

(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2004-355638

(P2004-355638A)

(43) 公開日 平成16年12月16日(2004.12.16)

(51) Int. Cl. ⁷	F I	テーマコード (参考)
G06F 9/46	G06F 9/46 360C	5B005
G06F 3/06	G06F 3/06 301A	5B014
G06F 12/00	G06F 3/06 301Z	5B045
G06F 12/08	G06F 12/00 514E	5B065
G06F 13/10	G06F 12/00 545A	5B082
審査請求 未請求 請求項の数 16 O L (全 27 頁) 最終頁に続く		

(21) 出願番号 特願2004-184406 (P2004-184406)
 (22) 出願日 平成16年6月23日 (2004.6.23)
 (62) 分割の表示 特願2000-238865 (P2000-238865) の分割
 原出願日 平成12年8月2日 (2000.8.2)
 (31) 優先権主張番号 特願平11-241024
 (32) 優先日 平成11年8月27日 (1999.8.27)
 (33) 優先権主張国 日本国 (JP)

(71) 出願人 000005108
 株式会社日立製作所
 東京都千代田区丸の内一丁目6番6号
 (74) 代理人 100075096
 弁理士 作田 康夫
 (72) 発明者 北村 学
 神奈川県川崎市麻生区王禅寺1099番地
 株式会社日立製作所システム開発研究所 内
 (72) 発明者 村上 達也
 神奈川県横浜市戸塚区戸塚町5030番地
 株式会社日立製作所ソフトウェア事業部 内

最終頁に続く

(54) 【発明の名称】 計算機システム及びそのデバイスの割り当て方法

(57) 【要約】

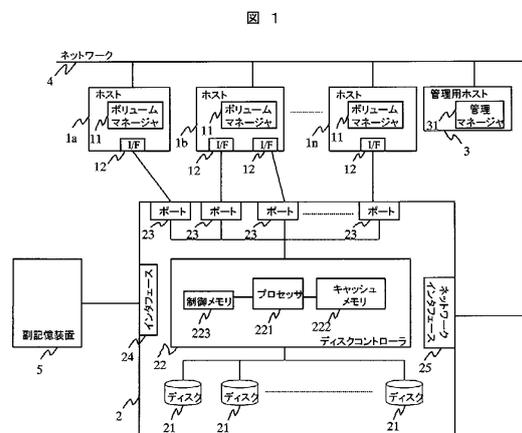
【課題】

複数の計算機が用途に合ったデバイスを必要に応じて利用できるように、デバイスの設定、各計算機への割付を動的に行う。

【解決手段】

記憶装置サブシステム3は、種々の構成、容量を持つ複数のデバイスを有し、複数の計算機1に接続される。管理計算機3は、計算機1と記憶装置サブシステム2の接続関係を記憶・管理し、計算機1のボリュームマネージャ11と通信しながらデバイスの管理を行う。計算機1が新たにデバイスを要求する際、管理マネージャ31にデバイスの数、容量、種類などの情報を通知する。管理マネージャ31は、要求に合ったデバイスを選択し、計算機1からそのデバイスを利用できるように記憶装置サブシステムに設定を指示する。管理マネージャ31は、その後、計算機1に新たなデバイスのアクセスに必要な情報を通知し、ボリュームマネージャ11は、その通知に従って計算機1の設定を変更する。

【選択図】 図1



【特許請求の範囲】**【請求項 1】**

第 1 の計算機と、

前記第 1 の計算機からアクセスされるデータを保持する記憶デバイスを有する記憶装置サブシステムと、

前記記憶装置サブシステムが有する前記記憶デバイスに関するデバイス管理情報と、前記記憶デバイスの前記第 1 の計算機への割り当ての状態を示すホスト管理情報とを有する第 2 の計算機とを有する計算機システムであって、

前記第 1 の計算機は、ユーザまたはアプリケーションプログラムからの新たな記憶デバイスの要求を受け付け、前記第 2 の計算機に新たな記憶デバイスの割り当てを要求する要求手段を有し、

前記第 2 の計算機は、前記要求手段からの要求に応じて、前記デバイス管理情報及び前記ホスト管理情報を参照して前記第 1 の計算機に割り当て可能な記憶デバイスを決定する手

段、及び該決定手段により決定された記憶デバイスを前記第 1 の計算機からアクセス可能となるように前記記憶装置サブシステムの設定を変更する変更手段とを有することを特徴とする計算機システム。

【請求項 2】

前記記憶デバイスは、前記記憶装置サブシステムが備える物理的な記憶装置に形成される少なくとも一部の記憶領域であることを特徴とする請求項 1 記載の計算機システム。

【請求項 3】

前記要求手段は、前記割り当ての要求とともに、割り当てられる記憶デバイスの条件を指定する情報を前記第 2 の計算機に送ることを特徴とする請求項 1 記載の計算機システム。

【請求項 4】

前記決定手段は、前記デバイス管理情報を参照し、前記要求手段により指定された前記条件を満たす記憶デバイスであって、オフライン状態にある記憶デバイスを前記割り当て可能な記憶デバイスとして選択することを特徴とする請求項 3 記載の計算機システム

【請求項 5】

前記条件は、要求される記憶デバイスの性能及び信頼性の少なくとも一方を指定するための情報を含むことを特徴とする請求項 4 記載の計算機システム。

【請求項 6】

前記要求手段は、既存の記憶デバイスの容量の変更の要求を受け付け、該要求を前記第 2 の計算機に送信する手段を有し、

前記第 2 の計算機は、前記変更の要求要求に応じて、容量の変更後における記憶デバイスに合致する記憶デバイスを選択する手段と、前記記憶装置サブシステムに対し、前記既存の記憶デバイスに保持されているデータを前記選択手段により選択された記憶デバイスに移動するよう指示する手段とを有し、

前記記憶装置サブシステムは、前記指示にตอบสนองしてデータを移動する手段を有することを特徴とする請求項 1 記載の計算機システム。

【請求項 7】

前記第 1 の計算機は、複数の記憶デバイスを論理的に一つのデバイスとして前記アプリケーションプログラムに提供する手段を有することを特徴とする請求項 1 記載の計算機システム。

【請求項 8】

前記要求手段は、すでに前記アプリケーションプログラムに提供しているデバイスの容量の拡張の要求に応じて、前記第 2 の計算機に前記拡張の要求で要求された拡張に必要な容量の新たな記憶デバイスの割り当てを要求し、

前記提供する手段は、新たに割り当てられた記憶デバイスを前記拡張が要求されたデバイスの一部とすることを特徴とする請求項 7 記載の計算機システム。

10

20

30

40

50

【請求項 9】

第 1 の計算機、

前記第 1 の計算機からアクセスされるデータを保持する記憶デバイスを有する記憶装置サブシステム、

及び前記記憶装置サブシステムが有する前記記憶デバイスの前記第 1 の計算機への割り当ての状態を管理する第 2 の計算機とを有する計算機システムにおける記憶デバイスの割り当て方法であって、

前記第 1 の計算機から前記第 2 の計算機に新たな記憶デバイスの割り当てを要求し、

前記第 2 の計算機において、前記要求に基づいて前記第 1 の計算機に割り当て可能な記憶デバイスを決定し、

前記第 2 の計算機から前記記憶装置サブシステムに対して決定された記憶デバイスを前記第 1 の計算機からアクセス可能となるように設定を変更するよう前記記憶装置サブシステムに指示し、

前記第 2 の計算機から前記第 1 の計算機に前記決定された記憶デバイスのアクセスに必要な情報を転送し、

前記第 1 の計算機において、前記アクセスに必要な情報に基づいて前記決定された記憶デバイスを利用できるように、前記第 1 の計算機の設定を変更することを特徴とする記憶デバイス割り当て方法。

【請求項 10】

前記割り当ての要求ステップは、前記新たな記憶デバイスに要求される条件を示す情報を含む要求を前記第 2 の計算機に送信するステップを含むことを特徴とする請求項 9 記載の記憶デバイス割り当て方法。

【請求項 11】

前記要求される条件は、前記新たな記憶デバイスの性能及び信頼性の少なくとも一方を指定するための情報を含むことを特徴とする請求項 10 記載の記憶デバイス割り当て方法。

【請求項 12】

前記決定するステップは、前記記憶装置サブシステム内の記憶デバイスを管理するデバイス管理情報を参照し、前記条件を満たす記憶デバイスであってオフライン状態にある記憶デバイスを前記割り当て可能な記憶デバイスとして決定するステップを含むことを特徴とする請求項 11 記載の記憶デバイス割り当て方法。

【請求項 13】

請求項 9 記載の記憶デバイス割り当て方法において、さらに、前記第 1 の計算機において、前記決定された記憶デバイスと、前記第 1 の計算機において使用されている論理的なストレージを構成する既存の記憶デバイスとを結合し、前記論理的なストレージの容量を拡張することを特徴とする記憶デバイス割り当て方法。

【請求項 14】

計算機と、

該計算機からアクセスされるデータを保持する記憶デバイスを有する記憶装置サブシステムとを有する計算機システムにおいて、

前記計算機は、ユーザまたはアプリケーションプログラムからの新たな記憶デバイスの要求を受け付け、前記記憶装置サブシステムに新たな記憶デバイスの割り当てを要求する要求手段を有し、

前記記憶装置サブシステムは、前記要求手段からの要求に応じて、前記デバイス管理情報及び前記ホスト管理情報を参照して前記第 1 の計算機に割り当て可能な記憶デバイスを決定する手段、及び

該決定手段により決定された記憶デバイスを前記第 1 の計算機からアクセス可能となるように前記記憶装置サブシステムの設定を変更する変更手段とを有することを特徴とする計算機システム。

【請求項 15】

複数の計算機と、

該複数の計算機がネットワークを介して接続されるファイルサーバと、

ファイルサーバに接続され、複数の記憶デバイスを備える記憶装置サブシステムとを有する計算機システムにおいて、

前記複数の計算機は、ユーザまたはアプリケーションプログラムからの新たなファイル領域の要求を受け付け、前記第2の計算機に新たなファイル領域の割り当てを要求する要求手段を有し、

前記ファイルサーバは、前記要求手段からの要求に応じて、前記デバイス管理情報及び前記ホスト管理情報を参照して前記ファイル領域を設定可能な記憶領域を有する記憶デバイスを決定する手段、

10

該決定手段により決定された記憶デバイスを前記ファイルサーバからアクセス可能となるように前記記憶装置サブシステムの設定を変更する変更手段、

前記決定された記憶デバイスを利用できるように、前記ファイルサーバの設定を変更する設定手段、

前記決定された記憶デバイス上に前記ファイル領域を作成するファイル管理手段を有することを特徴とする計算機システム。

【請求項16】

前記ファイル管理手段は、作成したファイル領域のアクセスに使われる情報を、前記要求を行った計算機に送信する手段を備え、

前記要求を行った計算機は、前記アクセスに使われる情報を利用して前記作成されたファイル領域へのアクセスを行うアプリケーションプログラムを有することを特徴とする請求項16記載の計算機システム。

