

(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2007-164794

(P2007-164794A)

(43) 公開日 平成19年6月28日(2007.6.28)

(51) Int. Cl.	F I	テーマコード (参考)
G06F 12/00 (2006.01)	G06F 12/00 545A	5B014
G06F 13/10 (2006.01)	G06F 12/00 514E	5B065
G06F 3/06 (2006.01)	G06F 13/10 340A	5B082
	G06F 3/06 301A	

審査請求 未請求 請求項の数 10 O L (全 33 頁)

(21) 出願番号 特願2006-336393 (P2006-336393)
 (22) 出願日 平成18年12月13日 (2006.12.13)
 (31) 優先権主張番号 05112058.2
 (32) 優先日 平成17年12月13日 (2005.12.13)
 (33) 優先権主張国 欧州特許庁 (EP)

(71) 出願人 501263810
 トムソン ライセンシング
 Thomson Licensing
 フランス国, エフ-92100 ブロー
 ニュ ビヤンクール, ケ アルフォンス
 ル ガロ, 46番地
 46 Quai A. Le Gallo
 , F-92100 Boulogne-
 Billancourt, France
 (74) 代理人 100061815
 弁理士 矢野 敏雄
 (74) 代理人 100094798
 弁理士 山崎 利臣
 (74) 代理人 100099483
 弁理士 久野 琢也

最終頁に続く

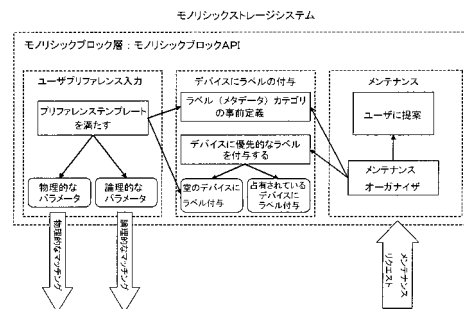
(54) 【発明の名称】 ネットワーク内のノードを自動的に分類する方法および装置

(57) 【要約】

【課題】 自立的な構成要素として機能し、中央のストレージ制御装置を要せずに分散ストレージシステムを形成することができ、よりインテリジェントなストレージデバイスを使用可能にする、ネットワーク内のノードを自動的に分類する方法および装置を提供する。

【解決手段】 少なくとも1つのデータクラスを検出し、データクラスに従いノードに格納されているデータを自動的に分類し、データクラスに従い分類されたデータの量を検出し、ここでファイルの数およびこのファイルの平均サイズを検出し、データクラスに分類されているデータの検出された量から優先値を計算し、この優先値はデータクラスに従いノードに格納されているデータの相対的な優先度に対する尺度であり、データクラスに従いデータタイプを表すリクエストを受信し、データクラスに従いデータの相対的な優先度に対する尺度である優先値をリクエストに応じて提供する。

【選択図】 図6



【特許請求の範囲】

【請求項 1】

ネットワーク内のノードを自動的に分類する方法において、

- 少なくとも 1 つのデータクラス (ビデオ、オーディオ、テキスト) を検出するステップを有し、

- 前記データクラスに従いノードに格納されているデータを自動的に分類するステップを有し、

- 前記データクラスに従い分類されたデータの量を検出するステップを有し、ここでファイルの数および該ファイルの平均サイズを検出し、

- 前記データクラスに分類されているデータの検出された量から優先値 (E_{dominant}) を計算するステップを有し、該優先値 (E_{dominant}) は前記データクラスに従いノードに格納されているデータの相対的な優先度に対する尺度であり、

- 前記データクラスに従いデータタイプを表すリクエストを受信するステップを有し、

- 前記データクラスに従いデータの相対的な優先度に対する尺度である前記優先値 (E_{dominant}) を前記リクエストに応じて提供するステップを有することを特徴とする、ネットワーク内のノードを自動的に分類する方法。

10

【請求項 2】

ファイルサイズとファイルの数の組み合わせに関してデータの量を検出する、請求項 1 記載の方法。

【請求項 3】

ストレージデバイスの残りの潜在的な記憶容量を自動的に推定し、さらに、

- 前記ストレージデバイスに既に格納されているデータのデータタイプを検出するステップを有し、

- 前記データタイプの既に格納されているデータのファイルサイズ統計を検出するステップを有し、

- 前記ストレージデバイスにおける残りの空き記憶容量を検出するステップを有し、

- 前記データタイプのデータを格納するためのリクエストを受信するステップを有し、該リクエストはファイルサイズを表さず、

- 検出された前記ファイルサイズ統計に従い、前記データタイプのデータの予測されるサイズを検出するステップを有し、

- 前記ストレージデバイスの前記残りの空き記憶容量が、前記データタイプのデータの予測されるサイズよりも大きい場合には、前記データの記憶を受諾するステップを有する、請求項 1 または 2 記載の方法。

20

30

【請求項 4】

データタイプに関連するノードの前記優先値 (E_{dominant}) を次式により計算し、

$$E_{\text{dominant, type}} = R P S C * A_{\text{total}} * k_1$$

ここで、

$$A_{\text{total}} = N_{\text{total}} * S_{\text{total}}$$

$$R P S C = k_2 - k_3 (k_4 + S_{\text{average}} + C_{\text{used}}) / C_{\text{total}}$$

ここで、 k_1 , k_2 , k_3 , k_4 は定数であり、 N_{total} は前記データタイプのデータファイルの総数であり、 S_{total} は前記データタイプのファイルの累算されたサイズであり、 S_{average} は前記データタイプのファイルの平均サイズであり、 C_{used} は前記ノードの占有されている記憶容量であり、 C_{total} は前記ノードの総記憶容量である、請求項 1 から 3 までのいずれか 1 項記載の方法。

40

【請求項 5】

データクラスの分類および / または検出をデータに関連するメタデータに従い行う、請求項 1 から 4 までのいずれか 1 項記載の方法。

【請求項 6】

前記ノードはピア・ツー・ピアネットワークにおけるピアである、請求項 1 から 5 までのいずれか 1 項記載の方法。

50

【請求項 7】

ノードの選択はノードの相対的な空き記憶容量 (R F S C) を基礎とし、該相対的な空き記憶容量を次式により計算し、

$$R F S C = c_1 - c_2 (c_3 + S_{in_data} + C_{used}) / C_{total}$$

ここで c_1 , c_2 , c_3 は定数であり、 S_{in_data} は格納すべきデータのサイズであり、 C_{used} は前記ノードの占有されている記憶容量であり、 C_{total} は前記ノードの総記憶容量である、請求項 1 から 6 までのいずれか 1 項記載の方法。

【請求項 8】

ネットワーク内のノードを自動的に分類する装置において、該装置は、

- 少なくとも 1 つのデータクラス (ビデオ、オーディオ、テキスト) を検出する手段を有し、
- 前記データクラスに従いノードに格納されているデータを自動的に分類する手段を有し、
- 前記データクラスに従い分類されたデータの量を検出する手段を有し、ここでファイルの数および該ファイルの平均サイズが検出され、
- 前記データクラスに分類されているデータの検出された量から優先値 ($E_{dominant}$) を計算する手段を有し、該優先値 ($E_{dominant}$) は前記データクラスに従いノードに格納されているデータの相対的な優先度に対する尺度であり、
- 前記データクラスに従いデータタイプを表すリクエストを受信する手段を有し、
- 前記データクラスに従いデータの相対的な優先度に対する尺度である前記優先値 ($E_{dominant}$) を前記リクエストに応じて提供する手段を有することを特徴とする、ネットワーク内のノードを自動的に分類する装置。

【請求項 9】

ファイルサイズとファイルの数の組み合わせに関してデータの量が検出される、請求項 8 記載の装置。

【請求項 10】

ストレージデバイスの残りの潜在的な記憶容量を自動的に推定し、さらに、

- 前記ストレージデバイスに既に格納されているデータのデータタイプを検出する手段を有し、
- 前記データタイプの既に格納されているデータのファイルサイズ統計を検出する手段を有し、
- 前記ストレージデバイスにおける残りの空き記憶容量を検出する手段を有し、
- 前記データタイプのデータを格納するためのリクエストを受信する手段を有し、該リクエストはファイルサイズを表さず、
- 検出された前記ファイルサイズ統計に従い、前記データタイプのデータの予測されるサイズを検出する手段を有し、
- 前記ストレージデバイスの前記残りの空き記憶容量が、前記データタイプのデータの予測されるサイズよりも大きい場合には、前記データの記憶を受諾する手段を有する、請求項 8 または 9 記載の装置。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、ネットワーク内のノードを自動的に分類する方法および装置に関する。

【背景技術】

【0002】

ストレージエリアネットワーク (S A N) は、多数のサーバを集中型のディスクストレージプールに接続する、ストレージディスクのネットワークとして公知である。接続されている全てのストレージを単一のリソースとして処理することによって、システムの保守は容易になる。S A N は集中型か分散型が考えられる。集中型の S A N は多数のサーバをディスクコレクションに接続し、これに対し分散型の S A N は建物内のノードを接続する

ために典型的に1つまたは複数のスイッチを使用する。しかしながら両方のタイプとも集中型の制御構造、すなわちクライアントサーバ構造を使用する。

【0003】

一般的な分散ストレージシステム(DSS)においては、データが格納されるべきデバイスをアプリケーションが決定することができる。このようなデバイスを本明細書においては「要求されたストレージターゲット」と称する。ユーザは特定のストレージデバイスの選択を要求されることが多い。DSS内の個々のストレージデバイスを自動的にする選択するインテリジェンスなやり方は存在しない。

【0004】

DSSに依存せずに、改善されたデータの安全性、可用性、アクセス可能性のためにストレージデバイスを複製することができる。データをパーティションに区切り、そのパーティションを異なるストレージデバイスに記憶するコンセプトをDSSの単一のノードにおいても適用することができる。

10

【0005】

自立的な構成要素として機能し、中央のストレージ制御装置を要することなく分散ストレージシステムを形成することができる、よりインテリジェントなストレージデバイスを使用できるようにすることが所望される。

【0006】

さらには、分散ストレージシステム全体が「モノリシックストレージブロック」のように振る舞うこと、すなわちアプリケーションおよびユーザには1つの(大きな)凝集したストレージファシリティとみなされるが、内部的には個々のデバイスを備えた構造を維持することが所望される。これは慣例のストレージクラスとは異なるDSSの直交的な振る舞いである。何故ならば、各ストレージデバイスはDSS内で別個のタスクを有する個々のデバイスのままであるが、外部からはDSSをデータの格納の際に1つのユニットとしてアドレッシングすることができる、すなわち外部からでなくDSS内において個々のデバイスをアドレッシングする必要はないからである。その結果、ストレージデバイスを自由にDSSに接続することができ、またDSSから切断することができ、ストレージデバイスは有効データ、例えば映画のようなオーディオビジュアル(AV)データを持ち込むまたは受け取ることができる。有利には、DSS内のデバイスはクライアントサーバ構造ではないが、例えばピア・ツー・ピア(P2P)ネットワークの場合のようにクライアントとサーバの両方の役割を果たすことができる。

20

30

【発明の開示】

【発明が解決しようとする課題】

【0007】

従って本発明の課題は、自立的な構成要素として機能し、中央のストレージ制御装置を要することなく分散ストレージシステムを形成することができ、よりインテリジェントなストレージデバイスを使用できるようにし、さらには、分散ストレージシステム全体が「モノリシックストレージブロック」のように振る舞うこと、すなわちアプリケーションおよびユーザには1つの(大きな)凝集したストレージファシリティとみなされるが、内部的には個々のデバイスを備えた構造を維持することができる、ネットワーク内のノードを自動的に分類する方法および装置を提供することである。

40

【課題を解決するための手段】

【0008】

方法に関する課題は、この方法が、少なくとも1つのデータクラスを検出するステップを有し、データクラスに従いノードに格納されているデータを自動的に分類するステップを有し、データクラスに従い分類されたデータの量を検出するステップを有し、ここでファイルの数およびこのファイルの平均サイズを検出し、データクラスに分類されているデータの検出された量から優先値を計算するステップを有し、この優先値はデータクラスに従いノードに格納されているデータの相対的な優先度に対する尺度であり、データクラスに従いデータタイプを表すリクエストを受信するステップを有し、データクラスに従いデ

