

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6209861号  
(P6209861)

(45) 発行日 平成29年10月11日 (2017.10.11)

(24) 登録日 平成29年9月22日 (2017.9.22)

(51) Int.Cl.		F I			
<b>G 0 6 F</b>	<b>3/06</b>	<b>(2006.01)</b>	<b>G 0 6 F</b>	<b>3/06</b>	<b>3 0 1 J</b>
<b>G 0 6 F</b>	<b>12/00</b>	<b>(2006.01)</b>	<b>G 0 6 F</b>	<b>12/00</b>	<b>5 0 1 A</b>

請求項の数 10 (全 16 頁)

(21) 出願番号	特願2013-109184 (P2013-109184)	(73) 特許権者	000005223
(22) 出願日	平成25年5月23日 (2013.5.23)		富士通株式会社
(65) 公開番号	特開2013-246827 (P2013-246827A)		神奈川県川崎市中原区上小田中4丁目1番1号
(43) 公開日	平成25年12月9日 (2013.12.9)	(74) 代理人	100107766
審査請求日	平成28年2月26日 (2016.2.26)		弁理士 伊東 忠重
(31) 優先権主張番号	13/482,832	(74) 代理人	100070150
(32) 優先日	平成24年5月29日 (2012.5.29)		弁理士 伊東 忠彦
(33) 優先権主張国	米国 (US)	(74) 代理人	100146776
			弁理士 山口 昭則
		(72) 発明者	スリブラシ・ルディ
			アメリカ合衆国, カリフォルニア州 94085, サニーヴェイル, ウェスト・デュエイン・アヴェニュー 244番

最終頁に続く

(54) 【発明の名称】 記憶メタデータのための拡張可能な方法及びシステム

(57) 【特許請求の範囲】

【請求項 1】

記憶メタデータを表現するために拡張可能言語を用いるシステムであって、前記システムは、

プラットフォーム固有のキー値で符号化された圧縮バイナリ形式の記憶メタデータを格納されたコンピュータ可読記憶媒体と、

処理装置と、

を有し、

前記記憶メタデータは、ハードディスクパーティションテーブル、前記コンピュータ可読記憶媒体に格納されたデータの位置を示すブロックアドレスデータ、及びブート命令コードを有し、

前記処理装置は、

前記ハードディスクパーティションテーブルを、前記ハードディスクパーティションテーブルのフォーマットを識別する第1のキー値ペアに変換し、

前記ブロックアドレスデータを、ブート命令コードが格納されている前記コンピュータ可読記憶媒体の開始アドレスを識別する第2のキー値ペアに変換し、

前記ブート命令コードを、前記ブート命令コードが格納されている前記コンピュータ可読記憶媒体の終了アドレスを識別する第3のキー値ペアに変換し、

前記第1、第2、第3のキー値ペアを含む拡張可能言語フォーマットの記憶メタデータを前記コンピュータ可読記憶媒体に書き込み、

10

20

前記拡張可能言語フォーマットの記憶メタデータを操作し、  
前記拡張可能言語フォーマットの記憶メタデータを転送する、  
よう構成される、

システム。

【請求項 2】

前記ハードディスクパーティションテーブルは、M B R (master boot record) パーティションテーブル、E B R (extended boot record) パーティションテーブル、G U I D (globally unique identifier) パーティションテーブル (G P T)、B S D (Berkeley Software Distribution) パーティションテーブル、又は A P M (Apple Partition Map) パーティションテーブルのうちの少なくとも 1 つを有する、請求項 1 に記載のシステム。

10

【請求項 3】

前記ブロックアドレスデータは、C H S (cylinder-head-sector) ブロックアドレスデータ又は L B A (logical block address) ブロックアドレスデータのうちの少なくとも 1 つを有する、請求項 1 に記載のシステム。

【請求項 4】

前記ブート命令コードは、B P B (BIOS Parameter Block) 又はブートセクタデータのうちの少なくとも 1 つを有する、請求項 1 に記載のシステム。

【請求項 5】

前記拡張可能言語フォーマットは、X M L (eXtensible Markup Language) フォーマット又は J S O N (JavaScript Object Notation) フォーマットを有する、請求項 1 に記載のシステム。

20

【請求項 6】

前記第 2 のキー値ペアは開始オブジェクトを有し、前記第 3 のキー値ペアは終了オブジェクトを有し、前記開始オブジェクト及び前記終了オブジェクトの各々は、前記コンピュータ可読記憶媒体に格納されたデータの位置を示すブロックアドレスデータのフォーマットを識別するフォーマットキー値ペアを含む複数のキー値ペアを有する、請求項 1 に記載のシステム。

【請求項 7】

前記開始オブジェクト及び前記終了オブジェクトの各々は、  
前記開始又は終了アドレスの個々のシリンダを識別するシリンダキー値ペア、  
前記開始又は終了アドレスの個々の先頭を識別する先頭キー値ペア、  
前記開始又は終了アドレスの個々のセクタを識別するセクタキー値ペア、  
を更に有する請求項 6 に記載のシステム。

30

【請求項 8】

記憶メタデータを表現するために拡張可能言語を用いる処理装置の作動方法であって、前記方法は、

記憶メタデータをプラットフォーム固有のキー値で符号化された圧縮バイナリ形式でコンピュータ可読記憶媒体に格納するステップであって、前記記憶メタデータは、ハードディスクパーティションテーブル、前記コンピュータ可読記憶媒体に格納されたデータの位置を示すブロックアドレスデータ、及びブート命令コードを有する、ステップと、

40

前記ハードディスクパーティションテーブルを、前記ハードディスクパーティションテーブルのフォーマットを識別する第 1 のキー値ペアに変換するステップと、

前記ブロックアドレスデータを、ブート命令コードが格納されている前記コンピュータ可読記憶媒体の開始アドレスを識別する第 2 のキー値ペアに変換するステップと、

前記ブート命令コードを、前記ブート命令コードが格納されている前記コンピュータ可読記憶媒体の終了アドレスを識別する第 3 のキー値ペアに変換するステップと、

前記第 1、第 2、第 3 のキー値ペアを含む拡張可能言語フォーマットの記憶メタデータを前記コンピュータ可読記憶媒体に書き込むステップと、

前記拡張可能言語フォーマットの記憶メタデータを操作するステップと、

50

前記拡張可能言語フォーマットの記憶メタデータを転送するステップと、  
を有する方法。

【請求項 9】

コンピューティングシステムに記憶メタデータを表現するために拡張可能言語を用いる動作を実行させるコンピュータ命令を実行するよう構成されるプロセッサであって、前記動作は、