20

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、計算機システム、及び、計算機システムにおける計算機への記憶デバイスの割り当て方法に係り、特に、複数の計算機により共用される記憶装置サブシステムを備えた計算機システムにおいて、計算機に対する記憶デバイスの割り当てを行うための方法に関する。

30

【背景技術】

【0002】

近年、企業などで利用される計算機システムで扱われる情報量は、飛躍的に増大しており、これに伴いデータを記憶するディスク装置などの容量も増加の一途をたどっている。例えば、磁気ディスク装置においては、数TB（テラバイト）の容量を持つ装置も珍しくなくなっている。このようなディスク装置に関して、例えば、特許文献1には、1台の記憶装置サブシステムを複数種類の論理的なディスク装置（以下では、デバイスとも呼ぶ）で構成する技術が開示されている。具体的には、ホストコンピュータからアクセスされるデバイス（論理ディスク）として、RAID（Redundant Arrays of Inexpensive Disks）におけるRAID5とRAID1のように、RAIDレベルの異なるものを混在させ、あるいは、論理ディスクを構成するための実際の磁気ディスク装置（物理ディスク装置）として、アクセス速度の異なるものを混在させたディスクサブシステムが開示されている。ユーザは、各デバイスのアクセス頻度に応じて、デバイスを使い分けることができる。

40

【0003】

一方、ホスト計算機とディスク装置などの周辺装置との間のインタフェースとして、ファイバチャネル技術が現れたことにより、複数のホスト計算機、複数の記憶装置を1つのファイバチャネルケーブルで接続して計算機システムを構成することも行われるようになってきた。このような計算機システムでは、それぞれのホスト計算機は、ファイバチャネル上の任意の記憶装置に直接アクセスできる。このため、従来のように、各ホスト計算機が

50

それぞれに記憶装置を持つ場合に比べ、ホスト計算機間でのデータの共有やネットワーク負荷の低減が期待できる。

【0004】

【特許文献1】特開平9-274544号公報

【発明の開示】

【発明が解決しようとする課題】

【0005】

上述した従来技術によれば、各ホスト計算機がアクセスできるデバイスの数や種類を飛躍的に増加させることができる。しかし、ホスト計算機がアクセスできるデバイスの数や種類が増加するに伴い、各ホスト計算機でのデバイスの管理が難しくなってくる。1つのホスト計算機から多くのデバイスにアクセスできる利点の一方で、ユーザにとっては、ある業務でどのデバイスを使用すれば良いか選択するといったことが困難となる。特に、ファイバチャネルで接続された計算機システムの場合、あるホスト計算機からそのホスト計算機が本来使用していないデバイスにまでアクセスできることとなる。このため、他のホスト計算機が使用しているデバイスに不正アクセスを起し、データを破壊してしまうといったことが生じ得る。

10

【0006】

特開平10-333839号公報には、このような問題を解決するため、ファイバチャネルで接続された記憶装置が、特定のホスト計算機からのアクセスだけを許可する方法が開示されている。しかし、記憶装置、デバイスが複数になった場合や、異なる種類のデバイスが混在する場合には、処理が複雑であることには変わりはなく、各ホスト計算機で、それぞれデバイスの種類などを常に意識する必要がある。

20

【0007】

本発明の目的は、各ホスト計算機が用途に合ったデバイスを必要な時に必要な分だけ利用できるように、デバイスの設定、デバイスの各ホスト計算機への割り当てを容易に行えるようにすることにある。

【課題を解決するための手段】

【0008】

本発明における計算機システムは、その好ましい一つの態様において、複数の計算機と、これら複数の計算機に接続する記憶装置サブシステムとを有する。記憶装置サブシステムは、複数の記憶デバイスと複数のインタフェースとを有し、各計算機に接続される。複数の計算機の1つは、記憶装置サブシステム内の記憶デバイスと、各計算機と記憶装置サブシステムの接続関係の情報を保持する管理手段を有する。各計算機は、新たにデバイスが必要となき、管理手段にその容量や種類を通知する。管理手段は、その通知を受けて、要求に合った記憶デバイスを選択する。そして選択したデバイスが計算機からアクセスできるように、記憶装置サブシステムに対し所定の情報を設定するように指示する。管理手段は、また、デバイス割り当ての要求のあった計算機に所定の情報を返却し、要求のあった計算機では、その情報に基づき計算機の設定を変更し、当該計算機で割り当てられたデバイスが使用できるようにする。

30

【0009】

本発明の他の態様において、複数の計算機と複数の記憶装置サブシステムがネットワークを介して接続される。任意の計算機上には、各記憶装置サブシステムが有する記憶デバイスと、各計算機と記憶装置サブシステムの接続関係の情報を保持する管理手段が設けられる。各記憶装置サブシステムは、管理手段から指定された計算機に対してアクセスを許可するための制御手段を有する。各計算機は、新たな記憶デバイスを必要とするとき、管理手段にその容量や種類を通知する。管理手段は、通知に応じて要求に合ったデバイスを選択し、選択したデバイスが新たな記憶デバイスを必要とする計算機からアクセスできるように、記憶装置サブシステムに対し当該計算機からのアクセスを許可するよう指示する。管理手段は、また、デバイス割り当てを要求した計算機に所定の情報を返却する。デバイス割り当てを要求した計算機では、管理手段から返却された情報に基づき、計算機の設定

40

50

定を変更し、その計算機から割り当てられたデバイスが使用できるようにする。

【発明の効果】

【0010】

本発明によれば、ホスト計算機に対する記憶デバイスの割り当てを必要に応じて動的に行うことができる。

【発明を実施するための最良の形態】

【0011】

図1は、本発明が適用された計算機システムの一実施形態における構成例を示す簡略なブロック図である。

【0012】

計算機システムは、複数のホスト計算機、ホスト計算機1a、ホスト計算機1b、...ホスト計算機1n（総称してホスト1と呼ぶ）、ホスト1に接続される記憶装置サブシステム2、管理用ホスト計算機3、ネットワーク4、及び遠隔地に配置される記憶装置である副記憶装置5を有して構成される。

【0013】

ホスト計算機1a、1b、...は、CPU、メモリなどを有する計算機であり、メモリに格納されたオペレーティングシステム、アプリケーションプログラムをCPUが読み出して実行することで、所定の機能を達成する。

【0014】

記憶装置サブシステム2は、複数のディスクユニット21、ディスクコントローラ22、ホスト計算機1と接続する複数のポート23、副記憶装置5と接続するためのインタフェース24、ネットワーク4と接続するネットワークインタフェース25を有している。本実施形態における記憶装置サブシステム2は、複数のディスクユニット21をまとめて1または複数の論理デバイスとしてホスト計算機1に見せかける。もちろん、個々のディスクユニット21を1つの論理デバイスとしてホスト計算機1に見せるようにしても良い。

【0015】

ポート23としては、たとえば、接続されるホスト計算機1が、いわゆるオープンシステムの計算機であればSCSI (Small Computer System Interface) などのインタフェースが用いられる。一方、ホスト計算機1が、いわゆるメインフレームであれば、ESCON (Enterprise System CONnection) などのチャネルインタフェースが用いられる。それぞれのポート23は、同一のインタフェースであっても異なるものが混在していてもかまわない。本実施形態では、全てのポート23にインタフェースとして、SCSIを用いるものとして説明する。

【0016】

ディスクコントローラ22は、プロセッサ221、キャッシュメモリ222、制御メモリ223を有している。プロセッサ221は、ホスト計算機1からのアクセスやディスクユニット21の制御を行う。プロセッサ221は、特に、記憶装置サブシステム2がホスト計算機1に対してディスクユニット21単体ではなく、ディスクアレイの様に複数のディスクユニットをまとめて1または複数の論理デバイスに見せかけている場合には、その処理や管理などを行う。ディスクコントローラ22は、ネットワークインタフェース25を介して管理用ホスト計算機3との通信を行う。

【0017】

キャッシュメモリ222は、ホスト計算機1からのアクセス処理速度を高めるため、頻りに読み出されるデータを格納したり、あるいはホスト計算機1からのライトデータを一時的に格納したりする。キャッシュメモリ222の一部を、1または複数の論理的なディスクに見せかけ、磁気ディスクユニットへのアクセスが不要なデバイスとして利用することもできる。

【0018】

制御メモリ223は、プロセッサ221が実行するプログラムを格納し、また、ディス

10

20

30

40

50

クユニット 2 1 や、複数のディスクユニット 2 1 を組み合わせる構成される論理デバイスの管理を行うための情報を格納するために使用される。

【 0 0 1 9 】

各ホスト計算機 1 a、1 b ... には、ボリュームマネージャ 1 1 と呼ばれるソフトウェア（プログラム）が配置される。ボリュームマネージャ 1 1 は、管理用ホスト計算機 3 に配置される管理マネージャ 3 1 と通信しあって動作する。各ホスト計算機 1 は、インタフェース（I/F）1 2 を持っており、インタフェース 1 2 により記憶装置サブシステム 2 のポート 2 3 と接続される。

【 0 0 2 0 】

次に、記憶装置サブシステム 2 内の論理デバイスの管理形態について説明する。

10

【 0 0 2 1 】

前述の通り、記憶装置サブシステム 2 は、複数のディスクユニット 2 1 を 1 または複数の論理デバイスとして、あるいは、個々のディスクユニット 2 1 を 1 つの論理デバイスとしてホスト計算機 1 に見せかける。記憶装置サブシステム 2 は、また、キャッシュメモリ 2 2 2 の一部の領域を 1 または複数の論理デバイスとしてホスト計算機 1 に見せる。記憶装置サブシステム 2 内のディスクユニット 2 1 の数と論理デバイスの数に相関関係はない。

【 0 0 2 2 】

図 2 は、記憶装置サブシステム 2 が論理デバイスを管理するために使用する情報を保持する論理デバイス管理テーブルの一例を示すテーブル構成図である。

20

【 0 0 2 3 】

論理デバイス管理テーブルは、論理デバイス番号 6 1 に対し、サイズ 6 2、構成 6 3、状態 6 4、パス 6 5、ターゲット ID 6 6、LUN 6 7 の項目の組を保持する。サイズ 6 2 には、論理デバイス番号 6 1 により特定される論理デバイスの容量を表す情報が設定される。

【 0 0 2 4 】

構成 6 3 には、当該論理デバイスの構成を示す情報、たとえばディスクユニット 2 1 により RAID (Redundant Arrays of Inexpensive Disks) が構成され、それが論理デバイスに割り当てられている場合、RAID 1、RAID 5 など、RAID の種別を示す情報が設定される。また、当該論理デバイスとして、キャッシュメモリ 2 2 2 の一部の領域が割り当てられていれば「キャッシュ」、単体のディスクユニットが割り当てられている場合には「単体ディスクユニット」を示す情報が構成 6 3 に設定される。