50

ータの相対的な優先度に対する尺度である優先値をリクエストに応じて提供するステップを有することにより解決される。

【0009】

装置に関する課題は、この装置が、少なくとも1つのデータクラスを検出する手段を有し、データクラスに従いノードに格納されているデータを自動的に分類する手段を有し、データクラスに従い分類されたデータの量を検出する手段を有し、ここでファイルの数およびこのファイルの平均サイズが検出され、データクラスに分類されているデータの検出された量から優先値を計算する手段を有し、この優先値はデータクラスに従いノードに格納されているデータの相対的な優先度に対する尺度であり、データクラスに従いデータタイプを表すリクエストを受信する手段を有し、データクラスに従いデータの相対的な優先度に対する尺度である優先値をリクエストに応じて提供する手段を有することにより解決される。

10

【発明を実施するための最良の形態】

【0010】

本発明により構成されており、且つモノリシックストレージブロックのように振る舞うDSSネットワークにおいては、ストレージデバイスに関連する優先的なデータタイプが求められる。この情報は特定のデータタイプのデータが格納されるべき場合にストレージデバイスを選択するために使用される。またこの情報は特定のデータタイプのデータがサーチされる場合にも使用することができる：このケースではそのような情報を特定のタイプのデータが大量に格納されているストレージデバイスにおいて優先的に/最初にサーチ

20

【0011】

本発明はネットワーク内のノードを自動的に分類する方法を提供し、この方法は以下のステップを有する：

- 少なくとも1つのデータクラス、例えばビデオ、オーディオまたはテキストを検出するステップ。
- 前述のデータクラスに従いノードに格納されているデータを自動的に分類するステップ。
- 前述のデータクラスに従い分類されたデータの量を検出するステップ。この際ファイルの数およびこのファイルの平均サイズが検出される。
- 前述のデータクラスに分類されているデータの検出された量からデータタイプ優先値を計算するステップ。この優先値は前述のデータクラスに従いノードに格納されているデータの相対的な優先度に対する尺度である。
- 前述のデータクラスに従いデータタイプを表すリクエストを受信するステップ。
- 前述のデータクラスに従いデータの相対的な優先度に対する尺度である前述の優先値を前述のリクエストに応じて提供するステップ。

30

【0012】

さらに本発明によれば、ネットワーク内のノードを自動的に編成する方法は以下のステップを有する：

- データを格納するためのリクエストを受信するステップ。ここでリクエストは所定のデータクラスに従いデータタイプを表す。
- 複数のストレージノードにリクエストを伝送するステップ。
- ストレージノードからそれぞれの優先値を受信するステップ。この優先値は前述のデータクラスに従いそれぞれのノードに格納されているデータの相対的な優先度に対する尺度である。
- それぞれのノードの優先値を比較するステップ。
- データを格納するために最も高い優先値を有するノードを選択するステップ。

40

【0013】

さらに本発明による方法を、ファイルサイズとファイルの数の組み合わせに関してデータの量が分類されることによりさらに特定することができる。

50

【0014】

さらに本発明による方法を、データタイプに関連するノードの優先値が次式により計算されることによって特定することができる。

$$E_{\text{dominant,type}} = R P S C * A_{\text{total}} * k_1$$

ここで、

$$A_{\text{total}} = N_{\text{total}} * S_{\text{total}}$$

$$R P S C = k_2 - k_3 (k_4 + S_{\text{average}} + C_{\text{used}}) / C_{\text{total}}$$

ここで、 k_1, k_2, k_3, k_4 は定数であり、 N_{total} は前述のデータタイプのデータファイルの総数であり、 S_{total} はデータタイプのファイルの累算されたサイズであり、 S_{average} はデータタイプのファイルの平均サイズであり、 C_{used} はノードの占有されている記憶容量であり、 C_{total} はノードの総記憶容量である。 10

【0015】

1つの実施形態においては、データクラスの分類および/または検出はデータに関連するメタデータに従い行われる。しかしながら別の実施形態においては、データクラスの分類および/または検出は他の指示、例えばファイル拡張またはファイルヘッダ分析に従い行われる。

【0016】

1つの実施形態においては、ノードがピア・ツー・ピアネットワークにおけるピアである。

【0017】

本発明の1つの実施形態においては、ノードの選択はノードの相対的な空き記憶容量 ($R F S C$) を基礎としており、この相対的な空き記憶容量は次式により計算される。 20

$$R F S C = c_1 - c_2 (c_3 + S_{\text{in_data}} + C_{\text{used}}) / C_{\text{total}}$$

ここで c_1, c_2, c_3 は定数であり、 $S_{\text{in_data}}$ は格納すべきデータのサイズであり、 C_{used} はノードの占有されている記憶容量であり、 C_{total} はノードの総記憶容量である。

【0018】

本発明によるネットワークプロトコルコマンドは優先値 ($E_{\text{dominant,type}}$) の要求または提供に適しており、ここで要求または提供された優先値はデータタイプおよびネットワークノードに関し、またコマンドはデータタイプの指示を包含する。特定のノードの優先値は次式に従い計算される。 30

$$E_{\text{dominant,type}} = R P S C * A_{\text{total}} * k_1$$

ここで、

$$A_{\text{total}} = N_{\text{total}} * S_{\text{total}}$$

$$R P S C = k_2 - k_3 (k_4 + S_{\text{average}} + C_{\text{used}}) / C_{\text{total}}$$

ここで、 k_1, k_2, k_3, k_4 は定数であり、 N_{total} はノードに格納された前述のデータタイプのデータファイルの総数であり、 S_{total} はノードに格納されたデータタイプのファイルの累算されたサイズであり、 S_{average} はノードに格納されたデータタイプのファイルの平均サイズであり、 C_{used} はノードの占有されている記憶容量であり、 C_{total} はノードの総記憶容量である。単純なケースにおいて、定数は $k_1 = k_2 = k_3 = 1$ 且つ $k_4 = 0$ である。 40

【0019】

ネットワーク内のノードを自動的に分類するための相応の装置は以下の手段を包含する：

- 少なくとも1つのデータクラス (例えばビデオ、オーディオまたはテキスト) を検出する手段。
- 前述のデータクラスに従いノードに格納されているデータを自動的に分類する手段。
- 前述のデータクラスに従い分類されたデータの量を検出する手段。ここで、ファイルの数およびこのファイルの平均サイズが検出される。
- 前述のデータクラスに分類されているデータの検出された量からデータタイプ優先値を計算する手段。この優先値は前述のデータクラスに従いノードに格納されているデータの 50

相対的な優先度に対する尺度である。

- 前述のデータクラスに従いデータタイプを表すリクエストを受信する手段。
- 前述のデータクラスに従いデータの相対的な優先度に対する尺度である前述の優先値を前述のリクエストに応じて提供する手段。

【0020】

さらに本発明によれば、ネットワーク内のノードを自動的に編成する装置は以下の手段を包含する：

- データを格納するためのリクエストを受信する手段。リクエストは所定のデータクラスに従いデータタイプを表す。
- 複数のストレージノードにリクエストを伝送する手段。
- ストレージノードからそれぞれの優先値を受信する手段。この優先値は前述のデータクラスに従いそれぞれのノードに格納されているデータの相対的な優先度に対する尺度である。
- それぞれのノードの優先値を比較する手段。
- データを格納するために最も高い優先値を有するノードを選択する手段。

10

【0021】

本発明の有利な実施形態は従属請求項、以下の説明ならびに図面に記載されている。

【実施例】

【0022】

分散ストレージシステム(DSS)が個々のストレージデバイスから構築される場合には、DSSの2つの異なる直交的な振る舞いを選択することができる。つまり、データが格納されるべきデバイスをアプリケーションが選択するか(要求されたストレージターゲット)、DSS全体が1つの大きな凝集したストレージファシリティとみなされ、1つのユニットのように振る舞うかである。これを本明細書においては「モノリシックストレージブロック」と称する。

20

【0023】

アプリケーションがデータを格納するために所定のストレージデバイスを選択することを意図している場合には、DSS内のノードが先ず自身のノードの状態およびレイバリティに関する情報をアプリケーションに提供し、それによりアプリケーションはデータを何処に格納するかを決定することができる。他方ユーザは何処にコンテンツが記憶されるかには関心がないので、ユーザは便利な自動的な「記憶管理」に関心を持つ可能性がある。このケースにおいては、DSSがモノリシックストレージブロックのように振る舞えば有利である。

30

【0024】

要求されたストレージターゲットのコンセプトにおいて、1つの明示的なノードがデータを格納するためのターゲットとして要求されると、DSS内のノードは自身のリソースに関する適切な情報を供給する。この場合にはアプリケーションは適切なやり方で決定することができるが、他方ではDSS内のノードは適切なシステム状態およびレイバリティデータを供給すべきである。

【0025】

モノリシックブロックのコンセプトにおいて明示的なノードがデータを格納するためのターゲットとして要求されていない場合には、DSSは1つのユニットのようにアクセス可能である。アプリケーションはDSSを記憶容量のモノリシックブロックとみなす。DSS自体は自律的に、空き記憶容量および十分な伝送レートにマッチするストレージデバイスを探す。これはネットワーク上で実現されており、且つネットワーク内の各(記憶)ノードによって遵守されている規則のセットに基づき実現される。

40

【0026】

アプリケーションはノードの中間制御層(ICL)を介してDSSにアクセスする。ICLはアプリケーションによって使用されるモードを収容し、また対応するアプリケーションインタフェースを提供する。ICLはそのインタラクションによってDSSの論理シ

50

エルを形成する。図1はアプリケーションがどのようにDSSを見るかを示す。モードは実際のところ制御メッセージシーケンスの明確なプロシージャである。以下では種々のモードを先ずは、選択されたノードに直接アクセスすることができる、いわゆる「要求されたストレージターゲット」に関して説明する。次に「ファイル分割」機能を説明する。このファイル分割機能は例えばレコーディング装置の容量が無い場合のレコーディングの実行を可能にする。

【0027】

ノードおよびエッジの数が増すに連れネットワークの複雑性も急激に増すので、そのような着実に増していく複雑性を検討し続けることは非現実的であるか、それどころか不可能なタスクである。消費者に実際に付加される値を構成するために、ストレージネットワークは自身を編成すべきである。以下では、アプリケーションが単一の凝集されたストレージブロックとしてのストレージネットワークにアクセスすることができる「モノリシックストレージの振る舞い」の概略を述べる。「モノリシックストレージの振る舞い」は最も複雑なシステムモードであるが、ユーザに対しては最善の利便性を提供する。何故ならば、このモードにおいてはシステムが可能な限り自己管理的なやり方で動作するからである。このモードにおいてはアプリケーションがストレージネットワークを1つの凝集されたストレージブロックと「みなす」。2つの極端なケースを考慮することができる。つまり、システムは「無秩序な」システムとして振る舞うか、「良好にソートされたノード自己ソート型の」システムとして振る舞う。前者の場合には、全てのコンテンツが可能とされる場所であればどこでも格納されるが、後者の場合にはコンテンツが以下例示的に定義するような規則に従いソートされて格納される。本発明は、単純性の原理に従い、これら両方の機能の態様を参照する。

10

20

【0028】

「要求されたストレージターゲット」モードは直接的なアプローチであり、ここでは完備性についてのみ言及する。ユーザがDSSに特定のストレージデバイスにデータを格納することを要求する場合には、所望のストレージデバイスに直接的にアドレッシングする手段を利用することができる。このモードが図2に示されており、ここではアプリケーション（またはユーザ）が特定のデバイスAにデータを格納することを決定する。この非常に基本的なモードはユーザにストレージネットワークを管理することを要求し、したがって煩雑である。

30

【0029】

以下では、モノリシックブロックアプリケーションを説明する。

【0030】

DSSにおいてはストレージデバイスを管理するための集中サーバが存在しないと想定する。各ストレージデバイスはサーバに依存せずに機能する。デバイスは例えばピア・ツー・ピアネットワークの場合のようなメッセージ、通常はリクエストおよび応答を介して通信する。制御メッセージを交換することによって非集中ネットワークデバイスは相互に通信することができ、また自身を編成することができる。「モノリシックブロック」のコンセプトは論理的なグループを形成するストレージノードのグループを参照し、ここでDSSの記憶容量は外部からは包括的なものとみなされ、また個々のノードは外部からは区別されない。その代わりに、ストレージは各ストレージデバイスが従う一定の規則に従い管理される。そのような規則は例示的に以下のように定義される。ストレージノードのグループが1つのユニット、すなわちモノリシックブロックのように振る舞うことによって、それらの規則をDSSにおいてソフトウェアで（中間層として）実現することができる。

40

【0031】

本発明はこのことをどのように実施することができるか、また記憶および検索を要求するアプリケーションとDSSとの間の協働がどのように行われるかを表す。さらに本発明は、モノリシックブロックの外部から到来するストレージリクエストがどのように処理されるか、モノリシックブロックにおけるノードの内の1つから到来するストレージリクエ

