(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第5715566号
(P5715566)

(45)発行日　平成27年5月7日(2015.5.7)　(24)登録日　平成27年3月20日(2015.3.20)

(51)Int.Cl.		F I		
G06F 12/08	(2006.01)	G06F	12/08	531F
G06F 12/00	(2006.01)	G06F	12/00	514K
		G06F	12/08	551H
		G06F	12/08	557

請求項の数 12　(全 27 頁)

(21)出願番号	特願2011-526951 (P2011-526951)
(86)(22)出願日	平成21年9月9日 (2009.9.9)
(65)公表番号	特表2012-503232 (P2012-503232A)
(43)公表日	平成24年2月2日 (2012.2.2)
(86)国際出願番号	PCT/US2009/056419
(87)国際公開番号	WO2010/030715
(87)国際公開日	平成22年3月18日 (2010.3.18)
審査請求日	平成24年8月2日 (2012.8.2)
(31)優先権主張番号	61/097,079
(32)優先日	平成20年9月15日 (2008.9.15)
(33)優先権主張国	米国(US)
(31)優先権主張番号	12/271,472
(32)優先日	平成20年11月14日 (2008.11.14)
(33)優先権主張国	米国(US)

(73)特許権者　500046438
マイクロソフト　コーポレーション
アメリカ合衆国　ワシントン州　９８０５２－６３９９　レッドモンド　ワン　マイクロソフト　ウェイ
(74)代理人　100140109
弁理士　小野　新次郎
(74)代理人　100075270
弁理士　小林　泰
(74)代理人　100080137
弁理士　千葉　昭男
(74)代理人　100096013
弁理士　富田　博行
(74)代理人　100119781
弁理士　中村　彰吾

最終頁に続く

(54)【発明の名称】キャッシュデータおよびメタデータの管理

(57)【特許請求の範囲】

【請求項１】

記憶域媒体(storage medium)（７０６）およびキャッシュデバイス（１１０）が結合される(coupled)コンピューター（７００）を操作するための方法であって、前記コンピューターはオペレーティングシステムを含む、コンピューターを操作する(operating)ための方法であって、

（Ａ）前記記憶域媒体上のアドレスにデータ項目(data item)を書き込む要求に応答し、前記データ項目を前記記憶域媒体上の前記アドレスに（１０１）および前記キャッシュデバイス上の対応する(corresponding)アドレスに（１０２）書き込ませる(to be written)ステップと、

（Ｂ）前記オペレーティングシステムを再起動(reboot)した後(subsequent to)、前記データ項目を前記記憶域媒体上の前記アドレスから読み取る要求を満たすために、前記キャッシュデバイス上の前記アドレスに記憶した前記データ項目を確実に(reliably)使用できる(employed)かどうかを判定する(determining)ステップ（２７０、３３０）と、

（Ｃ）（Ｂ）において、前記データ項目を前記記憶域媒体上の前記アドレスから読み取る要求を満たすために、前記キャッシュデバイス上の前記アドレスに記憶した前記データ項目を確実に使用できると判定される場合、前記データ項目を前記キャッシュデバイス上の前記アドレスから読み取らせる(to be read from)ステップ（２９９）と

を含み、

前記行為（Ｂ）は、前記記憶域媒体上の前記アドレスから前記データ項目を読み取る要

求に応答して実行され、前記キャッシュデバイス上の前記アドレスに記憶した前記データ項目が、前記行為(act)（Ａ）の完了(completion)後に修正されなかった(not modified)ことを検証する(verifying)ステップをさらに含み、

（Ａ）は、少なくとも前記データ項目の表記(representation)を生成(generating)し、前記表記を前記キャッシュデバイスに書き込むステップをさらに含み、（Ｂ）において検証する前記行為は、

（Ｂ１）前記キャッシュデバイスに書き込まれた(written)前記表記を取得する(retrieving)ステップと、

（Ｂ２）前記表記を再生成する(re-generating)ステップと、

（Ｂ３）前記データ項目を読み取る前記要求を満たすために、前記キャッシュデバイス上の前記アドレスに記憶した前記データ項目を確実に使用できる(employed)かどうかを判定するために、（Ｂ１）で取得した(retrieved)前記表記を（Ｂ２）で再生成した前記表記と比較するステップと

を含み、

前記キャッシュデバイスに対しては実行されない(not performed)、前記記憶域媒体に対して実行される書込み操作を識別する(identify)よう動作する書込みレコーダコンポーネント(write recorder component)を提供する行為(act of providing)をさらに含み、

当該書込みレコーダコンポーネントは、オペレーティングシステムのＩ／Ｏパス内のドライバーとして実装することができ、又は、当該書込みレコーダコンポーネントの機能を提供するディスク記憶域ハードウェアとして実装することができ、

前記行為（Ｂ）は、前記記憶域媒体上の前記アドレスに記憶した前記データ項目が、前記行為（Ａ）の完了後に修正された(modified)かどうかを、前記書込みレコーダコンポーネントにより判定する(determining)ステップをさらに含み、

もし、前記書込みレコーダコンポーネントがすべての書込み操作は追跡できなかった場合に、キャッシュコンテンツの全体、または、当該書込み操作は追跡できなかった特定の部分が、除去されるステップを更に含む、

方法。

【請求項２】

前記行為（Ａ）～（Ｃ）は、前記オペレーティングシステムによって実行される、請求項１に記載の方法。

【請求項３】

前記記憶域媒体にはディスク記憶装置が含まれる、請求項１に記載の方法。

【請求項４】

前記キャッシュデバイスは、前記コンピューターから取り外すことができる、請求項１に記載の方法。

【請求項５】

前記行為（Ａ）は、少なくとも前記データ項目の表記を生成し、前記表記を前記キャッシュデバイスに書き込むステップをさらに含み、（Ｂ）において判定する前記行為は、

（Ｂ１）前記キャッシュデバイスに書き込まれた前記表記を取得する(retrieving)ステップと、

（Ｂ２）前記表記を再生成するステップと、

（Ｂ３）前記データ項目を読み取る前記要求を満たすために、前記キャッシュデバイス上の前記アドレスに記憶した前記データ項目を確実に使用できるかどうかを判定するために、（Ｂ１）で取得した前記表記を（Ｂ２）で再生成した前記表記と比較するステップと

を含む、請求項１に記載の方法。

【請求項６】

前記行為（Ａ）は、前記データ項目が書き込まれる前記キャッシュデバイス上の前記アドレスを求めるためにキャッシュメタデータを使用するステップをさらに含み、前記行為（Ｂ）は、前記キャッシュメタデータが（Ａ）の後に修正されたかどうかを判定するステップをさらに含む、請求項１に記載の方法。

【請求項７】
データが記憶される少なくとも１つのアドレスを有する記憶域媒体（７０６）と、
データが記憶される少なくとも１つのアドレスを有するキャッシュデバイス（１１０）と、
オペレーティングシステムを実行する少なくとも１個のプロセッサー（７０３）であって、
前記記憶域媒体上の第１のアドレスにデータ項目を書き込む要求に応答し、前記データ項目を前記記憶域媒体上の前記第１のアドレスに（１０１）および前記キャッシュデバイス上の対応するアドレスに（１０２）書き込ませ、
前記キャッシュデバイス上のアドレスに対して実行されない、前記記憶域媒体上のアドレスに対する書込み操作の発生(occurrence)を識別し(identify)（３００）、
前記オペレーティングシステムを再起動した後、前記キャッシュデバイス上の前記対応するアドレスに対して実行されなかった、前記記憶域媒体上の前記第１のアドレスへの書込み操作が実行されたかどうかを判定することにより、前記データ項目を前記記憶域媒体上の前記第１のアドレスから読み取る要求を満たすために、前記キャッシュデバイス上の前記対応するアドレスに記憶した前記データ項目を確実に使用できるかどうかを判定する（３０）よう、
オペレーティングシステムによりプログラムされる少なくとも１個のプロセッサー（７０３）と
を備え
前記識別する機能が、オペレーティングシステムのＩ／Ｏパス内のドライバーとして実装され、
もし、前記キャッシュデバイス上のアドレスに対して実行されない、前記記憶域媒体上のアドレスに対する書込み操作の発生(occurrence)を識別する(identify)ステップ（３００）において、すべての書込み操作は追跡できなかった場合に、キャッシュコンテンツの全体、または、当該書込み操作は追跡できなかった特定の部分が、除去される、
コンピューター（７００）。
【請求項８】
前記少なくとも１個のプロセッサーは、前記キャッシュデバイス上のアドレスに対して実行されない、前記記憶域媒体上のアドレスに対する書込み操作の発生(occurrence)を、前記記憶域媒体に書き込まれ、前記オペレーティングシステムの再起動時にアクセスされるログ(log)ファイルを使用して識別する(identify)ようにプログラムされる、請求項７に記載のコンピューター。
【請求項９】
前記少なくとも１個のプロセッサーは、前記データ項目を前記記憶域媒体上の前記第１のアドレスから読み取る要求を満たすために前記キャッシュデバイス上の前記対応するアドレスに記憶した前記データ項目を確実に使用できないと判定される場合、前記キャッシュデバイス上の前記対応するアドレスから前記データ項目を除去(evict)し、前記記憶域媒体上の前記第１のアドレスから前記データ項目を読み取ることにより前記要求を満たすようにプログラムされる、請求項７に記載のコンピューター。
【請求項１０】
前記キャッシュデバイス上のアドレスに対して実行されない、前記記憶域媒体上のアドレスに対する書込み操作の前記識別は、前記オペレーティングシステムの第１の機能によって有効にされ(enabled)、前記オペレーティングシステムは、前記オペレーティングシステムがシャットダウンされている(being shut down)と判断される場合に前記第１の機能をアクティブ化する(activate)ように構成される、請求項７に記載のコンピューター。
【請求項１１】
前記少なくとも１個のプロセッサーは、前記キャッシュデバイス上のアドレスに対して実行されない、前記記憶域媒体上のアドレスに対する書込み操作の発生を防ぐようにプログラムされる、請求項７に記載のコンピューター。

【請求項１２】
前記少なくとも１個のプロセッサーが識別するようにプログラムされる前記書込み操作は、前記記憶域媒体上のファイルシステムに記憶されるデータに対するものであり、前記少なくとも１個のプロセッサーは、前記オペレーティングシステムを再起動する前に、前記ファイルシステムによって使用されるログを特定の状態に置き、前記オペレーティングシステムを再起動した後、前記ログが前記特定の状態(particular state)にあるかどうかを判定することにより、前記記憶域媒体上のアドレスに対する書込み操作の発生を識別するようにプログラムされる、請求項７に記載のコンピューター。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、リムーバブルおよび非リムーバブルの不揮発性ランダムアクセスメモリー（ＮＶＲＡＭ）デバイスなど、不揮発性メモリーデバイスを使用するための技法に関する。
【背景技術】
【０００２】
一部の従来のオペレーティングシステムは、より遅い記憶装置（例えばディスク記憶域媒体や、ネットワークを介してアクセス可能な１つまたは複数の記憶装置）用のブロックレベルキャッシュまたはファイルレベルキャッシュとして、不揮発性メモリーデバイス（すなわちフラッシュメモリーＵＳＢドライブなど、コンピューターに補助記憶域および／またはメモリーを与えるように動作する周辺装置）を使用して、そのオペレーティングシステムおよび／またはアプリケーションの性能を向上させる能力を提供する。この点において、不揮発性メモリーデバイス（以下、簡潔にするために「キャッシュデバイス」と呼ぶ）との間では、より遅い記憶装置との間でよりもはるかに速く読み書き操作を実行することができるので、そうしたより遅いデバイス上に記憶されるデータをキャッシュするためにキャッシュデバイスを使用することは、オペレーティングシステムおよび／またはアプリケーションの入出力（Ｉ／Ｏ）操作の速度を著しく改善する機会を与える。そのために、Ｒｅｄｍｏｎｄ，ＷＡのＭｉｃｒｏｓｏｆｔ Ｃｏｒｐｏｒａｔｉｏｎによって製造されるＭｉｃｒｏｓｏｆｔ Ｗｉｎｄｏｗｓ Ｖｉｓｔａオペレーティングシステムは、ＲｅａｄｙＢｏｏｓｔとして知られる機能を含み、このＲｅａｄｙＢｏｏｓｔ機能は、同様により遅い記憶装置（以下、便宜上「ディスク記憶域」または「ディスク」と呼ぶが、これらの用語は一般に、ネットワークを介してアクセス可能な記憶装置を含む、Ｉ／Ｏが概してキャッシュデバイスよりも遅く実行される相手先である任意の１つもしくは複数の記憶域機構および／または１つもしくは複数のデバイスを指すことを理解すべきである）の中に存在するデータをキャッシュするために、ユーザーがキャッシュデバイスを使用することを可能にする。
【０００３】
ディスク上に記憶されるデータをキャッシュするためにキャッシュデバイスを使用することは、一部の実装形態ではオペレーティングシステムのＩ／Ｏスタックに実装されるドライバーである、キャッシュマネージャーコンポーネントを使用して達成することができる。図１Ａ～図１Ｂは、キャッシュマネージャーコンポーネント１００がキャッシュデバイス１１０へのデータのキャッシングを管理する、高水準プロセス１０Ａ～１０Ｂの例を示す。キャッシュデバイス１１０は、ワイヤードおよび／またはワイヤレス通信インフラストラクチャーならびに１つもしくは複数のプロトコルを使用して、キャッシュマネージャー１００がその上に存在するコンピューター（不図示）に結合することができる。例えば、キャッシュデバイス１１０は、コンピューターから取外し可能（例えばフラッシュメモリーＵＳＢドライブを含む）、コンピューターにとって取外し不可能かつ／または１つもしくは複数のワイヤードおよび／またはワイヤレスネットワークを介してアクセス可能とすることができる。
【０００４】
プロセス１０Ａ（図１Ａ）の開始時に、データがディスク記憶域（すなわちキャッシュ