30

【 0 0 2 5 】

状態 6 4 には、当該論理デバイスの状態を示す情報が設定される。状態としては、「オンライン」、「オフライン」、「未実装」、「障害オフライン」が存在する。「オンライン」は、当該論理デバイスが正常に稼動し、ホスト計算機 1 からのアクセスが可能な状態であることを示す。「オフライン」は、当該論理デバイスは定義され、正常に稼動しているが、ホスト計算機 1 からのアクセスはできない状態にあることを示す。この状態は、以前ホスト計算機 1 で使用されていたが、ホスト計算機 1 でそのデバイスが不要となって使われなくなった場合に相当する。「未実装」は、当該論理デバイスが定義されておらずホストからのアクセスはできない状態にあることを示す。また「障害オフライン」は、当該論理デバイスに障害が発生してホストからのアクセスができないことを示す。

40

【 0 0 2 6 】

パス 6 5 には、当該論理デバイスが複数のポート 2 3 のどのポートに接続されているかを表す情報が設定される。各ポート 2 3 には、記憶装置サブシステム 2 内で一意な番号が割り振られており、「パス」欄には論理デバイスが接続されているポート 2 3 の番号が記録される。ターゲット ID 6 6 と LUN 6 7 は、論理デバイスを識別するための識別子である。ここでは、これらの識別子として、SCSI 上でホスト計算機 1 からデバイスをアクセスする場合に用いられる SCSI-ID、LUN が用いられる。

【 0 0 2 7 】

50

1つの論理デバイスを複数のポートに接続し、複数のホスト計算機1から同一の論理デバイスをアクセス可能とすることができる。この場合、論理デバイス管理テーブルには、当該論理デバイスに関する複数のエントリが作成される。例えば、図2に示す論理デバイス管理テーブルでは、論理デバイス番号2番のデバイスはポート番号0、及び1の2つのポート23に接続されている。このため、論理デバイス番号2番の項目が2つ存在している。1つの論理デバイスがこのように複数のポート23からアクセス可能とされる場合、それぞれのパス65に対応するターゲットID、LUNは同一である必要はなく、図2に示されるように異なっても良い。

【0028】

論理デバイス管理テーブルに保持された情報は、ネットワークインタフェース24を通じて、適当なタイミングで、あるいは記憶装置サブシステム2に障害が発生して構成が変化した、などの際に管理用ホスト計算機3に送られる。このため、管理用ホスト計算機3も、図2に示すテーブルと同様の論理デバイス管理テーブルを保持する。

10

【0029】

図3は、管理用ホスト計算機3の管理マネージャ31により保持されるホスト管理テーブルの一例を示すテーブル構成図である。

【0030】

ホスト管理テーブルは、管理用ホスト計算機3が各ホスト計算機1に対するデバイスの割り当てを管理するために、ホスト名71、ポート番号72、インタフェース番号73、論理デバイス番号74の組で構成される管理情報を保持する。

20

【0031】

ポート番号72と論理デバイス番号74は、記憶装置サブシステム2の内部で定義される番号で、記憶装置サブシステム2の各ポート23及び論理デバイスを識別するための情報である。ポート番号72と論理デバイス番号74には、ホスト名71に設定された識別子で識別されるホスト計算機1が接続されるポートのポート番号と、そのホスト計算機に割り当てられている論理デバイスのデバイス番号が設定される。

【0032】

インタフェース番号73は、各ホスト計算機1のインタフェース12を管理するために付けられた番号である。インタフェース番号73は、特に、1つのホスト計算機1が複数のインタフェース12を持つ場合に必要となる。ポート番号72とインタフェース番号73の組は、ホスト計算機1と論理デバイスの接続関係を示すために重要な要素である。例えば、図1に示すホスト計算機1bは、2つのインタフェース12を備え、各インタフェース12は、それぞれ異なるポート23に接続されている。このような場合、一方のインタフェース、あるいは、一方のインタフェースと記憶装置サブシステム2を接続するラインの障害などにより使えなくなっても、他方のインタフェースから論理デバイスへの接続がなされていれば処理を続行でき、信頼性を高めることができる。

30

【0033】

管理用ホスト計算機3は、ホスト管理テーブルと、記憶装置サブシステム2から送られる論理デバイス管理テーブルを参照して、各ホスト計算機1に論理デバイスを割り当てる。以下、デバイスの割り当ての処理について説明する。

40

【0034】

図4は、各ホスト計算機1のボリュームマネージャ11により実施される処理の流れを示すフローチャートである。この処理は、ホスト計算機1を使用するユーザ、またはホスト計算機1で稼動するアプリケーションプログラムなどが新規にデバイスを必要とするときに実施される。

【0035】

ステップ1001で、ボリュームマネージャ11は、ユーザ、またはアプリケーションプログラムから必要とされるデバイスの数とデバイスの種類の情報を得る。ユーザ、またはアプリケーションプログラムは、デバイスに関する情報として、その容量、性能条件、信頼性レベルなどの情報を指定する。デバイスの容量とは、先に説明したデバイスのサイ

50

ズのことである。性能条件としては、例えば、低速ディスクドライブ、高速ディスクドライブ、キャッシュ常駐ディスクドライブといった、デバイスのアクセス速度などの性能に関する情報が指定される。信頼性レベルとしては、例えば、RAID0、RAID1、RAID5、二重パス、リモートミラーなど、デバイスの信頼性に関する情報が指定される。二重パスでは、ホスト計算機1が複数のインタフェースを持つ場合に、それら複数のインタフェースを利用して同一デバイスにアクセスできるよう複数のパスが設けられる。二重パスでは、あるパスが利用できなくなった時でも、他のパスを使ってそのデバイスにアクセスすることが可能となる。リモートミラーは、副記憶装置5に記憶装置サブシステム2内のデバイスのコピーを持たせるものであり、地震、火災などの要因で記憶装置サブシステム2そのものが稼働できなくなった場合にも、副記憶装置5にデータが保持されているため、信頼性を高めることができる。

10

【0036】

次に、ステップ1002でボリュームマネージャ11は、当該ホスト計算機1のインタフェース12上で使用されていないターゲットID、LUNの組を検索する。

【0037】

ステップ1003でボリュームマネージャ11は、ステップ1001において指定された容量、性能条件、信頼性レベル、及びステップ1002で検索された未使用のターゲットID、LUNの組を管理用ホスト計算機3の管理マネージャ31に送信し、新たなデバイスの割付を要求する。管理マネージャ31は、受け取った情報に基づいて割り当てるべきデバイスを検索し、そのデバイスのアクセスに使用するホストインタフェース番号、ターゲットID、及びLUNを指定する情報を返却する。ここで行われる管理マネージャ31の処理については後述する。

20

【0038】

ステップ1004でボリュームマネージャ11は、管理マネージャ13からの情報を受け取る。ステップ1005では、管理マネージャ13から受け取った情報をもとに、新しいデバイスを使用できるようにホスト計算機1の設定変更が行われる。

【0039】

いわゆるオープン系のオペレーティングシステムの場合、ホスト計算機1が各デバイスにアクセスするために、デバイスファイルがデバイス毎に用意され、デバイスファイルに対してアクセスが実施される。通常、デバイスファイルは、ホスト計算機1のデバイスコンフィギュレーション処理を行った際に用意され、デバイスコンフィギュレーション処理時に存在しないデバイスについては、デバイスファイルが作成されていない。このため、ステップ1004では、新しく割り当てられたデバイスに関するデバイスファイルが作成される。具体的には、例えば、サンマイクロシステムズ社のSolarisオペレーティングシステムでは、“drvconfig”コマンド、あるいは“drives”コマンドにより、新規デバイスの認識、デバイスファイルの作成が行われ、ホスト計算機1から新たに割り当てられたデバイスに対してアクセスできるようになる。

30

【0040】

最後にステップ1006でボリュームマネージャ11は、割り当てられたデバイスファイル名、ターゲットID、LUNの情報をユーザまたはアプリケーションプログラムに通知して処理を終了する。

40

【0041】

図5は、新しいデバイスの割り当て時に管理用ホスト計算機3の管理マネージャ31による処理の流れを示すフローチャートである。

【0042】

管理マネージャ31は、ステップ1101でホスト計算機1から送られてきたデバイスサイズ、性能条件、信頼度レベルなどの情報を受け取ると、保持している論理デバイス管理テーブル、及びホスト管理テーブルに設定されている情報を参照し、要求に応じたデバイスが存在するかを検索する。ここで検索対象となるデバイスは、論理デバイス管理テーブルの状態64に「オフライン」が設定されているものである(ステップ1102)。管理

50

マネージャ 3 1 は、検索の結果、要求に合った「オフライン」状態のデバイスが見つかったかどうか判別する（ステップ 1 1 0 3）。

【 0 0 4 3 】

要求に合った「オフライン」状態のデバイスが見つかった場合、管理マネージャ 3 1 は、ホスト計算機 1 から受け取ったターゲット ID、LUN の情報と、論理デバイス管理テーブル及びホスト管理テーブルに設定されている情報に基づいて、当該デバイスをホスト計算機 1 に接続するために使用するポート番号、ターゲット ID、LUN を決定する（ステップ 1 1 0 4）。

【 0 0 4 4 】

次に、管理マネージャ 3 1 は、ステップ 1 1 0 3 で見つけた論理デバイス番号のデバイスを、ステップ 1 1 0 4 で決定したポート番号、ターゲット ID、LUN でアクセスできるように設定してオンライン状態にするよう記憶装置サブシステム 2 に指示する。記憶装置サブシステム 2 は、管理マネージャ 3 1 からの指示に従って設定を行い、その結果を管理マネージャ 3 1 に返却する（ステップ 1 1 0 5）。

【 0 0 4 5 】

管理マネージャ 3 1 は、記憶装置サブシステム 2 から結果を受け取ると（ステップ 1 1 0 6）、要求のあったホスト計算機 1 のボリュームマネージャ 1 1 に対して、インタフェース番号、ターゲット ID、LUN を返却する（ステップ 1 1 0 7）。

【 0 0 4 6 】

一方、ステップ 1 1 0 3 で「オフライン」状態のデバイスで要求に合ったものが存在しなかった場合、管理マネージャ 3 1 は、論理デバイス管理テーブルの状態 6 4 が「未実装」の論理デバイス番号が存在するか検索する（ステップ 1 1 0 8）。「未実装」の論理デバイス番号が存在する場合、管理マネージャ 3 1 は、記憶装置サブシステム 2 に対して、ホスト計算機 1 から要求のあったデバイスサイズ、性能条件、信頼度レベルなどの情報を伝えてデバイスの構築を要求する。記憶装置サブシステム 2 は、管理マネージャ 3 1 から要求に合わせて当該デバイス番号のデバイスを構築し、管理マネージャ 3 1 にその結果を返却する（ステップ 1 1 0 9）。管理マネージャ 3 1 は、その結果を受け取ると、上述したステップ 1 1 0 4 以降の処理を実施する（ステップ 1 1 1 0）。