50

ストがどのように処理されるか、またどのメッセージおよび制御メタデータをこのコンセプトの実現のために使用することができるかを説明する。

【0032】

本発明によれば、表1に示されている層を区別することができる。

【0033】

【表1】

アプリケーション層 (AL)
“モノシックブロック”層 (MBL)
制御言語層 (CLL)
ピア・ツー・ピア技術層 (PPL)
ネットワーク層、トランスポート層、リンク層および物理層

10

表1：層の概観

【0034】

ストレージリクエストはアプリケーション層 (AL) から開始される。これらのストレージリクエストは制御メッセージを使用して制御言語層 (CLL) において実施される。モノシックブロック層 (MBL) はこれらの層間のリンクを形成し、したがってノード間の通信を管理することによって、また制御メタデータを適切に使用することによってモノシックブロックのコンセプトを実現する。インタフェースは適切に層間に存在すると想定する。例えば AL - MBL インタフェースおよび CLL - PPL インタフェース。メッセージおよび制御メタデータはピア・ツー・ピア技術層 (PPL) に関連付けられている。

20

【0035】

MBL は全ての関連するストレージデバイスを単一のストレージブロックとみなす仮想効果をユーザに提供する。事前定義されたデフォルトマッチングストラテジに基づき、MBL は到来するデータをストレージデバイスに自動的に割り当てるために保存する。

30

【0036】

例えば、本発明の1つの実施形態においては、ネットワークに接続されている全てのストレージデバイスがモノシックブロック (MB) を形成し、ストレージリクエストはネットワークにおけるノードの内の1つから到来する。ネットワークは例えば、欧州特許明細書 EP1427149号に記載されているような、オーナーゾーン (OwnerZone) として実現されているホームネットワークでもよい。このケースにおいては、ネットワークまたはオーナーゾーンにおけるノードの内の1つにおいて実行されるアプリケーションが MB 上のコンテンツの部分を記録することを要求する。すなわち要求を行うノード自体も含めて、オーナーゾーン内のどこかにコンテンツの部分が記録され、この際アプリケーションはコンテンツが何処に格納されるかを知らない。

40

【0037】

外部からみるとホームネットワークでは、ホームネットワーク内の DSS にコンテンツの部分を記憶することが許可されているか否かという問題が生じる。この問題は例えば、携帯装置またはモバイル装置を介して遠隔地から家にアクセスする場合に解決する必要がある。遠隔装置またはノードを、少なくともホームネットワークにアクセスしている時間にわたり、許可されたものとして識別できる場合、例えばホームネットワークの一部として識別できる場合には、それらの遠隔装置またはノードはホームネットワーク内に存在し、またネットワークノード、したがって DSS にアクセスすることができる。

【0038】

50

以下では、MBアプリケーションのストラテジの概略を述べる。

【0039】

MBアプリケーションに関する一般的なタスクはデータ記憶およびデータ検索である。すなわち到来するデータをモノリシックストレージブロックに保存し、モノリシックストレージブロックから保存されたデータを検索する。MBアプリケーションのデータ記憶機能はデータ検索よりも高度なアルゴリズムを必要とする。

【0040】

MBシステムに到来するデータを保存している間に、理想的な場合にはデータを直接的に所定の宛先ストレージデバイス上に良好にソートして保存することができる。したがってこの保存を本明細書においては「ソートされた記憶」と称する。しかしながらユーザはむしろ「無秩序なストック管理」でデータを格納することを所望することもある。適切な時点において、またはユーザ取り決めに応じて、一時的な記憶データがより良い宛先デバイスへとさらにソートされる。

【0041】

無秩序なストック管理および良好にソートされた記憶は例示的に事前定義された2つのモノリシックストレージストラテジである。両方のストラテジをシステム定義型のストレージデバイスマッチング判定基準によって実現することができる。

【0042】

「無秩序な」ストック管理に関するデバイスのサーチは良好にソートされた記憶に関する場合よりも簡潔且つ迅速である。「無秩序な」ストック管理とはMB内のデータがいずれかの判定基準、例えば論理的なマッチング判定基準によってソートされていないことを意味している。ストレージデバイスの幾つかの基本的で物理的な特性（例えば記憶容量、伝送レート）のみが考慮される。したがって、無秩序なストレージは何らかのやり方で決定されるが、良好にソートされるケースよりも構造化されていない。しかしながら術語「無秩序」はここでは単純性に対して使用される。無秩序なストック管理に関しては、ストレージデバイスがMBアプリケーションの物理的なマッチング判定基準を満たすことができる場合にはこのストレージデバイスを選択することができる。

【0043】

良好にソートされた記憶に関するデバイスのサーチの場合、システムは例えばデータが付随するメタデータ(MD)を介して、格納されているデータを分析する。分析の結果に依存して、モノリシックな振る舞いを例えば「メタデータによりなされるソートされたストレージの振る舞い」に切り換えることができ、このことはMD内で定義されているような異なるコンテンツカテゴリが別個のデバイスに記憶されることを意味する。例えば全てのスポーツイベントはドライブAに格納され、映画はドライブBに格納される。続いて、システムはソートされたコンテンツをその新たなシステム振る舞いに従い適切に再配置することができる。動作フェーズの間に、時々(例えばデータ記憶/検索統計に依存して)、システムはソートされたコンテンツを分析し、実施されている振る舞いが良好にフィットしているか否かを判定する。

【0044】

順応できていない振る舞い特性の場合には、システムがその動作モードを変更し、より適した動作モードを選択することができる。

【0045】

良好にソートされた記憶とは、MB内のデータが物理的なマッチング判定基準および論理的なマッチング判定基準の両方によってソートされることを意味している。例えば、ストレージデバイスが所定のジャンルに関連付けられる、またはビデオデータをビデオにとって優先的なデバイス内に一緒に保存することができる、またはMP3をオーディオ収集デバイスに保存することができる、など。

【0046】

ストラテジをデフォルトによって、またはアプリケーションによって、またはユーザの代わりに、またはユーザによって選択することができる。ユーザが選択するケースは、(

10

20

30

40

50

据え置き型の媒体が取り外し可能な媒体を装備している) 所定の1つまたは複数のノードをユーザが関心のある所定の対象と関連付けることをユーザが所望する場合に生じる可能性がある。事前に割り当てられていないノードをデフォルトに従い、またはユーザの代わりにアプリケーションによって選択された他の規則に従い処理することができる。

【0047】

2つのストラテジをシステム要求に従い相互に変換することができる。MBシステムの無秩序なストック管理状態をメンテナンスプロセスによって良好にソートされた記憶状態に変換することができる。良好にソートされた記憶状態を、論理的なマッチングプロセスおよび特定の物理的なマッチングプロセスを放棄することによって無秩序なストック管理状態に変更することができる。

10

【0048】

無秩序なストック管理および良好にソートされた記憶の両方に関して、デバイスサーチプロセスはMBシステムのコアタスクである。適切な宛先デバイスを発見するために、到来するデータとストレージデバイスの両方の物理的なパラメータおよび論理的なパラメータが評価される。無秩序なストック管理に関しては、単純化された物理的なマッチング判定基準のみが必要とされる。良好にソートされた記憶に関しては、物理的なマッチングプロセスと論理的なマッチングプロセスの両方が実行される。

【0049】

以下では、図4に示されているような、到来するデータのために適切なストレージデバイスを評価するプロシージャを「デバイスマッチング」と称する。一般的にデバイスマッチングプロセスは物理的な判定と論理的な判定を含む。物理的な判定に関しては、到来するデータおよびストレージデバイスの物理的なパラメータが考慮される。論理的な判定に関しては、到来するデータ、ストレージデバイスのMDの特徴、また他のMDソースからのMDの特徴がデバイスサーチのために考慮される。

20

【0050】

物理的な判定の適用時には以下の物理的なパラメータが評価される：ノードの空き記憶容量、ノードの自由に利用可能な伝送レートおよびノードの優先的なデータタイプ。ここでは、これらの全てのパラメータが物理的なパラメータとして処理される。

【0051】

論理的な判定においては、記述的なメタデータが評価される。モノリシックなストレージシステムに関して割り当てられた記述的なMDカテゴリが規定され、またこのMDカテゴリを例えば以下詳細に説明するようにALにおいて使用されるパラメータテンプレートに包含させることができる。

30

【0052】

これら全ての実現は制御言語を用いることにより可能である。MBLソフトウェアは関連する全てのノード、すなわち(論理的な)ネットワーク、例えばオーナーゾーン内の全てのノードでは同一であると想定する。アプリケーションが動作を実行するノードにおけるMBLソフトウェアは関連するプロセスを管理する。すなわち、MBのコンセプトは各ノードのMBLソフトウェアとして実現されている規則のセットによって実現され、また一般的に動作を実行するノードによって適用される。

40

【0053】

モノリシックストレージシステム ストレージプリファレンス

モノリシックストレージシステムは、無秩序なストック管理および良好にソートされた記憶の両方に関して、システムプリファレンスおよびユーザプリファレンスの制御下で動作する。

【0054】

システムプリファレンスに従いシステムの振る舞いは事前に、例えば製造時または購入時に定義され、またシステムはデフォルトパラメータセッティングに基づき自動的に実行することができる。特定のシステムコンフィギュレーションの生存期間の間に、全体のシステムの振る舞い、つまり無秩序なストック管理が良好にソートされた記憶を制御するた

50

めに、1つのシステムプリファレンスのみを適用することができる。システムプリファレンスをユーザプリファレンスに置き換えることができる。

【0055】

システムのモノリシックな振る舞いに関する何らかのユーザプリファレンスが存在する場合には、プリファレンスを自動的なセッティングに可能なときに何時でも置き換える。システムの特徴的な振る舞いのユーザのプリファレンスを手動でALに入力することができる。

【0056】

モノリシックブロックアプリケーション セットアップ

無秩序なストック管理に関しては、MBアプリケーションのシステムセットアップがALにおいて実施される。図5に示されているように、システムセットアップはユーザプリファレンスの入力から構成される。

10

【0057】

良好にソートされた記憶に関しても、MBアプリケーションのシステムセットアップがALにおいて実施される。ここでシステムセットアップはユーザの好みの入力、デバイス定義およびシステムメンテナンスを包含する。システムセットアッププロシージャはユーザにモノリシックブロックシステムへのAPIを提供する。初期システム状態はシステムセットアップ段階の間に定義される。システムのフィードバックに応じて、MBアプリケーションの比較的長い生存期間を達成するために、規則的なシステムメンテナンスを実行する。

20

【0058】

ユーザプリファレンス入力

ユーザプリファレンス入力は無秩序なストック管理においても良好にソートされた記憶においても行われる。ユーザプリファレンスを入力している間にパラメータ制御テンプレートがユーザに提示され、このテンプレートはユーザの入力に従い満たされる。テンプレートのカテゴリは目下のMBシステム規則に従い事前定義されている。テンプレートのパラメータカテゴリを用いることにより、ユーザは好みの物理的なパラメータを入力することができるか、(良好にソートされた記憶に関しては)論理的なパラメータ、例えば記憶場所、デバイスプロパティ、メディアデータのタイプおよび記述的なMDを、ユーザ自身の関心に従いデバイスマッチングプロシージャを制御するために入力することができる。

ユーザの好みのパラメータが存在しない場合、または考慮されない場合には、テンプレートはシステム利用に関するデフォルトの物理的または論理的なパラメータのグループを使用することができる。テンプレートにおけるパラメータはシステムの物理的および論理的なマッチングプロセスによって使用される。

30

【0059】

無秩序なストック管理に関しては、単純化されたパラメータ制御テンプレートが物理的なパラメータを入力するために使用される。良好にソートされた記憶に関しては、物理的なパラメータも論理的なパラメータも包含する完全なパラメータ制御テンプレートが使用される。

【0060】

良好にソートされた記憶ストラテジに関してはユーザプリファレンスを入力している間に、ユーザは制御デバイス初期化プロシージャを決定することもできる。ユーザはデバイスラベルカテゴリを定義または変更することができ、また空のデバイスにラベルを付与することもできる。

40

【0061】

デバイス初期化

デバイス初期化機能は良好にソートされた記憶ストラテジにおいてのみ存在する。システム初期化の間に、目下の分散しているデバイスの全てのデバイスプロパティが収集され、全てのデバイスにはデバイスプリファレンスMD、例えばデータタイプおよび他の記述的なメタデータによってラベルが付与される。空のデバイスに関しては、デバイスプリフ

50

ァレンスラベルが目下のデバイスプロパティに従いユーザによって直接定義される。占有されているデバイスに関しては、デバイスプリファレンスがMBシステムによって計算される。