済みボリューム１２０）上のアドレスＸに書き込まれるべきことを指定する、書込み要求がキャッシュマネージャー１００によって受け取られる。キャッシュマネージャー１００は、操作１０１において、そのデータをキャッシュ済みボリューム１２０上のアドレスＸに書き込ませ、操作１０２において、キャッシュデバイス１１０上のアドレスＹにも書き込ませることによりその要求を処理する。プロセス１０Ｂ（図１Ｂ）は、キャッシュ済みボリューム１２０上のアドレスＸに記憶されるデータを読み取るべきことを指定する読取り要求が受け取られる、その後実行される操作を含む。キャッシュマネージャー１００は、そのデータがキャッシュデバイス１１０上のアドレスＹにキャッシュされていることを判断し、操作１０３において、そのアドレスＹのデータを読み取らせる。操作１０４において、そのデータがキャッシュデバイスから供給されて読取り要求が満たされる。

【０００５】

このキャッシュマネージャーは、ディスクアドレス（例えばアドレスＸ）の、対応するキャッシュアドレス（例えばアドレスＹ）に対するマッピングをメタデータ内に保持し、この「キャッシュメタデータ」は、キャッシュデバイスとの間で読み書きする際に通常使用される。典型的には、キャッシュメタデータはメモリーの中に保持され、Ｉ／Ｏ要求が受け取られるときにキャッシュマネージャーによってアクセスされる。そのため、ディスクオフセットＸを対象にする読取り要求がキャッシュマネージャーによって受け取られるとき、そのキャッシュマネージャーはキャッシュメタデータを使用してそのデータがキャッシュオフセットＹにも記憶されていることを判断し、そのデータをディスクオフセットＸではなく、キャッシュオフセットＹから読み取らせることにより要求を満たす。ディスクオフセットＸを対象にする書込み要求がキャッシュマネージャーによって受け取られるとき、キャッシュマネージャーはキャッシュメタデータを使用してそのディスクアドレスのそのデータがキャッシュにも記憶されているかどうかを判断する。そうである場合（例えばそのデータがキャッシュアドレスＹに記憶されている場合）、キャッシュマネージャーは、そのデータをキャッシュ内の適切なアドレスに書き込ませ、またはそのアドレスのキャッシュコンテンツを除去（ｅｖｉｃｔ）することができる。そうでない場合、キャッシュマネージャーはそのデータをキャッシュに書き込ませることができ、ディスクオフセットＸに対する将来の読取りを代わりにキャッシュ上に記憶されるデータから供給できるよう、キャッシュメタデータを更新することができる。

【０００６】

従来のオペレーティングシステムは、比較的限られた記憶容量のキャッシュデバイスをサポートすることができる。例えば、Ｗｉｎｄｏｗｓ　Ｖｉｓｔａオペレーティングシステムの ＲｅａｄｙＢｏｏｓｔ機能は、最高４ギガバイトまでの記憶容量のキャッシュデバイスをサポートする。（Ｗｉｎｄｏｗｓ　Ｖｉｓｔａがリリースされた時点では、キャッシュデバイスの最大記憶容量は約２ギガバイトだった）。キャッシュデバイスの記憶容量は近年急激に増加しており、一部のキャッシュデバイスは最高１６ギガバイトまでの記憶容量を提供し、この記憶容量は圧縮時には３２ギガバイト相当のデータを記憶することができる。

【発明の概要】

【課題を解決するための手段】

【０００７】

出願者らは、比較的より大きい記憶容量を有するキャッシュデバイスが、オペレーティングシステムおよびアプリケーションによって実行されるＩ／Ｏ操作の速度を改善する重大な機会を提供することが分かった。出願者らはさらに、従来のオペレーティングシステムが比較的限られた記憶容量のキャッシュデバイスしかサポートしない１つの理由は、特定の種類の電源切替え（例えばスタンバイ、休止状態（または非Ｍｉｃｒｏｓｏｆｔ　Ｗｉｎｄｏｗｓオペレーティングシステムによって使用される等価のモード）、または再起動）が生じるとき、キャッシュコンテンツが再作成（ｒｅｐｏｐｕｌａｔｅ）されなければならないことであることも分かった。比較的より大きな記憶容量を有するキャッシュデバイスでは、キャッシュコンテンツを再作成することは、時間が相当かかり、かなりの処

理リソースを消費する可能性がある。一例として、最高１６ギガバイトまでの圧縮データを保持することができる８ギガバイトのフラッシュメモリーデバイスは、ディスクからの約１０メガバイト／秒のバックグラウンドＩ／Ｏを使用して、再作成するために最高３０分かかる場合がある。このことは、キャッシュデバイスを使用することによって得られた可能性のある任意の性能上の利益を事実上打ち消すだけでなく、さらにシステムの動作を著しく遅くする場合がある。

【０００８】

特定の電源切替えにわたりキャッシュコンテンツを再作成しなければならない１つの理由は、キャッシュデバイスおよび／またはディスクのコンテンツが電源切替え中に修正された可能性があるので、コンピューターが再起動されるとき、キャッシュコンテンツがディスク記憶域のコンテンツを正確に表すことを確実に保証する方法がないことである。例えば、第１のコンピューターがシャットダウンされるとき、ハッカーはリムーバブルキャッシュデバイスを取り外し、それを別のコンピューターに接続し、その後そのデバイスが第１のコンピューターに再び接続される場合に、誤ったデータ（以下、「本物でない」データと呼ぶ）がキャッシュデバイスから供給されてＩ／Ｏ要求を満たすことができるよう、キャッシュコンテンツを修正することができる。ハッカーの悪意のある行為に加え、キャッシュコンテンツは、コンピューターまたはキャッシュデバイスのハードウェア障害に起因する電源切替え中に破損する可能性もある。

【０００９】

電源切替え中にディスク上のデータが更新されたので、キャッシュコンテンツはその切替え中に「古く」なる可能性もあり、そのためコンピューターが再起動されるとき、キャッシュコンテンツはもはやディスクコンテンツを正確に表さない場合がある。例えば、シャットダウン後にユーザーは、キャッシュデバイスを認識しない別のオペレーティングシステム内にディスクを起動し、キャッシュデバイス上にキャッシュされている、そのディスク上に記憶されたデータを修正することができ、そのためコンピューターが再起動されるとき、キャッシュコンテンツはもはやディスク上に記憶されるものを反映しない。別の例では、キャッシュデバイスがオペレーティングシステムにとってアクセスできなくされた後（例えばキャッシュデバイスがオフにされた後）、シャットダウン中にコンピューター上で特定の動作が生じる可能性があり、これにより、この時点の後にオペレーティングシステムによって実行されるディスクへの任意の書込みが、キャッシュコンテンツにより正確に反映されない場合がある。数多くのイベントのうちのいずれかが、電源切替えにわたりキャッシュコンテンツを古くさせることができる。

【００１０】

本発明の諸実施形態は、電源切替えにわたり、ディスク上に記憶されるデータを正確に反映するものとしてキャッシュコンテンツに依拠できるよう、これらのおよび他の問題を管理するための技法を提供する。例えば本発明の一部の実施形態は、キャッシュコンテンツが電源切替えにわたり本物のままであることを検証するための技法を提供する。さらに、一部の実施形態は、キャッシュコンテンツが電源切替えにわたり古くならないことを確実に保証するための技法を提供する。さらに、一部の実施形態は、キャッシュメタデータを電源切替えにわたってだけでなく通常（「定常状態」）動作中にも管理し、電源切替えが生じるとき、キャッシュメタデータに効率的にアクセスし、確実に保存／復元し得ることを保証するための技法を提供する。

【００１１】

電源切替えにわたり、ディスク上に記憶されるデータを正確に反映するものとしてキャッシュコンテンツを利用できることを保証する技法を提供することにより、本発明の一部の実施形態は、オペレーティングシステムおよび／またはアプリケーションによって実行されるＩ／Ｏ操作を著しく速めるために、かなりの記憶容量のキャッシュデバイスが使用されることを可能にすることができる。Ｉ／Ｏ操作について増加される速度は、コンピューターの通常の「定常状態」動作を促進するだけでなく、起動中に実行される操作を著しく速めることもでき、それによりそのコンピューターははるかに速く使える状態になる。

【００１２】
一部の実施形態では、記憶域媒体およびキャッシュデバイスが結合されるコンピューターを操作するための方法であって、そのコンピューターはオペレーティングシステムを含む、コンピューターを操作するための方法を提供する。この方法は、（Ａ）記憶域媒体上のアドレスにデータ項目を書き込む要求に応答し、そのデータ項目をその記憶域媒体上のアドレスおよびキャッシュデバイス上の対応するアドレスに書き込ませる行為と、（Ｂ）オペレーティングシステムを再起動した後、そのデータ項目を記憶域媒体上のアドレスから読み取る要求を満たすために、そのキャッシュデバイス上のアドレスに記憶したデータ項目を確実に使用できるかどうかを判定する行為と、（Ｃ）（Ｂ）において、そのデータ項目を記憶域媒体上のアドレスから読み取る要求を満たすために、そのキャッシュデバイス上のアドレスに記憶したデータ項目を確実に使用できると判定される場合、そのデータ項目をそのキャッシュデバイス上のアドレスから読み取らせる行為とを含む。
【００１３】
他の実施形態は、オペレーティングシステムを含みかつキャッシュデバイスが結合されるコンピューターによって実行されるとき、（Ａ）コンピューター記憶域媒体上のアドレスにデータ項目を書き込む要求に応答し、そのデータ項目をそのコンピューター記憶域媒体上のアドレスおよびキャッシュデバイス上の対応するアドレスに書き込ませる行為と、（Ｂ）オペレーティングシステムを再起動した後に、そのデータ項目をコンピューター記憶域媒体上のアドレスから読み取る要求に応答して、そのキャッシュデバイス上のアドレスに記憶したデータ項目が（Ａ）の後に修正されたかどうかを判定する行為と、（Ｃ）（Ｂ）において、そのキャッシュデバイス上のアドレスに記憶したデータ項目が（Ａ）の後に修正されなかったと判定される場合、そのデータ項目をそのキャッシュデバイス上のアドレスから読み取らせる行為とを含む方法を実行する命令が記憶された少なくとも１つのコンピューター記憶域媒体を提供する。
【００１４】
さらに他の実施形態は、データが記憶される少なくとも１つのアドレスを有する記憶域媒体と、データが記憶される少なくとも１つのアドレスを有するキャッシュデバイスと、オペレーティングシステムによりプログラムされる少なくとも１個のプロセッサーであって、記憶域媒体上の第１のアドレスにデータ項目を書き込む要求に応答し、そのデータ項目をその記憶域媒体上の第１のアドレスおよびキャッシュデバイス上の対応するアドレスに書き込ませ、そのキャッシュデバイス上のアドレスに対してはやはり実行されない、その記憶域媒体上のアドレスへの書込み操作の発生を識別し、オペレーティングシステムを再起動した後、そのキャッシュデバイス上の対応するアドレスに対してはやはり実行されなかった、その記憶域媒体上の第１のアドレスへの書込み操作が実行されたかどうかを判定することにより、そのデータ項目を記憶域媒体上の第１のアドレスから読み取る要求を満たすために、そのキャッシュデバイス上の対応するアドレスに記憶したデータ項目を確実に使用できるかどうかを判定するようオペレーティングシステムによりプログラムされる、少なくとも１個のプロセッサーとを備えるコンピューターを提供する。
【図面の簡単な説明】
【００１５】
【図１】図１Ａは、従来技術による、キャッシュデバイスとの間で読み書きするための技法を示すブロック図である。図１Ｂは、従来技術による、キャッシュデバイスとの間で読み書きするための技法を示すブロック図である。
【図２Ａ】本発明の一部の実施形態による、キャッシュデバイスとの間で読み書きするための例示的技法を示す図である。
【図２Ｂ】本発明の一部の実施形態による、キャッシュデバイスとの間で読み書きするための例示的技法を示す図である。
【図３】本発明の一部の実施形態による、電源を切り替えた後、キャッシュデータが、ディスク上に記憶されたデータを正確に反映することを保証するための例示的技法を示すブロック図である。

