エントリはファイル名としてネットワークストレージの障害情報通知用の仮想ファイルと認識できる情報を含む。この仮想的なファイルエントリはネットワークストレージ211、212、213には存在しないことを除き、通常のファイルと同様にアクセスできる。

40

【0061】

これにより、コントローラ202を管理するための専用のコンソールなどを用いることなく、クライアントはネットワークストレージの障害を知ることができる効果がある。

【0062】

障害処理ハンドラ117は障害の発生していたネットワークストレージが復帰した時の整合性管理を行う。ここで、「整合性」とは、複数NASに存在する同一ディレクトリ名の下に同一ファイル名が複数していない状態を「整合性」とあると言う。また、同一ディレクトリ上に同一ファイル名が複数存在する状態（実際のファイルは異なるNAS上に作られる）を「不整合」とあると言う。不整合が発生する原因としては、NASに障害が発生している場合などが考えられる。正常動作しているNASの下に障害の発生したNAS上

50

に存在するのと同じのファイル名を持つファイルが作られる可能性があるためである。本実施の形態によるネットワークストレージ仮想化方法では、クライアントがCREATE要求によりファイルを作成する際に、同一ディレクトリ上に同一ファイル名が高々一つしかないことを保証する。しかしネットワークストレージに障害が発生しアクセス不能になった場合には、ネットワークストレージが障害から復帰した際に同一ファイル名が存在する、整合性が取れていない状態になる可能性がある。この問題を解決するため、障害処理ハンドラ117は以下のように動作する。

【0063】

障害から復帰したネットワークストレージとその他のネットワークストレージについて同一ディレクトリ上に同一ファイル名が存在しないか調査する。同一ファイル名が存在する場合には、障害から復帰したネットワークストレージ上のファイル名を元のファイル名と障害復帰情報を含むファイル名に変更する。障害復帰情報としては、例えばネットワークストレージ固有の情報と復帰時刻などがある。別の手法としては、障害から復帰したネットワークストレージ上にディレクトリ名に障害復帰情報を含んだディレクトリを作成し、そこにファイルを移動する手法がある。

10

【0064】

コントローラ202にネットワークストレージを増設する際には、仮想化済みのネットワークストレージ群との間でファイル名の重複ないかを調べる整合性調査が必要になるが、このような場合にも障害処理ハンドラ117がそのまま適用できる。

【0065】

これらにより、コントローラ202を管理するための専用のコンソールなどを用いることなく、クライアントは障害から復帰したネットワークストレージに発生した整合性エラーを知ることができる効果がある。

20

【0066】

特殊エントリ管理ハンドラ119は前述の障害情報通知用の仮想的なエントリ管理の他に、ネットワークストレージ211、212、213のなかのいずれか一台を指定してアクセスする必要がある場合の仮想的なディレクトリエントリを提供する。この特殊エントリはディレクトリ名として特定のネットワークへのアクセス用の仮想ディレクトリと認識できる情報を含む。この仮想ディレクトリへの各種アクセス要求は全て特定のネットワークストレージに転送する。この仮想的なディレクトリエントリはネットワークストレージ211、212、213には存在しないことを除き、通常のファイルと同様にアクセスできる。

30

【0067】

これにより、コントローラ202を管理するための専用のコンソールなどを用いることなく、クライアントは特定のネットワークストレージを指定してファイルを移動することなどができる効果がある。この機能は例えばコントローラ202から特定のネットワークストレージを取り外す際に必要になる。本実施の形態では、NFSプロトコルを仮定しているが、NFS以外のプロトコルについても容易に適用可能である。

【0068】

本実施の形態では、ファイルハンドルをネットワークストレージ固有の情報より符号化しているが、任意の符号化を利用できる。

40

【0069】

本実施の形態では、負荷の要因としてディスク容量とネットワーク帯域を挙げているが、これら以外の負荷要素、例えばネットワークストレージのプロセッサ負荷や内部ディスク帯域など、任意の負荷要素に付いても同様に適用できる。