【0062】

自動的なシステムストレージマッピングに関しては、ストレージデバイスの優先的なデータにはシステム初期化段階の間にラベルが付与される。目下のシステムに対して使用することができるラベルの全てのカテゴリは事前定義されるべきである。通常の場合、これらのラベルのカテゴリは記述的なMDであり、この記述的なMDはデバイスの優先的なデータの種類、例えばデータフォーマット、ジャンル、コンテンツタイトルなどを表す。異なるMBシステムに関しては、異なる種類のラベルカテゴリを設計することができる。初期化段階の間に、各モノリシックブロックシステムはそのアプリケーションに対して1種類のラベルカテゴリを選択することができる。

10

【0063】

1つの実施形態においては、システム内の目下の全てのストレージデバイスに少なくとも1つのラベルカテゴリに従いラベルが付与される。空のストレージデバイスに関しては、ユーザはデバイスの記憶容量、伝送レートおよび他の特性を分析し、続いてデバイスにラベルカテゴリにおいて定義されているラベルを付与することができる。マルチメディアシステムアプリケーションに関しては、重要なラベルカテゴリは「データタイプ」であり、したがってこのラベルカテゴリがデフォルトとして使用される。例えばデータはそのデータタイプ、例えばビデオ、オーディオ、テキストに従いストレージデバイスにおいてソートされ、そのようなデータタイプはさらにDVD（映画）、MP3（オーディオ）、XML（汎用）、DOC（テキスト）などに細分化される。

20

【0064】

占有されているデバイス、すなわちデータが格納されているデバイスに関しては、その優先的なデータタイプが計算される。占有されているデバイスにはその優先的なデータタイプを表すラベルが付与される。以下ではこのことを詳述する。

【0065】

デバイス初期化段階はMBシステムの初期状態を形成する。後に、システムアプリケーションの間にシステム状態を変更し、システムメンテナンスプロセスによって操作することができる。

30

【0066】

システムメンテナンス

DSSのメンテナンスはMBシステムの動作状態を維持するために良好にソートされた記憶ストラテジにおいて使用される。2種類のシステムメンテナンスが存在する、すなわち能動的なシステムメンテナンスと受動的なシステムメンテナンスである。能動的なシステムメンテナンスは事前定義された周期またはシステム状態に従い規則的に実行される。受動的なメンテナンスは、いずれかのシステム限界に達したときに、MBシステムのリクエストによって起動される。

【0067】

メンテナンスブロックがメンテナンスリクエストを受信すると、図6に示されているように、メンテナンスオーガナイザが全てのストレージデバイスを自動的に分析し、新たなデバイスラベルをストレージデバイスに割り当てる。デバイスの分析後に、メンテナンスオーガナイザはユーザに対するメンテナンス指示を形成することもでき、ユーザは例えば古いデータ、ソートされているデータを削除することによって、もしくは新たな記憶ディスクを購入することによってシステムを「手動で」改善することができる。

40

【0068】

占有されているストレージデバイスにおける優先的なデータタイプ

良好にソートされた記憶に関するデバイス初期化の間の重要なタスクは占有されているストレージデバイスにおける優先的なデータタイプを発見することである。占有されているデバイスへのラベル付与は空のデバイスへのラベル付与とは異なり、そのデバイスの容

50

量および特性に応じてユーザによってラベルを付与することができる。占有されているストレージデバイスが既に、ユーザ自身の関心に従いユーザによって最適に使用されていると想定する。システムが実施しなければならないことは占有されているデバイスにおける優先的なデータタイプを発見することである。本発明によれば、以下の優先付けストラテジが使用される。

【0069】

ストレージデバイスが、例えばビデオ（例えばDVD）、オーディオ（例えばMP3）、テキスト（DOC）などのような1つ以上のデータタイプのデータを包含すると想定する。

【0070】

ストレージデバイスの優先的なデータタイプとはストレージデバイスにおいて優先的なファイル総数またはファイル総サイズを有するデータタイプである。つまりこれらは両方も優先度を表すものだからである。これら2つの値の積が計算され、また優先的なデータタイプはこの積の最も高い値を有するデータタイプと解される。優先的なデータタイプはそれぞれのデバイスの優先的なデータタイプとして定義され、またデバイスにはこのデータタイプを表すラベルが付与される。

【0071】

新たに到来するデータは優先的なデータタイプおよびこのタイプに関連する潜在的な空き記憶容量に従い格納される。新たなデータが格納されるべき場合には、以下のプロセスを実施することができる：先ず、データタイプまた可能であれば新たなデータのデータサイズが検出され、メッセージがデータタイプおよびデータサイズ（または算出されたデータサイズ）を表す、接続されているストレージデバイスに送信され、これに基づき受信デバイスがリクエストを評価し、マッチングデバイス（すなわち、その優先的なデータタイプが要求されたデータタイプに等しいデバイス）が例えばその利用可能な容量および伝送帯域幅を用いて応答し、これらの応答から最高にマッチするデバイスが記憶のために選択される。

【0072】

実際のアプリケーションである実施例においては、好みのデータタイプは次式に従い計算される：

$$\begin{aligned} A_{total} &= N_{total} \times S_{total} \\ S_{average} &= S_{total} / N_{total} \\ R P S C &= 1 - (S_{average} + C_{used}) / C_{total} \\ E_{dominant} &= R P S C \times A_{total} \end{aligned}$$

ここで、

N_{total} は所定のデータタイプのデータファイルの総数であり、

S_{total} [Gバイト] は所定のデータタイプの総ファイルサイズであり、

$S_{average}$ [Gバイト] は所定のデータタイプの平均ファイルサイズであり、

C_{used} [Gバイト] はデバイスの占有されている記憶容量、すなわち全てのデータタイプの全ての S_{total} の合計であり、

C_{total} [Gバイト] はデバイスの総記憶容量であり、

A_{total} はデータの優先的な「図表領域」（下記参照）であり、ファイル数とファイルサイズの上述の積であり、

$R P S C$ は所定のデータタイプのストレージデバイス相対的潜在的記憶容量であり、ここでの正の値は同一のデータタイプの別のファイルがデバイス内でフィットすることが期待されることを意味し、大きい値は同一のタイプの別のファイルのために多くの容量が残っていることを意味し、また、

$E_{dominant}$ は優先的なデータタイプを検出するための所定のデータタイプの評価値であり、このデータタイプは $E_{dominant}$ の最も高い値を有するタイプである。

【0073】

より一般化された形態では上記の式は、

$$E_{\text{dominant,type}} = R P S C_{\text{type}} * A_{\text{total,type}} * K_1$$

$$R P S C_{\text{type}} = k_2 - k_3 (k_4 + S_{\text{average}} + C_{\text{used}}) / C_{\text{total}}$$

であり、ここで、 k_1, k_2, k_3, k_4 は定数である（または異なるノードの $E_{\text{dominant,type}}$ 値の評価時間にわたり準定数である）。上記の式の形において定数は $k_1 = k_2 = k_3 = 1$ 且つ $k_4 = 0$ である。

【0074】

A_{total} の値は、総ファイルサイズとファイル数の両方を考慮したデータタイプの量子化された量を表し、また図7に示されている矩形の領域に対応する。例示的に図7においてはストレージデバイスがビデオデータ、オーディオデータおよびテキストデータを有している。横軸はファイルサイズを表し、縦軸はそれぞれのデータタイプの全てのファイルの数を表す。総ファイルサイズは最大デバイスストレージ容量内にある。各データタイプ（ビデオ、オーディオ、テキスト）の総データ「領域」（ A_{total} ）はこのストレージデバイスにおけるデータタイププリファレンスを表す。

10

【0075】

S_{average} は目下のデバイスにおける各データタイプの単位データの平均ファイルサイズを表す。到来する未知のサイズのデータストリームに関しては、そのサイズを関連する S_{average} から推定することができる。R P S C（相対的潜在的記憶容量）は、既に格納されている同一のタイプのデータの平均ファイルサイズに従い、データタイプの予期すべき到来するデータのための目下のストレージデバイス潜在的記憶容量を評価することができる。一般的に、R P S Cはそれぞれのデータタイプのファイルサイズ統計、例えば平均ファイルサイズ、平均ファイルサイズ偏差などから計算され、また未知のサイズの到来するデータを格納できるか否かを評価するためにこのR P S Cを使用することができる。到来するデータを評価できないことが判明すると、残りのデータを他のデバイスに保存するためにファイル分割を使用することができる。

20

【0076】

最終的には、目下のデバイス内の優先的なデータタイプをR P S Cと A_{total} の乗算によって評価することができる。結果として生じる値 $E_{\text{dominant,type}}$ は、占有されているストレージデバイスにおける優先的なデータタイプが、データファイルの総数と総データサイズの両方の点において最大値を有するデータタイプであり、また目下のデバイスは新たに到来する同一タイプのデータを潜在的に受信できることを意味する。

30

【0077】

ストレージデバイスには自身の優先的なデータタイプを表すラベルが付与される。このことは、そのデータタイプのデータがこのストレージデバイスに格納するには「歓迎すべき」ものであることを意味する。

【0078】

【表 2】

	データ タイプ	サイズ S_{total} [GB]	数 N_{total}	データ “領域” A_{total}	RPSC	優先タイプ $E_{dominant}$ に関する評価値
デバイスA 200 GB	ビデオ	50	10	500	0.70	347.50
	オーディオ	5	60	300	0.72	215.88
	テキスト	1	200	200	0.72	144.00
デバイスB 200 GB	ビデオ	50	10	500	0.67	335.50
	オーディオ	10	50	500	0.69	347.00
	テキスト	1	500	500	0.69	347.50

表2：優先的なデータタイプに関する計算例

【0079】

表2は占有されているストレージデバイスにおける優先的なデータタイプに関する計算例を示す。この例においてデバイスAは200Gバイトの記憶容量を有する。この記憶容量は3つのデータタイプのグループ、すなわちビデオデータ、オーディオデータおよびテキストデータによって占有されている。「ビデオタイプ」がデバイスAの最大 $E_{dominant}$ 値を有するので、上述の式を使用してデバイスAにおける優先的なデータタイプは「ビデオタイプ」であると評価することができる。

【0080】

デバイスBはデバイスAと同じ総記憶容量を有するが、ビデオデータ、オーディオデータおよびテキストデータの数およびサイズは異なる。全てのデータタイプグループは図7によるグラフにおいては同一の「データ領域」を有する。このケースにおいては、優先的なデータタイプが $E_{dominant}$ の値によって評価される。デバイスBにおいては優先的なデータタイプは「テキスト」である。何故ならばその値 $E_{dominant, text}$ はビデオおよびオーディオに対する値 $E_{dominant, video}$ 、 $E_{dominant, audio}$ よりも高いからである。他のタイプのデータが格納されていない場合、そのような他のタイプのデータの量がビデオ、オーディオおよびテキストに比べて少なければこの他のタイプのデータを無視することができるか、別個の評価値 $E_{dominant}$ が計算される。

【0081】

この例において示したように、この章における上述の式は占有されているストレージデバイスにおける優先的なデータタイプの評価にとって有効である。

【0082】

モノリシックブロックシステム

ストレージデバイスの検索がMBLにおいて実施される。この検索はALからの入力パラメータにより制御される。到来するデータをモノリシックなストレージブロックに保存する場合には、ユーザは関連するサーチストラテジおよびパラメータをALを介して定義することができる。続いて、このストラテジおよびパラメータに従い、MBLは適切なストレージデバイスを自動的にサーチすることができる。ユーザ入力パラメータが存在しな

10

20

30

40

50

い場合には、MBシステムは適切なストレージデバイスを探すためにデフォルトマッチングストラテジに従い実行されることになる。以下これをより詳細に説明する。

【0083】

上述のシステム初期化プロセスの後に、MBLはさらにストレージデバイスサーチアクティビティを実施することができる。サーチアクティビティはデフォルトパラメータまたはユーザが割り当てたパラメータに基づき自動的に実行される。図8はモノリシックブロック(MB)システム全体の概観を示す。MBシステムを上述した2つのストレージストラテジ、すなわち無秩序なストック管理および良好にソートされた記憶の内のいずれかに従い実現することができる。両方のストラテジ形態は「ストレージデバイスの物理的な決定」および「ユーザプリファレンスの決定」と称される類似する評価プロシージャを使用する。