記憶メタデータをプラットフォーム固有のキー値で符号化された圧縮バイナリ形式でコンピュータ可読記憶媒体に格納するステップであって、前記記憶メタデータは、ハードディスクパーティションテーブル、前記コンピュータ可読記憶媒体に格納されたデータの位置を示すブロックアドレスデータ、及びブート命令コードを有する、ステップと、

10

前記ハードディスクパーティションテーブルを、前記ハードディスクパーティションテーブルのフォーマットを識別する第 1 のキー値ペアに変換するステップと、

前記ブロックアドレスデータを、ブート命令コードが格納されている前記コンピュータ可読記憶媒体の開始アドレスを識別する第 2 のキー値ペアに変換するステップと、

前記ブート命令コードを、前記ブート命令コードが格納されている前記コンピュータ可読記憶媒体の終了アドレスを識別する第 3 のキー値ペアに変換するステップと、

前記第 1、第 2、第 3 のキー値ペアを含む拡張可能言語フォーマットの記憶メタデータを前記コンピュータ可読記憶媒体に書き込むステップと、

前記拡張可能言語フォーマットの記憶メタデータを操作するステップと、

前記拡張可能言語フォーマットの記憶メタデータを転送するステップと、

20

を有する、プロセッサ。

【請求項 10】

前記拡張可能言語フォーマットは、XML (eXtensible Markup Language) フォーマット又は JSON (JavaScript Object Notation) フォーマットを有する、請求項 9 に記載のプロセッサ。

【発明の詳細な説明】

【技術分野】

【0001】

本願明細書で議論される実施形態は、データ記憶フォーマットに関する。

30

【背景技術】

【0002】

パーソナルコンピュータは、1980年代に普及し、現在も重要性及び展開の範囲を増し続けている。広範な種類のオペレーティングシステム及び記憶フォーマットが展開され、インターネットを形成するために記憶装置の多様な使用及びハードウェアのネットワーク接続を可能にしている。このようなオペレーティングシステムの例は、Microsoft Windows (登録商標)、Linux (登録商標)、Mac OS/X を含み、数々のあまり知られていない変形も含む。これらのオペレーティングシステムの全ては、2つの共通の特徴を有する。つまり、それらはブート可能でなければならないこと、それらはディスクドライブに格納されたデータにアクセス可能でなければならないことである。残念ながら、分かりにくい相違点の多様すぎる賛辞 (compliment) は、先行する動作をサポートするために必要なメタデータを操作するためにオペレーティングシステム個別の自動ユーティリティが拡散したことから生じている。ブートシステム機能なしでは、オペレーティングシステムは起動できない。ディスクドライブのパーティション及びファイルシステムメタデータの記録機能なしでは、オペレーティングシステムは、自身が格納しているデータへのアクセスを提供できない。これら2つの動作をサポートするために多くのユーティリティが作成されているが、いずれも拡張性をサポートするために設計されていない。拡張性サポートの欠如は、多様なオペレーティングシステムとマシンアーキテクチャとの間の相互接続性を制限する。

40

【0003】

50

限られた相互接続性は、ホームユーザやサーバ技術者に対して、ブート及び初期ファイルシステムの実装の複雑性を著しく増大し且つ信頼性を低下させ、システム管理コストを有意に増大させてしまう。

#### 【 0 0 0 4 】

ある単純な例は、「fdisk」ユーティリティ又はコマンドである。このコマンドは、Windows及びLinuxの両方において同じ名称である。残念ながら、fdiskコマンドの動作は、WindowsとLinuxの間で変わる。例えば、fdiskコマンドの2つのバージョンは、MBR及びEBRフォーマットパーティション情報をハードディスクドライブに読み出し及び書き込むが、それらは、それぞれ異なるコマンドラインオプションと異なる対話式コマンドメニューを用いる。MBRパーティションテーブル内には、論理ブロックアドレス(logical block address : LBA)セクタ番号及びCHS(Cylinder-Head-Sector)アドレスの両方がある。CHSアドレスは、ことなるランタイム環境及びオペレーティングシステムで異なるように解釈される。ブートアップでは、基本入力/出力システム(BIOS)は、通常、CHSアドレスを用いる。最新のオペレーティングシステムは、パーティションテーブル内のCHS値を完全に無視するが、それらがLBAアドレスと一致しないときは分かりにくい診断メッセージを印刷する。Linux sfdiskコマンドのマニュアルページに記載されているように、特定バージョンのDOSの中のシリンダ番号の1つの解釈には相違がある。Linuxには、「fdisk」、「cfdisk」、「sfdisk」及び「gdisk」と称される少なくとも4個の有名なディスクパーティションの変形がある。これら全ては僅かに異なる動作を有し、名称が類似しWindowsの変形と類似するので初心者には区別するのが難しい。今やUnix(登録商標)シェルは、Cygwinのような類似システムを通じてWindows下で利用可能なので、ディスクドライブパーティションを操作するための自動シェルスクリプトの使用に関連する非常に多くの危険と混乱がある。自動パーティションテーブル操作の頻繁な出力は、共有ドライブ上の1又は複数のオペレーティングシステムを失うことである。

#### 【 0 0 0 5 】

年を追って、オペレーティングシステムの記憶装置の設置面積よりも速く、ハードドライブの容量が増大しコストが低減したので、アドバンスドユーザは彼らのコンピュータに1より多いオペレーティングシステムをインストールすることが一般的である。その結果、インストール又は復旧中に、ユーザがどのシステムが望ましいかを選択できるよう、ディスクドライブパーティション及びブートシーケンスに変更が必要であり得る。Intel Virtualization Technology (VT) CPU機能のような機能を通じてCPUハードウェアでサポートされるエミュレーションシステムの普及は、ブートブロック及びパーティションテーブルの変更の実行に伴う複雑性及び危険を増すだけである。オペレーティングシステムは、互いを承知し、これらのシステムの重要な領域を操作するときに協力すべきであるが、現在の多くの異なるコマンド、フォーマット及びバイナリデータの解釈は、このような相互動作を妨げてしまう。

#### 【 発明の概要 】

#### 【 発明が解決しようとする課題 】

#### 【 0 0 0 6 】

本願明細書で請求される主題は、上述のような欠点を解決する実施形態や上述のような環境でのみ機能する実施形態に限定されない。むしろ、この背景技術は、単に、本願明細書に記載される複数の実施形態が実施される技術分野の一例を説明するために提供される。

#### 【 課題を解決するための手段 】

#### 【 0 0 0 7 】

一実施形態の一態様によると、記憶メタデータを表現するために拡張可能言語を用いるシステムは、コンピュータ可読記憶媒体及び処理装置を有する。コンピュータ可読記憶媒体は、記憶メタデータを格納しても良い。処理装置は、記憶メタデータをコンピュータ可読記憶媒体に拡張可能言語フォーマットで書き込むよう構成されても良い。処理装置は、拡張可能言語フォーマットの記憶メタデータを操作するよう構成されても良い。処理装置