【 0 0 4 7 】

図 6 は、ホスト計算機 1 で不必要になったデバイスの返却処理においてボリュームマネージャ 1 1 が実施する処理の流れを示すフローチャートである。

【 0 0 4 8 】

デバイスの返却処理において、ボリュームマネージャ 1 1 は、まず、ユーザ、あるいは上位アプリケーションプログラムから不必要になったデバイスの情報、例えば、デバイスファイル名を受け取る（ステップ 1 2 0 1）。ボリュームマネージャ 1 1 は、受け取った情報をもとに、返却処理の対象となるデバイスに関連するインタフェース番号、ターゲット ID、LUN を取得する（ステップ 1 2 0 2）。

【 0 0 4 9 】

次に、ボリュームマネージャ 1 1 は、ホスト計算機 1 でそのデバイスを使わないようにするため、必要に応じてホスト計算機 1 の設定変更を行う。ここでは、具体的には、デバイスファイルの削除などの処理が行われる（ステップ 1 2 0 3）。続いて、ボリュームマネージャ 1 1 は、ステップ 1 2 0 2 で取得したインタフェース番号、ターゲット ID、LUN を管理マネージャ 3 1 に通知し、処理を終了する（ステップ 1 2 0 4）。

【 0 0 5 0 】

図 7 は、ホスト計算機 1 で不必要になったデバイスの返却処理において管理マネージャ 3 1 が実施する処理の流れを示すフローチャートである。

【 0 0 5 1 】

管理マネージャ 3 1 は、ホスト計算機 1 からインタフェース番号、ターゲット ID、LUN を受け取る（ステップ 1 3 0 1）。管理マネージャ 3 1 は、受け取ったインタフェース番号、ターゲット ID、LUN に基づいて、記憶装置サブシステム 2 に対し、返却の対

象とされているデバイスをオフラインにするよう指示する。この指示に応じて、記憶装置サブシステム 2 は、指定されたデバイスをオフラインとし、その結果を反映した論理デバイス管理テーブルを管理マネージャ 3 1 に返却する（ステップ 1 3 0 2）。記憶装置サブシステム 2 から、論理デバイス管理テーブルを受け取ると、それを保持して処理を完了する（ステップ 1 3 0 3）。

【0052】

上述した第 1 の実施形態では、管理用ホスト計算機を設け、そこに管理マネージャを配置しているが、管理マネージャの機能は必ずしも管理用ホスト計算機に存在する必要はない。例えば、ホスト計算機 1 a、1 b、... のいずれかに存在するように構成することも可能である。また、管理マネージャの機能を記憶装置サブシステムに設けることもできる。この場合、各ホスト計算機 1 a、1 b、... は、インタフェースを介して直接記憶装置サブシステムとの間で、要求の送付、情報受け取りを行えばよい。

10

【0053】

図 8 は、本発明の第 2 の実施形態における計算機システムの構成を示す簡略なブロック図である。

【0054】

本実施形態における計算機システムは、複数のホスト計算機 1（ホスト計算機 1 a、ホスト計算機 1 b、...、ホスト計算機 1 n）と、複数の記憶装置サブシステム 2 a、2 b、...、2 m、管理用ホスト計算機 3、ネットワーク 4、及びファイバチャネルスイッチ 6 を有して構成される。

20

【0055】

ホスト計算機 1 は、第 1 の実施形態と同じく、ボリュームマネージャ 1 1 を有する。ボリュームマネージャ 1 1 は、管理用ホスト計算機 3 におかれた管理マネージャ 3 1 と通信しあって動作する。さらに、ホスト計算機 1 は、インタフェース（I/F）1 2 を有し、インタフェース 1 2 によりファイバチャネルスイッチ 8 と接続される。

【0056】

記憶装置サブシステム 2 a、2 b、...、2 m は、それぞれ、第 1 の実施形態における記憶装置サブシステム 2 と同様に、ディスクユニット 2 1、ディスクコントローラ 2 2、ポート 2 3、ネットワークと接続するネットワークインタフェース（ネットワーク I/F）2 5 を含んで構成される。第 1 の実施形態と同じくディスクユニット 2 1、ポート 2 3 は複数あってもよいが、ここでは説明を簡単にするためディスクユニット、及びポートは、それぞれ 1 つであるものとして説明する。

30

【0057】

ファイバチャネルスイッチ 8 は、複数のポート 8 1 を有する。各ポート 8 1 には、ホスト計算機 1 a、1 b、... のインタフェース 1 2、及び、記憶装置サブシステム 2 a、2 b、... のポート 2 3 のいずれかが接続される。ファイバチャネルスイッチ 8 は、ネットワークインタフェース 8 2 を有しており、ネットワーク 4 にも接続されている。ファイバチャネルスイッチ 8 は、ホスト計算機 1 a、1 b、... が、記憶装置サブシステム 2 a、2 b、... を自由にアクセスできるようにするために使用される。この構成では、基本的にすべてのホスト計算機 1 が、すべての記憶装置サブシステム 2 にアクセスすることが可能である。

40

管理用ホスト計算機 3 は、第 1 の実施形態と同じく、管理マネージャ 3 1 を有する。管理マネージャ 3 1 は、各ホスト計算機 1 a、1 b、... のボリュームマネージャ 1 1 と通信しあって動作する。

【0058】

図 9 は、管理用ホスト計算機 3 が保持する論理デバイス管理テーブルの一例を示すテーブル構成図である。本実施形態における論理デバイス管理テーブルは、第 1 の実施形態において記憶装置サブシステム 2 が保持する論理デバイス管理テーブルと同様の情報を管理するために用いられる。以下、主に第 1 の実施形態における論理デバイス管理テーブルとの相違点について説明する。

50

【0059】

本実施形態において、管理用ホスト計算機3は、全ての記憶装置サブシステム2が有する全てのデバイスに一意的な番号をつけて管理する。この管理の目的のため、論理デバイス管理テーブルには、デバイス毎にそのサイズ103、構成104、状態105、LUN106、WWN(World Wide Name)102、及び接続ホスト名107を情報として持つ。

【0060】

サイズ103、構成104、状態105、LUN106については、第1の実施形態における論理デバイス管理テーブルが保持する情報と同じものである。WWN102は、記憶装置サブシステム2のポート23に設定されている情報で、各ポートを識別するために個々のファイバチャネルインタフェースに固有に割り付けられている情報である。WWN107は、N_PORT_NAMEとも呼ばれる。接続ホスト名107は、当該デバイスに接続が許可されているホスト計算機を識別するためのホスト名である。

【0061】

基本的に、ファイバチャネルスイッチ8に接続された複数のホスト計算機1が任意の記憶装置サブシステム2に自由にアクセスできると、システムの安全上問題となることがある。このようなシステムの安全性に関する問題を解決するために、例えば、特開平10333839号公報には、ファイバチャネルで接続された記憶装置に対し、特定のホスト計算機からのアクセスだけを許可することを可能とする技術が開示されている。本実施形態でも、システムの安全性を維持するために、記憶装置サブシステム2は、特開平10333839号公報に開示されているような安全性を維持するための手段を持っているものとする。ただし、これについては、本発明の本質とは直接関係するところではなく、ここでは詳しい説明を省略する。

【0062】

本実施形態において、WWN107は、各ホスト計算機1のインタフェース12にも与えられている。管理用ホスト計算機3は、図10に示すテーブルにより、ホスト名108とWWN109の組を管理している。

【0063】

以下、ボリュームマネージャ11、管理マネージャ31の動作について説明する。

【0064】

本実施形態において、ホスト計算機に新しいデバイスを割り当てる際にボリュームマネージャ11によって実施される処理は、基本的には、図4に示した第1の実施形態における処理と同様に実施される。すなわち、ボリュームマネージャ11は、ユーザまたはアプリケーションプログラムから必要とするデバイスの数や種類の情報を受けると、それをもとに管理マネージャ31に新たなデバイスの割付を要求する。管理マネージャ31において新たなデバイスの割り当てが終わると、ボリュームマネージャ11は、デバイスの設定変更を行い、新しいデバイスをホスト計算機1から利用できるようにする。

【0065】

図11に、本実施形態において、新しいデバイスの割り当て時に管理マネージャ31により実施される処理のフローチャートを示す。

【0066】

管理マネージャ31により行われる処理も、同様に、図5に示した第1の実施形態における管理マネージャの処理とほぼ同様に行われる。なお、図11において、図5に示したものと同様の処理が行われる部分については、図5と同一の参照番号を用いている。以下では、主に、図5と異なる処理が行われる部分について説明し、図5と同一の処理が行われる部分については説明を省略する。

【0067】

本実施形態において、記憶装置サブシステム2は、デバイスに対して、割り当てられていないホスト計算機から不用意にアクセスされることのないよう、初期状態では全てのホスト計算機1からのアクセスを禁止している。このため、管理マネージャ31は、ステップ1105で記憶装置サブシステム2にデバイスをオンラインにするよう指示する際、併

せて記憶装置サブシステム 2 に当該ホスト計算機 1 から新たに割り当てるデバイスへのアクセスを許可するように指示する。

【0068】

この指示において、管理マネージャ 31 は、記憶装置サブシステム 2 に対して、デバイスへのアクセスを許可すべきホスト計算機 1 の WWN を通知する。記憶装置サブシステム 2 は、ホスト計算機 1 によるデバイスのアクセス時に、管理マネージャ 31 から受け取った WWN に基づいて、そのアクセスの可否を判断する (ステップ 2105)。

【0069】

ステップ 2105 の処理に続いて、管理マネージャ 31 は、ファイバチャネルスイッチ 8 の設定変更を行う。例えば、図 12 に示すように、ホスト計算機 A、B は、ディスクユニット (デバイス) a、b にアクセスするが、ホスト計算機 C は、ディスクユニット (デバイス) c のみにアクセスする場合を考える。この場合、管理マネージャ 31 は、ホスト計算機 C に接続するポート c からは、ディスクユニット a、b に接続するポート (ポート d、ポート e) にはアクセスできないような経路設定をファイバチャネルスイッチ 8 に対して行う。これにより、あたかも 2 つのスイッチが存在するようにできる。このような経路設定を行うことをゾーニングと呼ぶ。ゾーニングを行うことによって、あるデバイスが本来アクセスが許されていないホスト計算機からアクセスされることを防ぐことができる。また、データの流が分離されるため、性能を向上させることもできる (ステップ 2106)。

10

【0070】

以上の処理の後、管理マネージャ 31 は、ステップ 1106、1107 の処理を実施する。

20

【0071】

図 13 は、本発明が適用された計算機システムの第 3 の実施形態における構成例を示す簡略なブロック図である。

【0072】

本実施形態の計算機システムは、複数のホスト計算機 1a、ホスト計算機 1b、... ホスト計算機 1n (総称してホスト 1 と呼ぶ) がネットワークインタフェース (I/F) 12、ネットワーク 4 を介して、ファイルサーバ 9 に接続されている。ファイルサーバ 9 は、インタフェース (I/F) 92 を介して記憶装置サブシステム 2 に接続される。記憶装置サブシステム 2、及び遠隔地に配置される記憶装置である副記憶装置 5 は第 1 の実施形態と同様のものである。

