

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6237230号
(P6237230)

(45) 発行日 平成29年11月29日(2017.11.29)

(24) 登録日 平成29年11月10日(2017.11.10)

(51) Int.Cl.

G06F 11/07 (2006.01)

F 1

G06F 11/07 178
G06F 11/07 140C

請求項の数 7 (全 20 頁)

(21) 出願番号 特願2013-273051 (P2013-273051)
 (22) 出願日 平成25年12月27日 (2013.12.27)
 (65) 公開番号 特開2015-127879 (P2015-127879A)
 (43) 公開日 平成27年7月9日 (2015.7.9)
 審査請求日 平成28年9月5日 (2016.9.5)

(73) 特許権者 000005223
 富士通株式会社
 神奈川県川崎市中原区上小田中4丁目1番
 1号
 (74) 代理人 100107766
 弁理士 伊東 忠重
 (74) 代理人 100070150
 弁理士 伊東 忠彦
 (74) 代理人 100146776
 弁理士 山口 昭則
 (72) 発明者 治部 将之
 神奈川県川崎市中原区上小田中4丁目1番
 1号 富士通株式会社内

最終頁に続く

(54) 【発明の名称】メモリ管理プログラム、メモリ管理方法、及びメモリ管理装置

(57) 【特許請求の範囲】

【請求項 1】

メモリへの書き込みを検知し、
 検知された前記書き込みの書き込み先のデータ領域に記憶された書き込み前のデータと、前記メモリへの書き込みを検知したときのプロセッサのコンテキスト情報を対応付けて所定の記憶領域に保存し、

コンテキストスイッチにより前記コンテキスト情報を保存がされていない場合、前記メモリにおける前記コンテキスト情報を保存する領域に変更を加える、
処理をコンピュータに実行させるメモリ管理プログラム。

【請求項 2】

前記所定の記憶領域は、前記コンテキスト情報を保存するときに変更され、
 前記書き込み前のデータは、変更された前記所定の記憶領域に対応付けられて順次保存される、請求項 1 に記載のメモリ管理プログラム。

【請求項 3】

前記コンテキスト情報は、コンテキストスイッチによる前記コンテキスト情報を保存がされていない場合には前記メモリのコンテキスト退避領域に明示的な変更を加えることにより前記メモリへの書き込みが検知されて保存される、請求項 1 又は 2 に記載のメモリ管理プログラム。

【請求項 4】

前記所定の記憶領域の変更は、予め設定可能な所定のタイミングで行われる、請求項 1

乃至 3 のいずれか一項に記載のメモリ管理プログラム。

【請求項 5】

保存される前記コンテキスト情報は、複数のプロセッサのコンテキスト情報である、請求項 1 乃至 4 のいずれか一項に記載のメモリ管理プログラム。

【請求項 6】

メモリへの書き込みを検知する処理と、

検知された前記書き込みの書き込み先のデータ領域に記憶された書き込み前のデータと、前記メモリへの書き込みを検知したときのプロセッサのコンテキスト情報を対応付けて所定の記憶領域に保存する処理と

コンテキストスイッチにより前記コンテキスト情報の保存がされていない場合、前記メモリにおける前記コンテキスト情報を保存する領域に変更を加える処理と、 10

をコンピュータが実行するメモリ管理方法。

【請求項 7】

メモリへの書き込みを検知する検知部と、

前記検知部により検知された前記書き込みの書き込み先のデータ領域に記憶された書き込み前のデータと、前記メモリへの書き込みを検知したときのプロセッサのコンテキスト情報を対応付けて所定の記憶領域に保存する保存部と、

を備え、

コンテキストスイッチにより前記コンテキスト情報の保存がされていない場合、前記メモリにおける前記コンテキスト情報を保存する領域に変更を加えるメモリ管理装置。 20

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、メモリ管理プログラム、メモリ管理方法、及びメモリ管理装置に関する。

【背景技術】

【0002】

従来、仮想マシンモニタが、ゲストOSに割り当てるメモリ領域に対応するページテーブルの全ページを書き込み禁止状態に設定し、ページ書き込み違反の例外に応答して、ページ書き込み違反が発生したページの更新前データを仮想マシンモニタが管理するメモリ領域に保存する情報処理装置がある。 30

【0003】

仮想マシンモニタは、更新前データを保存した後に、ページ書き込み違反が発生したページの書き込み禁止を解除することによってゲストOSによるページ書き込み違反が発生したページへの書き込みを継続させる。チェックポイントが取得されるたびに、仮想マシンモニタは、ゲストOSに割り当てるメモリ領域に対応するページテーブルの全ページを書き込み禁止状態に再設定する（例えば、特許文献1参照）。

【0004】

また、プログラムが実行時に使用するメモリ情報を時間的経過に従って収集および保存し、後に任意の時点の全メモリ情報を復元する方法がある（例えば、特許文献2または3参照）。 40

【先行技術文献】

【特許文献】

【0005】

【特許文献1】特開2009-245216号公報

【特許文献2】特開昭62-000509号公報

【特許文献3】特開2009-146381号公報

【発明の概要】

【発明が解決しようとする課題】

【0006】

しかしながら、従来の情報処理装置では、CPU(Central Processing Unit)コンテ

50

キスト情報を取得する場合、コンテキスト情報取得のプロセスがコンテキスト情報を保存し、さらにC P Uを通常の動作状態にする必要があったため、コンテキスト情報の取得処理に時間が掛かってしまう。

【0007】

また、メモリデータとC P Uのコンテキスト情報との対応が取れなかつたために、障害によりコンピュータシステムが停止した場合に、コンテキスト情報によりC P Uの動作状況を把握して、障害の原因を過去に遡って調査することはできない。