【図４】本発明の一部の実施形態による、キャッシュメタデータを記憶するための例示的技法を示すブロック図である。
【図５】本発明の一部の実施形態による、キャッシュメタデータのための例示的記憶操作を示すブロック図である。
【図６】本発明の一部の実施形態による、キャッシュデバイスを使用して読取り要求に応えるための例示的技法を示す流れ図である。
【図７】本発明の諸態様を実施するために使用することができるコンピューターの一例を示すブロック図である。
【図８】本発明の諸態様を実施する命令を記録することができるコンピューターメモリーの一例を示すブロック図である。
【発明を実施するための形態】
【００１６】
本発明の一部の実施形態は、キャッシュコンテンツが、電源切替えにわたりディスク記憶域のコンテンツを正確に反映することを保証するための技法を提供する。例えば、一部の実施形態は、電源切替えにわたりキャッシュコンテンツが本物のままでありかつ／または古くなっていないことを保証する能力を提供する。さらに、一部の実施形態は、電源切替え中にメタデータが改ざんされていないことを保証するためにキャッシュメタデータを管理するための技法を提供する。さらに、一部の実施形態はキャッシュメタデータを記憶する能力を提供し、この能力は、電源切替えおよび通常動作の両方を実行し得る効率を改善することができる。以下のセクションは、これらの実施形態を詳細に説明する。
Ｉ．キャッシュデータの信頼性の検証
コンピューターが電源切替えを経験する（例えばスタンバイモードもしくは休止状態モードにされ、または再起動される）とき、そのコンピューターからキャッシュデバイスを取り外すことができ、そのキャッシュデバイスのコンテンツが（例えば悪意のあるハッカーにより）改変される可能性がある。例えば、コンピューターがスタンバイモードまたは休止状態モードにされるとき、そのコンピューターからフラッシュメモリードライブなどのリムーバブルキャッシュデバイスを取り外し、そのリムーバブルキャッシュデバイスのコンテンツを修正することができる。オペレーティングシステムが再起動される（すなわち再度読み込まれ、それによりコンピューターを再起動する）とき、内蔵ＮＶＲＡＭデバイスなどの非リムーバブルデバイスでさえ、取り外し、そのコンテンツを変更することができる。その結果、コンピューターにそのキャッシュデバイスが再び接続されるとき、そのキャッシュデバイスはユーザーが思うのとは違う情報を記憶している可能性がある（すなわち、キャッシュに記憶されているデータは「本物」ではない可能性がある）。入出力（Ｉ／Ｏ）要求を満たすために、本物でないデータがキャッシュから供給される場合、そのコンピューターの動作は悪い方に影響を受ける場合がある。
【００１７】
本発明の一部の実施形態は、Ｉ／Ｏ要求がキャッシュからの本物でないデータを使用して満たされないことを確実にするために、電源切替え中にキャッシュコンテンツに生じる修正を検出するための技法を提供する。一部の実施形態では、電源切替え中にキャッシュコンテンツに生じ、そのキャッシュコンテンツを本物でないものにする任意の「オフライン修正」を検出するための能力を提供する。
【００１８】
一部の実施形態は、この能力をデータの少なくとも一部分の表記（ｒｅｐｒｅｓｅｎｔａｔｉｏｎ）を使用して提供する。例えば、表記は、データおよび／または他の情報の少なくとも一部分から所定の方法で計算しもしくは導き出すことができる。この表記は、データがキャッシュに書き込まれ、１つまたは複数の位置に記憶されるときに初めて生成することができる。例えば、表記は、データとともにキャッシュにまたは他の何らかの１つもしくは複数の位置に書き込むことができる。キャッシュに記憶する場合、この表記は、この表記をデータに関連させる方法で記憶することができる（例えば、この表記はデータが書き込まれるアドレスに隣接するキャッシュアドレスに書き込み、データに関連しかつ

／または他の何らかの様式で関連するキャッシュメタデータに書き込むことができる）。その後、そのデータをキャッシュから読み取るとき、その表記も取得することができる。この表記は所定の方法で再生成することができ、再生成される表記は取得される表記と比較することができる。これらの表記が一致する場合、キャッシュから取得されるデータは本物であると判定され、読取り要求を満たすために供給される。一致しない場合、その読取り要求を満たすために代わりにディスク記憶域からデータを読み取るための要求が発行され、キャッシュ上に記憶された本物でないデータは除去（例えば削除）することができる。

【００１９】

図２Ａ～図２Ｂは、このプロセスをさらに詳細に示す。具体的には、図２Ａに示すプロセス２０Ａは、キャッシュにデータが書き込まれるときに実行される操作を含み、図２Ｂに示すプロセス２０Ｂは、その後キャッシュからそのデータが読み取られるときに実行される操作を含む。プロセス２０Ａ～２０Ｂは、（各図面の左側にある）キャッシュ済みボリューム１２０、（真ん中の）キャッシュマネージャー１００、および（右にある）キャッシュデバイス１１０によって実行される操作をそれぞれ含む。

【００２０】

プロセス２０Ａ（図２Ａ）の開始時に、行為２０５で、ディスク記憶域上のアドレスＸを対象にする書込み要求がキャッシュマネージャー１００によって受け取られる。行為２１０で、キャッシュマネージャー１００は、キャッシュメタデータ（不図示）を使用して、同様にデータが書き込まれるべきキャッシュデバイス１１０上のアドレスＹを求める。キャッシュマネージャーは、データの少なくとも一部分の表記を生成することも行う。

【００２１】

本発明の諸実施形態は、この表記を任意の適切な技法を使用して生成することができる。一部の実施形態では、この表記を生成するために１つまたは複数の暗号の認証技法を使用することができる。例えば一部の実施形態では、この表記は、データおよび１組の秘密キーならびにデータ項目ごとのシーケンス番号から生成される、メッセージ認証コード（ＭＡＣ）を含むことができる。ただし、本発明はそうした実装形態に限定されず、この表記を生成するための任意の適切な技法を使用することができる。例えば、暗号の認証技法を使用する必要はない。一例として、ストロングハッシュ（ｓｔｒｏｎｇ　ｈａｓｈ）および／または巡回冗長コード（ＣＲＣ）を、データを表すために代わりに使用することができ、キャッシュに記憶される個々のデータ項目からまたはデータ項目の１つもしくは複数のグループについて生成することができる。この点で、出願者らは、データの信頼性を検証する目標が単にハードウェア破損のインスタンスを検出することである（すなわちデータのハッキングは問題でない）場合、ＣＲＣを使用することで十分な場合があり、データ項目ごとにＭＡＣを生成するよりも少ない処理リソースを消費できることが分かった。しかし、その目標がハッカーや悪意のある行為者がキャッシュコンテンツを修正することを防ぐことである場合、暗号の解決策が好ましい場合があり、そのためＭＡＣなどの表記を使用することができる。本発明はどんな特定の実装形態にも限定されず、任意の適切な技法を使用することができる。

【００２２】

行為２１５で、キャッシュマネージャー１００が、キャッシュアドレスＹにデータを書き込むための要求をキャッシュデバイス１１０に対して発行する。キャッシュマネージャーは、表記を書き込むための要求もキャッシュデバイス１１０に対して発行する。例えば、キャッシュマネージャー１００は、その表記がキャッシュアドレスＹに隣接する１つもしくは複数の位置に書き込まれ、またはデータのキャッシュメタデータ内におよび／または他の任意の技法を使用して記憶されるべきことを指定することができる。上記で論じたように、本発明はこの表記をどんな特定の位置に書き込むことにも限定されない（例えば、この表記はキャッシュデバイス１１０に書き込まれる必要はない）。キャッシュに書き込まれる場合、この表記は所望の任意の方法でデータに関連させることができる。

【００２３】

行為２２０で、キャッシュデバイス１１０がその要求を受け取り、行為２２５で、そのデータおよび表記を書き込むことによりその要求を処理する。

行為２３０で、キャッシュマネージャー１００が、データをディスクアドレスＸに書き込むための、対応する要求をキャッシュ済みボリューム１２０に対して発行する。行為２３５で、キャッシュ済みボリューム１２０がこの要求を受け取り、行為２４０で、そのデータをアドレスＸに書き込むことによりその要求を処理する。図２Ａには、行為２１５～２２５の、キャッシュへの書込み後に実行されるものとして示すが、行為２３０～２４０は、行為２１５～２２５と同時に、または他の任意の適切な１つもしくは複数の時点において実行することができ、本発明はどんな特定の実装形態にも限定されない。これでプロセス２０Ａは完了する。

【００２４】

プロセス２０Ｂ（図２Ｂ）は、キャッシュに記憶されたデータを読み取るために実行される操作を含む。行為２４５で、キャッシュマネージャー１００が、キャッシュ済みボリューム１２０上のアドレスＸに記憶されたデータを読み取る要求を受け取る。行為２５０で、キャッシュメタデータ（不図示）を使用し、キャッシュマネージャー１００は、そのデータがキャッシュデバイス１１０上のアドレスＹに記憶されていることを判定する。行為２５５で、キャッシュマネージャー１００が、アドレスＹに記憶されたデータおよび関連する表記の両方を取得するための読取り要求をキャッシュデバイス１１０に対して発行する。この要求は行為２６０でキャッシュデバイス１１０によって受け取られ、行為２６５で処理され、それからキャッシュデバイス１００はその結果をキャッシュマネージャー１００に返す。

【００２５】

行為２７０で、キャッシュマネージャー１００は、キャッシュから取得したデータを検証できるかどうかを判定する。一部の実施形態では、この判定することは、データの表記を再生成し、その再生成した表記を行為２１５で当初生成した表記と比較することによって行われる。例えば、行為２７０は、データのＭＡＣまたはＣＲＣを再生成し、その再生成したＭＡＣまたはＣＲＣを行為２６５でキャッシュから取得した表記と比較することを含むことができる。

【００２６】

行為２７０で、その表記が検証できると判定される場合、このプロセスは行為２７５に進み、行為２７５では、読取り要求を満たすためにキャッシュデバイス１１０から取得されるデータが供給され、これでプロセス２０Ｂは完了する。行為２７０で、その表記が検証できないと判定される場合、このプロセスは行為２８０に進み、行為２８０では、キャッシュマネージャー１００が、アドレスＹに記憶されたデータを除去する（例えば、消去しまたは別の方法でアクセスできなくする）要求をキャッシュデバイス１１０に対して発行する。次いで行為２８５で、キャッシュマネージャー１００が、ディスク上のアドレスＸからそのデータを読み取る要求をキャッシュ済みボリューム１２０に対して発行する。この要求は行為２９０で受け取られ、行為２９５で処理され、それからそのデータがキャッシュマネージャー１００に返される。次いで行為２９９で、読取り要求を満たすためにアドレスＸから読み取られるデータが供給される。これでプロセス２０Ｂは完了する。

【００２７】

データの信頼性を検証するための暗号の解決策を使用する場合、ハッカーがキーにアクセスして、変更されるデータ項目のための表記を再生成することを防ぐために、電源切替えの期間の間、表記を生成するために使用される任意の１つまたは複数のキーをキャッシュデバイス以外の位置に書き込むことができる。例えば一部の実施形態では、（例えばコンピューターがシャットダウンされるとき）キーをディスク記憶域に記憶して不正アクセスを防ぐことができる。ただし、本発明はそうした実装形態に限定されず、キーを記憶する必要はなく、記憶する場合は任意の適切な位置にあることができる。例えば、記憶されるキーは、システムの起動中に使用可能な、オペレーティングシステムによって提供される任意の構成ストア（例えばＷｉｎｄｏｗｓのシステムレジストリー）内に配置し、また

はキーを記憶する必要がないよう、何らかのユーザー入力（例えばパスワード）に基づいて再生成することができる。

【００２８】

キャッシュデバイス上に記憶されるデータ項目の信頼性を検証するための上記に記載した実施形態は例に過ぎず、信頼性は任意の適切な技法を使用して検証することができることを理解すべきである。例えば、データ項目の信頼性は、そのデータ項目が書き込まれるとき、そのデータ項目を読み取るときに後で再生成されるそのデータ項目の少なくとも一部分の表記を生成することにより検証する必要はない。キャッシュから読み取られるデータ項目が本物であり、前にキャッシュに書き込まれたデータ項目に一致することを確実に保証する任意の適切な技法を使用することができる。本発明はどんな特定の実装形態にも限定されない。

ＩＩ．キャッシュデータが古くなること（ｓｔａｌｅｎｅｓｓ）を防ぐ

上記で論じたように、従来のオペレーティングシステムは、電源切替え中に、ディスク上に記憶されたデータ項目に対して書込み操作が実行されるときを検出することができず、このことはキャッシュコンテンツを古くする。本発明の一部の実施形態は、これらの「オフライン書込み」が生じるときを検出するための機構を提供し、それにより、電源切替えが生じた後にキャッシュコンテンツがディスク上に記憶されたデータを正確に反映することを保証する。