は、拡張可能言語フォーマットの記憶メタデータを転送するよう構成されても良い。

【0008】

実施形態の目的及び利点が理解され、少なくとも特に特許請求の範囲で指摘された要素、特徴及び組合せを用いて達成されるだろう。

【0009】

上述の全体的説明及び以下の詳細な説明の両方は、例示及び説明のためであり、本発明の範囲を限定しないことが理解される。

【図面の簡単な説明】

【0010】

例示的な実施形態は、添付の図面を用いて、更なる特異性及び詳細事項と共に記載され説明される。

【図1】記憶メタデータを表すために拡張可能言語を用いるよう構成された例示的なコンピューティングシステムを示すブロック図である。

【図2】図1の記憶メタデータの例示的な実施形態を示す。

【図3】圧縮バイナリフォーマットの記憶メタデータの例示的な実施形態を示す。

【図4】拡張可能言語フォーマットの図3の記憶メタデータの例示的な実施形態を示す。

【図5A】記憶メタデータを表現する拡張可能言語を用いる例示的な方法のフローチャートである。

【図5B】記憶メタデータを表現する拡張可能言語を用いる別の例示的な方法のフローチャートである。

【発明を実施するための形態】

【0011】

本発明の実施形態を、添付の図面を参照して以下に説明する。

【0012】

図1は、本開示による、記憶メタデータを表すために拡張可能言語を用いるよう構成された例示的なコンピューティングシステム100を示すブロック図である。記憶メタデータは、例えばシステムクリティカルメタデータであっても良い。非常に基本的な構成102では、コンピューティングシステム100は、1又は複数のプロセッサ104及びシステムメモリ106を有しても良い。メモリバス108は、プロセッサ104とシステムメモリ106との間の通信のために用いられても良い。

【0013】

所望の構成に依存して、プロセッサ104は、マイクロプロセッサ(μP)、マイクロコントローラ(μC)、デジタル信号プロセッサ(DSP)又はそれらの任意の組合せを含むがこれらに限定されない任意の種類であっても良い。プロセッサ104は、レベル1キャッシュ110及びレベル2キャッシュ112のような1又は複数のキャッシュ、プロセッサコア(又は複数のコア)114、及びレジスタ116を有しても良い。例示的なプロセッサコア114は、算術論理演算ユニット(ALU)、浮動小数点演算ユニット(FPU)、デジタル信号プロセッサコア(DSPコア)又はこれらの任意の組合せを有しても良い。例示的なメモリ制御部118は、プロセッサ104と共に用いられても良い。または、幾つかの実施形態では、メモリ制御部118は、プロセッサ104の内部の一部であっても良い。プロセッサ104は、例えばシステムメモリ106にロードされたコンピュータ命令又はコードを実行することにより、及び/又はシステムメモリ106を使用しないでコンピュータ命令又はコードを1行ずつ実行することにより、本願明細書に記載された1又は複数の動作を実行するよう構成されても良い。

【0014】

所望の構成に依存して、システムメモリ106は、(RAM(Random Access Memory)のような)揮発性メモリ、(ROM(Read Only Memory)、フラッシュメモリ等のような)不揮発性メモリ又はそれらの任意の組合せを含むがこれらに限定されない任意の種類であっても良い。システムメモリ106は、オペレーティングシステム120、1又は複数のアプリケーション122、プログラムデータ124を有しても良い。アプリケーシ

10

20

30

40

50

ョン１２２は、図５Ａ－５Ｂに関して記載される１又は複数の機能を含む、本願明細書に記載される機能を実行するよう構成される拡張可能アルゴリズム１２６を有しても良い。例えば、アプリケーション１２２は、プロセッサ１０４により実行され、コンピューティングシステム１００に本願明細書に記載される機能を実行させても良い。プログラムデータ１２４は、例えばプロセッサ１０４により本願明細書に記載される拡張可能言語フォーマットで書かれ、操作され及び／又は転送され得る記憶メタデータ１２８を有しても良い。幾つかの実施形態では、アプリケーション１２２は、オペレーティングシステム１２０上でプログラムデータ１２４により動作するよう構成されても良い。したがって、記憶メタデータ１２８は、拡張可能言語フォーマットを用いて表現されても良く、及び／又は拡張可能言語フォーマットで書かれ、操作され及び／又は転送されても良い。記憶メタデータ１２８のための拡張可能言語フォーマットの使用を更に詳細に説明する前に、幾つかの実施形態による図１のコンピューティングシステム１００の種々の任意のコンポーネント、特徴及び／又は機能が記載される。

10

#### 【００１５】

コンピューティングシステム１００は、基本構成１０２と他の装置及びインタフェースとの間の通信を実現するために、追加の特徴又は機能、及び追加インタフェースを有しても良い。例えば、バス／インタフェース制御部１３０は、記憶装置インタフェースバス１３４を介して基本構成１０２と１又は複数のデータ記憶装置１３２との間の通信を実現するために用いられても良い。データ記憶装置１３２は、取り外し可能記憶装置１３６、非取り外し可能記憶装置１３８又はそれらの組合せであっても良い。取り外し可能記憶装置及び非取り外し可能記憶装置の例は、少数を列挙すると、フレキシブルディスクドライブ及びハードディスクドライブ（ＨＤＤ）のような磁気ディスク装置、ＣＤ（compact disk）又はＤＶＤ（digital versatiledisk）のような光ディスクドライブ、ＳＳＤ（solid state drive）及びテープドライブを含む。例示的なコンピュータ記憶媒体は、コンピュータ可読命令、データ構造、プログラムモジュール又は他のデータのような情報を記憶する方法又は技術で実装される揮発性及び不揮発性、取り外し可能及び非取り外し可能媒体を含み得る。

20

#### 【００１６】

幾つかの実施形態では、記憶メタデータは、以下に更に詳細に記載するように、圧縮バイナリフォーマットで及び／又は拡張可能言語フォーマットで１又は複数のデータ記憶装置１３２に書き込まれ、及び／又はそれから転送されても良い。まとめると、例えば、コンピューティングシステム１００は、データ記憶装置１３２のうちの１つに圧縮バイナリフォーマットで格納され、システムメモリ１０６及び／又はプロセッサ１０４へ転送されて拡張可能言語フォーマットに変換され、次に拡張可能言語フォーマットでデータ記憶装置１３２のうちの１つに書き込まれる記憶メタデータを有しても良い。