【0070】**【発明の効果】**

本発明を実施した場合の効果としては、専用ではなく汎用のネットワークストレージを用いてネットワークストレージ仮想化が可能になる効果がある。

【0071】

50

また、本発明の範疇に含まれる種々の構成要件を付加することにより、以下のような効果も実現可能となる。例えば、集中管理型サーバや分散ディレクトリを用いずにネットワークストレージの仮想化が可能になる効果がある。

【0072】

また、ネットワークストレージの障害発生時に障害の影響が他のサーバに伝播せずサービスを続行できる効果がある。

【0073】

また、ネットワークストレージの仮想化を行うコントローラでファイルの存在場所の管理をしないため、コントローラで障害が発生した場合には、コントローラを交換するだけで障害から復帰できる効果がある。

【0074】

また、READ/WRITE要求処理の際にネットワークストレージ仮想化を行うコントローラが行うのはREAD/WRITE要求と応答の書き換えだけであり、ネットワークストレージが実際のREAD/WRITE要求を処理するため、コントローラへの負荷を低減できる効果がある。

【0075】

また、ファイルを新規に作成する際に、負荷の少ないネットワークストレージにCREATE要求を転送することにより、ネットワークストレージの負荷を均等化する効果がある。

【0076】

また、アクセス頻度の高いファイルが特定のネットワークストレージ上に保存されている場合、それらのファイルを他のネットワークストレージに移動することにより、ネットワークストレージの負荷を均等化する効果がある。

【0077】

また同時に、障害情報通知用の仮想的なファイルエントリを作成することにより、クライアントは専用のコンソールを用いずにネットワークストレージの障害を知ることができる効果がある。

【0078】

また、ネットワークストレージが障害から復帰する時にファイル名の整合性にエラーが発生した場合、障害のあったネットワークストレージ上のファイル名を障害復帰情報を含んだファイル名に変更することにより、クライアントは専用のコンソールを用いずにネットワークストレージに発生した整合性エラーを知ることができる効果がある。

【0079】

また、ネットワークストレージ特定のネットワークストレージへのアクセスを指定する仮想的なディレクトリエントリを作成することにより、クライアントは専用のコントローラを用いずに特定のネットワークストレージを指定してファイルを移動することなどができる効果がある。

【図面の簡単な説明】

【図1】本発明によるネットワークストレージ仮想化方法の実施の形態のアーキテクチャのブロック図。

【図2】本発明によるネットワークストレージ仮想化方法の実施の形態のネットワーク接続図

【図3】本発明によるネットワークストレージ仮想化方法の実施の形態のREADDIR処理ハンドラ111のフロー図。

【図4】本発明によるネットワークストレージ仮想化方法の実施の形態のLOOKUP処理ハンドラ112のフロー図。

【図5】本発明によるネットワークストレージ仮想化方法の実施の形態のREAD処理ハンドラ113のフロー図。

【図6】本発明によるネットワークストレージ仮想化方法の実施の形態のWRITE処理ハンドラ114のフロー図。

10

20

30

40

50

【図7】本発明によるネットワークストレージ仮想化方法の実施の形態のCREATE処理ハンドラ115のフロー図。

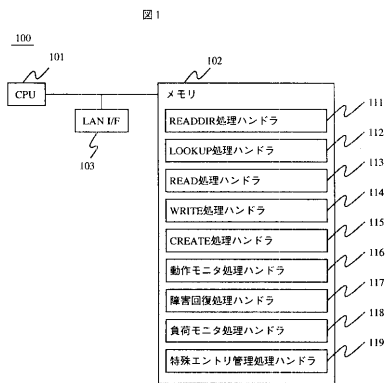
【図8】本発明によるネットワークストレージ仮想化方法の実施の形態の負荷モニタ処理ハンドラ118のフロー図。