30

【0073】

ファイルサーバ 9 は各ホスト計算機 1 と接続するネットワークインタフェース 91、記憶装置サブシステム 2 と接続する複数のインタフェース 32、管理マネージャ 93、及びサーバプログラム 94 を備える。

【0074】

管理マネージャ 93 は、第 1 の実施形態におけるの管理マネージャ 31 と同じく、要求に応じたデバイスの割り当てなどを実施する。サーバプログラム 94 は、NFS (Network File System) などの、ネットワーク経由でのファイルアクセスを提供するファイルサーバプログラムである。サーバプログラム 94 は、ファイルサーバ 9 が記憶装置サブシステム 2 に作ったファイルシステムをホスト計算機 1 からアクセスするための手段を提供する。

40

【0075】

記憶装置サブシステム 2 とファイルサーバ 9 は、各ホスト計算機 1 からそれらがひとつの記憶装置として見える、いわゆる NAS (Network Attached Storage) になっている構成もありうる。

【0076】

ホスト計算機 1 のクライアントプログラム 11 は、ファイルサーバ 9 上のサーバプログラム 94 と通信しあって、ホスト 1 上で動作するアプリケーションプログラムから

50

、ファイルサーバ 9 が記憶装置サブシステム 2 に作ったファイルシステムを使用できるようにするプログラムである。クライアントプログラム 11 は、システムの構成によっては、ホスト 1 上の図示しないオペレーティングシステムに組み込まれていてもよい。クライアントプログラム 11 は、管理マネージャ 93 に対し、新たにファイルシステムを作るよう要求し、あるいは既存のファイルシステムのサイズの変更を要求する。

【0077】

ホスト計算機 1 の稼働中に、既存のファイルシステムのサイズの変更を可能とするため、本実施形態の記憶装置サブシステムは、ある論理デバイスに存在するデータをその論理デバイスが形成されている物理的なディスクユニットとは別の物理的なディスクユニットに移動する機能を備える。このような機能を実現する具体的な技術手段については、例えば、特開平 9 - 274544 号公報に開示された公知の技術を適用することができる。したがって、本明細書ではその詳細な説明は省略する。

10

【0078】

図 14 は、ホスト計算機 1 のクライアントプログラム 11 が新規にファイルシステムを構築する際に行われる処理の流れを示すフローチャートである。

【0079】

この処理は、ホスト計算機 1 を使用するユーザ、またはホスト計算機 1 で稼働するアプリケーションプログラムなどが新規にファイル領域を必要とするときに実施される。

【0080】

クライアントプログラム 11 は、ユーザ、またはアプリケーションプログラムからの要求に応じて必要とするデバイスについての情報の指定を受け付ける。ここで取得する情報には、図 4 に示した第 1 の実施形態におけるステップ 1001 と同様に、必要とするデバイスの容量、性能条件、信頼性レベルなどの情報が含まれる（ステップ 2001）。

20

【0081】

次に、クライアントプログラム 11 は、ステップ 2001 で指定された容量、性能条件、信頼性レベルなどの情報を管理マネージャ 93 に送信し、新たなファイルシステムの領域を要求する。管理マネージャ 93 は、クライアントプログラム 11 から受け取った情報に基づいて、割り当てることができるデバイスの領域を検索して用意し、その結果をクライアントプログラム 11 に返却する。このとき行われる管理マネージャ 93 の処理については後述する（ステップ 2002）。

30

【0082】

クライアントプログラム 11 は、新たな領域の要求に対する管理マネージャ 93 からの応答を受け取る。このときに受け取る応答には情報は、マウントポイント、例えば、NFS の場合にはファイルサーバのホスト名、あるいは、ホスト IP アドレス、及びディレクトリ名などが含まれる（ステップ 2003）。クライアントプログラム 11 は、管理マネージャ 93 受け取った情報をもとに、ファイルシステムをマウントする（ステップ 2004）。最後に、クライアントプログラム 11 は、割り当てられたマウントポイントをユーザまたはアプリケーションプログラムに通知して処理を終了する（ステップ 2005）。

【0083】

図 15 は、クライアントプログラム 11 からの新たな領域の要求に応答して行われる管理マネージャ 93 による処理の流れを示すフローチャートである。

40

【0084】

この処理は、基本的に図 5 に示した第 1 の実施形態における管理マネージャ 31 の処理と同様に行われる。ただし、図 5 におけるステップ 1107 の処理が、ステップ 2107、2111、2112 の処理に変更される。

【0085】

図 5 のステップ 1107 では要求のあったホスト計算機に対してターゲット ID などの情報が渡されるが、本実施形態では、これらの情報に対する加工が施される。このために、管理マネージャ 93 は、ターゲット ID などデバイスについての情報をサーバプログラ

50

ム 9 4 に渡し (ステップ 2 1 0 7)、サーバプログラム 9 4 からマウントポイントの情報を受け取る (ステップ 2 1 1 1)。そして、サーバプログラム 9 4 から受け取ったマウントポイントの情報を要求のあったクライアントプログラム 1 1 に渡して処理を終了する (ステップ 2 1 1 2)。

【0086】

図 1 6 は、デバイスについての情報を管理マネージャから受け取ったサーバプログラムが実施する処理の流れを示すフローチャートである。

【0087】

管理マネージャ 9 3 からデバイスについての情報が渡されると (ステップ 2 2 0 1)、サーバプログラム 9 4 は、ファイルサーバ 9 のデバイスリコンフィグを実施する。この処理は、具体的には、図 4 に示した第 1 の実施形態におけるステップ 1 0 0 5 の処理と同様の処理である (ステップ 2 2 0 2)。

10

【0088】

続いて、サーバプログラム 9 4 は、新たにできたデバイスにファイルシステムを作成し (ステップ 2 2 0 3)、管理マネージャ 9 3 にそのファイルシステムのマウントポイントを示す情報を返却する (ステップ 2 2 0 4)。

【0089】

以上の処理により、ホスト計算機 1 から利用可能な新しいファイルシステムを追加することができる。

【0090】

20

図 1 7 は、既存のファイルシステムのサイズを変更する際に管理マネージャ 9 3 により実施される処理の流れを示すフローチャートである。図 1 5 に示した新たなファイルシステムの要求時の処理とは、以下の点において相違する。

【0091】

既存のファイルシステムのサイズを変更しようとするとき、ユーザ、あるいは、アプリケーションプログラムは、クライアントプログラム 1 1 に対して、サイズを変更するファイルシステムのマウントポイント、拡張、あるいは、縮小しようとするサイズなどに関する情報を指定した要求を発行する。クライアントプログラム 1 1 は、ユーザ、あるいは、アプリケーションプログラムから指定された情報を用いて管理マネージャ 9 3 にファイルシステムのサイズの変更を要求する。管理マネージャ 9 3 は、クライアントプログラム 1 1 から送られてくる、処理の対象とするファイルシステムのマウントポイント、拡張したいサイズ等の情報を受け取る (ステップ 2 3 0 1)。

30

【0092】

管理マネージャ 9 3 は、クライアントプログラム 1 1 から受け取ったマウントポイントに基づいて処理の対象となっているファイルシステムを格納している論理デバイスのターゲット ID、LUN などの情報を得て、論理デバイスを判別する。そして、管理マネージャ 9 3 は、この論理デバイスの種類、すなわち、信頼性、性能等の情報を得る (ステップ 2 3 0 2)。続いて、管理マネージャ 9 3 は、ステップ 2 3 0 1、2 3 0 2 で得た情報に基づき、新たなファイルシステムを追加するときと同様にして、変更後のファイルシステムのサイズの空き領域を持ち、元の論理デバイスと同じ種類の論理デバイスを確保する (ステップ 1 1 0 2 ~ 1 1 1 0)。

40

【0093】

この後、管理マネージャ 9 3 は、ステップ 2 3 0 4 において、記憶装置サブシステム 2 に対し、これまでファイルシステムが記録されていた論理デバイスから新たに確保した論理デバイスにデータの移動を指示する。データの移動はファイルサーバプログラム 9 4 から透過的に行われる。ホスト計算機 1 はファイルサーバプログラム 9 4 を介して記憶装置サブシステム 2 にアクセスするため、この処理は、ホスト計算機 1 から透過的な処理となる。したがって、データの移動中、ホスト計算機 1 は処理を停止する必要はない。

【0094】

50

データの移動が終わると、管理マネージャ 9 3 は、サーバプログラム 9 4 にファイルシステムの拡張を指示する。実際のデバイス容量が増加してもファイルシステムを構築しなおさなければ、ファイルシステムとしては拡張された容量のすべてを使うことはできない。サーバプログラム 9 4 に対してファイルシステムの拡張を指示した後、管理マネージャ 9 3 は、処理の完了をクライアントプログラム 1 1 に通知して処理を終了する（ステップ 2 3 0 5）。

【0095】

以上の処理により、既存のファイルシステムのサイズの変更を、ホスト計算機 1 を稼働させたまま行うことが可能となる。なお、既存のファイルシステムのサイズを変更する場合、クライアントプログラム 1 1 は、管理マネージャからの通知を受けた後、そのまま拡張されたファイルシステムを使用することができる。したがって、この場合には、図 1 4 におけるステップ 2 0 0 4、及びステップ 2 0 0 5 の処理は実施する必要がない。

【0096】

図 1 8 は、本発明が適用された計算機システムの第 4 の実施形態における構成例を示す簡略なブロック図である。

【0097】

本実施形態における計算機システムは、複数のホスト計算機 1（ホスト計算機 1 a、1 b、・・・、1 n）、管理用ホスト計算機 3、記憶装置サブシステム 2、副記憶装置 5 を有している。各ホスト計算機 1 と記憶装置サブシステム 2 は、ファイバチャネルスイッチ 8 を介して接続される。また、ホスト計算機 1、記憶装置サブシステム 2、ファイバチャネルスイッチ 8 は、ネットワーク 4 を介して相互に接続されている。

【0098】

ファイバチャネルスイッチ 8 は、複数のポート 8 1 を備えており、これらのポート間の接続を切り替え、ポート 8 1 に接続された機器間でのデータ転送を実現する。ファイバチャネルスイッチ 8 は、また、ネットワーク 4 を介して通信を行うためのネットワークインタフェース 8 2 を備える。ホスト計算機 1 は、それぞれ、ボリュームマネージャ 1 1、1 または複数のインタフェース 1 2 を備えている。ホスト計算機 1 のインタフェース 1 2 は、ファイバチャネル 8 が有する複数のポート 8 1 のいずれかと接続される。