10

【0084】

ストレージデバイスの物理的な決定は、MBアプリケーションのためのストレージデバイスの選択において最初の役割を果たす。このストレージデバイスの物理的な決定は基本的に物理的なパラメータを評価し、また以下のステップを含む：デバイスの空き記憶容量の計算、デバイス伝送レートの計算、(良好にソートされた記憶のケースにおいては)デバイスの優先的なデータタイプの問合せ、また必要に応じてファイル分割の起動。データの量が格納の前に未知である場合、またはいずれのストレージデバイスも十分な空き容量を有していない場合には、データを種々のストレージデバイスに分散させることができる。このいわゆる「ファイル分割」モードを、図3に示されているように、DSSによって支援することができる。

20

【0085】

無秩序なストック管理と良好にソートされた記憶とではストレージデバイスの物理的な決定の異なるバージョンを使用する。つまり、無秩序なストック管理は良好にソートされた記憶に比べて単純化されたバージョンの物理的な決定を使用する。

【0086】

ユーザプリファレンスの決定はアプリケーションに関する最終的な選択の役割を果たし、したがって最終的な決定はユーザプリファレンスに残される。最終的な選択を行うために、事前定義されたデフォルトパラメータまたはユーザアップデートの論理的なパラメータ(記述的なメタデータ)が使用される。このパラメータを物理的なストレージマッチングおよび(良好にソートされた記憶に関しては)論理的なストレージマッチングのステップから構成することができる。

30

【0087】

物理的なストレージマッチングは事前定義された公式およびストラテジに基づき、デバイスの物理的なパラメータを評価する。「ストレージデバイスの物理的な決定」と「物理的なストレージマッチング」の主な相違点は、前者がデバイスの物理的なパラメータの絶対値を評価するのに対し、後者はデバイスの物理的なパラメータの相対値を評価することである。

【0088】

物理的なストレージマッチングは以下のステップを含む：到来するデータのための相対的な記憶容量の計算、(良好にソートされた記憶に関しては)事前定義されたストラテジによるシステム平衡評価、および(ユーザによって起動される場合には、良好にソートされた記憶に関しては)セーフコピーアクションの起動。

40

【0089】

良好にソートされた記憶に関して、論理的なストレージマッチングは(ALから受信した)ユーザ入力メタデータのみに従いデバイスを評価する。論理的なストレージマッチングに基づき、モノリシックなストレージシステムはユーザの全ての種類のストレージ要求を満たすことができる。

【0090】

モノリシックブロック層に関するさらなる詳細を以下説明する。

50

【0091】

無秩序なストック管理

図8に示されているように、「無秩序なストック管理」状態のデバイスをサーチするプロセスを以下の2つのステップに基づき実現することができる。すなわち、ストレージデバイスの物理的な決定およびユーザプリファレンスの決定。良好にソートされた記憶状態のデバイスに比べて、無秩序なストック管理状態のデバイスのサーチはこれらのステップに関する単純化されたプロセスのみを使用する。例えば、ストレージデバイスの物理的な決定プロセスにおいては、デバイスの絶対的な空き記憶容量および伝送レートのみが評価され、ユーザプリファレンスの決定においては、(サブカテゴリの物理的なマッチングのもとで)デバイスの相対的な空き記憶容量のみが評価される。

10

【0092】

無秩序なストック管理のデバイスに関する基本的なタスクは到来するデータのための記憶空間を提供し、他方ではよりメタデータに依存する最適なデバイス評価およびデータソートは考慮されない。

【0093】

一時的な無秩序なストック管理を適切な時点において、良好にソートされた記憶にさらに変形することができる。

【0094】

良好にソートされた記憶

良好にソートされた記憶状態のデバイスのサーチをストレージデバイスの物理的な決定およびユーザプリファレンスの決定の2つのステップで実現することができる。オペレーションのステップは無秩序なストック管理のステップと類似するが、関連するプロセスの完全なバージョンを使用する。ストレージデバイスの物理的な決定プロセスにおいては、絶対的なデバイス空き記憶容量およびデバイスの伝送レートに加えて、デバイスの優先的なデータタイプが評価され、ユーザプリファレンスの決定プロセスにおいては、(サブカテゴリの物理的なマッチングのもとでの)相対的なデバイス空き記憶容量に加え、(物理的なマッチングにおける)システムパラメータ平衡評価および論理的なマッチングが実施される。論理的なマッチングにおいては、より記述的なメタデータの特徴が考慮される。「セーフコピー」プロセスは最終的に割り当てられるストレージに対しても可能である。

20

【0095】

ストレージデバイスの物理的な決定

ストレージデバイスの物理的な決定は無秩序なストック管理および良好にソートされた記憶の両方の状態のデバイスに関する初期選択プロセスである。このプロセスは、図9に示されているように、4つのオペレーション、すなわちデバイス空き記憶容量の評価、デバイス伝送レートの評価、(良好にソートされた記憶に関しては)デバイスの優先的なデータタイプの評価、また必要に応じてファイル分割コマンドの形成により形成される。

30

【0096】

全てのストレージストラテジはデバイス空き記憶容量の評価およびデバイス伝送レートの評価を使用することができる。絶対的な空き記憶容量および伝送レートという物理的なパラメータが計算される。

40

【0097】

デバイスの優先的なデータタイプの評価は、そのデータタイプ、例えばビデオ、オーディオ(MP3)などに従い到来するデータを分類する、良好にソートされた記憶状態のデバイスをサーチする場合にのみ使用される。

【0098】

到来するデータのサイズが単一のデバイスにロードするには過度に大きい場合には、ファイル分割コマンドが形成される。このプロセスをユーザ定義によりALにおいて起動することができる。ファイル分割コマンドが形成されると、MBシステムは残りのデータを保存するための他のストレージデバイスを検出する。

【0099】

50

通常の場合、ファイル分割は自動的な機能である。しかしながら、場合によっては、ユーザがデータを分割するためのコマンドをトリガすることもできる。

【0100】

ユーザプリファレンスの決定

ユーザプリファレンスの決定は無秩序なストック管理および良好にソートされた記憶状態のデバイスのサーチの両方に関する最終的な選択プロセスである。このプロセスが図10に示されており、また物理的なストレージマッチングおよび（良好にソートされた記憶のための）論理的なストレージマッチングから構成することができる。

【0101】

これらのマッチングプロセスに対してデフォルトパラメータおよびアップデートされたユーザ入力パラメータの両方を適用することができる。デフォルトパラメータはMBアプリケーションによって事前定義されている。これらの事前定義されたパラメータをALにおいて使用されるパラメータ制御テンプレートに含ませることができる。これらのデフォルトパラメータに基づき、MBシステムはいかなるユーザコマンドを要することなくマッチングタスクを自動的に実行することができる。ユーザはパラメータテンプレートにおける関連付けられた値を変更することによりデフォルトパラメータをアップデートすることができる。目下のユーザ入力パラメータにより、ユーザはこのユーザの優先的なストレージデバイスを発見するためにシステムを制御することができる。

【0102】

物理的なストレージマッチング

物理的なストレージマッチングは、図11に示されているように、以下の機能を含む。すなわち、相対的な空き記憶容量（RFSC）の評価、システムパラメータ平衡の評価、およびセーフコピーコマンドの形成。

【0103】

相対的な空き記憶容量（RFSC）の評価プロセスは、無秩序なストック管理および良好にソートされた記憶の両方によって使用される。上述のRPSC（相対的潜在的記憶容量）と同様に、RFSCは到来するデータのデータタイプに関する以下の式により計算される：

$$RFSC = 1 - (S_{in_data} + C_{used}) / C_{total}$$

ここで、

C_{used} [GB] はデバイスの使用されている記憶容量であり、

C_{total} [GB] はデバイスの総記憶容量であり、

S_{in_data} [GB] は所定のデータタイプを有する到来するデータのサイズであり、

RFSCはこのデータタイプに対するストレージデバイスの相対的な空き記憶容量である。

【0104】

より一般化された形では上記の式は以下ようになる。

$$RFSC = c_1 - c_2 (c_3 + S_{in_data} + C_{used}) / C_{total}$$

ここで、 c_1 、 c_2 、 c_3 は定数である（または異なるノードのRFSC値の評価時間にわたり準定数である）。

【0105】

上記の式の形においては、定数は $k_1 = k_2 = k_3 = 1$ 且つ $k_4 = 0$ である。

【0106】

システムパラメータ平衡の評価は良好にソートされた記憶によってのみ使用される。このシステムパラメータ平衡の評価は基本的なパラメータに基づき複雑なデバイスマッチングストラテジを定義することができる。例えば、上述の相対的な空き記憶容量の評価においては、最大のRFSC値を有するデバイスが到来するデータに対して最高の伝送レートを提供しない可能性がある。このケースにおいては、より低いRFSC値を有するが、適切な伝送レートを有する他のデバイスを到来するデータにとっての最善のストレージ候補として選択することができる。システムパラメータ平衡の評価においては、ユーザがMB

10

20

30

40

50

アプリケーションに対して異なる実用的なストラテジを定義することができる。

【0107】

物理的なストレージマッチングの場合においてセーフコピー機能がユーザによってALにおいて起動される場合には、このセーフコピー機能はセーフコピーコマンドを形成し、到来するデータを保存するための他のストレージデバイスを発見するために上述のデバイスサーチプロセスを再び実行する。この付加的なデータ記憶は、必要に応じて、後のディザスタリカバリに対して有効である。

【0108】

表3は相対的な空き記憶容量に従いデバイスを選択するための例を示す。デバイスAは200GBバイトの総記憶容量を有し、またその優先的なデータタイプは「ビデオ」である。デバイスBは100GBバイトの総記憶容量を有し、またその優先的なデータタイプはやはり「ビデオ」である。10GBバイトの到来するデータをデバイスAに保存する場合には、デバイスAにおける到来するデータのRPSC値は0.67である。10GBバイトの到来するデータをデバイスBに保存する場合には、デバイスBにおける到来するデータのRPSC値は0.72である。したがって、到来するデータに対してデバイスBはデバイスAよりも多くの相対的な空き記憶容量を提供するので、デバイスBが選択されることになる。

【0109】

【表3】

	データ タイプ	サイズ S_{total} [GB]	数 N_{total}	データ “領域” A_{total}	RPSC または RFSC	優先タイプ $E_{dominant}$ に関する評価値
デバイスA 200 GB	ビデオ	50	10	500	0.70	347.50
	オーディオ	5	60	300	0.72	215.88
	テキスト	1	200	200	0.72	144.00
到来するデータ		10	1		0.67	
デバイスB 100 GB	ビデオ	15	5	75	0.79	59.25
	オーディオ	2	30	60	0.82	49.16
	テキスト	1	50	50	0.82	40.99
到来するデータ		10	1		0.72	

表3：デバイスの相対的な空き記憶容量（RFSC）を評価するための例

【0110】

上記の例は、上述の式があらゆる到来するデータに対するストレージデバイスの選択に関する量的な評価を提供することを示す。

【0111】

論理的なストレージマッチング

物理的なストレージマッチングの後に、候補ストレージデバイスを論理的なストレージマッチングプロセスによってさらにフィルタリングすることができる。論理的なストレージマッチングは、到来するデータを適切なデバイスに配置するための記述的なメタデータを使用するために設計されている。

【0112】

原理的にはMDベースの全ての種類のマッチングが許容される。事前定義されたMDマッチングストラテジおよびアップデートされたMDマッチングストラテジの両方を論理的なストレージマッチングに適用することができる。論理的なストレージマッチングストラテジの目的は、ユーザが良好にソートされた記憶状態のデバイスにデータを保存することができる最大限の機会を提供することにある。

【0113】

実際のシステムアプリケーションに関してはメタデータベースのマッチング方法が考えられ、これは図12に示されている。システム初期化段階においては、ユーザは好みのMDを入力することができるか、好みのMDコンテナに直接リンクすることができるか、その結果コンテナ内のMDコンテンツはユーザの好みのMDとして読み取られることになる。アプリケーション層においては、ユーザは上述のMDコンテンツを入力するためにパラメータ制御テンプレートを使用することができる。システムが許容できる全ての記述的なMDカテゴリをパラメータテンプレートにおいて事前定義することができる。パラメータテンプレートはデフォルトMDコンテンツも含むことができる。アップデートされたユーザの好みのMDが設定されていない場合には、システムは論理的なストレージマッチングプロセスを実行するためにデフォルトMDを使用する。

【0114】

図12に示されているように、MBシステムはユーザプリファレンスと同じメタデータによってラベルが付与されているストレージデバイスをサーチする。最終的に、到来するデータは物理的なストレージ要求および論理的なストレージ要求の両方を満たすストレージデバイスに保存される。