30

#### 【００１７】

システムメモリ１０６、取り外し可能記憶装置１３６及び非取り外し可能記憶装置１３８は、コンピュータ記憶媒体の例である。コンピュータ記憶媒体は、ＲＡＭ、ＲＯＭ、ＥＥＰＲＯＭ、フラッシュメモリ又は他のメモリ技術、ＣＤ－ＲＯＭ、ＤＶＤ（digital versatile disk）又は他の光記憶装置、磁気カセット、磁気テープ、磁気ディスク記憶装置又は他の磁気記憶装置、又は所望の情報を格納するために用いることができコンピューティングシステム１００によりアクセスできる任意の他の媒体を含むがこれらに限定されない。任意のこのようなコンピュータ記憶媒体は、コンピューティングシステムの一部であっても良い。

40

#### 【００１８】

コンピューティングシステム１００は、バス／インタフェース制御部１３０を介して種々のインタフェース装置（例えば、出力装置１４２、周辺機器インタフェース１４４、及び／又は通信装置１４６）から基本構成１０２への通信を実現するインタフェースバス１４０を有しても良い。例示的な出力装置１４２は、グラフィック処理ユニット１４８及びオーディオ処理ユニット１５０を有する。これらは、ディスプレイ又はスピーカのような

50

種々の外部装置と1又は複数のA/Vポート152を介して通信するよう構成されても良い。例示的な周辺機器インタフェース144は、シリアルインタフェース制御部154又はパラレルインタフェース制御部156を有する。これらは、入力装置(例えば、キーボード、マウス、ペン、音声入力装置、タッチ入力装置等)又は他の周辺機器(例えば、プリンタ、スキャナ等)のような外部装置と通信するよう構成されても良い。例示的な通信装置146は、ネットワーク制御部160を有する。ネットワーク制御部160は、1又は複数の通信ポート164を介したネットワーク通信リンクを介して1又は複数の他のコンピューティングシステム162との通信を実現するよう構成されても良い。

#### 【0019】

ネットワーク通信リンクは、通信媒体の一例であっても良い。通信媒体は、通常、コンピュータ可読命令、データ構造、プログラムモジュール又は搬送波又は他のトランスポート機構のような変調されたデータ信号内の他のデータにより実現されても良く、任意の情報配信媒体を有しても良い。「変調されたデータ信号」は、1又は複数の特性セットを有する信号であるか、又は信号内の情報をエンコードするために変更されても良い。例として限定ではなく、通信媒体は、有線ネットワーク又は直接有線接続のような有線媒体、並びに音響、無線周波数(RF)、マイクロ波、赤外線(IR)のような無線媒体及び他の無線媒体を含み得る。本願明細書で用いられる用語コンピュータ可読媒体は、記憶媒体及び通信媒体の両方を含み得る。

#### 【0020】

コンピューティングシステム100は、携帯電話機、PDA(personal data assistant)、個人向けメディアプレーヤ装置、無線ウェブ閲覧装置、個人向けヘッドセット装置、特定用途装置、又は上述の機能のうちの任意のものを有するハイブリッド装置のような小型ポータブル(又はモバイル)電子機器の一部として実装されても良い。コンピューティングシステム100は、ラップトップコンピュータ及び非ラップトップコンピュータ構成を有するパーソナルコンピュータとして実装されても良い。

#### 【0021】

図2は、図1の記憶メタデータ128の例示的な実施形態を示す。上述のように、記憶メタデータ128は、取り外し可能記憶装置136及び/又は非取り外し可能記憶装置138内にあるような1又は複数のデータ記憶装置132に格納されても良い。代替で又は追加で、記憶メタデータ128は、システムメモリ106内にロードされても良い。さらに、記憶メタデータ128は、圧縮バイナリフォーマットで表現されても良い。この例は、図3に関して詳細に記載される。また、記憶メタデータ128は、拡張可能言語フォーマットで表現されても良い。この例は、図4に関して詳細に記載される。あるいは、記憶メタデータ128は、圧縮バイナリフォーマット及び拡張可能言語フォーマットの両方のような複数のフォーマットで表現されても良い。

#### 【0022】

図2に図示した実施形態では、記憶メタデータ128は、1又は複数のハードディスクパーティションテーブル202、1又は複数のブロックアドレスデータ204、ブート命令コード206を有する。他の実施形態では、記憶メタデータ128は、ハードディスクパーティションテーブル202、ブロックアドレスデータ204、及び/又はブート命令コード206のうちの一部を有するが全部を有しなくても良く、及び/又は記憶メタデータ128は、図2に示したものと異なる幾つかのコンポーネントを有しても良い。

#### 【0023】

ハードディスクパーティションテーブル202は、図1の1又は複数のデータ記憶装置132に含まれるようなディスクドライブ又は他の記憶装置のパーティションを記述するデータを有しても良い。ハードディスクパーティションテーブル202は、MBR(master boot record)パーティションテーブル、EBR(extended boot record)パーティションテーブル、GUID(globally unique identifier)パーティションテーブル(GPT)、BSD(Berkeley Software Distribution)ディスクラベル、APM(Apple Partition Map)等、又はそれらの任意の組合せを有し得るが、これらに限定され

10

20

30

40

50

ない。

【 0 0 2 4 】

ブロックアドレスデータ 2 0 4 は、図 1 の 1 又は複数のデータ記憶装置 1 3 2 に含まれるようなディスクドライブ又は他の記憶装置に格納されるデータの位置及びブロックを記述するよう構成されたデータを有しても良い。ブロックアドレススキームは、C H S ( cylinder-head-sector ) ブロックアドレスデータ、L B A ( logical block address ) ブロックアドレスデータ等又はこれらの任意の組合せを有し得るが、これらに限定されない。

【 0 0 2 5 】

ブート命令コード 2 0 6 は、図 1 のコンピューティングシステム 1 0 0 のようなコンピューティングシステムのブートを制御するよう構成されるデータを有しても良い。ブート命令コードは、B I O S ( Basic Input/Output System ) パラメータブロック ( B P B ) 等又はこれらの任意の組合せを有し得るが、これらに限定されない。

【 0 0 2 6 】

図 3 は、圧縮バイナリフォーマット 3 0 0 ( 以下では「記憶メタデータ 3 0 0」と表す ) の記憶メタデータの例示的な実施形態を示す。記憶メタデータ 3 0 0 のような圧縮バイナリフォーマットの記憶メタデータは、従来のコンピューティングシステム及び / 又は他のコンピューティングシステムに含まれても良い。以下に更に詳述するように、記憶メタデータを圧縮バイナリフォーマットで表現することから生じる問題を実質的に除去するために、記憶メタデータ 3 0 0 は拡張可能言語フォーマットに変換され表現されても良い。