【0099】

記憶装置サブシステム 2 は、複数のクラスタ 2 6 とクラスタ 2 6 を相互に接続するコントローラ間接続機構 2 7 を有する。各クラスタ 2 6 は、チャンネルプロセッサ 2 3、ドライブプロセッサ 2 2、複数のディスクユニット 2 1 を備える。同じクラスタ内のチャンネルプロセッサ 2 3 とドライブプロセッサ 2 2 とは、コントローラ間接続機構 2 7 よりも高速なバス 2 8 で結合されている。各チャンネルプロセッサ 2 3 は、1、または複数のポート 2 3 1 を備えており、副記憶装置 5、あるいは、ファイバチャネル 8 を介してホスト計算機 1 と接続される。ドライブプロセッサ 2 2 には複数のディスクユニット 2 1 が接続されている。本実施形態では、これら複数のディスクユニット 2 1 を組み合わせることで 1 または複数の論理デバイスが構成され、あるいは、1 つのディスクユニット 2 1 により 1 または複数の論理デバイスが構成される。なお、ある 1 つの論理デバイスを構成するにあたって複数のクラスタ 2 6 が備えるディスクユニット 2 1 を組み合わせることはできないものとする。

【0100】

チャンネルプロセッサ 2 3 は、各ホスト計算機 1 に対し、1 または複数の論理デバイスを見せ、各ホスト 1 からのアクセスを受け付ける。原則として、チャンネルプロセッサ 2 3 は、そのチャンネルプロセッサ 2 3 が属するクラスタ 2 6 内のドライブプロセッサ 2 2 に接続されたディスクユニット 2 1 により構成される論理デバイスを管理対象とする。これは同一クラスタ 2 6 内のチャンネルプロセッサ 2 3 とドライブプロセッサ 2 2 の通信は、クラスタをまたがった通信よりも高速に行うことができることによる。ただし、障害などの要因により、あるクラスタ 2 6 のチャンネルプロセッサ 2 3 が動作しない場合には、他クラスタ 2 6 のチャンネルプロセッサ 2 3 がその処理を肩代わりする。チャネ

10

20

30

40

50

ルプロセッサ 2 3 はホスト計算機 1 から指定された論理デバイスがどのドライブプロセッサ 2 2 に接続されたディスクユニット 2 1 に構成されているか判別し、然るべきドライブプロセッサ 2 2 に処理の要求を渡す。ドライブプロセッサ 2 2 は、チャンネルプロセッサ 2 3 からの要求を解釈して、論理デバイスの置かれている各ディスクユニット 2 1 に対するディスクアクセス要求を生成して、該当するディスクユニット 2 1 にそのディスクアクセス要求を送る。

【0101】

ホスト計算機 1 は、第 1 の実施形態におけるホスト計算機 1 とほぼ同様の構成を有するが、その上で動作するボリュームマネージャ 1 1 の機能に若干の相違がある。ボリュームマネージャ 1 1 は、第 1 の実施形態においてボリュームマネージャ 1 1 が行う論理デバイスの割り当て、返却の処理に加え、複数の論理デバイスをまとめて、別の論理デバイスとして上位のアプリケーションプログラムに見せる機能を持つ。以下、ボリュームマネージャ 1 1 が作る論理デバイスを、記憶装置サブシステム 2 が管理する論理デバイスと区別するため L V O L と表記する。ボリュームマネージャ 1 1 は、複数の論理デバイスを見かけ上結合してより大きな 1 つの L V O L を形成し、あるいは、1 つの論理デバイスを複数の領域に分割し、それらの領域を L V O L としてホスト計算機 1 上のアプリケーションプログラムに利用させることができる。また、すでに存在する L V O L に新たな論理デバイスを結合して、L V O L の容量を拡張することもできる。

【0102】

図 1 9 は、本実施形態においてボリュームを新規に割り当てる際にボリュームマネージャ 1 1 により実施される処理の流れを示すフローチャートである。ここで説明する処理は、図 4 に示した第 1 の実施形態におけるデバイスの割り当て処理のステップ 1 0 0 2 がステップ 1 0 0 2 に、ステップ 1 0 0 6 がステップ 1 0 0 5 、ステップ 1 0 0 6 に置き換えたものである。このほかのステップでは、図 4 の対応するステップと同様の処理が行われる。以下、ステップ 1 0 0 2 、1 0 0 5 、及び 1 0 0 6 で行われる処理について説明する。

【0103】

ステップ 1 0 0 2 では、未使用の W W N 、L U N の組をボリュームマネージャ 1 1 が管理している L V O L 管理テーブルより検索する。L V O L 管理テーブルの一例を図 2 0 に示す。L V O L 管理テーブルには、L V O L 名 1 5 1 、デバイスファイル名 1 5 2 、サイズ 1 5 3 、そして各デバイスの W W N 1 5 4 、L U N 1 5 5 の組からなる情報が登録される。L V O L 名 1 5 1 は、ボリュームマネージャ 1 1 によりアプリケーションプログラムに提供されている L V O L を識別するために付与される識別子である。デバイスファイル名 1 5 2 は、L V O L を構成する論理デバイスの名称である。ボリュームマネージャ 1 1 は、各 L V O L に属する論理デバイスをデバイスファイル名に基づいて管理している。サイズ 1 5 3 は、その L V O L を構成する各論理デバイスの容量を示す。1 つの L V O L が複数の論理デバイスで構成されることもあるため、1 つの L V O L 名に複数のデバイスファイルが属することもある。

【0104】

ステップ 1 0 0 5 でボリュームマネージャ 1 1 は、管理マネージャ 3 1 によって割り当てられた論理デバイスを用いて新たに L V O L を作成し、その内容を L V O L 管理テーブルに登録する。ステップ 1 0 0 6 では、割り当ての済んだ L V O L 名がユーザに通知されて処理は終了する。

【0105】

図 2 1 は、ユーザ、あるいは、アプリケーションプログラムの要求に応じて L V O L の容量を拡張する際のボリュームマネージャの処理を示すフローチャートである。

【0106】

L V O L の容量を拡張する際には、新たに論理デバイスが用意され、その論理デバイスを拡張しようとする L V O L を構成している論理デバイスと組み合わせられて新しい L V O L が構築される。このとき新たに用意される論理デバイスは、拡張しようとする L V O L

10

20

30

40

50

を構築している論理デバイスと同じ種類の論理デバイスであることが普通である。本実施形態では、ボリュームマネージャ11が拡張しようとするLVO Lを構成している論理デバイスの種類を判別し、同じ種類の論理デバイスを確保する。

【0107】

本処理において、ボリュームマネージャ11は、まず、ユーザあるいは、アプリケーションプログラムから拡張対象のLVO LのLVO L名と拡張すべき容量についての情報を受け取る(ステップ2501)。次に、ボリュームマネージャ11は、拡張対象のLVO Lを構成する論理デバイスの種類を管理マネージャ31に問い合わせる(ステップ2502)。ボリュームマネージャ11は、未使用のWWN、LUNの組をLVO L管理テーブルより検索する(ステップ2503)。ボリュームマネージャ11は、ステップ2502、2503で取得した論理デバイスの種類、及び未使用のWWNとLUNの組み合わせを含む情報を管理マネージャ31に送信する。(ステップ2504)。管理マネージャ31から新たに割り当てられた論理デバイスについての情報を受信すると(ステップ2505)、ボリュームマネージャ11は、ホスト計算機1のリコンフィグを実施し、新たに割り当てられた論理デバイスをホスト計算機1から認識できるようにする(ステップ2506)。最後に、ボリュームマネージャ11は、拡張対象のLVO Lに新たに割り当てられた論理デバイスを追加してLVO Lの容量を拡張し、処理を終了する(ステップ2507)。

10

【0108】

図19のステップ1003、及び図21のステップ2504でボリュームマネージャ11から新しい論理ボリュームの割り当ての要求があると、管理マネージャ31は、いずれの場合もボリュームマネージャ11から要求されたデバイスの種類、容量に見合ったデバイスを検索して割り当てる。この処理のために、管理マネージャ31は、図9に示すような論理デバイス管理テーブルの他、記憶装置サブシステム2内のクラスタ26に関する情報が設定されたクラスタ情報テーブルを備える。

20

【0109】

図22は、クラスタ情報管理テーブルの一例を示すテーブル構成図である。

【0110】

クラスタ情報管理テーブルは、各クラスタ26に対応したエントリを有し、クラスタ26ごとにそのクラスタを識別するクラスタ番号161、クラスタが有するポートのポート番号162、及びポートに割り当てられているWWN163が設定されている。図に示すように、1つのクラスタ26に複数のポートが存在する場合、それぞれのポート番号とWWNがそのクラスタに対応したエントリに設定される。先に説明したように、あるドライブプロセッサ22に接続されたディスクユニット21に論理デバイスを構築したときは、その論理デバイスに対しては同一のクラスタ内のポート231からアクセスできるようにすることが性能上の観点から望ましい。管理マネージャ31は、クラスタ情報テーブルに基づいてホスト計算機1からのアクセスに用いられるポート231と、新しく割り当てられる論理デバイスが構築されるディスクユニット21が接続するドライブプロセッサ22が同じクラスタとなるようにデバイスの設定を行う。

30

【0111】

図23は、管理マネージャ31によるデバイスの割り当て処理の流れを示すフローチャートである。

40

【0112】

本実施形態におけるデバイスの割り当て処理は、図11に示した第2の実施形態における処理とほぼ同様に行われるが、記憶装置サブシステムの構成上の相違から一部の処理が第2の実施形態とは異なる。具体的には、ステップ1109において、管理マネージャ31から記憶装置サブシステム2に対して新規にデバイスの構築が要求されると、記憶装置サブシステム2は、要求に合わせてデバイスを構築する。記憶装置サブシステム2において論理デバイスが構築されると、管理マネージャ31には、新しく構築された論理デバイスに関し、その論理デバイスがどのクラスタ26に構築されたかを示す情報を受け

50

取る（ステップ2610）。

【0113】

管理マネージャ31は、記憶装置サブシステム2から受け取った論理デバイスに関する情報と、クラスタ情報管理テーブルを参照して、どのポートから当該デバイスをアクセスできるようにするか決定する。管理マネージャ31は、さらに、未使用LUNの情報によって、新たに割り当てるデバイスのLUNを決定する（ステップ2604）。また、管理マネージャ31は、最後に、ボリュームマネージャ11に対し、新しく割り当てる論理ボリュームのアクセスに必要なWWN、LUNなどの情報を送り、処理を終了する（ステップ2607）。

【0114】

ここで説明した以外の処理については、図11に示した第2の実施形態における管理マネージャの処理と同様であり、図では、図11と同一の参照番号を用い、詳細な説明は省略する。

【0115】

以上説明した実施の形態によれば、ホスト計算機が稼働中であっても、要求にあわせてデバイスの割り当てを行うことができる。さらにファイバチャネルスイッチで構成された、複数のデバイスが混在するような環境でも、要求に合ったデバイスを容易にホスト計算機に割り当てることが可能となる。