【0115】

例示的なシナリオおよびそのメッセージ並びにメタデータの流れ

以下では、分散的なストレージ管理のために要求されるシステムモデルおよび制御言語の実現可能性を実証するためにDSSの考えられるアプリケーションを表す幾つかの単純なシナリオを、例示的な関連するメッセージおよび制御メタデータも含めて説明する。このシナリオは、図13に示されているように、分散型のストレージ用の例示的なネットワークを基礎としている。このネットワークは据え置き型のストレージデバイス/ノードS0, . . . , S (個人用デジタルレコーダ、例えばHDD、光ディスク)およびポータブルなストレージデバイス/ノードPを包含する。各ノードはアプリケーションを実行することができるか、またこれらのノードはユーザインタフェース/遠隔制御部を有し、これらのユーザインタフェース/遠隔制御部を別個のデバイス/ノードとみなすこともできる。ホームネットワークに対して考えられる拡張として、チューナ/受信装置(例えばDVB-SまたはDVB-Cなど)、AVディスプレイ/出力装置、インターネットアクセス用のADSLモデムまたはゲートウェイ等を挙げることができる。ここで例示されているシナリオにおいては、ノードS0が一般的にDSSと対話するために使用される。

【0116】

シナリオ1: 容量が制限されているケースにおけるコンテンツのコピーおよびネットワーク内の記憶容量のバランスの良い使用。

【0117】

ユーザはデバイスPに格納されているコンテンツを据え置き型のストレージデバイスS0, . . . , S3のいずれかにコピーしようとする。コンテンツは最も大きい空き記憶容量を提供する据え置き型のデバイスにコピーされる。他の実施形態においては、コンテンツが空き記憶容量の最高のマッチング量を提供するデバイス、すなわち残りの容量が最も少ないデバイスにコピーされ、ギャップが最適に使用される。

【0118】

シナリオ2: 休止および再開ならびにトリックモード(早送り、スローモーション)

(例えば表示装置Pにおいてコンテンツを見るために) デバイスS1~Pに格納されているコンテンツを伝送している間に、ユーザは(S0を介して)伝送(例えば再生)を休

10

20

30

40

50

止し、数分後に再生を再開する。後に再び通常で再生している間に、ユーザは早送りまたはスローモーションのような何らかのトリック再生（サムネイルのブラウズ、ビデオグループシーンのブラウズなども考えられる）を選択する。デバイス S 1 がトリックモードケイパビリティについて検査されて、そのケイパビリティが確認されると、デバイス S 0 が早送りリクエストメッセージをデバイス S 1 に送信し、所望の早送りモードおよび速度に変更される。

【0119】

シナリオ 3：ストレージデバイス S 1 からの読み出しとストレージデバイス S 1 への書き込み（レコーディングと再生）を同時に行うマルチタスク

ノード S 1 は 2 つのタスク、すなわちノード S 2 から受信したコンテンツの部分のレコーディング、およびノード P についてコンテンツの別の部分の再生を同時に実行する。 10

【0120】

シナリオ 4：タスクの衝突および伝送遅延を伴う同時的な読み出しと書き込み

シナリオ S 2 と同様に、ユーザはデバイス S 2 から S 1 へのレコード伝送を設定しようとし、またデバイスケイパビリティはこのレコード伝送の実施することができる。しかしながら伝送を開始する前に、別のユーザがデバイス S 1 から P への別の再生伝送を開始すると、デバイス S 1 の自由に利用可能な伝送レートはデバイス S 2 から S 1 への伝送を容易にするにはもはや十分なものではなく、したがってこの伝送は拒否される。ユーザは伝送が後の時点に行われるようにスケジューリングすることを決定する。したがってデバイス S 2 から S 1 への伝送が設定される。 20

【0121】

シナリオ 5：デバイスの接続および切断

ユーザは自身のポータブルデバイス（P）をデバイス S 0，．．．，S 3 から構成されている既存のネットワークに接続し、後の時点に切断する。ソフトウェアによってトリガされる「ソフト切断」と強制的に切断される「ハード切断」は区別される。

【0122】

シナリオ 6：タスクの衝突および非リアルタイムの伝送を伴う同時的な読み出しと書き込み

伝送遅延の代わりに非リアルタイム伝送が実施される点を除いてシナリオ 4 に類似する。コンテンツはファイルとして比較的低いビットレートで伝送され、したがって比較的長い時間にわたり伝送され、記憶空間である積は一定のままである。 30

【0123】

シナリオ 7：コンテンツの分割を伴うレコーディング

いずれのノードもコンテンツ全体を格納することができないので、コンテンツの部分が種々のノードにわたりレコードおよび分散される。

【0124】

シナリオ 8：分割されたコンテンツの再生

種々のノードにわたり分散しているコンテンツの部分がシームレスに再生される。

【0125】

シナリオ 9：事後的なコンテンツの分割を伴う瞬間的なレコーディング 40

記録は 1 つのノードにおいて開始されるが、いつ終了するかは分からず、デバイスがほぼ一杯になると、レコーディングは別のノードにおいて継続される。

【0126】

シナリオ 10：コンテンツの消去

サーチおよび発見されたコンテンツの一部が消去される。

【0127】

シナリオ 11：マルチキャスト/ブロードキャスト

コンテンツの部分が種々のノードに伝送される（マルチキャスト）か、全てのノードに同時に伝送される（ブロードキャスト）。

【0128】

シナリオ 1 2 : 伝送の中断および再開

ソースまたは宛先がハード切断に起因して一時的に利用不能になるので、コンテンツの部分の伝送が中断される。3と2のケースが区別され、これらの内の2においては伝送が中断された点から再開される。

【0129】

以下の構造は例示的に上述のピア・ツー・ピア技術層、または他の対応する層を参照する。上記のシナリオの実現にあたり、異なるメッセージまたはタスクが、メッセージパラメータまたは引き数として包含されている固有の制御メタデータと共に使用される。表記を容易にするために、メッセージはメッセージ名および引き数、例えばDeviceCapabilitiesInformation(Sender, Receiver, TaskID, DeviceType, DeviceServices, MaxBitRate, FreeStorageCapacity, ...)によって表される。全てのメッセージは自身の固有のメッセージID (MessageID) を有しているが、見やすくため上記の表記においてこのメッセージIDは省略されている。

10

【0130】

各シナリオに関して、先ずシナリオの初期条件、次にシナリオの詳細な説明、最後にメッセージフローが挙げられている。

【0131】

シナリオ 1 : 容量が制限されている場合のコンテンツのコピーおよびオーナーゾーン内の記憶容量のバランスの良い使用

初期条件 : ノード S 0 , . . . , S 3 および P から成るネットワークが図 1 4 に示されているように構成され実行される。コンテンツの伝送は実施されず、全てのノードはアイドル状態にある。

20

【0132】

シナリオ : ユーザはデバイス P に格納されているコンテンツをストレージデバイス S 1 , S 2 または S 3 のいずれかにコピーしようとする。最も大きい空き記憶容量を提供する据え置き型のストレージデバイスが検出され、そのストレージデバイスにコンテンツがコピーされる。

【0133】

ユーザはコンテンツの所望の部分をサーチするためにデバイス S 0 を使用する。したがって、デバイス S 0 はサーチリクエストメッセージをネットワーク / オーナーゾーン内の全てのデバイスに送信する。デバイス P はコンテンツを有し、したがってデバイス S 0 に応答する。ただしこのシナリオに対する変形実施形態「A」として、コンテンツのサーチおよびコピーのタスクを開始するためにデバイス P をデバイス S 0 の代わりに使用することができる。このケースにおいては、ノード P はリクエストに整合するコンテンツに関する応答を自身には送信せず、ノード P は相応の情報を自身のコンテンツデータベースから受け取ることになる。

30

【0134】

ユーザはいずれかの据え置き型のストレージデバイスにコンテンツを格納しようとするので、デバイス S 0 がデバイス S 1 , S 2 および S 3 にそれらの記憶容量および伝送キャパビリティを問い合わせるために使用される。S 1 , S 2 , S 3 はそれらのデバイスキャパビリティに関して S 0 に情報を提供し、それらの全てのデバイスは自由に利用可能である十分な伝送レートを有する。この場合、デバイス S 1 に対しては空き記憶容量の制限が観測されているのに対し、デバイス S 3 は最大空き容量を提供する。したがってデバイス S 0 はデバイス P に対してコンテンツをデバイス S 3 に伝送するようリクエストし、したがってこれによりネットワーク内でよくバランスのとれた形で記憶容量を利用できるようになる。関連するデータ伝送が終了すると、デバイス P は S 3 にメッセージを通知する。コンテンツのレコーディング後には、デバイス S 3 がうまく完了したことに関する情報を S 0 に通知する。

40

【0135】

オーナーゾーン内での記憶容量のバランスのとれた利用、すなわち各ノードの記憶空間

50

の管理とは例えば、(このシナリオにおけるように)最大の絶対的な空き記憶容量または相対的な空き記憶容量または自由に利用できる最高の伝送レートを実施形態に依存して提供するノードにおいてコンテンツの部分をレコーディングすることを意味する。しかしながら、バランスのとれた記憶容量の利用は、ネットワーク内の記憶容量を管理するための1つの可能なやり方にすぎない。例えば容量が制限されているケースにおいてコンテンツをコピーする場合に、他のストラテジも同様に適用することができる。

【0136】

メッセージフロー：このシナリオにおいてはメッセージの以下のシーケンスが生じる。全てのメッセージは送信側 (Sender) および受信側 (Receiver) を含む。これらはまたそれぞれのメッセージタイプに固有のパラメータも包含する。

10

【0137】

ユーザはコンテンツの所定の部分またはタイプをサーチしようとする。ユーザ入力の結果として、デバイス S0 は以下のサーチリクエストを全てのデバイスに送信し、その際、S0 は S2 に関していくつかの事前の知識があるという理由から、あるいは殊に S2 に関心があるという理由から、S0 は特に S2 に対しメッセージを送信する。

- **ContentInfoRequest** (*Sender=NodeID(S0), Receiver=all, Receiver=NodeID(S2), TaskID=abc, TaskInitTime=2002-12-01-18:10:08.012-GMT, MessageMode="search", SearchString={Title="Octopussy"}*)

20

【0138】

全てのデバイスは、タスク ID (TaskID) の関係およびタスクに関連するパラメータを一時的に格納し、自身のデータベースをサーチする。P は要求されたコンテンツの部分を発見し、したがって以下のメッセージをデバイス S0 に送り返す。

- **ContentInfoResponse** (*Sender=NodeID(P), Receiver=NodeID(S0), TaskID=abc, MessageMode="found content", ContentID=UUID, LocationID=UUID, ContentDescription={Title="Octopussy", Summary="...", Actor="Roger Moore", Actor="Maud Adams", Actor="...", Genre="Action", Keyword="James Bond", ..., AspectRatio="16:9", ...}, Duration=2:05 h, BitRate=7 Mbps, [, more information about the content])*

30

【0139】

S0 の代わりにデバイス P がコンテンツのサーチおよびコピーのタスクを開始するために使用される、このシナリオについての変形実施形態「A」においては、このメッセージは発生せず、ノード P は単にそのコンテンツデータベースから発見されたコンテンツに関する情報を取得しようとするに過ぎない。

【0140】

「全ての」受信側が ContentInfoRequest ("search") メッセージでアドレッシングされているので、受信側はリクエストに整合するコンテンツが発見されるまでリクエストに回答する必要はない。しかしながら S2 は明示的に受信側として挙げられているので、所望のコンテンツを保持しているか否かというリクエストに対しては回答する必要がある。S2 は時々自身のデータベースをサーチする必要があり、またサーチを開始するとき以下のメッセージを S0 に送信する：

40

- **ContentInfoResponse** (*Sender=NodeID(S2), Receiver=NodeID(S0), TaskID=abc, MessageMode="searching"*)

【0141】

この場合デバイス S2 は要求されたコンテンツの部分を発見しない。デバイス S2 は「回答しなければならない」受信側として ContentInfoRequest ("search") メッセージでアド

50

レッシングされているので、デバイス S 2 内で所望のコンテンツが発見されなかったにもかかわらず、このデバイスは以下のメッセージをデバイス S 0 へ送り返す。

- **ContentInfoResponse** (Sender=NodeID(S2), Receiver=NodeID(S0), TaskID=abc, MessageMode="found content", LocationID="none")

【 0 1 4 2 】

全てのデバイスのサーチプロセスが完了する前に、ユーザが所望の何らかのコンテンツを発見した場合には、ユーザは S 0 に以下のメッセージを使用してサーチプロセスをキャンセルさせることができる。

- **CancelTaskRequest** (Sender=NodeID(S0), Receiver=all, TaskID=abc)