【0008】

そこで、一側面では、装置が障害により停止した場合に、コンテキスト情報の取得処理の時間を短縮することができるシステム管理プログラムを提供することを目的とする。

10

【課題を解決するための手段】

【0009】

一つの案では、メモリ管理プログラムは、メモリへの書き込みを検知し、検知された前記書き込みの書き込み先のデータ領域に記憶された書き込み前のデータと、前記メモリへの書き込みを検知したときのプロセッサのコンテキスト情報とを対応付けて所定の記憶領域に保存し、コンテキストスイッチにより前記コンテキスト情報の保存がされていない場合、前記メモリにおける前記コンテキスト情報を保存する領域に変更を加える、処理をコンピュータに実行させる。

【発明の効果】

【0010】

20

一様によれば、装置が障害により停止した場合に、コンテキスト情報の取得処理の時間を短縮することができる。

【図面の簡単な説明】

【0011】

【図1】メモリ管理装置のハードウェア構成を示す図

【図2】メモリ管理装置のソフトウェア構成を示す図

【図3】メモリ管理装置の機能の概略を示す図

【図4】ページ管理テーブルを示す図

【図5】メモリ変更テーブルを示す図

【図6】差分メモリ取得方法の設定を示す図

30

【図7】書き込みトラップとメモリデータの保存を示す図

【図8】差分メモリの確保とメモリデータの保存を示す図

【図9】メモリダンプの復元を示す図

【図10】差分メモリとコンテキスト情報との対応を示す図

【図11】ページング発生時のメモリデータの保存を示す図

【図12】差分メモリの削除方法の設定を示す図

【図13】差分メモリの削除を示す図

【図14】メモリ管理動作を示すフローチャート

【図15】メモリデータ取得の動作を示すフローチャート

【図16】メモリダンプ復元ポイントの作成動作を示すフローチャート

40

【図17】メモリダンプの復元動作を示すフローチャート

【発明を実施するための形態】

【0012】

以下、図面に基づいて本発明の実施の形態を説明する。

【0013】

図1は、本実施の形態におけるメモリ管理装置のハードウェア構成の一例を示す図である。

【0014】

図1において、メモリ管理装置1は、C P U 2、メモリ3、H D D (Hard Disk Drive) 4、入力装置5、出力装置6、およびネットワークI / F (Interface) 7を備える。

50

C P U (Central Processing Unit) 2、メモリ3、H D D (Hard Disk Drive) 4、入力装置5、出力装置6、およびネットワークI / F (Interface) 7は、システムバス8によって相互に接続される。

【0015】

C P U 2は、H D D 4に記憶されたプログラムやデータをメモリ3に読み出して、プログラムによる処理を実行する中央演算処理装置である。C P U 2は、例えば複数のプロセッサコアを有するマルチコアプロセッサである。マルチコアプロセッサは複数のプロセッサの間でプロセッサ間通信を行うことができる。

【0016】

メモリ3は、メモリ管理装置1の主記憶装置であり、例えばR A M (Random Access Memory)によって構成される。メモリ3に記憶されたプログラムはC P U 2によって実行される。メモリ3は、例えばR O M (Read Only Memory)やフラッシュメモリを含んでいてもよい。

10

【0017】

H D D 4は、メモリ管理装置1の2次記憶である。H D D 4には、プログラムやデータが格納されて、適宜メモリ3に読み出される。また、メモリ3に確保された主メモリ領域のアドレスが不足した場合、ページング処理にてメモリ3のデータがH D D 4に一時的に退避される。本実施の形態では2次記憶装置としてH D D 4を例示したが、H D D 4の代わりに、例えばS S D (Solid State Drive)を用いることもできる。

【0018】

20

入力装置5は、メモリ管理装置1の外部から内部にデータの入力を行う機器である。入力装置5は、例えば、キーボード、マウス、タッチパネル等である。出力装置6は、メモリ管理装置1の内部のデータを外部に出力する装置である。出力装置6は、例えば、ディスプレイ、スピーカである。

【0019】

ネットワークI / F 7は、ネットワークを介した外部装置との通信を制御する。ネットワークI / F 7は、接続するネットワークのプロトコルに対応したN I C (Network Interface Controller)を有する。

【0020】

30

次に、本実施の形態におけるメモリ管理装置1のソフトウェアの構成を、図2を用いて説明する。図2は、メモリ管理装置のソフトウェア構成の一例を示す図である。図2で説明するソフトウェアは、図1で説明したメモリ3に読み出されて、C P U 2によって実行される。

【0021】

図2において、メモリ管理装置1は、ソフトウェアの構成として、O S (Operating System)層とアプリケーション層と、O S層の下位に位置し、図示しないハードウェア層からなる階層構造のソフトウェアを備える。O S層は、カーネル10とデバイスドライバ11とを備える。

【0022】

40

デバイスドライバ11は、ハードウェア層のハードウェアを制御するプログラムである。デバイスドライバ11は、ハードウェアの構成に対応した複数のプログラムモジュールを有し、それぞれのプログラムモジュールはハードウェアの構成に応じてカーネル10に対して動的に追加してもよい。

【0023】

アプリケーション層は、O S層の上で動作する複数のアプリケーションプログラム(以下、「アプリ」と省略する。)を備える。メモリ管理装置1は、アプリの一例として複数のサービス13と、複数のユーザプロセス14とを備える。ユーザプロセス14は、例えば、O S上で実行され、ユーザに対して定められた目的を果たすための機能を提供するプログラムである。また、サービス13は、例えば、ユーザプロセス14に対して、サービス13を介してO S層またはハードウェア層を利用したサービスを提供するプログラムで

50

ある。

【0024】

カーネル10は、OS層の中核となる機能を備え、例えば、メモリ管理部102、プロセス管理部103、I/O (Input Output) 管理部104、ネットワーク管理部105、及びファイルシステム管理部106を備える。

【0025】

メモリ管理部102は、物理的メモリであるメモリ3に対して仮想アドレッシングによるメモリマッピングを行い、それぞれのアプリがマッピングしたメモリ領域を安全に利用できるように管理する。プロセス管理部103は、それぞれのアプリに対して実行を許可し、ハードウェアへのアクセスのためのインターフェースを提供する。例えば、プロセス管理部103は、アプリからのシステムコールに対して、ハードウェアの利用を提供する。また、プロセス管理部103は、アプリを実行するのに必要なスタックを提供する。