【００２９】

一部のオペレーティングシステム（例えばＭｉｃｒｏｓｏｆｔ　Ｃｏｒｐｏｒａｔｉｏｎによって提供されるＷｉｎｄｏｗｓファミリーのオペレーティングシステム）では、一定の電源切替え（例えばスタンバイモードや休止状態モード）のセマンティックは、電源切替えの間、非リムーバブル記憶域デバイス（例えばディスク記憶域）上のデータを修正できないことである。そのため、そうした非リムーバブル媒体上のデータに対応するキャッシュコンテンツは概して古くならない。しかし、コンピューターがシャットダウンされるとき、ディスク上のデータが修正されることを可能にするいくつかのことが起こる可能性がある。例えば、ユーザーはそのコンピューター上の別のオペレーティングシステム内にそのディスクを起動し、またはそのディスクを別のコンピューターに接続し、ディスク上に記憶されたデータを修正することができる。さらに、上記で論じたように、多くの従来のオペレーティングシステムのシャットダウンの仕組みは、シャットダウン中の何らかの時点においてキャッシュデバイスがオフにされ、オペレーティングシステムによってもはやアクセスできなくなるが、そのオペレーティングシステムはディスクに引き続きアクセスできるものである。そのため、そのオペレーティングシステムは、キャッシュデバイス上にキャッシュされる、ディスク上のデータ項目を更新することができる。キャッシュデバイスはオフにされていたので、オペレーティングシステムがこれらのキャッシュコンテンツも更新するすべはなく、そのためこれらのキャッシュコンテンツは古くされる。

【００３０】

これらのおよび他の発生を管理するために、本発明の一部の実施形態はシャットダウンが開始された後の、ディスク上に記憶されたデータに対する修正を検出するための技法を提供し、それによりそうした修正によって古くされるキャッシュコンテンツを更新し、キャッシュから除去し、または他の方法で処理することができる。

【００３１】

キャッシュデバイスがシャットオフされた後に発生する、シャットダウン操作中にディスク記憶域に対して実行される書込みを検出するために、本発明の一部の実施形態は書込みレコーダーコンポーネントを使用する。書込みレコーダーコンポーネントは、例えばオペレーティングシステムのＩ／Ｏパス内のドライバーとして実装することができるが、本発明はそうした実装形態に限定されない。例えば、書込みレコーダーコンポーネントはハードウェアベースとすることができる。一例として、ディスク記憶域ハードウェアは、一定の期間中に生じた１組の修正を識別し、または一定の期間中に修正が生じたかどうかを識別する能力を与える１つもしくは複数のインターフェイスを提供することができる。例

えば、ディスク記憶域ハードウェアは、少なくとも一部の記憶済みデータ項目が更新されたことであって、その場合ディスク上に記憶されるデータに対応するキャッシュコンテンツを除去することができる（この除去することは頻繁に生じるべきでなく、そのためキャッシュデバイスを使用することは依然としてかなりの利益をもたらすことになる）、少なくとも一部の記憶済みデータ項目が更新されたことを推論するために使用することができる、スピンアップ／パワーアップ／起動カウンターを提供することができる。本発明はどんな特定の実装形態にも限定されない。

【００３２】

一部の実施形態では、この書込みレコーダーコンポーネントは、シャットダウンが開始されるときにアクティブになり、シャットダウンが完了するまで、ディスク記憶域に対して実行されるすべての書込みを追跡するように構成される。その結果、後にコンピューターが再起動されるとき、キャッシュコンテンツにこれらの書込みを施すことができる。例えばコンピューターが再起動され、ディスクボリュームがオンラインになるとき、キャッシュマネージャーを起動することができ、ディスクに対する書込みを追跡し始めることができる。キャッシュマネージャーは、書込みレコーダーコンポーネントにクエリして、キャッシュデバイスがシャットオフされた後に生じたオフライン書込みを判断し、これらの書込みをキャッシュマネージャーが起動中に追跡した書込みとマージし、キャッシュコンテンツにそのマージした１組の書込みを施すことができる。キャッシュコンテンツに書込みを施すことは、例えば書込みの対象にされたディスク上のデータに対応するキャッシュコンテンツを更新すること（例えばこれらのキャッシュコンテンツに対して同じ書込み操作を実行すること）、これらのキャッシュコンテンツを除去すること、その２つの組合せ（例えば特定のキャッシュコンテンツに書込み操作を施し、他のものを除去すること）、または他の何らかの１つもしくは複数の操作を実行することを含むことができる。キャッシュコンテンツにオフライン書込みが施された後、書込みレコーダーコンポーネントをシャットダウンすることができ、キャッシュデバイスはＩ／Ｏ要求に応えることを開始することができる。

【００３３】

図３は、オフライン書込みを追跡し、キャッシュコンテンツにこれらの書込みを施すためのプロセスの一例３０を示す。具体的には、プロセス３００は、コンピューターのシャットダウンおよびその後の再起動中にキャッシュマネージャー１００、書込みレコーダー３００、キャッシュデバイス１１０、およびキャッシュ済みボリューム１２０によって実行される操作を含む。

【００３４】

コンピューターのシャットダウン中に生じる行為３０５で、キャッシュマネージャー１００が書込みレコーダー３００をアクティブ化し、書込みレコーダーによって追跡されるべき書込み操作の組（すなわち世代）を識別する「保持識別子（ｐｅｒｓｉｓｔｅｎｃｅ　ｉｄｅｎｔｉｆｉｅｒ）」を書込みレコーダー３００に供給する。（保持識別子に関する使用の例は以下に詳細に記載する）。行為３１０で、キャッシュマネージャー１００がその保持識別子、ならびにメモリーの中に記憶されるキャッシュメタデータをキャッシュデバイス１１０に書き込む。シャットダウンプロセスのこの時点において、キャッシュデバイス１１０がオフにされ、キャッシュマネージャー１００にとってアクセスできなくなる。

【００３５】

行為３１５で、書込みレコーダー３００が、行為３０５で自らに渡された保持識別子をキャッシュ済みボリューム１２０に書き込み、シャットダウンの間にキャッシュ済みボリューム１２０に対して実行される任意の書込み操作を追跡し始める。例えば、書込みレコーダー３００は、書込み操作が実行されるディスク上のアドレスおよび／またはそれらのアドレスに書き込まれるデータを指示するために、キャッシュ済みボリューム１２０上にまたは他の何らかの１つもしくは複数の位置にログファイルまたは１つもしくは複数の他のデータ構造を作成することができる。行為３１５の完了時に、このコンピューターのシ

ャットダウン操作は完了している。

【００３６】

その後、このコンピューターが再起動される。この起動プロセスの一部として、キャッシュ済みボリューム１２０がオンラインにされ、書込みレコーダー３００およびキャッシュマネージャー１００が再起動される。次いで、キャッシュマネージャー１００は、キャッシュ済みボリューム１２０に対して実行される書込み操作の追跡を開始することができる。例えば、キャッシュマネージャー１００はログファイルを作成し、そのログファイルをキャッシュデバイス１１０、キャッシュ済みボリューム１２０、および／またはコンピューターのメモリー（図３には不図示）上に記憶することができる。行為３２０で、書込みレコーダー３００が、行為３１５でログされたボリュームの変更、ならびに行為３１５でキャッシュ済みボリューム１２０に書き込まれた保持識別子を読み取る。次いで行為３２５で、このボリュームの変更および保持識別子がキャッシュマネージャー１００に渡される。

【００３７】

この書込みレコーダー３００は、キャッシュデバイス１１０がオフにされた後の、ディスクに対するすべての書込みを追跡する能力はなくてよいことを理解すべきである。例えば、ハードウェアのデータ破損、タイミングの悪い電源障害、および／またはログファイルを書き込む上での問題は、書込みレコーダー３００がディスクボリュームに対して実行されるすべてのオフライン書込みを追跡することをできなくする可能性がある。そのような場合、行為３２５で、書込みレコーダー３００は、そのログが、実行されたすべてのオフライン書込みの完全かつ正確な記録であると確実に判断できないことをキャッシュマネージャー１００に知らせることができる。これが生じる場合、キャッシュマネージャー１００は、潜在的に信頼できないものであるとして、キャッシュコンテンツの全体または（例えば書込みレコーダーがすべての書込み操作は追跡できなかった特定のディスクボリュームに対応する）その一部分を除去することができる。図３についての記載の残りの部分は、書込みレコーダー３００がすべてのオフライン書込みを追跡する能力があるとみなす。

【００３８】

行為３３０で、キャッシュマネージャー１００が、キャッシュメタデータおよび保持識別子をキャッシュデバイス１１０からメモリーの中に読み取る。キャッシュマネージャー１００は、その保持識別子を検証できるかどうかを判定する（これについては以下にさらに記載する）。検証できない場合、キャッシュマネージャー１００は、キャッシュデバイス１１０のコンテンツの全体または（例えば保持識別子を検証できなかった特定のディスクボリュームに対応する）その一部分を除去することができる。保持識別子を検証することができる場合、キャッシュマネージャー１００は、コンピューターが再起動されてからディスク記憶域に対して実行される任意の書込み操作を、書込みレコーダー３００によって追跡される任意の書込み操作とマージする。例えば、１つまたは複数のログが、ディスク上のそれぞれのアドレスに書き込まれるデータを示す場合、キャッシュマネージャー１００はそれぞれのアドレスに対して実行される最新の更新を選択し、それをメモリーに書き込むことができる。

【００３９】

一部の実施形態では、書込みレコーダー３００は、コンピューターが再起動された後に書込みを記録し続けるよう構成することができ、それにより、キャッシュマネージャー１００は、その時点の後に実行される書込みを記録し、それらの書込みを書込みレコーダー３００によって追跡される書込みとマージする必要はない。代わりに、書込みレコーダー３００は、すべての書込みの記録をキャッシュマネージャー１００に単純に提供することができる。

【００４０】

行為３３０で読み取ったキャッシュメタデータを使用し、行為３３５で、キャッシュマネージャー１００はその１組の書込みをキャッシュデバイス１１０のコンテンツに施す。

上記に記載したように、書込みを施すことは、キャッシュコンテンツを除去すること、キャッシュコンテンツを更新すること、その両方を行うこと、または他の何らかの１つもしくは複数の操作を実行することを含むことができる。例えば、行為３１５で書込みレコーダー３００によって追跡されるオフライン書込みは、対応するキャッシュコンテンツを除去することによって施すことができる一方で、コンピューターが再起動されてからキャッシュマネージャー１００によって追跡される書込みは、それらの書込みを反映させるために対応するキャッシュコンテンツを更新することによって施すことができる。キャッシュコンテンツに書込み操作を施すことは任意の適切な方法で行うことができ、本発明はどんな特定の実装形態にも限定されない。

【００４１】

行為３３５の完了時に、図３のプロセスは完了する。

本発明は、シャットダウンが開始されるときにアクティブになるように構成される書込みレコーダーコンポーネントを使用することに限定されず、シャットダウン中に発生しない書込み操作も追跡しまたはそうした書込み操作を代わりに追跡し得ることを理解すべきである。例えば、一部の実装形態では、キャッシュデバイスがある期間にわたりアクセス不能になりやすい可能性がある。例えば、そのキャッシュデバイスが１つまたは複数のネットワークを介してアクセスされる場合は接続性が失われる可能性があり、またはそのキャッシュデバイスがコンピューターから取外し可能な場合は不意の（例えば意図的でない）取外しが生じる可能性がある。その結果、一部の実施形態は、シャットダウン中に発生する書込みだけでなくディスクに対して実行されるすべての書込み（またはその一部分）を追跡するための書込みレコーダーと、引き続きオンラインでありながら、定期的にキャッシュの「スナップショット」をキャプチャするように構成されるキャッシュデバイスとを使用する場合がある。そのため、キャッシュがある期間にわたりアクセス不能になり、その後再び接続される場合、キャッシュの最新のスナップショットは、完全に再構築しなければならないのではなく、書込みレコーダーによって追跡される書込み操作を使用して更新することができる。

【００４２】

図３のプロセスの一例３０は、シャットダウン中にオペレーティングシステムによって実行されるオフライン書込みを検出することができるが、シャットダウンが完了した後にディスクに対して実行されるオフライン書込みを検出するために他の手段が必要とされ得る場合があることも理解すべきである。そうした書込みは、例えばユーザーがシャットダウン後に別のオペレーティングシステム内にディスクを起動し、またはシャットダウン後にコンピューターからディスクを取り外し、それを別のコンピューターに接続し、ディスク上に記憶されたデータを修正する場合に生じることができる。

【００４３】

シャットダウン後に発生する（例えば別のオペレーティングシステムによる）オフライン書込みを追跡しようとする試みに関連する困難を認識し、本発明の一部の実施形態は、代わりにそれらのオフライン書込みが発生することを防ごうとする。例えば一部の実施形態は、シャットダウン後に書込みレコーダーコンポーネントを提供しないオペレーティングシステムに対し、特定のディスクボリュームをアクセス不能にしようと試みる。これは数多くの方法のうちのいずれかによって達成することができる。