10

【 0 1 4 3 】

このメッセージを受信した後に全てのデバイスは自身のサーチプロセスを停止する。デバイス S 2 は「応答しなければならない」受信側としてContentInfoRequest("search")メッセージでアドレッシングされているので、CancelTaskRequest("search")リクエストの受領確認としてデバイス S 2 は以下のメッセージをデバイス S 0 へ送り返す。

- **CancelTaskResponse** (Sender=NodeID(S2), Receiver=NodeID(S0), TaskID=abc)

【 0 1 4 4 】

ContentInfoResponseメッセージをデバイス S 0 に送信した後に、ノード P および S 2 はタスク ID および関連するパラメータを自身のテンポラリメモリから削除する。同じことはCancelTaskResponseメッセージを送信した全てのデバイスについてもあてはまる。

20

【 0 1 4 5 】

この場合ユーザはサーチ結果に満足し、それらのデバイスの空き記憶容量および伝送レートを取得するために、デバイス S 0 はこのときリクエストメッセージをデバイス S 1 , S 2 , S 3 に送信し、それらのデバイスレイパビリティについて問い合わせる。デバイス S 1 , S 2 , S 3 は、自身のデバイスレイパビリティに関する情報をデバイス S 0 に提供することによって応答する。

- **DeviceCapabilitiesInfoRequest** (Sender= NodeID(S0), TaskID=bcd, Receiver=NodeID(S1))
- **DeviceCapabilitiesInformation** (Sender=NodeID(S1), Receiver=NodeID(S0), TaskID=bcd, DeviceCapabilityInformation{ DeviceType=stationary storage device, DeviceServices=record or playback, MaxCapacity=100 GB, FreeCapacity=5 GB, MaxTransferRate=30 Mbps, FreeTransferRate=20 Mbps, MaxStreams=2 [, ActiveStreams=1, Until=20:15:00:00]})
- **DeviceCapabilitiesInfoRequest** (Sender= NodeID(S0), Receiver=NodeID(S2), TaskID=cde) 10
- **DeviceCapabilitiesInformation** (Sender=NodeID(S2), Receiver=NodeID(S0), TaskID=cde, DeviceCapabilityInformation{ DeviceType=stationary storage device, DeviceServices=record or playback, MaxCapacity=50 GB, FreeCapacity=40 GB, MaxTransferRate=30 Mbps, FreeTransferRate=30 Mbps, MaxStreams=2})
- **DeviceCapabilitiesInfoRequest** (Sender= NodeID(S0), Receiver=NodeID(S3), TaskID=def)
- **DeviceCapabilitiesInformation** (Sender=NodeID(S3), Receiver=NodeID(S0), TaskID=def, DeviceCapabilityInformation{DeviceType=stationary storage device, DeviceServices=record or playback, MaxCapacity=300 GB, FreeCapacity=200 GB, MaxTransferRate=40 Mbps, FreeTransferRate=40 Mbps, MaxStreams=2}) 20

【 0 1 4 6 】

択一的に、デバイス S 0 は以下のようなRequestDeviceCapabilityメッセージをこれら 3 つの全てのノードに送信することができる。

DeviceCapabilitiesInfoRequest (Sender= NodeID(S0), Receiver=NodeID(S1), Receiver=NodeID(S2), Receiver=NodeID(S3), TaskID=bcd).

30

【 0 1 4 7 】

デバイス S 0 はデバイス S 1 , S 2 , S 3 の空き記憶容量および伝送レート进行评估する。デバイス S 1 は十分な空き記憶容量をもっていないのに対し、デバイス S 3 は最大記憶容量を提供する。ネットワーク内の据え置き型のストレージデバイスの記憶容量をバランスよく利用するために、デバイス S 0 はコンテンツをデバイス P からレコーディングするために、ユーザの対話を必要とすることなく自動的にデバイス S 3 を選択し、伝送を実行するようデバイス S 3 および P に要求する。

40

- **InitiateTransferRequest** (Sender=NodeID(S0), Receiver=NodeID(S3), Receiver=NodeID(P), TaskID=fgh, TransferPurpose="Record", Destination=NodeID(S3), Source=NodeID(P), ContentID=UUID, LocationID=UUID, ContentDescription={Title="Octopussy"}, Duration=2:05 h, [Start=00:00:00:00, End=02:05:00:00,] RequestedBitRate =7 Mbps, UseKey=Key(John's James Bond friends))

【 0 1 4 8 】

このシナリオに対する変形実施形態「A」においては、1つの受信側が省略され、メッセージが即座にスタートすることである。

50

InitiateTransferRequest (Sender = NodeID(P), Receiver = NodeID(S3), TaskID = fgh, ...).このケースでは、ノード P において必要とされるリソースを利用できる場合にのみ、ノード P に対しこの InitiateTransferRequest の送信が許可される。

【 0 1 4 9 】

このメッセージはノード P のロケーションにあるコンテンツの部分をノード S 3 に伝送し、そこにレコードすることを要求する。コンテンツ ID (ContentID) は、ノード P におけるデータのロケーションを特定する UUID である。タスク ID (TaskID) は UUID であり、これは例えば関与するデバイスのノード ID (NodeID) と、伝送すべきデータのロケーションと、タスクが開始された時点とに基づき定義することができる。デバイス P およびノードまたはデバイス S 3 がそれらの FreeTransferRate (自由に利用可能な伝送レート) によればその時点で過度にビジー状態である場合、それらは InitiateTransferResponse("denied") メッセージ (伝送応答開始「拒否」) をデバイス S 0 に送信することになり、ついで S 0 により CancelTaskRequest メッセージがデバイス P とデバイス S 3 に送信されることによりタスクがキャンセルされ、それらのデバイスは CancelTaskResponse メッセージをデバイス S 0 に送ることにより返答するか、レコーディングが後の時点で再び試みられることになるか、または DeviceCapabilitiesInformation メッセージから得られる Until (いつまで) に従い After (以降) パラメータを使用してレコーディングがスケジューリングされることになる。

10

【 0 1 5 0 】

上記のメッセージの受信後に、デバイス S 3 およびデバイス P はリクエストに対し受領確認を行い、個々のリソースを割り当てる。このシナリオに対する変形実施形態「A」においては、ノード P が受領確認メッセージを自身とデバイス S 3 には送信しない。何故ならば、それらには十分なリソースを利用可能であることが暗黙的に既知であり、したがって第 2 の以下の 2 つの InitiateTransferResponse メッセージは発生しない。この場合、ユーザは自身が管理する所定の人員グループに、自身で定義した "John's James Bond friends" というラベルでコンテンツのコピーに対するアクセスの許可を与えたいと望んでおり、それに応じて S 0 に命令する。

20

- **InitiateTransferResponse** (Sender=NodeID(S3), Receiver=NodeID(S0), Receiver=NodeID(P), TaskID=fgh, MessageMode="confirmed", TransferPurpose="Record", Destination=NodeID(S3), Source=NodeID(P), ContentID=UUID, LocationID=UUID, ContentDescription={Title="Octopussy"}, [Start=00:00:00:00, End=02:05:00:00,] ReservedBitRate=7 Mbps, UseKey=Key(John's James Bond friends))
- **InitiateTransferResponse** (Sender=NodeID(P), Receiver=NodeID(S0), Receiver=NodeID(S3), TaskID=fgh, MessageMode="confirmed", TransferPurpose="Record", Destination=NodeID(S3), Source=NodeID(P), ContentID=UUID, LocationID=UUID, ContentDescription={Title="Octopussy"}, [Start=00:00:00:00, End=02:05:00:00,] ReservedBitRate=7 Mbps, UseKey=Key(John's James Bond friends))

30

40

【 0 1 5 1 】

TransferPurpose パラメータの値が「Record」であるので、宛先ノード S 3 はデータ転送プロセスを制御することになる。続いて (または After パラメータに従い後の時点において) デバイス S 3 はデバイス P に対してそれぞれのコンテンツデータを自身に送信することを要求する。

- **ForwardDataRequest** (*Sender=NodeID(S3), Receiver=NodeID(P), TaskID=fgh, ContentID=UUID, LocationID=UUID, [ContentDescription={Title="Octopussy"},] [Start=00:00:00:00, End=02:05:00:00]*)

【 0 1 5 2 】

デバイス P はデバイス S 3 からのリクエストを受け取り、要求されたコンテンツと共に以下の応答メッセージをデバイス S 3 に送信し、これによってデバイス P からデバイス S 3 へのコンテンツデータの伝送が開始される。

- **ForwardDataResponse** (*Sender=NodeID(P), Receiver=NodeID(S3), TaskID=fgh, ContentID=UUID, LocationID=UUID, [ContentDescription={Title="Octopussy"},] [Start=00:00:00:00, End=02:05:00:00,] BitRate=7 Mbps, Content*)

10

【 0 1 5 3 】

デバイス S 3 は次に S 0 に対しレコーディングプロセスの開始に関する情報を提供し、これによってユーザに対し以下のことを知らせることができる。

- **TransferStatusInformation** (*Sender=NodeID(S3), Receiver=NodeID(S0), TaskID=fgh, MessageMode="starting" [, TransferPurpose="Record", Destination=NodeID(S3), Source=NodeID(P), ContentID=UUID, LocationID=UUID, ContentDescription={Title="Octopussy"}] [, Start=00:00:00:00, End=02:05:00:00] [, BitRate=7 Mbps]*)

20

【 0 1 5 4 】

デバイス S 3 は (ForwardDataRequest メッセージにより開始される) 伝送を制御するので、デバイス S 3 は TransferStatusInformation ("starting") メッセージ (伝送ステータス情報「スタート」) をデバイス S 0 へ送信する。デバイス P がデータ伝送を完了すると、このデバイスは以下の情報メッセージをデバイス S 3 に送信し、したがって全てのデータが伝送されたことが確認される (このメッセージを受け取らなかった場合には、デバイス S 3 はそのことを、例えば強制的にデバイスが切断されるなど何らかの理由で伝送が完了しなかったことの合図として用いることができる) :

30

- **TransferStatusInformation** (*Sender=NodeID(P), Receiver=NodeID(S3), TaskID=fgh, MessageMode="end of data", ContentID=UUID, LocationID=UUID [, ContentDescription={Title="Octopussy"}] [, Start=00:00:00:00, End=02:05:00:00]*)

【 0 1 5 5 】

デバイス S 3 はレコーディングを完了し、レコーディングがうまく完了したことに關する以下の情報メッセージをデバイス S 0 に送信し、これによりそのことをユーザに通知することができる。

40

- **TransferStatusInformation** (*Sender=NodeID(S3), Receiver=NodeID(S0), TaskID=fgh, MessageMode="completed" [, TransferPurpose="Record", Destination=NodeID(S3), Source=NodeID(P), ContentID=UUID, LocationID=UUID, ContentDescription={Title="Octopussy"}] [, Start=00:00:00:00, End=02:05:00:00] [, Duration=02:05 h, BitRate=7 Mbps] [, StorageSpace=6.11 GB]*)

【 0 1 5 6 】

デバイス P およびデバイス S 3 は自身のリソースの割り当て解除を行い、ここでデバイス S 0 はユーザに対して伝送タスクがうまく完了したことに關する通知を行う。

【 0 1 5 7 】

50

これらのステップを図 15 に示されているように、メッセージのシーケンス図によって表すことができる。

【0158】

本発明を有利には自己完結型のネットワークに適用することができ、このネットワークにおいてはネットワークデバイスがある程度の自律性を有する。殊に、ネットワークデバイスはクライアントサーバアーキテクチャにおけるクライアント以上のものであるとみなされる。例えば欧州特許明細書EP1427149号に記載されているように、有利にはネットワークデバイスはクライアントとサーバの役割を同時に果たすことができ、例えばP2Pアーキテクチャにおけるピアとして機能することができる。さらにネットワークを他のネットワークと接続することができるが、このコネクションは明示的に「内部」ネットワークと定義されない限り、一般的には「外部」ネットワークとみなされる。外部コネクションを介して接続されているストレージデバイスは外部デバイスとみなされ、また一般的にはモノリシックブロック(MB)に含まれるものではない。

10

【0159】

ネットワーク内の複数のストレージデバイスを1つのMBとみなし、そこではユーザがそれらのデバイスを区別する必要がない場合は考慮すべき問題である。1つの可能性は全てのネットワークデバイスがMBの部分であるということである。このことは、僅かな記憶容量しか有さない小さいデバイスも考慮されるということの意味する。