【0026】

I/O管理部104は、デバイスドライバ11を介して、メモリ管理装置1に接続される入出力デバイスによる入出力データの管理を行う。ネットワーク管理部105は、メモリ管理装置1に接続されるネットワークの管理を行う。ファイルシステム管理部106は、ファイルシステムによるデータの取り扱いを管理する。

【0027】

メモリ管理部102は、メモリ管理プログラム12を備える。メモリ管理プログラム12は、メモリ管理部102の機能に対して、本実施の形態において新たに提供する機能であり、詳細は図3を用いて後述する。なお、本実施の形態において、メモリ管理プログラム12は、メモリ管理部102の機能を拡張することによって実装している。但し、メモリ管理プログラム12を、例えばサービス13として、メモリ管理部102とは別のプログラムとして実装してもよい。

【0028】

次に、メモリ管理装置1の機能を、図3を用いて説明する。図3は、メモリ管理装置1の機能の概略の一例を示す図である。

【0029】

図3において、メモリ管理装置1は、図2で説明したメモリ管理プログラム12と、ダンプ復元ポイント作成プロセス（以下、「復元プロセス」と省略する。）131のプログラムが動作している。メモリ管理プログラム12は、検知部121、確保部122、及び保存部123を備える。復元プロセス131はOS層の上で動作するサービス13として実装する。但し、復元プロセス131を、例えばカーネル10の機能を拡張して実装してもよい。

【0030】

メモリ管理装置1は、メモリ管理部102によってメモリ3にマッピングされた仮想アドレス空間において、主メモリ領域31と差分用メモリ領域32を備える。主メモリ領域31は、プログラムの実行においてカーネル10により利用される領域である。

【0031】

差分用メモリ領域32は、ページ管理テーブル33と差分メモリ34を備える。ページ管理テーブル33は、後述するページアウト時の差分メモリデータの変更を管理するテーブルである。差分メモリ34は、コンテキスト、差分メモリデータ、メモリ変更テーブルを有している。差分用メモリ領域32には複数の差分メモリ34を備えることができる。図3で例示する差分メモリ34は、差分メモリ1(341)、差分メモリ2(342)、差分メモリ3(343)が保存された状態を図示している。それぞれ所定の差分メモリ34nには、コンテキスト34n1、差分メモリデータ34n2、およびメモリ変更テーブル34n3が対応付けられて保存される。例えば、差分メモリ1(341)には、コンテキスト3411、差分メモリデータ3412、メモリ変更テーブル3413が対応付けられて保存される。

【0032】

10

20

30

40

50

検知部 121 は、主メモリ領域 31 を監視して、主メモリ領域 31 への書き込みを検知する。検知部 121 は、例えば、メモリ管理部 102 がメモリのデータを変更する書き込み処理をトラップして、主メモリ領域 31 への書き込みを検知してもよい。検知部 121 は、書き込み処理をトラップすることにより、主メモリ領域への書き込みをページ単位で検知することができる。また、検知部 121 は、ページング処理を検知してメモリへの書き込みを検知してもよい。

【0033】

ここで、ページング処理とは、例えばユーザプロセス 14 が、物理メモリとしてマップされていない仮想アドレス空間上のページにアクセスしたときに発生するページフォルト処理を行うものである。メモリ管理部 102 は、不要なページをハードディスク等に書き出しつつ物理メモリから消去（ページアウト）し、また、必要なページをハードディスク等から読み出して物理メモリ上に配置（ページイン）する。10

【0034】

確保部 122 は、所定のタイミングで差分用メモリ領域 32 に差分メモリ 34 の記憶領域を順次確保する。図 3 では、差分メモリ 1 (341) から差分メモリ 3 (343) の 3 つの差分メモリが確保されていることを示している。

【0035】

確保部 122 が差分メモリ 34 を確保する所定のタイミングとは、例えば、所定の時間間隔が経過したときである。所定のタイミングとしては他に、メモリの変更量が所定の量に達したとき、CPU の使用率が所定値以上になったとき、ハードディスク等の記憶装置の単位時間あたりの使用量が所定の回数以上となったとき、あるいは、ネットワークの単位時間あたりの使用量が所定値以上となったとき等であってもよい。20

【0036】

確保部 122 が差分メモリ 34 を確保する所定のタイミングは予め設定することができる。所定のタイミングの設定について、図 6 を用いて説明する。図 6 は、差分メモリ取得方法の設定の一例を示す図である。

【0037】

図 6において、「設定項目」及び「パラメータ」は、差分メモリ 34 の確保のタイミングを表している。ユーザは、設定項目とパラメータを指定できる。設定項目がパラメータで指定された条件となったときに、差分メモリ 34 が確保される。例えば、設定項目で「時間」を選択して、そのパラメータが「10秒ごとに取得」であった場合、10秒ごとに、差分メモリ 34 が順次確保されることになる。30

【0038】

設定項目は複数を選択可能である。設定項目を複数選択したときには、いずれかの設定項目がパラメータで指定された条件となったときに、差分メモリ 34 が確保される。また、パラメータは適宜変更ができる。

【0039】

図 6 で示す設定項目及びパラメータの指定は、例えばメモリ管理プログラム 12 が、出力装置 6 のディスプレイを通じて、図 6 で示した設定画面の UI (User Interface) を提供する。また、設定項目を記述したテキスト等にて、設定ファイルとして予め保存しておき、メモリ管理プログラム 12 の起動時に設定ファイルを読み込んでもよい。確保部 122 は、設定された条件に従い差分メモリ 34 を確保する所定のタイミングを設定又は変更することができる。40

【0040】

保存部 123 は、確保部 122 によって確保された差分メモリ 34 の記憶領域に検知部 121 で検知された書き込み先のデータ領域に記憶された差分メモリデータをメモリダンプとして順次保存する。ここで、メモリダンプとは、メモリに記憶されたデータを書き出したものである。保存部 123 が保存する差分メモリデータは、例えば、主メモリ領域 31 のページ単位で保存してもよい。また、複数のページを一つの差分メモリデータとして保存してもよい。50