【００４４】

一部の実施形態では、書込みレコーダー３００は、オフライン書込みを追跡するための書込みレコーダーコンポーネントを提供しないオペレーティングシステムによってディスクボリュームがマウント不能になるような方法でディスクボリュームをマークすることができる。例えば、書込みレコーダー３００は、ボリューム上で使用されるファイルシステムの種類を示すボリューム識別子を修正することができる。この点で、ボリューム識別子は、ボリューム上にデータを記憶するために使用されるファイルシステムの種類をオペレーティングシステムが識別できるようにし、それにより、そのボリューム上に記憶されるデータの構造、ファイルを探す場所等をオペレーティングシステムが理解できるようにす

ることを当業者なら理解されよう。例えば、ボリューム上にデータを記憶するためにＮＴファイルシステム（ＮＴＦＳ）ファイルシステムが使用されていたことをボリューム識別子が示す場合、そのボリュームをマウントしようと試みる別のオペレーティングシステムは、そのボリューム上のデータを解析し、アクセスするためにＮＴＦＳファイルシステムが必要になることを理解する。ボリューム上にデータを記憶するために使用されるファイルシステムの種類をボリューム識別子が示さなかった場合、ほとんどのオペレーティングシステムはそのボリュームをマウントすることに失敗し、そのボリューム上に記憶されるデータの構造を知る確実な方法はなくなる。そのため、本発明の一部の実施形態は、ディスクボリュームのボリューム識別子を修正してそのディスクボリュームをアクセスできなくし、それにより、ユーザーが別のオペレーティングシステム内にそのディスクボリュームを起動し、そのボリューム上に記憶されたデータをオフラインで変更することを防ぐ。

【００４５】

たとえボリューム識別子が修正されていても、一部のオペレーティングシステムは、ボリューム上にデータを記憶するために使用されるファイルシステムの種類を識別できる可能性があることを認識し、本発明の一部の実施形態は、オペレーティングシステムがそのボリュームをマウントするときを検出するための機構を提供する。この点で、ディスクボリュームをマウントするために、どんなオペレーティングシステムもボリューム識別子を（例えばそのボリューム上にデータを記憶するために、ＮＴＦＳファイルシステムが使用されていたことを示すために）更新して、そのディスクボリューム上のデータがアクセスされることを可能にする必要がある。そうした任意の更新は、再起動時に容易に検出可能である。そうした更新が検出された場合、本発明の一部の実施形態は、最後のシャットダウンからそのボリュームのコンテンツが修正されていたとみなし、そのボリューム上に記憶されるデータに対応するキャッシュコンテンツを除去することができる。

【００４６】

本発明の一部の実施形態は、同じく書込みレコーダーコンポーネントを使用する別のオペレーティングシステム内にディスクボリュームを起動することができる能力を提供する。例えば、書込みレコーダーコンポーネントを提供するオペレーティングシステムを実行するあるコンピューターからディスクを取り外し、そのディスクを書込みレコーダーコンポーネントを提供する別のオペレーティングシステム内に起動する場合、そのボリュームをキャッシュできることを、変えられたボリューム識別子が示すことを認識するよう、そのもう一方のオペレーティングシステムを構成することができる。その結果、そのもう一方のオペレーティングシステムは、最初のオペレーティングシステムによって作成された、（例えばそのボリューム上に記憶される）オフライン書込みのログを増やすことができる。

【００４７】

一定のオペレーティングシステムによりディスクボリュームをマウントできなくすることを目的とする上記に記載した実施形態は、特定の機能を実行するためにボリューム識別子に依拠する特定のアプリケーション（例えばバックアップアプリケーション）に問題を引き起こす場合がある。これらのアプリケーションでは、ボリューム識別子が変えられた場合、そのボリュームを認識できなくなり、したがってバックアップされない可能性がある。よって、本発明の一部の実施形態は、シャットダウン後にファイルシステムがマウントされたかどうかを判定するための機構を提供する。ファイルシステムがマウントされた場合、そのファイルシステム内のデータに変更が加えられたとみなされ、そのファイルシステム内のデータに対応するすべてのキャッシュコンテンツを除去することができる。

【００４８】

一部の実施形態は、シャットダウン時にファイルシステムのログを、そのファイルシステムをマウントしようと試みる任意のオペレーティングシステムがそのログを何らかの方法で修正すること（例えばそのログの位置を変更する、新たなエントリーを追加する等）を必要とする状態に置くことにより、シャットダウン後のファイルシステムのマウンティングを検出することができる。例えば、書込みレコーダー３００は、ファイルシステムが

マウント解除されるとき、オフライン書込みをログするタスクの一部として、そのファイルシステムのログの位置および／またはコンテンツを（例えばログ自体の中に）メモ（ｎｏｔｅ）することができる。ファイルシステムをマウントしようと試みる任意のオペレーティングシステムはログを変更しなければならない（例えばファイルシステムがＮＴＦＳファイルシステムであった場合、そのファイルシステムをマウントしようと試みるオペレーティングシステムはログにエントリーを追加することになる）ので、再起動時にログが変更されていない場合、そのファイルシステムは電源切替え中に別のオペレーティングシステムによってマウントされず、そのため、そのファイルシステムに記憶されるデータに対応するキャッシュコンテンツは古くされていないとみなされる。逆に、ログが何らかの方法で変更されている場合（例えばそのログの位置が変更され、エントリーが追加されている等）、そのファイルシステムは別のオペレーティングシステムによってマウントされ、かつそのファイルシステム内に記憶されるデータが変わっており、そのファイルシステムに記憶されるデータに対応するキャッシュコンテンツを古くしているとみなされる。そのため、これらのキャッシュコンテンツは除去することができる。

【００４９】

オフライン書込みを防ぐための機構を提供することに加え、本発明の一部の実施形態は、矛盾した世代のキャッシュコンテンツを管理する能力を提供する。矛盾した世代のキャッシュコンテンツは、数多くの理由のうちのいずれかのために作成される可能性がある。一例は、第１のキャッシュデバイスおよび第２のキャッシュデバイスが接続される、第１のコンピューターおよび第２のコンピューターが、電源切替えにわたりキャッシュコンテンツを保持するために本明細書に記載の技法を使用するときに生じることができる。第１のコンピューターに第２のキャッシュデバイスが接続され（または第２のコンピューターに第１のキャッシュデバイスが接続され）、第１のコンピューターが再起動される場合、Ｉ／Ｏ要求を満たすために誤ったデータが第２のキャッシュデバイスから供給される可能性がある。これは、第１のコンピューターのオペレーティングシステムが、第２のキャッシュデバイスのコンテンツを（キャッシュから返されるデータの再生成された表記が当初生成された表記に一致することができるため）本物であり、（キャッシュコンテンツにオフライン書込みを施すことができるため）古くないとみなすことができるからである。別の例は、第１のキャッシュデバイスがコンピューターに接続され、そのコンピューターがシャットダウンされ（それによりキャッシュコンテンツを保持し）、次いでそのコンピューターが再起動され、第２のキャッシュデバイスが接続され、そのコンピューターが再びシャットダウンされる（それにより再びキャッシュコンテンツを保持する）場合に生じることができる。その後、コンピューターが再び再起動され、第１のキャッシュデバイスが接続される場合、その第１のキャッシュデバイスが最新の世代のキャッシュコンテンツを記憶しないことを判断するための確実な方法がないため、Ｉ／Ｏ要求を満たすために誤ったデータが供給される可能性がある。

【００５０】

一部の実施形態は、最後のシャットダウンより前に保持されたキャッシュコンテンツがＩ／Ｏ要求を満たすために間違って使用されないよう、矛盾した世代のキャッシュコンテンツを識別する能力を提供する。一部の実施形態では、この能力は、（一例として）シャットダウンが開始されるときに数多くの方法のうちのいずれかによって生成することができる、一意の保持識別子によって提供される。例えば、このためにＧＵＩＤおよび／または暗号の乱数ジェネレーターを使用することができる。図３に関して上記に述べたように、この保持識別子はキャッシュデバイス上に（例えばキャッシュメタデータ内にまたはキャッシュメタデータとともに）、ならびにコンピューター上（例えばディスクおよび／またはメモリー上）に記憶し、そのコンピューターが起動されるときに（例えば２つのバージョンを比較することにより）検証することができる。検証に失敗する場合、前に保持したキャッシュの世代に相当するものとしてキャッシュコンテンツを除去することができる。

【００５１】

上記で論じた認証キーのように、保持識別子を生成するために使用されるどんなキーも、電源切替えの期間にわたりキャッシュデバイス以外の位置に書き込むことができる。例えば一部の実施形態では、書込みレコーダーコンポーネントが、キーならびに保持識別子を（例えばシャットダウン時に）ディスク記憶域に書き込むことができる。しかし、本発明はそうした実装形態に限定されず、当業者はキーを保存することができる数多くの代わりの位置を思い描くことができよう。キーは、例えばシステムの起動中に使用可能な、オペレーティングシステムによって提供される任意の構成ストア（例えばＷｉｎｄｏｗｓのレジストリー）内に保つことができる。

ＩＩＩ．キャッシュメタデータ

上記に述べたように、キャッシュメタデータは、データ項目が記憶されるディスクアドレスと、それらのデータ項目がキャッシュされるキャッシュデバイス上の対応するアドレスとの間のマッピングを提供することができる。本発明の一部の実施形態はキャッシュメタデータを記憶するための能力を提供し、この能力は、システムのランタイム動作中にキャッシュメタデータを記憶するために必要なメモリーの量を大幅に減らす。

【００５２】

さらに、一部の実施形態は、コンピューターが再起動されかつ／またはキャッシュデバイスがオンラインにされるときにキャッシュコンテンツに確実にアクセスできるよう、電源切替えまたはキャッシュデバイスをオフラインにする他の任意のイベント（例えばコンピューターからキャッシュデバイスを取り外すこと、ネットワークキャッシュデバイスにアクセスできなくするネットワーク故障等）にわたり、キャッシュメタデータが依拠されることを可能にする技法を提供する。この点で、一定の種類の電源切替え（例えばスタンバイモードおよび休止状態モード）では、スタンバイ切替えおよび休止状態切替えの間にメモリーのコンテンツは保たれるため、キャッシュメタデータを単純にメモリー（すなわちＲＡＭ）に記憶することが許容できることを理解すべきである。しかし、再起動中は、システムメモリーのコンテンツは保たれない。そのため、本発明の一部の実施形態は、シャットダウン中にキャッシュメタデータを何らかの１つまたは複数の不揮発性媒体上に記憶し、再起動時に復元することを可能にする。例えば、キャッシュメタデータは、キャッシュデバイス上に、および／または１つもしくは複数の別個の不揮発性媒体上に記憶することができる。さらに、一部の実施形態は、キャッシュメタデータのある一部を他のキャッシュメタデータから導き出すことができ、そのためすべてのキャッシュメタデータを記憶することは不要な場合がある。

【００５３】

一部の実施形態は、キャッシュデバイスがオフラインになるとき（例えばコンピューターのシャットダウン中、コンピューターからキャッシュデバイスを取り外すこと、ネットワークキャッシュデバイスにアクセスできなくするネットワーク故障等）のメタデータに対する不注意のまたは悪意のある修正を検出し防ぐよう、キャッシュメタデータの信頼性を検証するために、上記のセクションＩに記載した技法を使用することができる。例えば、キャッシュデバイスがオンラインになるとき、キャッシュマネージャーは、メモリーにメタデータがロードされるとき、図２Ａ～図２Ｂに関して上記に記載した技法を使用してそのメタデータの信頼性を検証することができる。キャッシュメタデータの信頼性を検証できない場合、対応するキャッシュコンテンツはディスク上に記憶されるデータに基づいて更新し、除去し、または上記に記載したように他の方法で処理することができる。

【００５４】

一部の実施形態では、シャットダウン中に保存し再起動時にロードするメタデータの量を減らすために、キャッシュメタデータを圧縮することができる。メタデータを圧縮することは、そのメタデータに関する情報を含む別個の情報（例えばキャッシュ内のヘッダー）を保存することを必要とする場合があるので、再起動時にこの情報の信頼性も検証するために上記に記載した技法を使用することができる。

【００５５】

本発明の一部の実施形態は、メモリーに記憶されるキャッシュメタデータの量を常に大

幅に減らし、それにより（例えばランタイム動作中および起動／シャットダウン動作中に）メモリーにキャッシュメタデータをロードし、メモリーからキャッシュメタデータをオフロードするのに必要な時間を減らし、キャッシュメタデータのメモリー「フットプリント」を大幅に減らす方法でキャッシュメタデータを記憶するための技法を提供する。この点で、キャッシュデバイスが比較的大きな記憶容量を有する状態では、キャッシュコンテンツを管理するためにかなりの量のメタデータが必要な場合があることを理解すべきである。例えば、１６ギガバイトの記憶容量を有するキャッシュデバイスは、最高３２ギガバイトの圧縮データを記憶する能力があり得る。一部の実装形態では、４キロバイトのディスク記憶域に相当する「データ単位」で、ディスクアドレスをキャッシュメタデータ内に反映させることができる。そのため、３２ギガバイトのデータの位置を追跡するために、８００万個の互いに異なるデータ単位が必要である。その８００万個のデータ単位のそれぞれが、１６バイトマッピングを使用してキャッシュメタデータ内に表される場合（すなわちディスクアドレスからキャッシュアドレスまで）、これらのマッピングは１２８メガバイトの記憶域を必要とする。出願者らは、１２８メガバイトのキャッシュメタデータをメモリーに記憶することは、多くのコンピューターにおいて不必要に多くのメモリーの部分を占有することが分かった。さらに、シャットダウン中に１２８メガバイトのキャッシュメタデータをメモリーから不揮発性媒体に書き込み、再起動時に１２８メガバイトのキャッシュメタデータを不揮発性媒体からメモリーに復元するのに必要な時間はひどく時間がかかり、過度の量の処理リソースを消費する。