【 0 0 2 7 】

圧縮バイナリフォーマットの記憶メタデータ 3 0 0 のような記憶メタデータを提供することは、誤りを生じ、及び / 又はデータ損失をもたらし得る。これらの一例は、以下の図 3 の議論で説明される。圧縮バイナリフォーマットは、概して、記憶メタデータに可能な限り少ないデータを含むことにより、記憶メタデータの記憶装置の専有面積を最小化する。オペレーティングシステム、バックアップシステム、再パーティショニングユーティリティ又は他のプログラム若しくはシステムが記憶メタデータのフォーマット、位置等を理解する限り、記憶メタデータの圧縮特性は問題がない。しかしながら、記憶メタデータを様々に理解し解釈する多様なオペレーティングシステム、バックアップシステム、再パーティショニングユーティリティ又は他のプログラム若しくはシステムが同じコンピューティングシステムで動作するとき、問題が生じ得る。

【 0 0 2 8 】

例えば、図 3 の記憶メタデータ 3 0 0 を検討する。幾つかのオペレーティングシステム、バックアップシステム、再パーティショニングユーティリティ又は他のプログラム若しくはシステムは、記憶メタデータ 3 0 0 を M B R ハードディスクパーティションテーブル及びブート命令コードが格納され得るコンピュータ可読記憶媒体内の C H S 型の開始アドレス 3 0 2 及び C H S 型の終了アドレス 3 0 4 を含む C H S ブロックアドレスデータとして解釈し得る。代替として又は追加で、他のオペレーティングシステム、バックアップシステム、再パーティショニングユーティリティ又は他のプログラム若しくはシステムは、記憶メタデータ 3 0 0 を G P T ハードディスクパーティションテーブル及びブート命令コードが格納され得るコンピュータ可読記憶媒体内の L B A 型の開始アドレス 3 0 6 及び L B A 型の終了アドレス 3 0 8 を含む L B A ブロックアドレスデータとして解釈し得る。更に他のオペレーティングシステム、バックアップシステム、再パーティショニングユーティリティ又は他のプログラム若しくはシステムは、記憶メタデータ 3 0 0 自体を認識もしないかも知れない。記憶メタデータ 3 0 0 が誤って解釈されるか又は認識されないとき、誤ったデータが誤って解釈された記憶アドレスからロードされ、記憶メタデータ 3 0 0 は上書きされ及び / 又は他の問題が生じ得る。類似した問題は、記憶メタデータが独自又は非拡張可能フォーマットで表現されるとき、生じ得る。したがって、記憶メタデータを圧縮バイナリフォーマットで又は独自若しくは非拡張可能フォーマットで表現することは、記憶メタデータが理解されない及び / 又はそれ自体認識されない程度の誤り及び / 又はデ



ータ損失を生じ得る。

【 0 0 2 9 】

しかしながら、本願明細書に記載する幾つかの実施形態によると、拡張可能言語フォーマットが、記憶メタデータを表現するために用いられ、特に異種環境内のオペレーティングシステムの継続的成長を可能にする安全且つプログラミング的に都合の良い方法で、根本的な圧縮バイナリフォーマットの解釈における不一致を解決し得る。代替又は追加で、幾つかの実施形態は、圧縮バイナリフォーマットで格納される等価な記憶メタデータよりも僅かに大きな記憶装置占有面積を代償にして、記憶メタデータを明確に識別し、記憶メタデータの意味を明確化し得る。

【 0 0 3 0 】

拡張可能言語フォーマットは、XML (eXtensible Markup Language) フォーマット、JSON (JavaScript(登録商標) Object Notation) フォーマット等又はそれらの組合せを含み得るがこれらに限定されない。代替又は追加で、拡張可能言語フォーマットは、キー値ペアを有する記憶メタデータを表現し得る。キー値ペアの各キーは、記憶メタデータの所与のパラメータ、属性又は態様を識別するストリングを有しても良く、対応する値は、識別されるパラメータ、属性又は態様の値を有しても良い。上述及び他の実施形態では、拡張可能言語フォーマットは、人間に理解可能であっても良い。

【 0 0 3 1 】

図4は、拡張言語フォーマット400(以下では「記憶メタデータ400」と表す)の図3の記憶メタデータ300の例示的な実施形態を示す。図4の例では、記憶メタデータ400は、1又は複数のキー値ペアを含むJSONフォーマットで表現されても良い。例えば、第1、第2及び第3のキー値ペア402、404、406が図4に示されるが、他の実施形態では、3個のキー値ペア402、404、406より多数又は少数が含まれても良い。第1、第2及び第3のキー値ペア402、404、406の各々は、それぞれ「フォーマット」、「開始」、「終了」を含むストリングを有するキーを含む。

【 0 0 3 2 】

第1のキー値ペア402は、コンピュータ可読記憶媒体のハードディスクパーティションテーブルの「フォーマット」を識別し得る。図示した実施形態では、フォーマットは、第1のキー値ペア402の値により「MBR」フォーマットとして識別できる。

【 0 0 3 3 】

第2のキー値ペア404は、コンピュータ可読記憶媒体の「開始」アドレスを識別し得る。また、第3のキー値ペア406は、ブート命令コードが格納されているコンピュータ可読記憶媒体の「終了」アドレスを識別し得る。図示した実施形態では、第2のキー値ペア内の値は開始オブジェクト408を有し、第3のキー値ペア内の値は終了オブジェクト410を有し得る。

【 0 0 3 4 】

開始オブジェクト408は、図4の例では、第2のキー値ペア404により特定のブロックアドレスフォーマット(本例では「CHS」)を有するとして識別される開始アドレスの「フォーマット」を識別するフォーマットキー値ペア、及び特定の開始アドレスを識別する1又は複数の他のキー値ペア有しても良い。CHSブロックアドレスフォーマットの例では、例えば、開始オブジェクト408は、開始アドレスの「シリンダ」を特定のシリンダ(本例では「03」)として識別するシリンダキー値ペア、開始アドレスの「先頭」を特定の先頭(本例では「57」)として識別する先頭キー値ペア、及び開始アドレスを特定のセクタ(本例では「22」)として識別するセクタキー値ペアを更に有しても良い。

【 0 0 3 5 】

同様に、終了オブジェクト410は、図4の例では、第3のキー値ペア406により特定のブロックアドレスフォーマット(本例では「CHS」)を有するとして識別される終了アドレスの「フォーマット」を識別するフォーマットキー値ペア、及び特定の終了アドレスを識別する1又は複数の他のキー値ペア有しても良い。CHSブロックアドレスフォ

10

20

30

40

50

ーマットの例では、例えば、終了オブジェクト 4 1 0 は、終了アドレスの「シリンダ」を特定のシリンダ（本例では「0 5」）として識別するシリンダキー値ペア、終了アドレスの「先頭」を特定の先頭（本例では「1 0」）として識別する先頭キー値ペア、及び終了アドレスを特定のセクタ（本例では「0 0」）として識別するセクタキー値ペアを更に有しても良い。