【符号の説明】

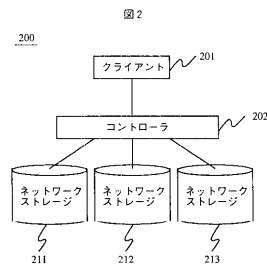
100	本発明によるネットワークストレージ仮想化方法の実施の形態のアーキテクチャ	
101	演算装置	
102	メモリ	
103	LAN接続インタフェース	
111	READDIR処理ハンドラ	10
112	LOOKUP処理ハンドラ	
113	READ処理ハンドラ	
114	WRITE処理ハンドラ	
115	CREATE処理ハンドラ	
116	動作モニタ処理ハンドラ	
117	障害処理ハンドラ	
118	負荷モニタ処理ハンドラ	
119	特殊エントリ管理処理ハンドラ	
200	本発明によるネットワークストレージ仮想化方法の実施の形態のネットワーク接続図	20
201	クライアント	
202	本発明のネットワークストレージ仮想化方法の実施の形態によるネットワークストレージ仮想化用コントローラ	
211	第1のネットワークストレージ	
212	第2のネットワークストレージ	
213	第3のネットワークストレージ	
301	クライアントからのREADDIR要求受信ステップ	
302	READDIR要求の各ネットワークストレージへの転送ステップ	
303	各ネットワークストレージからのREADDIR応答受信ステップ	
304	各ネットワークストレージからのREADDIR応答結合ステップ	30
305	クライアントへのREADDIR応答送信ステップ	
401	クライアントからのLOOKUP要求受信ステップ	
402	LOOKUP要求の各ネットワークストレージへの転送ステップ	
403	各ネットワークストレージからのLOOKUP応答受信ステップ	
404	クライアントへの正常なLOOKUP応答転送ステップ	
501	クライアントからのREAD要求受信ステップ	
502	READ要求の適切なネットワークストレージへの転送ステップ	
503	ネットワークストレージからのREAD応答受信ステップ	
504	クライアントへのREAD応答転送ステップ	
601	クライアントからのWRITE要求受信ステップ	40
602	WRITE要求の適切なネットワークストレージへの転送ステップ	
603	ネットワークストレージからのWRITE応答受信ステップ	
604	クライアントへのWRITE応答転送ステップ	
701	クライアントからのCREATE要求受信ステップ	
702	CREATE要求の適切なネットワークストレージへの転送ステップ	
703	ネットワークストレージからのCREATE応答受信ステップ	
704	クライアントへのCREATE応答転送ステップ	
801	ネットワークストレージのディスク容量調査ステップ	
802	ネットワークストレージのネットワーク帯域調査ステップ	
803	ネットワークストレージの負荷評価ステップ	50

- 804 特定ファイルへのアクセス状況調査ステップ
- 805 ファイルの再配置ステップ。

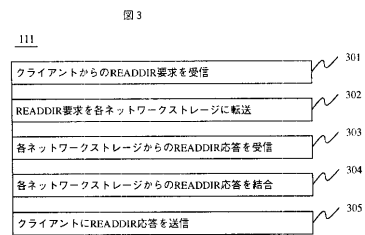
【図1】



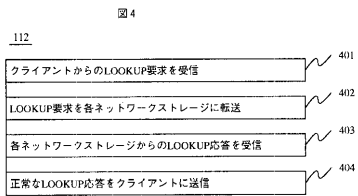
【図2】



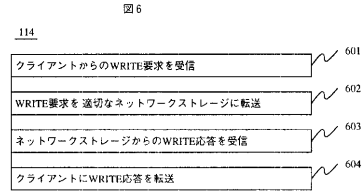
【図3】



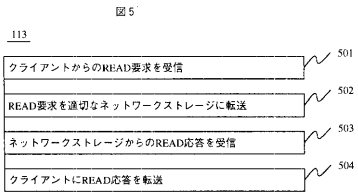
【 図 4 】



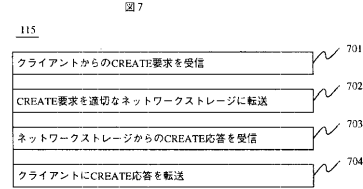
【 図 6 】



【 図 5 】



【 図 7 】



【 図 8 】