【００５６】

キャッシュメタデータの量は容易に減らすことはできないことを認識し、本発明の一部の実施形態は、キャッシュメタデータを記憶するために必要な記憶域リソース、ならびにシャットダウン時および起動時にキャッシュメタデータを保存／復元するために必要な時間および処理リソースを減らすことを目的とした技法を提供する。

【００５７】

一部の実施形態では、これは、キャッシュメタデータを１つまたは複数の階層型データ構造（例えばツリー、マルチレベルアレイ等）に記憶することによって達成される。階層型データ構造を使用することは、その階層のより低いレベルを不揮発性媒体（例えばキャッシュデバイス）上に記憶することを可能にできる一方、メモリーにはその階層のより高いレベルだけが記憶される。例えば一部の実施形態では、かなりの記憶容量を有するキャッシュデバイスをサポートするために必要な量のキャッシュメタデータを全体として記憶しながらでさえ、キャッシュメタデータ内でメモリーによって占有される「フットプリント」を大幅に減らすことができるよう、階層のより高いレベルだけがメモリーに記憶される。当然、キャッシュメタデータにおける階層のより高いレベルだけを記憶し、一部の実施形態は、階層のより低いレベルに保たれる何らかの情報に繰り返しアクセスすることに関連するＩ／Ｏオーバーヘッドを減らすよう、そのより低いレベルに保たれる情報もメモリーに記憶することを可能にすることができる。本発明は、どんな特定の様式で実装されることにも限定されない。

【００５８】

システムの動作中、読取り要求が処理されるとき、その読取り操作を実行するために不揮発性媒体から（すなわち階層のより低いレベルから）読み取られるキャッシュメタデータは、メモリーに「ページイン」する（すなわち記憶域媒体からメモリーに読み取る）ことができ、そのため、同じディスクアドレス／キャッシュアドレスに対するその後の読取り要求に関し、そのキャッシュメタデータにより速くアクセスすることができる。コンピューターが後にシャットダウンされ、および／またはキャッシュデバイスがオフラインにされるとき、階層のより高いレベルに記憶されるキャッシュメタデータ、およびメモリーにページインされた、階層のより低いレベルに記憶されるキャッシュメタデータのみを不揮発性媒体に保存する必要があり得る。そのため、シャットダウン時にキャッシュメタデータをメモリーから不揮発性記憶域に移動させ、再起動時にキャッシュメタデータを不揮発性記憶域からメモリーに復元するのに必要な時間を大幅に短縮することができる。

【００５９】
本発明の一部の実施形態は、キャッシュメタデータの少なくとも一部分を記憶するためにＢ＋ツリーを使用する。当業者なら分かるように、Ｂ＋ツリーは大きな分岐数（ｂｒａｎｃｈｉｎｇ　ｆａｃｔｏｒ）を使用し、その結果、使用される階層のレベルの数を減らすことができる。上記に挙げた例を使用し、キャッシュメタデータ内に８００万個のデータ単位が表され、分岐数が２００のＢ＋ツリーが使用される（そのため階層内の各ノードは２００個の「子」ノードを有する）場合、そのメタデータを記憶するには３つのレベルしか有さないデータ構造で十分であり、その３つのレベルとはつまり、最高レベルにある単一の「ルート」ノード、２番目のレベルにある２００個のノード、および３番目のレベルにある４万個のノードであり、その４万個のノードのそれぞれが２００個のデータ単位（または合計８００万個のデータ単位）へのポインターを含む。

【００６０】
図４は、ルートノード４００、レベル２ノード$４１０_{1\sim200}$、およびレベル３ノード$４２０_{1\sim200}$を含むこのＢ＋ツリーの例を示す。各ノードは、階層内のより低いレベルにあるノードへのポインターによりそれぞれが分けられる２００個のエレメントを含む。例えば、ルートノード４００の中のエレメント４０２は、ポインター４０１および４０３によって区切られる。ノードの中のエレメントの左側または右側のポインターをたどることにより、所与のキー（例えばディスクアドレス）に関連する値（例えばキャッシュアドレス）を求めることができ、キーがエレメントよりも少ない場合はそのエレメントの左側のポインターがたどられ、キーがエレメントよりも大きい場合は右側のポインターがたどられる。例えば、エレメント４０２よりも少ないキーの値を求めるために、ポインター４０１をレベル２ノード４１０_{1}までたどり、エレメント４０２より大きいがエレメント４０４よりも少ないキーの値を求めるために、ポインター４０３をレベル２ノード４１０_{2}（不図示）までたどり、その後も同じように続く。同様に、レベル２ノードにおいて、（そのキーがノードの中のエレメントより少ないか大きいかに応じて）エレメントの左側または右側のポインターをレベル３ノードまでたどる。レベル３において、（この場合もやはり、そのキーがノードの中のエレメントより少ないか大きいかに基づいて）最後のポインターがその値までたどられ、レベル３における各ポインターは、キャッシュメタデータ内の８００万個のデータ単位のうちの１つを参照する。

【００６１】
大きい分岐数を有するＢ＋ツリーは、比較的「平らな」階層を提供し、ほぼすべてのノードがその階層の底辺レベルに位置することを理解すべきである。つまり、このツリーの中の計４０，２０１個のノードのうち、４０，０００個が最も低いレベルにある。本発明の一部の実施形態は、起動時にこの階層の上位２レベルだけをメモリーに復元する一方、この階層の最も低いレベルのキャッシュメタデータを必要になるまでキャッシュデバイス上に記憶しておくことにより、このことをうまく利用する（例えば最も低いレベルのキャッシュメタデータは、読取り要求が処理されるときにオンデマンドでメモリーにロードすることができ、遅延ロード等をすることができる）。階層型データ構造の一部分しかメモリーに記憶されないので、このキャッシュメタデータは、キャッシュメタデータの全部またはそのより大きい部分をメモリーに保持する場合にさもなければ必要になるよりも、はるかに小さいメモリーの部分を占有することができる。さらに、コンピューターがシャットダウンされるとき、上位２レベルのデータおよび動作中にメモリーにロードされるデータしかキャッシュデバイス上に記憶する必要はない。その結果、起動動作およびシャットダウン動作のどちらも、速く効率的に行うことができる。

【００６２】
したがって本発明の一部の実施形態は、階層のあるレベル（上記の例ではレベル３）における、キャッシュデバイス上に記憶されるノードを参照する、この階層の別のレベル（上記の例ではこの階層のレベル２）における、メモリーに記憶されるノードの中のポインターを提供する。例えば、キャッシュ済みのデータ項目を得るための読取り要求が受け取られるとき、本発明の諸実施形態は、メモリーに記憶される階層の１つまたは複数のレベ

ルにわたりポインターをたどり、キャッシュに記憶されるその階層のより低いレベルにあるメタデータまでたどり、キャッシュの中でそのデータ項目が記憶されるアドレスを求める。一部の実施形態では、データ項目のキャッシュアドレスが求められると、その項目を読み取るためのその後の要求をキャッシュデバイスからキャッシュメタデータを読み取る必要なく実行できるよう、そのキャッシュアドレスをメモリーに記憶することができる。

【００６３】

図５は、本発明の一部の実施形態による、キャッシュメタデータを管理するためのシステムの一例５０を示す。図５は、その両方がコンピューター（不図示）にとってアクセス可能な、メモリー５００およびキャッシュデバイス１１０を示す。コンピューターが起動されると、動作５０５で、Ｂ＋ツリーなどの階層型データ構造の１つまたは複数のレベルを含むキャッシュメタデータが、メモリー５００にロードされる。説明するために上記の例を使用すると、キャッシュメタデータ内に８００万個のデータ単位が表され、そのためそのキャッシュメタデータを記憶するために３レベルの階層型データ構造を使用することができる場合、階層の上位２レベルをメモリー５００にロードすることができる。当然、メタデータ内に８００万個を上回るまたは下回るデータ単位が表され、３を上回るまたは下回るレベルを有する階層型データ構造が使用される場合、その階層について異なる数のレベルをメモリー５００にロードすることができる。

【００６４】

その後、読取り要求がキャッシュの中に保持されるデータ項目を対象にする場合、キャッシュデバイス１１０の中に記憶される階層の１つまたは複数のレベルの中に記憶されるキャッシュメタデータにアクセスすることにより、そのデータ項目が記憶されるキャッシュアドレスが求められる。次いで、このデータ項目が記憶されるキャッシュアドレスを求めるためにキャッシュデバイス上に記憶されるキャッシュメタデータを読み取る必要なく、このデータ項目についてのその後の読取りまたは書込みを実行できるよう、このキャッシュメタデータがメモリー５１０に記憶される。代わりに、キャッシュアドレスをメモリーから読み取ることができ、その読取りは、キャッシュに対する読取りよりも速く行うことができる。

【００６５】

後にコンピューターがシャットダウンされるとき、行為５１５で、メモリーに記憶されるキャッシュメタデータ（すなわち操作５０５でメモリーにロードされる、階層のレベルに記憶されるメタデータおよび操作５１０でメモリーに書き込まれる、読取り要求を満たすために使用される任意のメタデータ）がキャッシュデバイス５００にロードされる。比較的少ない量のキャッシュメタデータがメモリーに記憶される結果、かなりの処理リソースを必要とせずにシャットダウンを速く実行することができる。

【００６６】

Ｂ＋ツリーは、キャッシュメタデータを記憶するために使用することができる数多くの種類のデータ構造のうちの１つに過ぎず、他の種類のデータ構造（例えばＡＶＬツリー、赤黒ツリー、バイナリー検索ツリー、Ｂツリーなどの階層型構造ならびに／または他の階層型および非階層型データ構造）を使用することができることを理解すべきである。本発明は、キャッシュデータを記憶するためにどんな１つのデータ構造またはデータ構造の組合せを使用することにも限定されない。

【００６７】

一部の実施形態は、任意の時点においてメモリーに保たれる、キャッシュメタデータの「目標量」を定めることができる。この目標量は、任意の適切な様式で決定することができる。例えば目標量は、コンピューターにとって利用可能な物理メモリーの量の割合とすることができる。例えば、コンピューターが１ギガバイトのメモリーを有する場合、任意の時点において、（一例として）２メガバイトのキャッシュメタデータをメモリーに記憶することができる。したがって、そのコンピューターがシャットダウンされるとき、２メガバイトのキャッシュメタデータしかキャッシュデバイスにロードする必要がない。

【００６８】

一部の実施形態では、キャッシュメタデータをメモリーの内外に循環させることができる。例えば、キャッシュメタデータの目標量が既にメモリーに記憶されており、キャッシュメタデータがキャッシュデバイスから読み取られることを必要とする読取りが実行される場合、そのメタデータをメモリーに「ページイン」することができ、他の（例えば最も長い間アクセスされていない）キャッシュメタデータを消去することができる。例えば、キャッシュメタデータは、キャッシュデバイスに書き込まれた後に消去することができる。あるいは、このシステムは、キャッシュメタデータが最後に書き出されてからそのキャッシュメタデータが変わっているかどうかを判定することができ、変わっていない場合、そのキャッシュメタデータは単純に消去することができ、こうしてそのキャッシュメタデータを書き込むためにさもなければ必要な時間および処理リソースをなくす。上記に記載した技法を使用して、メモリー内でキャッシュメタデータによって占有される小さい「フットプリント」を維持することができる。

【００６９】

図６は一例を示す。具体的には、図６に示すプロセス６０は、上記に記載した技法を使用してキャッシュメタデータを読み取るために、キャッシュマネージャー１００によって実行することができる操作を含む。

【００７０】

プロセス６００の開始時に、行為６０５で、ディスクアドレスＸに記憶されるデータを読み取る要求が受け取られる。行為６１０で、メモリーに記憶されるキャッシュメタデータから、そのデータが記憶されるキャッシュアドレスを識別できるかどうかの判定が行われる。識別できる場合、このプロセスは行為６１５に進み、行為６１５では、識別されたキャッシュアドレスが求められ、行為６２０でキャッシュデバイス１１０に対して読取り要求を発行するために使用される。これでプロセス６０は完了する。メモリーに記憶されるキャッシュメタデータを使用してキャッシュアドレスを識別できない場合、このプロセスは行為６２５に進み、行為６２５では、そのデータが記憶されるキャッシュアドレスを求めるために、キャッシュデバイス１１０からキャッシュメタデータが読み取られる。この行為６２５で識別されるキャッシュオフセットを使用して、行為６２０で、識別されたキャッシュオフセットに対して読取り要求が発行され、これでプロセス６０は完了する。