#### 【0 0 3 6】

図 5 A は、記憶メタデータを表現する拡張言語を用いる例示的な方法 5 0 0 のフローチャートである。方法 5 0 0 及び / 又はその変形は、本願明細書に記載されたコンピューティングシステム 1 0 0 のようなコンピューティングシステムにより全体又は一部が実施されても良い。代替又は追加で、方法 5 0 0 及び / 又はその変形は、図 1 のプロセッサ 1 0 4 のようなプロセッサ又は他の処理装置により全体又は一部が実施されても良い。別個のブロックとして示したが、所望の実装に依存して、種々のブロックは、更なるブロックに分割され、少ないブロックに結合され、又は除去されても良い。

10

#### 【0 0 3 7】

方法 5 0 0 は、ブロック 5 0 2 で開始する。ブロック 5 0 2 で、記憶メタデータは、拡張可能言語フォーマットで書かれても良い。例えば、記憶メタデータは、図 1 のシステムメモリ 1 0 6、取り外し可能記憶装置 1 3 6 又は非取り外し可能記憶装置 1 3 8 のようなコンピュータ可読記憶媒体に拡張可能言語フォーマットで書かれ、

ブロック 5 0 4 で、記憶メタデータは、拡張可能言語フォーマットで操作されても良い。幾つかの実施形態では、拡張可能言語フォーマットの記憶メタデータを操作することは、拡張可能言語フォーマットで表現されたハードディスクパーティションテーブルを再パーティショニングすることを含んでも良い。記憶メタデータは拡張可能言語フォーマットで表現されるので、記憶メタデータが圧縮バイナリフォーマットであった場合に生じる可能性のある、異なるシステム等による記憶メタデータの解釈に関する不一致は、最初から解決されるか又は部分的に回避され得る。

20

#### 【0 0 3 8】

ブロック 5 0 6 で、拡張可能言語フォーマットの記憶メタデータは、転送されても良い。幾つかの実施形態では、拡張可能言語フォーマットの記憶メタデータを転送することは、拡張可能言語フォーマットの記憶メタデータをサーバからクライアントへネットワークを介して転送することを含んでも良い。記憶メタデータを拡張可能言語フォーマットで転送することにより、1 より多いプログラム又はシステムが記憶メタデータの同じピースを操作するために用いられるとき、異なるシステム間の互換性は改善でき及び / 又はソフトウェアバグが低減又は回避できる。

30

#### 【0 0 3 9】

当業者は、この処理及び本願明細書に開始した他の処理及び方法において、その処理及び方法で実行される機能が異なる順序で実施されても良いことを理解するだろう。さらに、概略のステップ及び動作は、単に例として提供され、幾つかのステップ及び動作は、開示の実施形態の本質から逸脱することなく、任意であり、より少ないステップ及び動作に組み合わせられ、又は追加ステップ及び動作に拡張されても良い。

#### 【0 0 4 0】

例えば、方法 5 0 0 は、記憶メタデータをコンピュータ可読記憶媒体に拡張可能言語フォーマットで書き込む前に、記憶メタデータを圧縮バイナリフォーマットから拡張可能言語フォーマットに変換するステップを更に有しても良い。上述及び他の実施形態では、圧縮バイナリフォーマットの記憶メタデータは、予めコンピュータ可読記憶媒体に格納されても良く、将来の使用で記憶メタデータの解釈に関する不一致を回避するために、例えば図 1 の拡張可能アルゴリズム 1 2 6 により拡張可能言語フォーマットに変換されても良い。

40

#### 【0 0 4 1】

代替又は追加で、拡張可能言語フォーマットの記憶メタデータをコンピュータ可読記憶媒体に書き込む前に、記憶メタデータは、最初に、プラットフォーム固有のキー値符号化

50

フォーマットで又は独自フォーマットでコンピュータ可読記憶媒体に格納されても良い。上述及び他の実施形態では、方法 500 は、拡張可能言語フォーマットの記憶メタデータをコンピュータ可読記憶媒体に書き込む前に、プラットフォーム固有のキー値符号化フォーマット（又は独自若しくは他のフォーマット）で格納された記憶メタデータを拡張可能言語フォーマットに変換するステップを更に有しても良い。

【0042】

図 5 B は、記憶メタデータを表現するために拡張可能言語を用いる別の例示的な方法 510 のフローチャートである。方法 510 及び / 又はその変形は、本願明細書に記載されたコンピューティングシステム 100 のようなコンピューティングシステムにより全体又は一部が実施されても良い。代替又は追加で、方法 510 及び / 又はその変形は、図 1 のプロセッサ 104 のようなプロセッサ又は他の処理装置により全体又は一部が実施されても良い。別個のブロックとして示したが、所望の実装に依存して、種々のブロックは、異なるブロックに分割され、少ないブロックに結合され、又は除去されても良い。

【0043】

方法 510 は、ブロック 512 で開始し得る。ブロック 512 で、コンピューティングシステムが設けられ、該コンピューティングシステムは記憶メタデータを有する。

【0044】

ブロック 514 で、記憶メタデータは、拡張可能言語フォーマットで表現されても良い。記憶メタデータを拡張可能言語フォーマットで表現するステップは、記憶メタデータの複数のコンポーネントの各々を表現するためにキー値ペア、又はより詳細にはキー及び値を用いるステップを有しても良い。

【0045】

ブロック 516 で、拡張可能言語フォーマットの記憶メタデータは、コンピューティングシステムに格納されても良い。

【0046】

図 5 B には図示しないが、方法 510 は、拡張可能言語フォーマットを用いて記憶メタデータを表現する前に、記憶メタデータを圧縮バイナリフォーマットから拡張可能言語フォーマットに変換するステップを更に有しても良い。

【0047】

幾つかの実施形態では、図 5 B の方法 510 は、拡張可能言語フォーマットで格納された記憶メタデータを読み出す拡張可能言語フォーマットを理解する再パーティショニングユーティリティを有しても良い。次に、再パーティショニングユーティリティは、コンピュータ可読記憶媒体に格納されたデータを失うことなく及び記憶メタデータに上書きすることなく、コンピューティングシステムのコンピュータ可読記憶媒体を再パーティショニングしても良い。

【0048】

幾つかの実施形態では、コンピューティングシステムは、コンピュータ可読記憶媒体を有しても良い。また、記憶メタデータは、コンピュータ可読記憶媒体内に格納されたデータの位置を示す第 1 及び第 2 のアドレスデータを有しても良い。第 1 のブロックアドレスデータは、第 2 のブロックアドレスデータを用い且つ拡張マークアップ言語を理解しないコンピューティングシステムのオペレーティングシステムに格納され及び / 又はインスタンスを作成することにより、危険に晒され得る。上述及び他の実施形態では、図 5 B の方法 510 は、第 1 のブロックアドレスデータを用いるコンピューティングシステムの第 1 のオペレーティングシステムを格納し及び / 又はインスタンスを生成するステップを有しても良い。第 1 のオペレーティングシステムは、拡張可能言語フォーマットを理解するよう構成されても良い。第 2 のオペレーティングシステムもコンピューティングシステムに格納され及び / 又はインスタンスを生成されても良い。第 2 のオペレーティングシステムは、拡張マークアップ言語フォーマットを理解するよう構成されても良く、コンピューティングシステムに第 2 のオペレーティングシステムのインスタンスを生成することにより、第 1 のブロックアドレスデータが危険に晒されないようにしても良い。