【0041】

復元プロセス 131 は、差分メモリ 34 の確保に対応して、CPU2 のコンテキスト情報をそれぞれの差分メモリ 34 に順次保存する。

【0042】

コンテキスト情報は、CPU2 が複数のプロセスにて共有されてコンテキストスイッチが発生したときに保存される CPU2 の使用状況を示す設定情報である。差分メモリ 34 に保存された差分メモリデータは、対応するコンテキスト情報と併せて保存することにより、過去の差分メモリデータとそのときの CPU2 の状態と整合性を図ることができる。

【0043】

ここで、整合性を図るとは、クラッシュ発生時より過去の時点でのメモリのデータと、そのときの CPU2 の状態とが時系列的に対応が取れていることをいう。

10

【0044】

主メモリ領域 31 のメモリデータと差分用メモリ領域 32 のメモリデータは、CPU がエラーによる停止をしたときに完全メモリダンプ及び差分メモリダンプとして保存される。保存された完全メモリダンプと差分メモリダンプは、エラーの解析のために、例えば CPU の再起動時に読み出すことができる。読み出された完全メモリダンプと差分メモリダンプによって、CPU2 がエラーで停止する前の過去のメモリダンプを復元することができる。メモリダンプ復元の詳細は後述する。

【0045】

次に、ページ管理テーブル 33 の詳細を、図 4 を用いて説明する。ページ管理テーブル 33 は、ページアウト時の差分メモリデータの変更を管理するテーブルである。図 4 は、ページ管理テーブルの一例を示す図である。

20

【0046】

図 4において、ページ管理テーブル 33 は、差分用メモリ領域番号、保存先のアドレス、ページアウト領域使用場所情報 (1 ~ n) の各フィールドを有する。差分用メモリ領域番号は、ページアウトした主メモリ領域の情報が、何番目の差分メモリに保存されているのかを示す情報である。差分用メモリ領域番号は、1 から n までの整数が入力される。

【0047】

保存先のアドレスは、主メモリ領域 31 に書き込みがされて変更があった場合に保存する、変更前の主メモリ領域 31 の情報を保存した差分用メモリ領域 34n の開始アドレスである。なお、主メモリ領域 31 が変更されていない場合は (0xffffffffffff) が入力されている。ページ管理テーブル 33 を設けることによって、増減する差分メモリ 34 を一元的に管理することが可能となる。

30

【0048】

ページアウト領域使用場所情報 (1 ~ n) は、ページアウトした主メモリ領域 31 の情報と過去の差分用メモリ領域の情報が一致した際に追加する情報である。ページアウト領域使用場所情報は、差分メモリ 34 と主メモリ領域 31 のアドレスとを対応させて保存する。ページアウト領域使用場所情報は、一つの差分メモリ 34 に対して、複数の差分メモリデータを保存することができる。例えば、図 4 では、差分用メモリ領域番号が 1 の領域では m 個の差分メモリデータが保存されている。一方、差分用メモリ領域番号が 2 の領域では、一つの差分メモリデータ、差分用メモリ領域番号が 3 の領域では、二つの差分メモリデータが保存されていることを表している。

40

【0049】

次に、それぞれの差分用メモリ領域 34n が有するメモリ変更テーブル 34n3 の詳細を、図 5 を用いて説明する。図 5 は、メモリ変更テーブルの一例を示す図である。

【0050】

図 5において、それぞれのメモリ変更テーブルは、メモリ領域のアドレス、変更フラグ、および保存先のアドレスの各フィールドを有する。メモリ領域のアドレスは、主メモリ領域 31 をページ単位で分割したときの先頭アドレスを示す。変更フラグは、主メモリ領域 31 に対する変更の有無を示す。変更フラグの "1" は、変更があったことを示す。また

50

、変更フラグの"0"は、変更が無かったことを示す。保存先のアドレスは、差分用メモリ領域34nのそれぞれの先頭アドレスである。この保存先のアドレスは、図4で説明した保存先のアドレスと同じであり、増減する差分メモリのそれぞれにおいてもメモリ領域のアドレスと対応付けて記録しておく。

【0051】

次に、図3で説明したメモリ管理プログラム12による、主メモリ領域31のメモリデータの差分用メモリ領域32への保存について、図7を用いて説明する。図7は、書き込みトラップとメモリデータの保存の一例を示す図である。

【0052】

図7において、(1)～(5)は時間経過の順番を示す。(1)において、先ずは、全ての主メモリ領域31を書き込み禁止にする。(1)で図示する主メモリ領域31の網点は、主メモリ領域31が書き込み禁止の状態であることを示している。主メモリ領域31を書き込み禁止にすることにより、検知部121は主メモリ領域31に対する全ての書き込み処理をトラップする。(2)において、検知部121が主メモリ領域に対する書き込みをともなう処理をトラップして割込処理を発生させる。

【0053】

(3)において、確保部122は、所定の記憶領域として差分用メモリ領域32に差分メモリ34を確保して、書き込み処理の対象となる主メモリ領域31のメモリデータ(データn)をページ単位で差分メモリ34にデータmとして保存する。なお、保存される主メモリ領域31のデータnと、差分メモリ34に保存されるデータmは、全く同一のデータであってもよいし、後述する完全メモリダンプの復元において、データmからデータnを復元できるのであれば、例えばデータmはデータnを圧縮したものであってもよい。

【0054】

(4)において、保存されたデータnのメモリ領域は書き込み禁止が解除されて割込処理が終了する。(5)において、トラップされた書き込み処理が実行されて、書き込み禁止が解除された主メモリ領域31にデータn'が書き込まれる。

【0055】

なお、主メモリ領域31への書き込み禁止と解除は、例えばメモリ管理プログラム12がメモリ管理部102に対して行ってもよい。

【0056】

次に、図6で説明した差分メモリ取得方法において、設定項目が「時間」である場合の差分メモリ34の確保とメモリデータの保存の詳細を、図8を用いて説明する。図8は、差分メモリ34の確保とメモリデータの保存の一例を示す図である。