【００７１】

キャッシュデバイス上にキャッシュメタデータを記憶することは、起動中およびシャットダウン中にキャッシュメタデータをロード／復元するプロセスを速めることができるだけでなく、起動中およびシャットダウン中に実行されるシステム動作も速めることができることを理解すべきである。この点で、シャットダウンおよび起動は一定のデータ項目への複数のアクセスを多くの場合伴い、キャッシュデバイスに対して２つの読取り操作を実行することは、典型的にはディスク記憶域に対して１つの読取り操作を実行するより速い。その結果、シャットダウン中および／または起動中にアクセスされるデータ項目と、そのデータ項目の位置を指定するメタデータとがどちらもキャッシュに記憶される場合、キャッシュに対する２つの読取り（すなわち、その項目の位置を求めるためのキャッシュメタデータにアクセスするために１つ、およびその項目自体にアクセスするためにもう１つ）は、典型的にはディスクに対する１つの読取りよりも速く実行することができるため、そのデータ項目は、そのデータ項目がディスク上に記憶される場合よりも速くアクセスすることができる。そのため、シャットダウン中および起動中に実行される個々の操作を促進することができる。さらに、キャッシュからキャッシュメタデータを最初に読み取る間に、その項目が記憶されたアドレスがメモリーにページされる場合、メモリーに対する読取りは、典型的にはキャッシュに対する読取りよりも速く実行することができるため、そのデータ項目のその後の読取りをさらに速く実行することができる。

【００７２】

本発明の特徴を実施するためのシステムおよび方法の様々な態様は、図７に示す例示的コンピューターシステム７００などの１つまたは複数のコンピューターシステム上に実装することができる。コンピューターシステム７００は、１つまたは複数の入力デバイス７

０２、１つまたは複数の出力デバイス７０１、プロセッサー７０３、メモリーシステム７０４、および記憶域７０６を含み、そのすべてが、１つまたは複数のバス、スイッチ、ネットワークおよび／または他の任意の適切な相互接続を含み得る、相互接続機構７０５を介して直接または間接的に結合される。１つまたは複数の入力デバイス７０２は、ユーザーもしくはマシン（例えば人間のオペレーター）から入力を受け取り、１つまたは複数の出力デバイス７０１は、ユーザーもしくはマシン（例えば液晶ディスプレイ）に情報を表示しまたは伝送する。プロセッサー７０３は、典型的には、他のコンピュータープログラムの実行を制御するオペレーティングシステムと呼ばれるコンピュータープログラム（例えばＭｉｃｒｏｓｏｆｔのＷｉｎｄｏｗｓファミリーのオペレーティングシステムや他の任意の適切なオペレーティングシステム）を実行し、スケジューリング、入出力や他のデバイス制御、アカウンティング、コンパイル、記憶域の割当、データ管理、メモリー管理、通信およびデータフロー制御を行う。集合的に、プロセッサーおよびオペレーティングシステムは、アプリケーションプログラムおよび他のコンピュータープログラム言語が書かれる対象のコンピュータープラットフォームを定義する。

【００７３】

プロセッサー７０３は、様々な機能を実施するために１つまたは複数のコンピュータープログラムを実行することもできる。これらのコンピュータープログラムは、手続き型プログラミング言語、オブジェクト指向プログラミング言語、マクロ言語、またはそれらの組合せを含む、任意の種類のコンピュータープログラム言語で書くことができる。これらのコンピュータープログラムは、記憶域システム７０６に記憶することができる。記憶域システム７０６は、揮発性媒体または不揮発性媒体上に情報を保持することができ、固定型またはリムーバブルとすることができる。図８に、記憶域システム７０６をさらに詳細に示す。

【００７４】

記憶域システム７０６は、典型的には、コンピュータープログラムまたはそのプログラムによって使用される情報を定義する信号がその上に記憶される、コンピューター可読／書込み可能不揮発性記録媒体８０１を含む。媒体は、例えばディスクやフラッシュメモリーとすることができる。典型的には動作、プロセッサー７０３は、不揮発性記録媒体８０１から、この媒体８０１よりも、プロセッサー７０３による情報へのより速いアクセスを可能にする揮発性メモリー８０２（例えばランダムアクセスメモリーすなわちＲＡＭ）にデータを読み取らせる。このメモリー８０２は、図８に示すように記憶域システム７０６の中に位置することができ、または図７に示すようにメモリーシステム７０４の中に位置することができる。プロセッサー７０３は、概して集積回路メモリー７０４、８０２内のデータを操作し、処理を完了した後にそのデータを媒体８０１にコピーする。媒体８０１と集積回路メモリー要素７０４、８０２との間のデータの移動を管理するための様々な機構が知られており、本発明はそれらに限定されない。本発明は、特定のメモリーシステム７０４または記憶域システム７０６にも限定されない。

【００７５】

さらに、本発明の実施形態は、オペレーティングシステムのＩ／Ｏスタック内のドライバーとして実装される、キャッシュマネージャーコンポーネントを使用することにも限定されない。そのそれぞれがオペレーティングシステムまたは１つもしくは複数のスタンドアロンコンポーネントによって実装され得る、任意の適切なコンポーネントまたはコンポーネントの組合せを、代わりにまたは追加的に使用することができる。本発明はどんな特定の実装形態にも限定されない。

【００７６】

上記に記載した本発明の実施形態は、数多くの方法のうちのいずれかによって実施することができる。例えば、上記で論じた機能性は、ハードウェア、ソフトウェア、またはその組合せを使用して実装することができる。ソフトウェアによって実装されるとき、そのソフトウェアコードは、単一のコンピューターの中に設けられるにせよ、複数のコンピューターの間で分散されるにせよ任意の適切なプロセッサーまたはプロセッサーの集まりの

上で実行することができる。この点で、本明細書に記載した機能を実行する任意のコンポーネントまたはコンポーネントの集まりは、上記で論じた機能を制御する１つまたは複数のコントローラーと総称的にみなすことができることを理解すべきである。この１つまたは複数のコントローラーは、専用ハードウェアを用いてや、上記に列挙した機能を実行するためのマイクロコードまたはソフトウェアを使用してプログラムされる１個もしくは複数個のプロセッサーを使用することによってなど、数多くの方法で実装することができる。コントローラーが、システム動作のためのデータを記憶しまたは提供する場合、そうしたデータは中央リポジトリ内、複数のリポジトリ内、またはその組合せに記憶することができる。

【００７７】

さらに、（クライアントまたはサーバー）コンピューターは、ラックマウント式のコンピューター、デスクトップコンピューター、ラップトップコンピューター、タブレットコンピューター、他の種類のコンピューターなど、いくつかの形態のうちのいずれかによって実施できることを理解すべきである。さらに、（クライアントまたはサーバー）コンピューターは、ＰＤＡ（携帯情報端末）、スマートフォン、または他の任意の適切なポータブル／固定電子デバイスを含む、概してコンピューターとみなされていないが適切な処理能力を備えるデバイスに埋め込むことができる。

【００７８】

また、（クライアントまたはサーバー）コンピューターは、１つまたは複数の入出力デバイスを有することができる。これらのデバイスは、とりわけユーザーインターフェイスを提示するために使用することができる。ユーザーインターフェイスを提供するために使用することができる出力デバイスの例には、出力を視覚的に提示するためのプリンターやディスプレイ画面、および出力を可聴的に提供するためのスピーカーや他の音声を発生させるデバイスが含まれる。ユーザーインターフェイスに使用することができる入力デバイスの例には、キーボードや、マウス、タッチパッド、デジタイジングテーブルなどのポインティングデバイスが含まれる。別の例として、コンピューターは、音声認識によりまたは他の可聴形式で入力情報を受け取ることができる。

【００７９】

そうしたコンピューターは、エンタープライズネットワークやインターネットなど、ローカルエリアネットワークまたはワイドエリアネットワークを含む、任意の適切な形態の１つまたは複数のネットワークによって相互接続することができる。そうしたネットワークは任意の適切な技術に基づくことができ、任意の適切なプロトコルに従って動作することができ、ワイヤレスネットワーク、ワイヤードネットワーク、または光ファイバーネットワークを含むことができる。また、本明細書で概説した様々な方法またはプロセスは、多岐にわたるオペレーティングシステムまたはプラットフォームのいずれか１つを使用する、１個または複数個のプロセッサー上で実行可能なソフトウェアとしてコード化することができる。

【００８０】

さらに、ソフトウェアはいくつかの適切なプログラミング言語および／または従来のプログラミングツールもしくはスクリプトツールのいずれかを使用して書くことができ、実行可能な機械言語コードまたはフレームワークもしくは仮想マシン上で実行される中間コードとしてコンパイルすることもできる。

【００８１】

この点で、本発明は、１つまたは複数のコンピューターもしくは他のプロセッサー上で実行されるとき、上記で論じた本発明の様々な実施形態を実施する方法を実行する、１つまたは複数のプログラムを用いてエンコードされる記憶域媒体（または複数の記憶域媒体）（例えば、コンピューターメモリー、１つまたは複数のフロッピーディスク、コンパクトディスク、光ディスク、磁気テープ、フラッシュメモリー、書替え可能ゲートアレイや他の半導体デバイス内の回路構成、または他のコンピューター記憶域媒体）として実施することができる。この１つまたは複数の記憶域媒体は可搬式とすることができ、そのため

その上に記憶される１つまたは複数のプログラムを、１台または複数台の別のコンピューターもしくは他のプロセッサー上にロードして、上記で論じた本発明の様々な態様を実施することができる。

【００８２】

用語「プログラム」または「ソフトウェア」は、上記で論じた本発明の様々な態様を実施するようコンピューターまたは他のプロセッサーをプログラムするために使用することができる、任意の種類のコンピューターコードまたは１組のコンピューター実行可能命令を指すための包括的な意味で本明細書において使用する。さらに、この実施形態の１つの側面によれば、実行されるときに本発明の方法を実行する１つまたは複数のコンピュータープログラムは、単一のコンピューターもしくはプロセッサー上にある必要はなく、本発明の様々な態様を実施するためにいくつもの異なるコンピューターまたはプロセッサーの間でモジュール方式で分散させることができることを理解すべきである。

【００８３】

コンピューター実行可能命令は、１台または複数台のコンピューターもしくは他のデバイスによって実行されるプログラムモジュールなど、多くの形式で提供することができる。概して、プログラムモジュールには、特定のタスクを実行しまたは特定の抽象データ型を実装するルーチン、プログラム、オブジェクト、コンポーネント、データ構造等が含まれる。典型的には、プログラムモジュールの機能は、様々な実施形態で望まれるようにまとめ、または分散させることができる。

【００８４】

本発明の様々な態様は、単独で、組合せで、または上記に記載した諸実施形態では特に論じていない様々な構成で使用することができ、したがってその適用において、上記の説明に記載しまたは図面に示した詳細およびコンポーネントの構成に限定されることはない。例えば、ある実施形態で記載した態様を、別の実施形態で記載した態様と任意の方法で組み合わせることができる。

【００８５】

請求項の要素を修飾するために、特許請求の範囲の中で「第１の」、「第２の」、「第３の」などの序数を表す用語を使用することは、別の請求項の要素に優先する１つの請求項の要素のどんな優先順位、順位、もしくは順序、または方法の行為が実行される時間的順序もそれ自体では意味しないが、請求項の各要素を区別するために、特定の名前を有する１つの請求項の要素と同じ名前を有する（が序数を表す用語を使用するための）別の要素とを区別するためのラベルとして単に使用する。

【００８６】

また、本明細書で使用した表記および用語は説明を目的としており、限定的であるとみなすべきでない。「含む（ｉｎｃｌｕｄｉｎｇ）」、「備える（ｃｏｍｐｒｉｓｉｎｇ）」、または「有する（ｈａｖｉｎｇ）」、「含有する（ｃｏｎｔａｉｎｉｎｇ）」、「伴う（ｉｎｖｏｌｖｉｎｇ）」、およびその改変形態を本明細書で使用することは、その後に列挙される項目およびその等価物ならびに追加項目を包含することを意図する。

【００８７】

こうして本発明の少なくとも１つの実施形態のいくつかの側面を説明し終え、当業者は様々な変更形態、修正形態、および改善形態を容易に思いつくことが理解される。そうした変更形態、修正形態、および改善形態は本開示の一部であることが意図され、本発明の趣旨および範囲に含まれることが意図される。したがって上記の説明および図面は例に過ぎない。