【0116】

なお、本発明は、上述した実施形態に限定されるものではなく、本発明の趣旨の範囲内において種々の態様をとり得るものであることは言うまでもない。

【図面の簡単な説明】

【0117】

【図1】本発明の第1の実施形態における計算機システムの構成例を示すブロック図である。

【図2】記憶装置サブシステムにより保持される論理デバイス管理テーブルの一例を示すテーブル構成図である。

【図3】管理マネージャにより保持されるホスト管理テーブルの一例を示すテーブル構成図である。

【図4】ホスト計算機のボリュームマネージャにより実施され処理の流れを示すフローチャートである。

【図5】管理マネージャにより実施される処理の流れを示すフローチャートである。

【図6】デバイスの返却処理におけるボリュームマネージャによる処理の流れを示すフローチャートである。

【図7】デバイスの返却処理における管理マネージャによる処理の流れを示すフローチャートである。

【図8】本発明の第2の実施形態における計算機システムの構成例を示すブロック図である。

【図9】管理マネージャにより保持される論理デバイス管理テーブルの一例を示すテーブル構成図である。

【図10】管理マネージャにより保持されるホスト計算機とWWNの対応関係を管理するためのテーブルの一例を示すテーブル構成図である。

【図11】管理マネージャによる処理の流れを示すフローチャートである。

【図12】ファイバチャネルスイッチのゾーニングの機能を示す説明図である。

【図13】本発明の第3の実施形態における計算機システムの構成例を示すブロック図である。

【図14】クライアントプログラムによる処理の流れを示すフローチャートである。

【図15】ファイルサーバの管理マネージャによる処理の流れを示すフローチャートである。

10

20

30

40

50

【図16】ファイルサーバのサーバプログラムによる処理の流れを示すフローチャートである。

【図17】ファイルシステムを拡張する際に管理マネージャにより実施される処理の流れを示すフローチャートである。

【図18】本発明の第4の実施形態における計算機システムの構成例を示すブロック図である。

【図19】ボリュームマネージャによる処理の流れを示すフローチャートである。

【図20】L V O L管理テーブルの一例を示すテーブル構成図である。

【図21】L V O Lを拡張するときにボリュームマネージャにより実施される処理の流れを示すフローチャートである。

10

【図22】クラスタ情報テーブルの一例を示すテーブル構成図である。

【図23】L V O Lを拡張するときに管理マネージャにより実施される処理の流れを示すフローチャートである。

【符号の説明】

【0118】

- 1 ... ホスト計算機、
- 2 ... 記憶装置サブシステム
- 3 ... 管理用ホスト計算機
- 4 ... ネットワーク
- 5 ... 副記憶装置
- 8 ... ファイバチャネルスイッチ
- 11 ... ボリュームマネージャ
- 21 ... ディスクユニット
- 22 ... ディスクコントローラ
- 23 ... ポート
- 31 ... 管理マネージャ
- 81 ... ポート

20

【 図 1 】

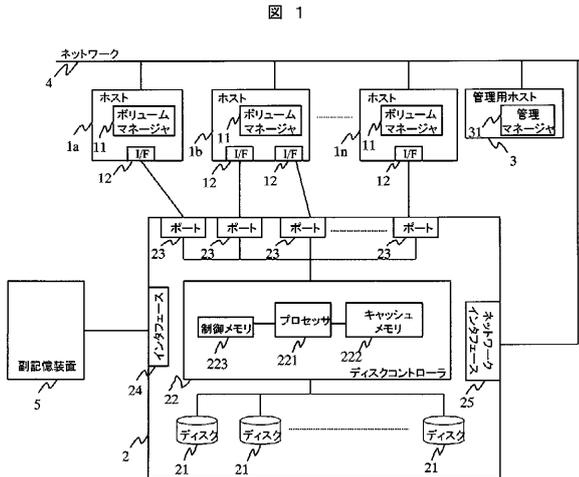


図 1

【 図 2 】

図 2

論理デバイス番号	サイズ	構成	状態	バス	Target ID	LUN
0	1000000	RAID1	オンライン	0	0	0
1	1000000	RAID5	オフライン	-	-	-
2	35000	キャッシュ	オンライン	0	0	2
2	35000	キャッシュ	オンライン	1	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮
k	-	-	未実装	-	-	-
k+1	1000000	単体ディスク	障害オフライン	0	1	0
k+2	1000000	RAID1	オフライン	-	-	-
⋮	⋮	⋮	⋮	⋮	⋮	⋮
n	-	-	未実装	-	-	-

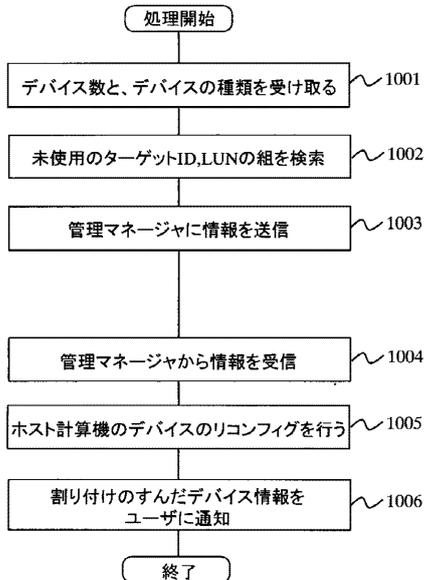
【 図 3 】

図 3

ホスト名	ポート番号	インタフェース番号	論理デバイス番号
host1	0	0	0,1
host2	1	0	1
host2	2	1	3
⋮	⋮	⋮	⋮
hostN	0	0	n

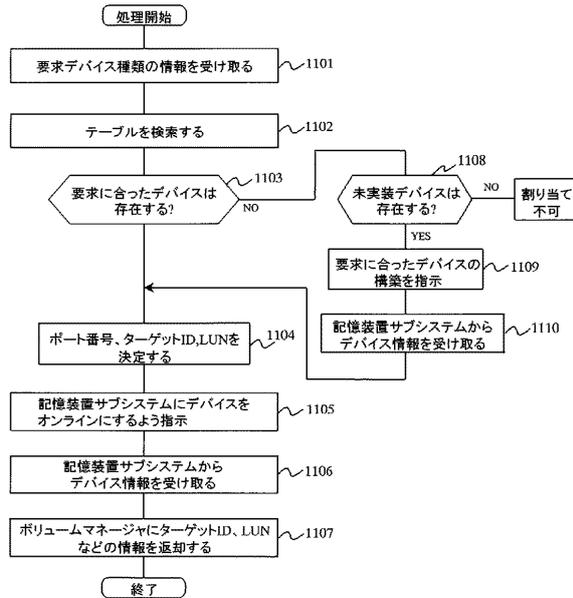
【 図 4 】

図 4



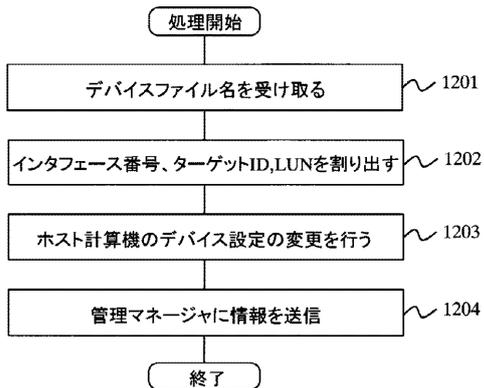
【 図 5 】

図 5



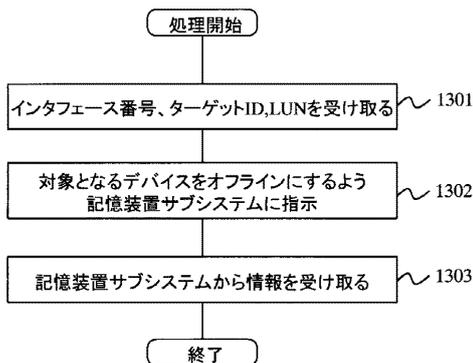
【 図 6 】

図 6



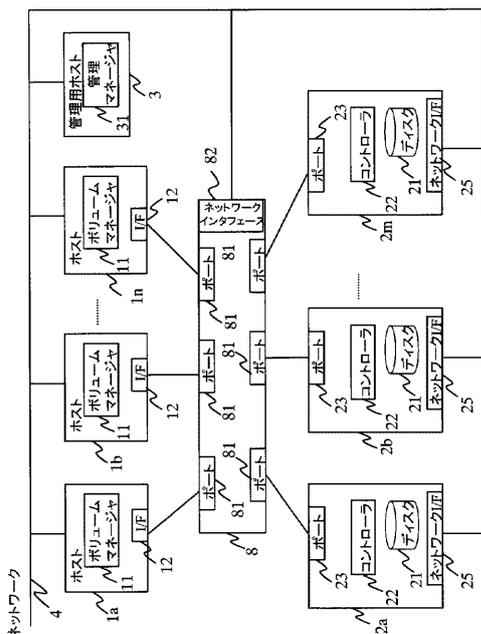
【 図 7 】

図 7



【 図 8 】

図 8



【 図 9 】

図 9

デバイスNo.	WWN	サイズ	構成	状態	LUN	接続ホスト名
0	xxxxxxx	1000000	RAID1	オンライン	0	hostA
1	xxxxxxx	1000000	RAID5	オフライン	-	
2	xxxxxxx	35000	キャッシュ	オンライン	2	hostA,hostB
...
n	xxxxxxx	1000000	単体ディスク	障害オフライン	0	