## 【 0 0 4 9 】

したがって、本願明細書に記載した幾つかの実施形態は、JSONフォーマット又は他の拡張可能言語フォーマットの記憶メタデータを読み出し及び書き込むディスクパーティショニング及びブートブロック操作プログラムを有しても良い。上述及び他の実施形態は、記憶メタデータが圧縮バイナリフォーマット、プラットフォーム固有キー値ペアフォーマット、及び/又は独自フォーマットで読み出され及び書き込まれるシステムと比べて、複数の言語、オペレーティングシステム及びランタイム環境に渡り大きな互換性を提供できる。

## 【 0 0 5 0 】

幾つかの実施形態では、記憶メタデータを拡張可能言語フォーマットで表現することは、記憶メタデータ又は他のデータを危険に晒すことなく、記憶メタデータに関して異種の及び時には矛盾する前提を有する複数のオペレーティングシステムを同じコンピューティングシステムで動作可能にし得る。代替又は追加で、記憶メタデータに関して異種の及び場合によっては矛盾する前提を有するエミュレータ、仮想機械、再パーティショニングユーティリティ又は他のプログラムは、記憶メタデータを拡張可能言語フォーマットで表現することにより、記憶メタデータ又は他のデータを危険に晒すことなく、同じコンピューティングシステムで動作され得る。

## 【 0 0 5 1 】

上述及び他の実施形態は、記憶メタデータが圧縮バイナリフォーマットで表現されるシステムと比べて、ブートメニュー調整、オペレーティングシステムのインストール、データ記憶装置の追加、データ統合及び/又は再パーティショニングに含まれる危険を除去又は実質的に低減できる。代替又は追加で、記憶メタデータを圧縮バイナリフォーマットではなく拡張可能言語フォーマットで表現することにより、データセンタ及び/又は他のシステムは、高度なパーティションテーブルフォーマットを用い、異なるオペレーティングシステムに渡って分散する可能性のある大規模且つ増大するディスクドライブアレイを、データ損失又はダウンタイムの少ない危険で管理できる。クラウドサービスは、種々のオペレーティングシステムを用い、圧縮バイナリフォーマットとは対照的に拡張可能言語フォーマットを用いることにより、ハードウェアアップグレードの危険及びダウンタイムを低減し得る。

## 【 0 0 5 2 】

前述のように、幾つかの実施形態で、拡張可能言語フォーマットの記憶メタデータは、人間に理解可能であっても良い。記憶メタデータを人間に理解可能に作成することにより、記憶メタデータは、比較的ユーザフレンドリであり、ユーザに理解し易くなり、異なるオペレーティングシステム及びハードウェアに渡り安全の一貫性の道を開き得る。

## 【 0 0 5 3 】

代替又は追加で、拡張可能言語フォーマットを理解するソフトウェアエージェントは、推測を行うことなくより多くを理解できるので、拡張可能言語フォーマットの記憶メタデータに対して、圧縮バイナリフォーマットの記憶メタデータに対するより多くを自動的に行うことができても良い。

## 【 0 0 5 4 】

本願明細書に記載した実施形態は、以下に更に詳細に議論するように、種々のコンピュータハードウェア又はソフトウェアモジュールを備えた特定用途又は汎用コンピュータの使用を含み得る。

## 【 0 0 5 5 】

本願明細書に記載した実施形態は、コンピュータにより実行可能な命令又はデータ構造を伝える又は格納しているコンピュータ可読媒体を用いて実施され得る。このようなコンピュータ可読媒体は、汎用又は特定目的コンピュータによりアクセスできる利用可能な媒体であり得る。例として且つ限定ではなく、このようなコンピュータ可読媒体は、RAM、ROM、EEPROM、CD-ROM又は他の光ディスク記憶装置、磁気ディスク記憶装置又は他の磁気記憶装置を含むコンピュータ可読記憶媒体、又はコンピュータにより実

10

20

30

40

50

行可能な命令若しくはデータ構造の形式で所望のプログラムコード手段を伝える若しくは格納するために用いられ汎用若しくは特定目的コンピュータによりアクセス可能な他の媒体を有し得る。上述の組合せも、コンピュータ可読媒体の範囲に包含され得る。

#### 【 0 0 5 6 】

コンピュータにより実行可能な命令は、例えば、汎用コンピュータ、特定目的コンピュータ又は特定目的処理装置に特定の機能又は機能グループを実行させる命令及びデータを有する。本発明の主題は構造的特徴及び／又は方法論的動作に特有の言葉で記載されたが、本発明の主題は、特許請求の範囲に定められる上述の特定の機能又は動作に限定されないことが理解されるべきである。むしろ、上述の特定の機能及び動作は、特許請求の範囲の実施の例示的携帯として開示されたものである。

10

#### 【 0 0 5 7 】

本願明細書で用いられるように、用語「モジュール」又は「コンポーネント」は、コンピューティングシステムで実行されるソフトウェアオブジェクト又はルーチンを表し得る。本願明細書に記載されたのと異なるコンポーネント、モジュール、エンジン及びサービスは、（例えば、別個のスレッドとして）コンピューティングシステムで実行されるオブジェクト又は処理として実施されても良い。本願明細書に記載されたシステム及び方法はソフトウェアで実施されることが望ましいが、ハードウェアによる実装又はソフトウェアとハードウェアの組合せも、可能であり想定される。この説明では、「コンピュータエンティティ」は、本願明細書で先に定められたようにコンピューティングシステム、又はコンピューティングシステムで実行されるモジュール若しくはモジュールの組合せであっても良い。

20

#### 【 0 0 5 8 】

本願明細書に記載された全ての例及び条件文は、教育上の目的で、読者が本発明の原理及び発明者により考案された概念を理解するのを助け、技術を促進させるためであり、これらの特に記載された例及び条件に限定されないものと考えられるべきである。本発明の実施形態が詳細に記載されたが、種々の変更、置換及び修正が本発明の精神及び範囲から逸脱することなく行われうることが理解されるべきである。