【0057】

図8(1)において、(1)から(3)は、時刻t1からt3におけるダンプ復元ポイント1から3の主メモリ領域31のメモリの状態を示している。時刻t1において、メモリの状態は、それぞれのデータ領域におけるメモリデータが「A、B、C、D、E」である。

【0058】

時刻t2において、主メモリ領域31に対する書き込み処理により書き込み先のデータ領域のメモリデータ「B、E」が「Z、E」に書き換えられる。このとき、所定の記憶領域として差分メモリ1には、書き込み先のデータ領域の書き込み処理の前のメモリデータ「B、E」が差分メモリデータ(3412-1、3412-2)に保存される。差分メモリ1(341)には、図7で説明した差分メモリ1(341)の先頭アドレス等が保存される。

【0059】

次に、時刻t3において、書き込み先のデータ領域の書き込み処理の前のメモリデータ「Z、C」が、差分メモリ1(342)から変更された差分メモリ2(342)の差分メモリデータ(3422-1、3422-2)として保存されて、主メモリ領域31が「G、F」に書き換えられる。時刻t4において、書き込み先のデータ領域の書き込み処理の

10

20

30

40

50

前のメモリデータ「G、D」が、差分メモリ2(342)から変更された差分メモリ3(343)の差分メモリデータ(3432-1、3432-2)に保存されて、主メモリ領域31が「H、D」に書き換えられる。

【0060】

時刻t4の状態にて、例えばOSがクラッシュして、CPUが停止したとすると、時刻t4における完全メモリダンプと、差分用メモリ領域32に保存された差分メモリ1(342)～差分メモリ3(343)までの情報が保存される。また、それぞれの差分メモリには、メモリ変更テーブル(3413、3423、3433)が保存されている。

【0061】

次に、OSがクラッシュした原因を解析する際の、各復元ポイントにおけるメモリダンプ復元の詳細を、図9を用いて説明する。図9は、メモリダンプの復元の一例を示す図である。10

【0062】

図9において、(1)は、図8で説明したOSクラッシュ時に保存された完全メモリダンプと差分用メモリ領域32を示す。

【0063】

(2)において、先ず、完全メモリダンプを、OSの再起動等により読み出して、仮ダンプファイルを作成する。仮ダンプファイルの作成及び復元は、例えばダンプファイル解析用の別のコンピュータで行ってもよい。

【0064】

(3)において、クラッシュ前に最後に保存された差分メモリ3(343)の差分メモリデータ(3432-1、3432-2)のメモリデータ「G、D」が、メモリ変更テーブル3433に保存された主記憶領域31のアドレスに従い、基のアドレスに書き込まれて、復元ポイント3における完全メモリダンプが復元される。20

【0065】

(4)において、差分メモリ2(342)の差分メモリデータ(3422-1、3422-2)のメモリデータ「Z、C」が、メモリ変更テーブル3423に保存された主記憶領域31のアドレスに従い、基のアドレスに書き込まれて、復元ポイント2における完全メモリダンプが復元される。

【0066】

図9では、復元ポイント2までの完全メモリダンプの復元を図示したが、同様の方法によって、順次保存された差分メモリのデータに基づいて過去の完全メモリダンプを復元することができる。したがって、OSのクラッシュがクラッシュ直前ではない過去の原因によるものである場合には、それぞれの復元ポイントにおけるメモリダンプを時系列的に遡って復元することによりクラッシュの原因解析を容易にすることができる。30

【0067】

次に、図3で説明した復元プロセス131による主メモリ領域31のメモリデータとコンテキスト情報との整合について、図10を用いて説明する。図10は、差分メモリとコンテキスト情報との対応の一例を示す図である。

【0068】

図10において、本実施の形態におけるメモリ管理装置1は、CPU0(20)～CPU3(23)の4つのCPUを備えるマルチプロセッサの場合を例示している。復元プロセス131は、CPU0によって実行されるものとする。40

【0069】

(1)において、復元プロセス131が起動されると、主メモリ領域31のダンプ用コンテキスト保存領域(図示網点の領域)のメモリデータがコンテキストスイッチにより保存されていない場合、明示的にコンテキスト保存領域に変更を加える。変更を加えることで、図3で説明した検知部121は、主メモリ領域31への書き込みを検知して、所定の記憶領域としての差分メモリ領域(i-1)のコンテキスト保存領域34(i-1)1にメモリデータを保存する。(1)に示す処理によって、コンテキスト情報と対応付けて保50

存する書き込み先のデータ領域のメモリのデータが保存可能となる。また、コンテキスト情報の保存は、図7又は図8で説明したメモリ管理プログラム12によって行うことができる。復元プロセス131がコンテキスト情報を保存する場合に比べてシステム停止時間を短縮することができる。

【0070】

(2)において、復元プロセス131は、CPU1(21)～CPU3(23)を待ち合わせ処理(ループ処理)にして停止するとともに、CPU1(21)～CPU3(23)のコンテキスト情報をダンプ用コンテキスト保存領域に退避させる。

【0071】

(3)において、復元プロセス131は、CPU0のコンテキスト情報をダンプ用コンテキスト保存領域に退避させる。10

【0072】

(4)において、復元プロセス131は、全ての主メモリ領域31を書き込み禁止にして、主メモリ領域31への書き込み処理をトラップする。主メモリ領域31への書き込み処理をトラップして、図7および図8で説明したとおり、書き込み先のデータ領域の書き込み前のメモリのデータを差分メモリ領域($i - 1$)に保存する。保存されたメモリのデータは、(1)および(2)で保存されたコンテキスト情報と同じ所定の記憶領域としての差分メモリ領域($i - 1$)に対応付けられて保存される。

【0073】

(5)において、復元プロセス131は、差分メモリ*i*にコンテキスト保存領域34*i*1を確保して、コンテキスト保存領域を差分メモリ34の中のコンテキスト保存領域34*n*3に変更する。例えば、*n* = *i*の場合は、コンテキスト保存領域は34*i*3に変更される。20