【0160】

1つの実施形態においては、自動的に検出されたストレージデバイスがプロポーザルとみなされ、ユーザによって確認されることが必要とされる。

20

【0161】

別の可能性はこれらのネットワークデバイスのみが、少なくとも最小限のリソース、例えば総記憶容量、データ伝送レート、または所定のサービスを提供するMBに含まれることである。そのようなデバイスに対する最小限の要求を局所的に定義することができ、またユーザによって修正することができ、さらにはこの要求はデバイスが接続されると自動的に検出される。

【0162】

例えば新たに接続されたデバイスには自身のレイバビリティを他のデバイスまたは特定のサービスに提供することが要求される。そのデバイスが例えば1Gバイトまたはそれ以上の総記憶容量を有し、そのコネクション帯域幅が他の目的のためには決して割り当てられない場合には、そのデバイスはMBに含まれることになる。そのデバイスが接続されたときに既にデータが格納されている場合には、(保護されていなければ)それらのデータをネットワーク内でMBアプリケーションを介して利用することができる。デバイスレイバビリティが十分でない場合には、このデバイスはMBに関してストレージノードとはみなされないが、例えば入力/出力ノードまたは他の特別な目的のノードとみなすことはできる。しかしながら、明示的にMB内のデバイスを排除するか、MB内に包含させるため、および/または、どのデバイスがMBの一部として考慮されるかを定める規則を修正するために、ユーザは自身のネットワークをコンフィギュレートすることができる。

30

【図面の簡単な説明】

40

【0163】

【図1】アプリケーションが分散ストレージシステムをどのように見るかを示す。

【図2】「要求されたストレージターゲット」モードを示す。

【図3】ファイル分割の原理を示す。

【図4】モノリシックブロックシステムのコアタスクである適切なストレージデバイスの識別(「デバイスマッチング」)を示す。

【図5】無秩序なストック管理に関するモノリシックブロックAPIを示す。

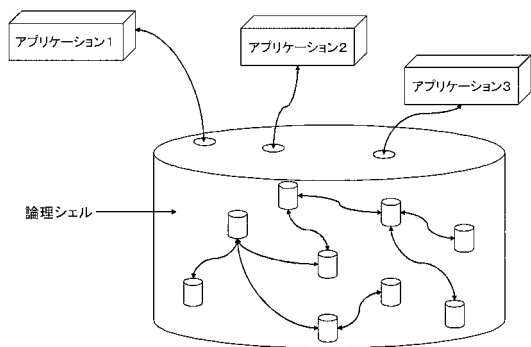
【図6】良好にソートされたストレージに関するモノリシックブロックのセットアップを示す。

【図7】ストレージデバイス内のデータタイプの総「領域」を示す。

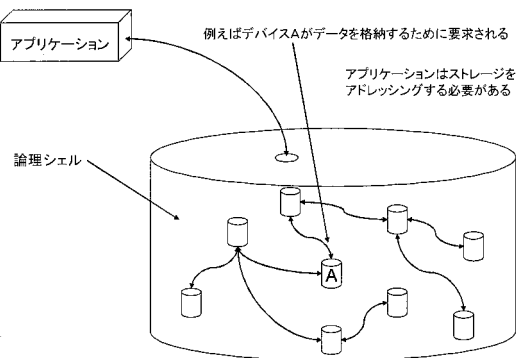
50

- 【図8】モノリシックブロックシステムの概観を示す。
- 【図9】ストレージデバイスの物理的な決定を示す。
- 【図10】ユーザプリファレンスの決定を示す。
- 【図11】物理的なストレージマッチングを示す。
- 【図12】論理的なストレージマッチングを示す。
- 【図13】分散ストレージシナリオにおける例示的なホームネットワークを示す。
- 【図14】容量が制限されているケースでのコンテンツのコピーを示す。
- 【図15】容量が制限されているケースでのコンテンツのコピーに関するメッセージのシーケンス図を示す。

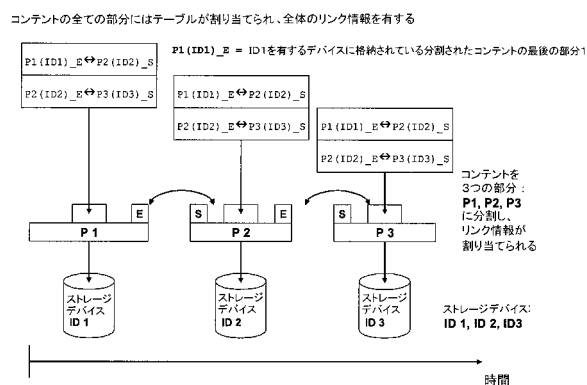
【図1】



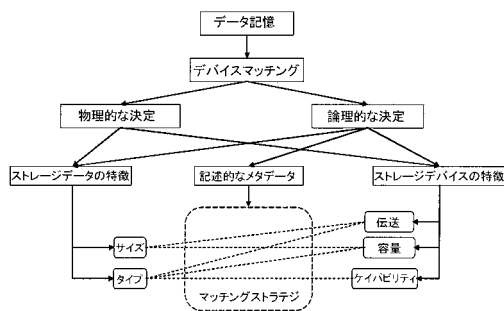
【図2】



【図3】

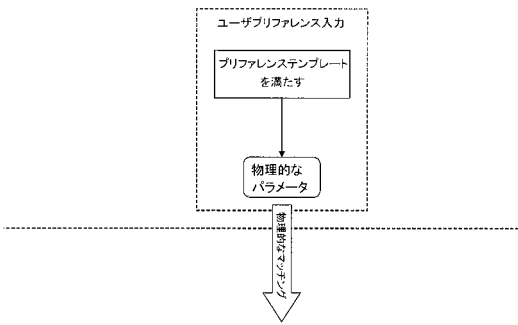


【図4】

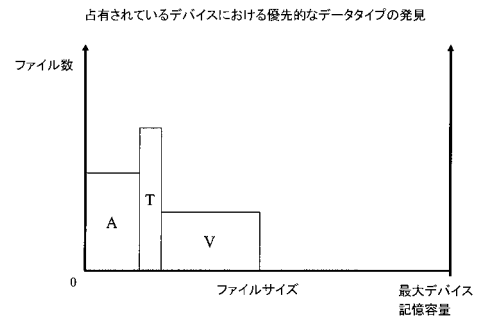


【 図 5 】

無秩序なストック管理に関するモニシックブロックAPI

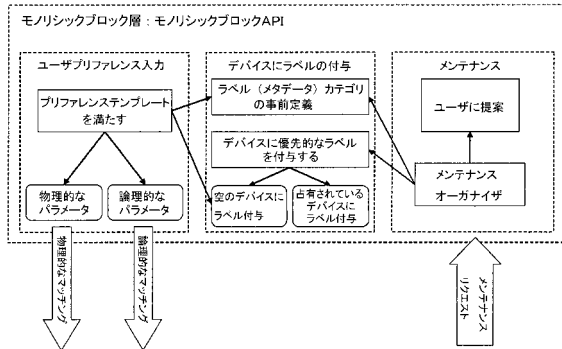


【 図 7 】

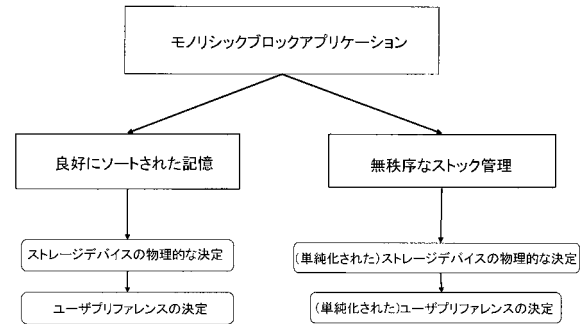


【 図 6 】

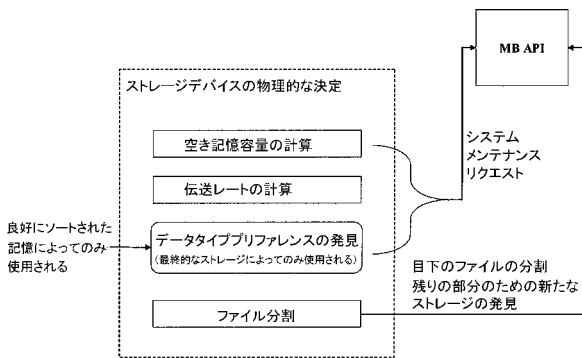
モニシックストレージシステム



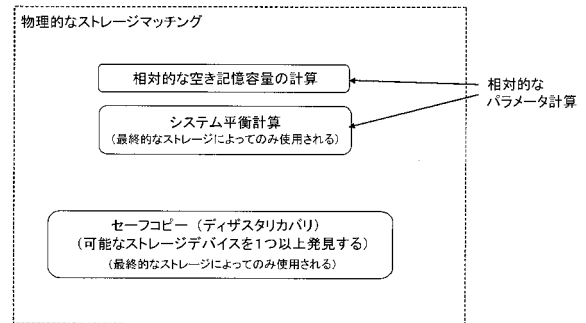
【 図 8 】



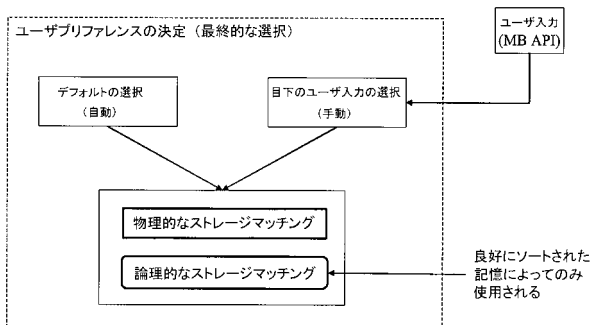
【 図 9 】



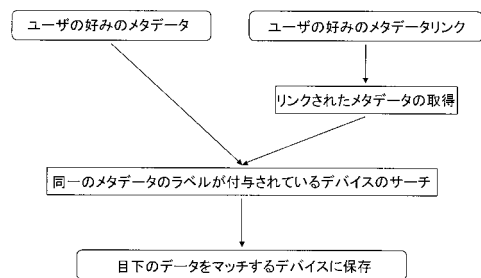
【 図 1 1 】



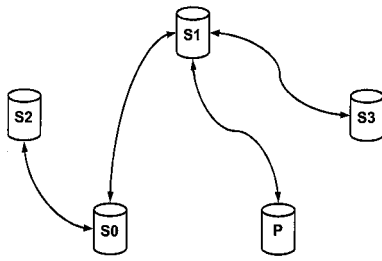
【 図 1 0 】



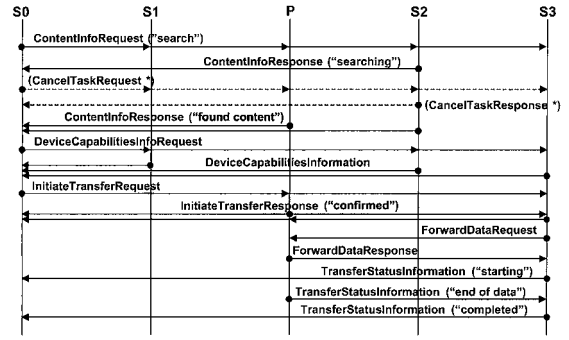
【 図 1 2 】



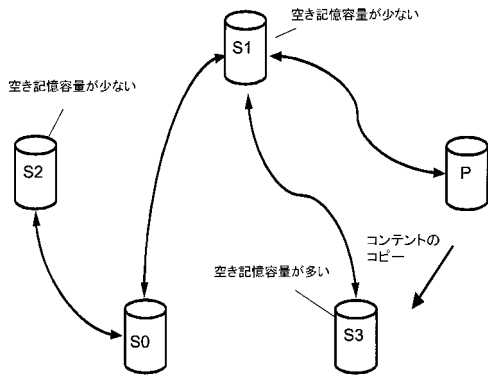
【 図 1 3 】



【 図 1 5 】



【 図 1 4 】



フロントページの続き

(74)代理人 100110593

弁理士 杉本 博司

(74)代理人 100114890

弁理士 アインゼル・フェリックス＝ラインハルト

(72)発明者 フイ リー

ドイツ連邦共和国 ハノーファー シュヴァルツェ ヴォルト 4 0

(72)発明者 マイノルフ プラヴァート

ドイツ連邦共和国 ハノーファー ズーテルシュトラッセ 5 4 ツェー

(72)発明者 ディートマー ヘッパー

ドイツ連邦共和国 ハノーファー ヴェルニゲローダー ヴェーク 4 1

Fターム(参考) 5B014 EB04

5B065 BA01 CA11 CE21 ZA16

5B082 CA11 CA16 FA07 HA08