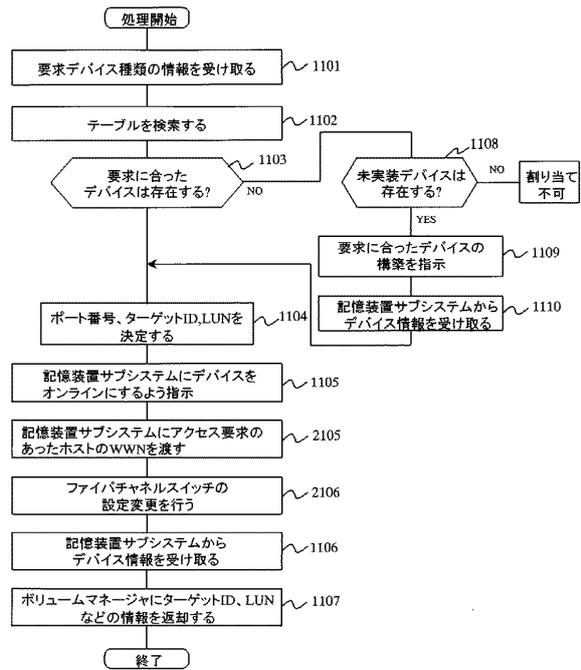
【 図 1 0 】

図 10

108 ホスト名	109 WWN
host1	XXXX
host2	XXXX
host2	XXXX
⋮	⋮
hostN	XXX

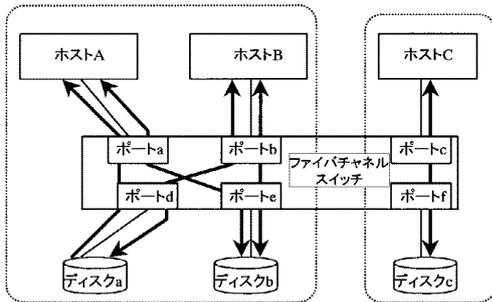
【 図 1 1 】

図 11



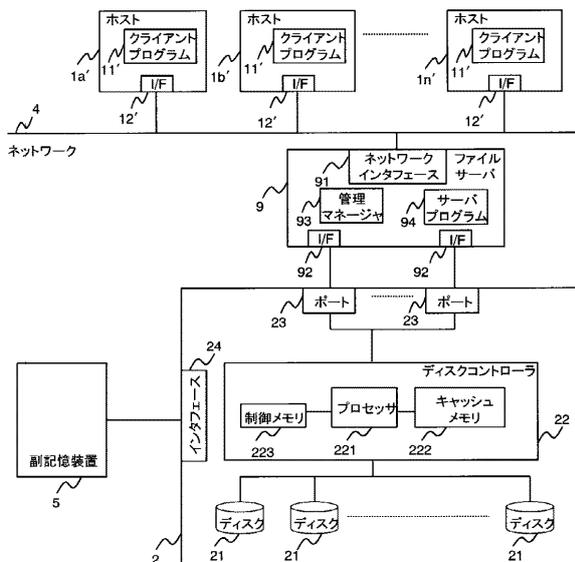
【 図 1 2 】

図 12

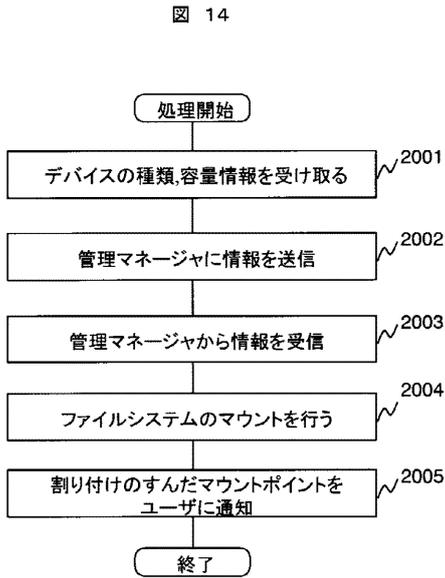


【 図 1 3 】

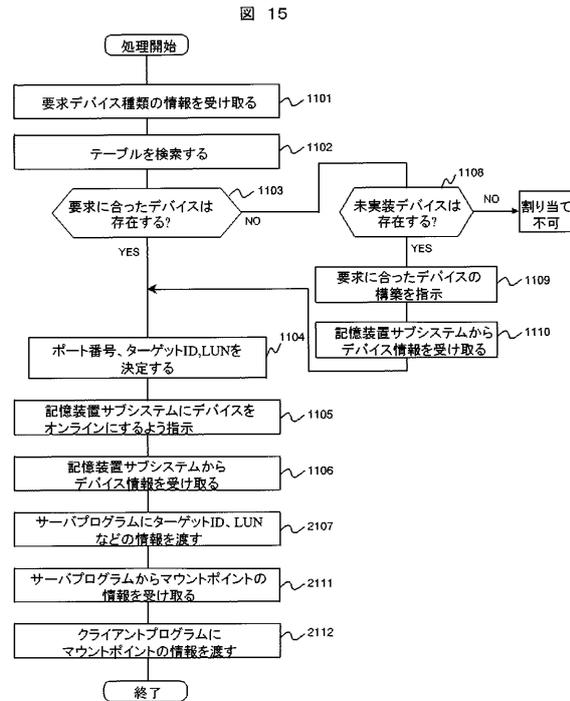
図 13



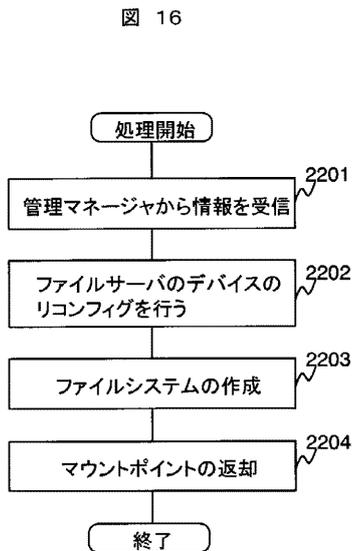
【 図 1 4 】



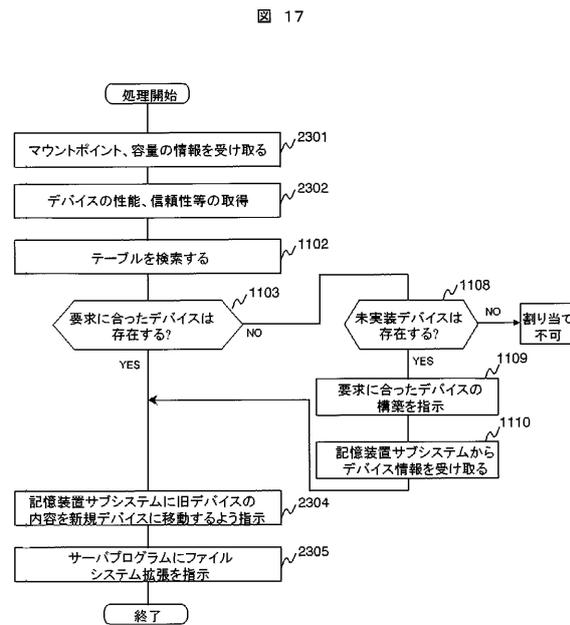
【 図 1 5 】



【 図 1 6 】

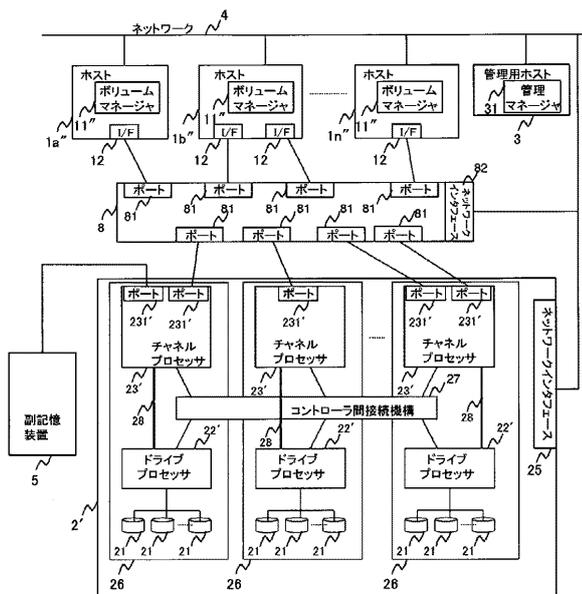


【 図 1 7 】



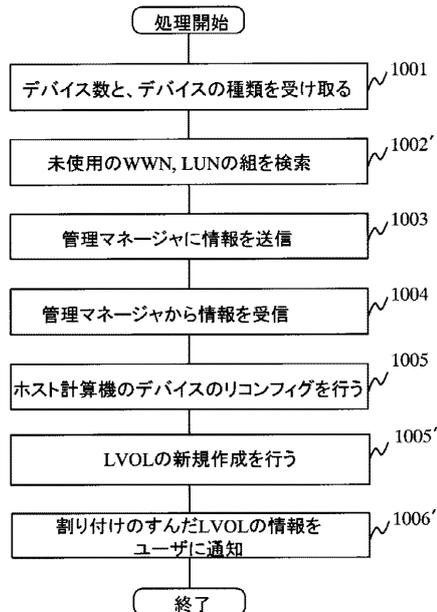
【 図 1 8 】

図 18



【 図 1 9 】

図 19



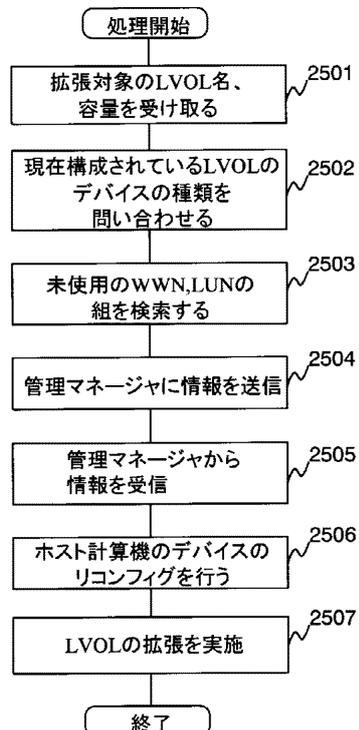
【 図 2 0 】

図 20

LVOL名	デバイスファイル	サイズ	WWN	LUN
lvol1	/dev/rdisk/c0t0d0	1000000	xxxx	0
lvol2	/dev/rdisk/c0t0d1	500000	xxxx	1
lvol3	/dev/rdisk/c1t0d0	35000	xxxx	0
	/dev/rdisk/c1t0d1	500000	xxxx	1
⋮	⋮	⋮	⋮	⋮
lvolN			xxx	n

【 図 2 1 】

図 21



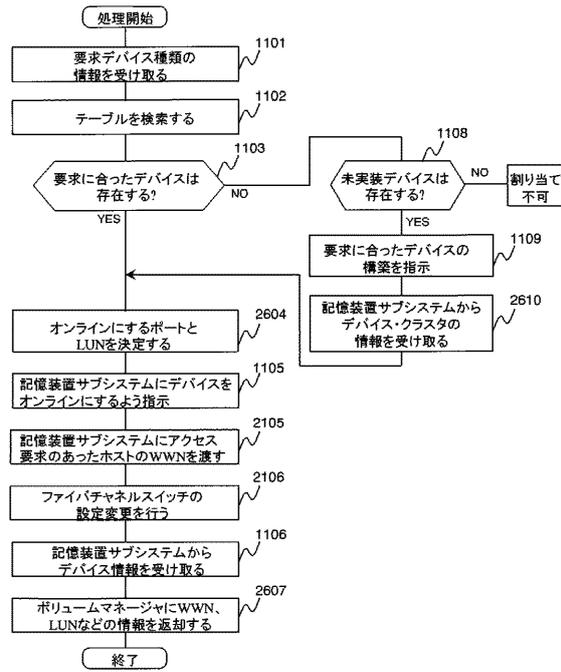
【 図 2 2 】

図 22

クラスタ番号	ポート番号	WWN
1	1	XXXX
2	3	XXXX
3	4	
	5	
⋮	⋮	⋮
N		XXX

【 図 2 3 】

図 23



フロントページの続き

(51)Int.Cl. ⁷	F I	テーマコード(参考)
G 0 6 F 15/177	G 0 6 F 12/08 5 2 3 E	5 B 0 9 8
	G 0 6 F 12/08 5 5 7	
	G 0 6 F 13/10 3 4 0 A	
	G 0 6 F 15/177 6 7 0 C	

(72)発明者 山神 憲司

神奈川県川崎市麻生区王禅寺 1 0 9 9 番地 株式会社日立製作所システム開発研究所内

F ターム(参考) 5B005 JJ00 MM11 UU41
5B014 EB04 FA05 FB04
5B045 BB29 EE02
5B065 BA01 CA11 ZA01
5B082 FA05 FA07 HA08
5B098 AA03 AA10 GA01 GD03 GD12 GD14

【要約の続き】