【0074】

(6)において、復元プロセス131は、停止しているCPU1(21)～CPU3(23)を起動して処理を再開し、自身のプロセスを終了する。

【0075】

以上の動作によって、マルチプロセッサの場合であっても、同じタイミングの全プロセッサのコンテキスト情報と差分メモリデータと対応付けて、整合させて保存することができる。また、コンテキスト情報を差分用メモリ領域32に保存することにより、コンテキスト情報の保存は、差分メモリの保存と対応付けられて行われることとなる。したがって、コンテキスト情報を復元プロセス131が割込処理にて行う必要がなくなり、割込処理によるシステムの停止時間を短くすることができる。30

【0076】

次に、ページング処理を考慮した差分メモリの保存について、図11を用いて説明する。図11は、ページング発生時のメモリデータの保存の一例を示す図である。

【0077】

図11において、図3で説明した検知部121が、ページアウトによって主メモリ領域31のメモリデータCが保存されていたメモリ領域が変更されることを検知すると、ページ管理テーブル33に保存された情報から、これからページアウトされるメモリデータCと差分メモリに過去に保存されたメモリデータと比較する。(図11では、メモリデータCが既に保存されている場合を図示している。)40

一致する情報が格納されておらず、メモリデータCが検出されなかった場合、差分メモリ2に変更される前の情報Cを保存するとともに、ページ管理テーブル33に差分メモリ3の差分メモリデータの保存先のアドレス(図示「0x####3000」)を新規に登録する。

【0078】

一方、メモリデータCと一致するメモリデータが差分メモリ領域(3442-2)に既に格納されていることが検出された場合、差分メモリ領域への情報保存は行わず、ページ管理テーブル33の一致した情報の記録に対して、本来情報が保存されるべきであった場所の情報を追加する。図11では、ページアウト領域使用場所情報_1に対して、「領域50

3、0x####3000」が追加されることを図示している。

【0079】

以上の動作によって、ページングの発生による主メモリ領域31が書き換わる場合であっても差分用メモリ領域32による差分メモリの保存ができる。

【0080】

次に、差分メモリの削除について、図12及び図13を用いて説明する。図12は、差分メモリの削除方法の設定の一例を示す図である。図13は、差分メモリの削除の一例を示す図である。

【0081】

差分メモリ34は、OSがクラッシュして停止等したときのみならず、稼働中においてもメモリデータとコンテキスト情報を順次保存していくため、例えば差分用メモリ領域32の記憶容量が大きくなる場合がある。そこで、本実施の形態においては、差分メモリ34を所定の条件にて削除して、障害調査に関連するデータを残しつつ、差分用メモリ領域32の記憶容量の肥大化防止を図る。10

【0082】

図12において、差分メモリ34の削除の方法は、設定項目とパラメータによって設定される。例えば、現在時刻より一定時間経過して古くなったメモリデータを削除する設定ができる。システムが安定して稼働している場合には、一定時間以上経過して古くなったメモリデータは、トラブルの原因解析に使用される可能性が低くなるため、クラッシュ前の所定の時間のみの記録を残して、差分メモリダンプの容量を小さくすることができる。20

【0083】

また、差分メモリ34を所定数毎に削除して間引く設定ができる。例えば、システムが安定して稼働している場合と、不安定となっている場合において、保存する差分メモリの時間的な粒度を変更することができる。

【0084】

さらに、CPUのコンテキスト情報に特定の命令、または特定のアドレスが含まれる場合に差分メモリ34を残して他を削除する設定をすることができる。トラブルの原因となる可能性の高い命令やデータの取り扱いに対して、監視を強化することができる。

【0085】

なお、図12で説明した差分メモリ34の削除の設定は、複数の設定項目を同時に設定することができる。設定はメモリ管理プログラム12が提供するUIによって設定をするようにしてもよい。また、テキストファイル30

図13において、削除前は差分メモリ1～5までが存在しているものとする。この中で差分メモリ2(342)及び差分メモリ4(344)を削除する。削除対象である差分メモリ2(342)に含まれるメモリデータ「Z、C」を、差分メモリ2(342)の直前に保存された差分メモリ1(341)に保存する。

【0086】

なお、削除対象の差分メモリがページアウトにより保存されたメモリ領域である場合には、ページ管理テーブル33に記録された差分メモリの差分メモリデータの保存先のアドレスの記載も変更する。40

【0087】

次に、削除対象である差分メモリ4(344)に含まれるメモリデータ「U」を、差分メモリ4(344)の直前に保存された差分メモリ3(343)に保存する。一方、差分メモリ4(344)に含まれるメモリデータ「D」は、差分メモリ3に同一領域のメモリデータ「D」が存在しているため、復元時に使用されることがない「D」は保存しない。

【0088】

次に、差分メモリ2(342)と差分メモリ4(344)を削除して削除の処理を終了する。

【0089】

50

20

30

40

50

次に、以上説明したメモリ管理装置1の動作を、フローチャートにて説明する。

【0090】

先ず、メモリ管理プログラム12の動作を、図14を用いて説明する。図14は、メモリ管理動作の一例を示すフローチャートである。

【0091】

図14において、メモリ管理プログラム12の検知部121は、メモリ差分監視の機構として主メモリ領域31の監視を開始する(S11)。なお、ステップS11の動作の詳細は、図15を用いて後述する。

【0092】

次に、メモリ管理プログラム12は復元ポイントの取得の要求を受けたか否かを判断する(S12)。復元ポイントの取得の要求を受けた場合(S12でYES)、ダンプ復元ポイントの作成を行う(S13)。復元ポイントの取得の要求がない場合(S12でNO)、ステップS12をループする。なお、ステップS13のダンプ復元ポイントの作成の詳細は、図16を用いて後述する。

【0093】

メモリ管理プログラム12は、メモリ管理装置1が稼働中は常駐するプログラムであり、S12～S13をループして動作する。

【0094】

次に、図11で説明したページアウトによるメモリデータの取得動作の詳細を、図15を用いて説明する。図15は、メモリデータ取得の動作の一例を示すフローチャートである。