#### 【 符号の説明 】

#### 【 0 0 5 9 】

- 1 0 0 コンピューティング装置
- 1 0 2 基本構成
- 1 0 4 プロセッサ
- 1 0 6 システムメモリ
- 1 0 8 メモリバス
- 1 1 0 L 1 キャッシュ
- 1 1 2 L 2 キャッシュ
- 1 1 4 プロセッサコア
- 1 1 6 レジスタ
- 1 1 8 メモリ制御部
- 1 2 0 O S
- 1 2 2 アプリケーション
- 1 2 4 プログラムデータ
- 1 2 6 拡張可能アルゴリズム
- 1 2 8 記憶メタデータ
- 1 2 8 記憶メタデータ
- 1 3 0 バス／インタフェース制御部
- 1 3 2 記憶装置
- 1 3 4 記憶装置インタフェースバス
- 1 3 6 取り外し可能記憶装置
- 1 3 8 非取り外し可能記憶装置

30

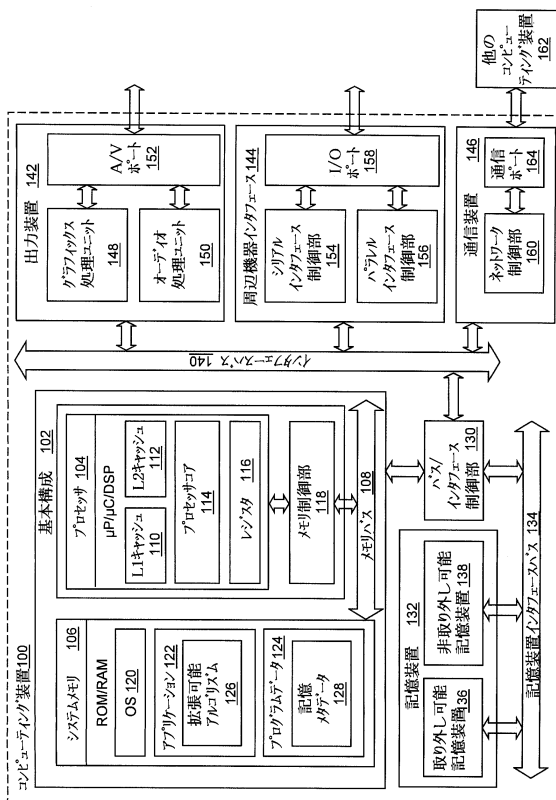
40

50

- |       |                    |
|-------|--------------------|
| 1 4 0 | インタフェースバス          |
| 1 4 2 | 出力装置               |
| 1 4 4 | 周辺機器インタフェース        |
| 1 4 6 | 通信装置               |
| 1 4 8 | グラフィックス処理ユニット      |
| 1 5 0 | オーディオ処理ユニット        |
| 1 5 2 | A / V ポート          |
| 1 5 4 | シリアルインタフェース制御部     |
| 1 5 6 | パラレルインタフェース制御部     |
| 1 5 8 | I / O ポート          |
| 1 6 0 | ネットワーク制御部          |
| 1 6 2 | 他のコンピューティング装置      |
| 1 6 4 | 通信ポート              |
| 2 0 2 | ハードディスクパーティションテーブル |
| 2 0 4 | ブロックアドレスデータ        |
| 2 0 6 | ブート命令コード           |

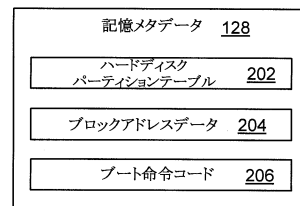
【 図 1 】

記憶メタデータを表すために拡張可能言語を用いるよう構成された  
例示的なコンピューティングシステムを示すブロック図



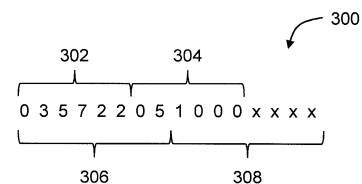
【圖 2】

図1の記憶メタデータの例示的な実施形態を示す図



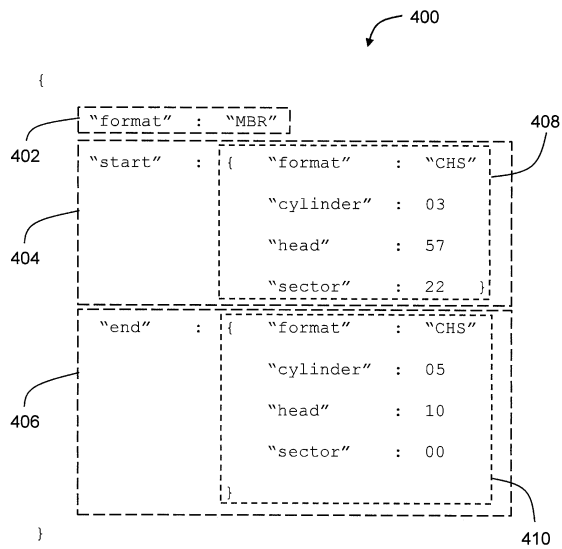
【 図 3 】

圧縮バイナリフォーマットの記憶メタデータの例示的な実施形態を示す図



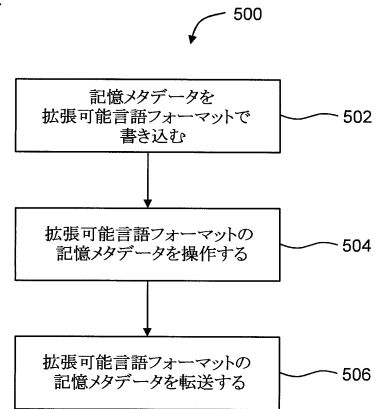
## 【図 4】

拡張可能言語フォーマットの図3の記憶メタデータの例示的な実施形態を示す図



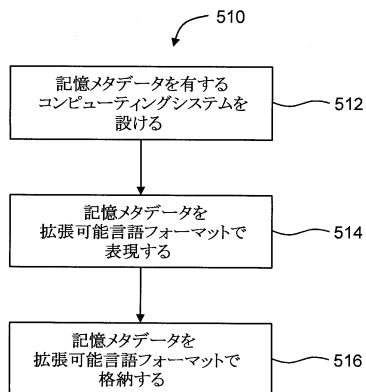
## 【図 5 A】

記憶メタデータを表現する拡張可能言語を用いる例示的な方法のフローチャート



## 【図 5 B】

記憶メタデータを表現する拡張可能言語を用いる別の例示的な方法のフローチャート



---

フロントページの続き

(72)発明者 マーヴィット・デイヴィド エル  
アメリカ合衆国，カリフォルニア州 94121，サンフランシスコ，トゥエンティフォース・ア  
ヴェニュー 815番

審査官 田中 啓介

(56)参考文献 特開2005-004870(JP,A)  
特開2011-258103(JP,A)  
特開2009-116873(JP,A)  
特開2009-043083(JP,A)

(58)調査した分野(Int.Cl.，DB名)  
G06F3/06-3/08、12/00