【図１】

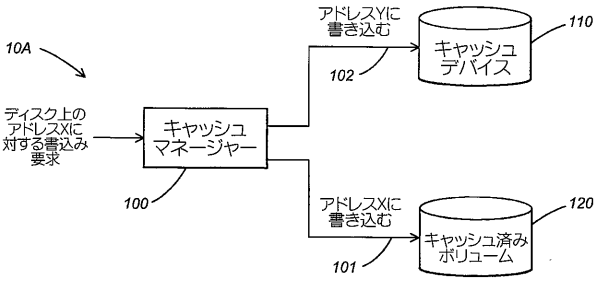


FIG. 1A

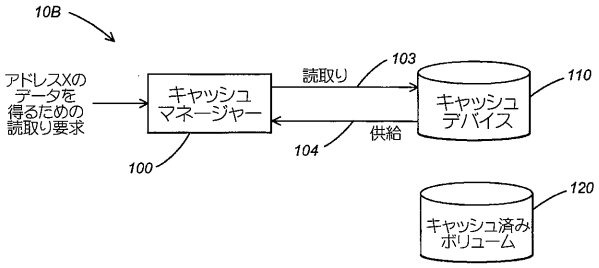
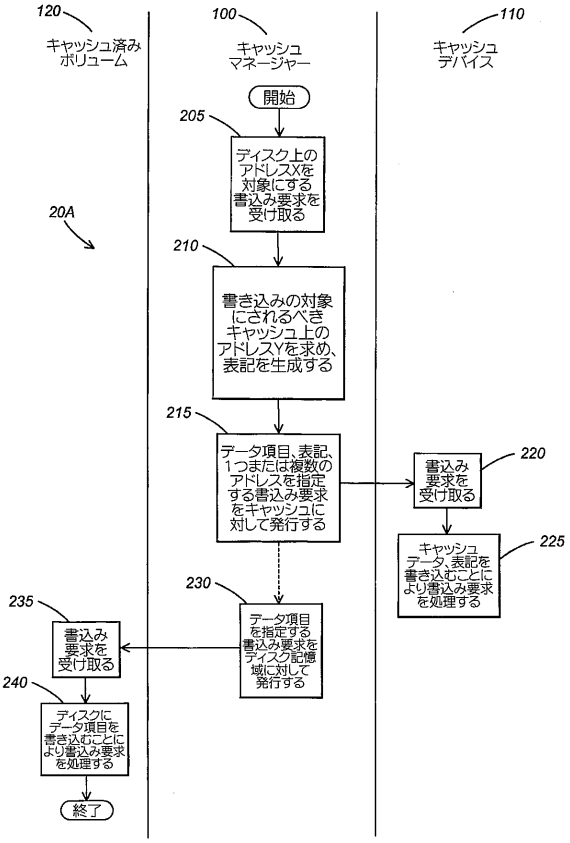
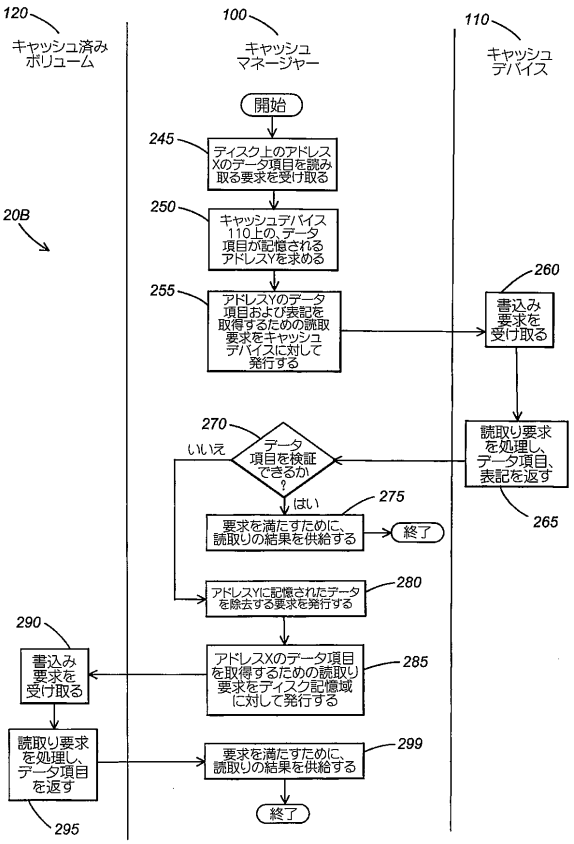


FIG. 1B

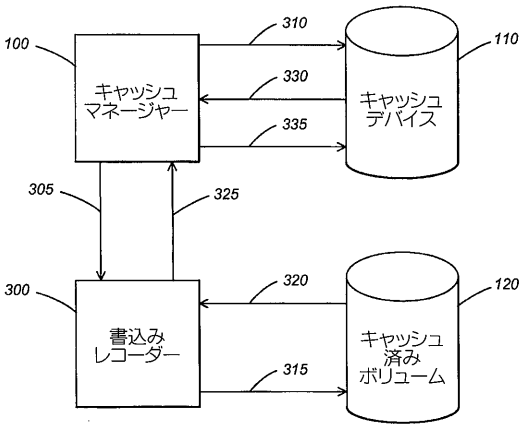
【図２Ａ】



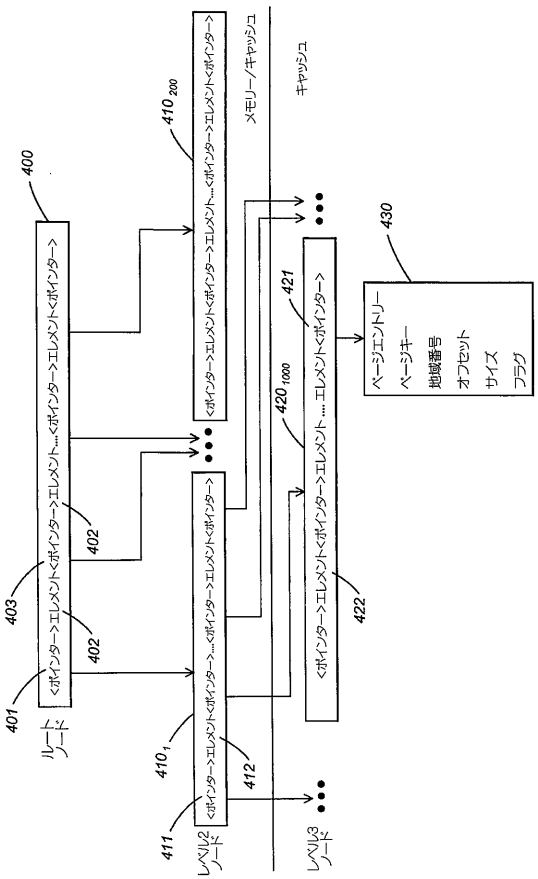
【図２Ｂ】



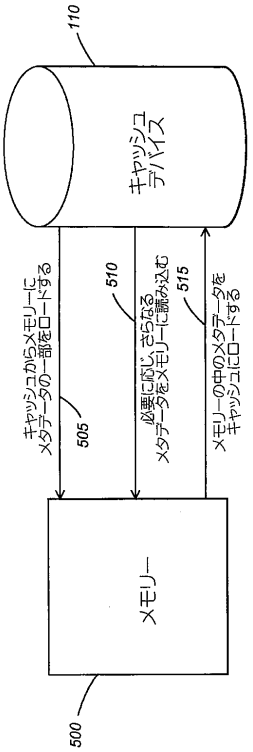
【図３】



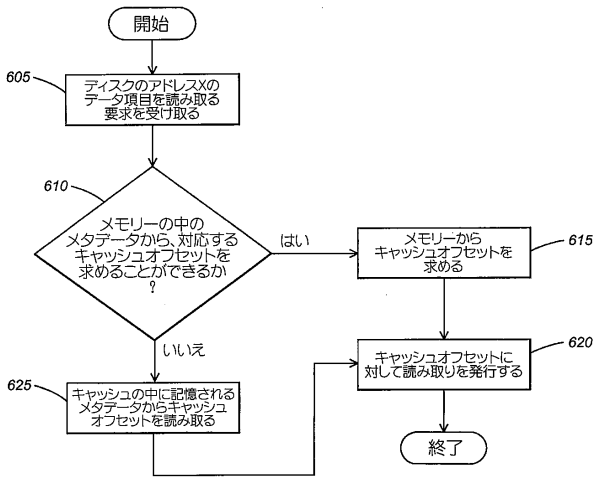
【図４】


400
401
ルートノード
<ポインター->エレメント<ポインター->エレメント....<ポインター->エレメント<ポインター>
402
403
410 200
<ポインター->エレメント<ポインター->エレメント....<ポインター->エレメント<ポインター>
メモリー/キャッシュ
キャッシュ
410 1
<ポインター->エレメント<ポインター->....<ポインター->エレメント<ポインター>
411
レベル2ノード
412
420 1000
<ポインター->エレメント<ポインター->エレメント.... エレメント<ポインター>
421
422
430
ページエントリー
ページキー
地域番号
オフセット
サイズ
フラグ
レベル3ノード

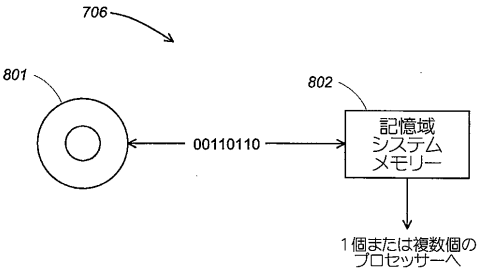
【図５】


110
キャッシュデバイス
キャッシュからメモリーにメタデータの一部をロードする
505
510
必要に応じ、さらなるメタデータをメモリーに読み込む
515
メモリーの中のメタデータをキャッシュにロードする
メモリー
500

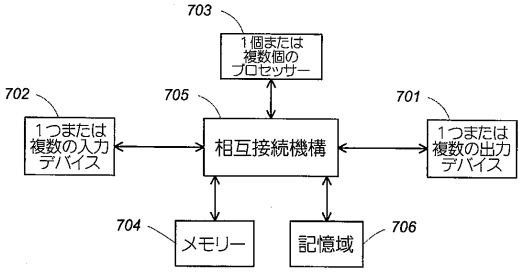
【図６】


開始
605
ディスクのアドレスXのデータ項目を読み取る要求を受け取る
610
メモリーの中のメタデータから、対応するキャッシュオフセットを求めることができるか？
はい
615
メモリーからキャッシュオフセットを求める
いいえ
620
キャッシュオフセットに対して読み取りを発行する
625
キャッシュの中に記憶されるメタデータからキャッシュオフセットを読み取る
終了

【図８】


706
801
802
00110110
記憶域システムメモリー
1個または複数個のプロセッサーへ

【図７】


703
1個または複数個のプロセッサー
702
1つまたは複数の入力デバイス
705
相互接続機構
701
1つまたは複数の出力デバイス
704
メモリー
記憶域
706

フロントページの続き

(72)発明者 イーイガン, メーメト
アメリカ合衆国ワシントン州９８０５２－６３９９, レッドモンド, ワン・マイクロソフト・ウェイ, マイクロソフト コーポレーション, エルシーエイ－インターナショナル・パテンツ
(72)発明者 バク, イェヴゲニイ
アメリカ合衆国ワシントン州９８０５２－６３９９, レッドモンド, ワン・マイクロソフト・ウェイ, マイクロソフト コーポレーション, エルシーエイ－インターナショナル・パテンツ
(72)発明者 フォーティン, マイケル
アメリカ合衆国ワシントン州９８０５２－６３９９, レッドモンド, ワン・マイクロソフト・ウェイ, マイクロソフト コーポレーション, エルシーエイ－インターナショナル・パテンツ
(72)発明者 フィールズ, デーヴィド
アメリカ合衆国ワシントン州９８０５２－６３９９, レッドモンド, ワン・マイクロソフト・ウェイ, マイクロソフト コーポレーション, エルシーエイ－インターナショナル・パテンツ
(72)発明者 アーガン, センク
アメリカ合衆国ワシントン州９８０５２－６３９９, レッドモンド, ワン・マイクロソフト・ウェイ, マイクロソフト コーポレーション, エルシーエイ－インターナショナル・パテンツ
(72)発明者 カーシェンバウム, アレクサンダー
アメリカ合衆国ワシントン州９８０５２－６３９９, レッドモンド, ワン・マイクロソフト・ウェイ, マイクロソフト コーポレーション, エルシーエイ－インターナショナル・パテンツ

審査官 滝谷 亮一

(56)参考文献 特開２００３－０８５０４１（ＪＰ, Ａ）
米国特許出願公開第２００７／０１５６９５４（ＵＳ, Ａ１）
特開２００５－２２２３８３（ＪＰ, Ａ）
特表２００１－５００２９３（ＪＰ, Ａ）
特開２００８－０５２３２１（ＪＰ, Ａ）
三華 和夫, こんな“ほら話”真に受けてない？ Ｖｉｓｔａ8つの都市伝説, Ｗｉｎｄｏｗｓ Ｓｅｒｖｅｒ Ｗｏｒｌｄ, 日本, （株）ＩＤＧジャパン, ２００７年 7月 1日, Ｖｏｌ．１２ Ｎｏ．７, Ｐ．４２－４３

(58)調査した分野(Int.Cl., ＤＢ名)
Ｇ０６Ｆ １２／０８
Ｇ０６Ｆ １２／００