【0095】

図15において、主メモリ領域31が書き込み禁止状態か否かを判断する(S21)。書き込み禁止状態において、検知部121は、主メモリ領域31の書き込み処理をトランプする。書き込み禁止で無い場合(S21でNO)、主メモリ領域31のメモリ変更が実行され(S28)、メモリ変更テーブル34n3が更新されて(S29)、処理を終了する。

【0096】

一方、書き込み禁止であった場合(S21でYES)、ページアウトの発生か否かを判断する(S22)。ページアウトの発生であった場合(S22でYES)、変更前の主メモリ領域31におけるメモリデータが登録済みか否かを判断する(S23)。登録済みであった場合(S23でYES)、図11で説明したページ管理テーブル33のページアウト領域使用場所情報を更新する。

【0097】

一方、登録済みで無かった場合(S23でNO)、ページ管理テーブル33の保存先のアドレスの情報に新たに差分メモリ領域を追加する更新を行い(S25)、書き込み先のデータ領域のデータを追加した差分メモリ領域に保存して退避させる(S26)。また、ページアウトで無い場合(S22でNO)、同様に書き込み先のデータ領域のデータを差分メモリ領域に保存する(S26)。

【0098】

次に、ページ書き込みの禁止を解除して(S27)、メモリ変更を実行する(S28)。

【0099】

以上の動作により、主メモリ領域31に対する書き込みは、差分メモリとして保存される。

【0100】

次に、図10で説明したメモリダンプ復元ポイントの作成動作の詳細を、図16を用いて説明する。図16は、メモリダンプ復元ポイントの作成動作の一例を示すフローチャートである。

【0101】

10

20

30

40

50

図16において、主メモリ領域31のコンテキスト情報が待避済みか否かを判断する(S31)。コンテキスト情報が待避済みの場合(S31でYES)、他のCPUを停止する(S33)。一方、コンテキスト情報が待避済みで無い場合(S31でNO)、コンテキスト情報を差分メモリのコンテキスト情報保存領域に保存して(S32)、他のCPUを停止する(S33)。停止したCPUのコンテキスト情報を主メモリ領域31に保存して(S34)、次に、復元プロセス131が動作するCPU0のコンテキスト情報を主メモリ領域31に保存する(S35)。

【0102】

主メモリ領域31を書き込み禁止にして(S36)、差分用メモリ領域32に新しい差分メモリ領域を用意し(S37)、さらに停止したCPUの動作を再開して(S38)、メモリダンプ復元ポイントの作成動作を終了する。10

【0103】

次に、図9で説明したメモリダンプの復元動作の詳細を、図17を用いて説明する。図17は、メモリダンプの復元動作の一例を示すフローチャートである。

【0104】

図17において、仮ダンプファイルとして、完全メモリダンプをコピーする(S41)
。

【0105】

次に、ステップS42からS45までの繰り返し動作を、目的の復元ポイントのメモリダンプになるまで行う。先ず、各復元ポイントに対する差分メモリのメモリ変更テーブルの情報を基に、差分メモリのメモリデータで仮ダンプファイルを書き換えて、各復元ポイントにおけるメモリダンプを復元する(S43)。20

一方、差分メモリに対応する情報が無い場合には、ページ管理テーブル33に保存されたページアウト領域使用場所情報の情報を基に、メモリデータを読み出して仮ダンプファイルを書き換えてメモリダンプを復元する(S44)。

目的の復元ポイントにおけるメモリダンプまで復元してメモリダンプの復元動作を終了する。

以上説明した本実施の形態の動作によって、CPUがOSクラッシュ等により停止する前であっても、差分メモリデータとそれに対応してコンテキスト情報とを整合性をたもつたまま保存するので、エラーで停止したときの完全メモリダンプに加えて、エラーで停止前の過去のCPU2の状態とメモリダンプを復元することができる。これにより、トラブルの原因を解析しやすくなる。30

【0106】

以上、本発明の実施例について詳述したが、本発明は斯かる特定の実施形態に限定されるものではなく、特許請求の範囲に記載された本発明の要旨の範囲内において、種々の変形・変更が可能である。

【0107】

以上の説明に関し、更に以下の項を開示する。

(付記1)

メモリへの書き込みを検知し、40

検知された前記書き込みの書き込み先のデータ領域に記憶された書き込み前のデータと、前記メモリへの書き込みを検知したときのプロセッサのコンテキスト情報を対応付けて所定の記憶領域に保存する

処理をコンピュータに実行させるメモリ管理プログラム。

(付記2)

前記所定の記憶領域は、前記コンテキスト情報を保存するときに変更され、

前記書き込み前のデータは、変更された前記所定の記憶領域に対応付けられて順次保存される、付記1に記載のメモリ管理プログラム。

(付記3)

前記コンテキスト情報は、コンテキストスイッチによる前記コンテキスト情報の保存が50

されていない場合には前記メモリのコンテキスト退避領域に明示的な変更を加えることにより前記メモリへの書き込みが検知されて保存される、付記 1 又は 2 に記載のメモリ管理プログラム。

(付記 4)

前記所定の記憶領域の変更は、予め設定可能な所定のタイミングで行われる、付記 1 乃至 3 のいずれか一に記載のメモリ管理プログラム。

(付記 5)

保存される前記コンテキスト情報は、複数のプロセッサのコンテキスト情報である、付記 1 乃至 4 のいずれか一に記載のメモリ管理プログラム。

(付記 6)

10

メモリへの書き込みを検知する処理と、

検知された前記書き込みの書き込み先のデータ領域に記憶された書き込み前のデータと、前記メモリへの書き込みを検知したときのプロセッサのコンテキスト情報を対応付けて所定の記憶領域に保存する処理と

をコンピュータが実行するメモリ管理方法。

(付記 7)

前記所定の記憶領域は、前記コンテキスト情報を保存するときに変更され、

前記書き込み前のデータは、変更された前記所定の記憶領域に対応付けられて順次保存される、付記 6 に記載のメモリ管理方法。

(付記 8)

20

前記コンテキスト情報は、コンテキストスイッチによる前記コンテキスト情報の保存がされていない場合には前記メモリのコンテキスト退避領域に明示的な変更を加えることにより前記メモリへの書き込みが検知されて保存される、付記 6 又は 7 に記載のメモリ管理方法。

(付記 9)

前記所定の記憶領域の変更は、予め設定可能な所定のタイミングで行われる、付記 6 乃至 8 のいずれか一に記載のメモリ管理方法。

(付記 10)

30

保存される前記コンテキスト情報は、複数のプロセッサのコンテキスト情報である、付記 6 乃至 9 のいずれか一に記載のメモリ管理方法。

(付記 11)

メモリへの書き込みを検知する検知部と、

前記検知部により検知された前記書き込みの書き込み先のデータ領域に記憶された書き込み前のデータと、前記メモリへの書き込みを検知したときのプロセッサのコンテキスト情報を対応付けて所定の記憶領域に保存する保存部と

を備えたメモリ管理装置。

(付記 12)

前記所定の記憶領域は、前記コンテキスト情報を保存するときに変更され、

前記書き込み前のデータは、変更された前記所定の記憶領域に対応付けられて順次保存される、付記 11 に記載のメモリ管理装置。

40

(付記 13)

前記コンテキスト情報は、コンテキストスイッチによる前記コンテキスト情報の保存がされていない場合には前記メモリのコンテキスト退避領域に明示的な変更を加えることにより前記メモリへの書き込みが検知されて保存される、付記 11 又は 12 に記載のメモリ管理装置。

(付記 14)

前記所定の記憶領域の変更は、予め設定可能な所定のタイミングで行われる、付記 11 乃至 13 のいずれか一に記載のメモリ管理装置。

(付記 15)

保存される前記コンテキスト情報は、複数のプロセッサのコンテキスト情報である、付

50

記 1 1 乃至 1 4 のいずれか一に記載のメモリ管理装置。

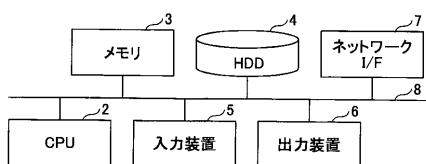
【符号の説明】

【 0 1 0 8 】

1	メモリ管理装置	
2	C P U	
3	メモリ	
4	H D D	
5	入力装置	
6	出力装置	
7	ネットワーク I / F	10
8	システムバス	
1 0	カーネル	
1 0 2	メモリ管理部	
1 0 3	プロセス管理部	
1 0 4	I O 管理部	
1 0 5	ネットワーク	
1 0 6	ファイルシステム	
1 1	デバイスドライバ	
1 2	メモリ管理プログラム	
1 2 1	検知部	20
1 2 2	確保部	
1 2 3	保存部	
1 3	サービス	
1 3 1	ダンプ復元ポイント作成プロセス	
1 4	ユーザプロセス	
3 1	主メモリ領域	
3 2	差分用メモリ領域	
3 3	メモリ管理テーブル	
3 4 1、3 4 2、3 4 3	差分メモリ	
3 4 1 1、3 4 2 1、3 4 3 1	コンテキスト保存領域	30
3 4 1 2、3 4 2 2、3 4 3 2	差分メモリデータ保存領域	
3 4 1 3、3 4 2 3、3 4 3 3	メモリ変更テーブル	

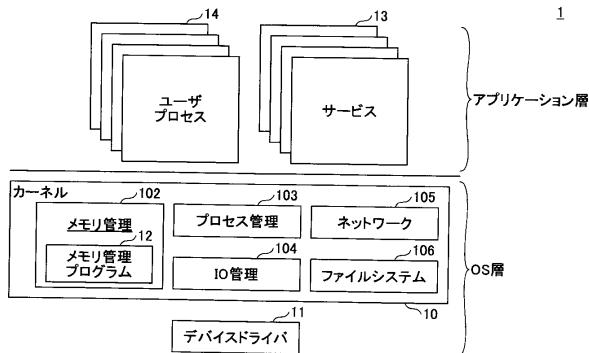
【図1】

メモリ管理装置のハードウェア構成を示す図



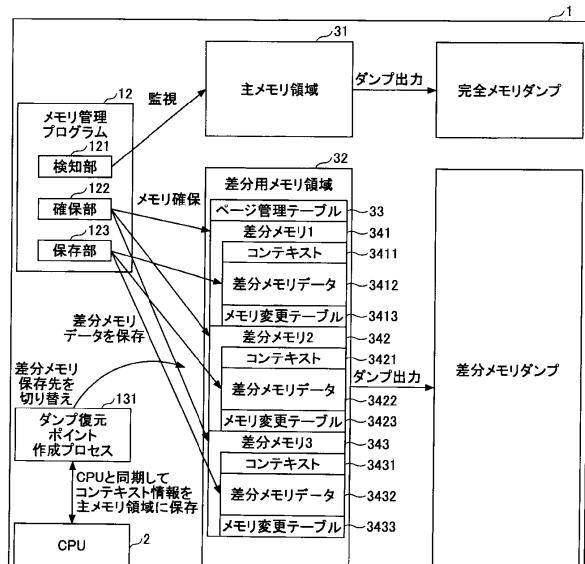
【図2】

メモリ管理装置のソフトウェア構成を示す図



【図3】

メモリ管理装置の機能の概略を示す図



【図4】

ページ管理テーブルを示す図

差分用メモリ領域番号	保存先のアドレス	ページアウト領域使用場所情報1	ページアウト領域使用場所情報2	ページアウト領域使用場所情報n
1	0x10003000	領域3, 0x1000f000	領域4, 0x20000000	領域m, 0x40000000
2	0x0000f000	-	-	-
3	0x10006000	領域5, 0x30000000	-	-

【図6】

差分メモリ取得方法の設定を示す図

設定項目(複数選択可)	パラメータ(変更可能)
時間	10秒ごとに取得
メモリ変更量	1000ページ変更されるごとに取得
CPU使用率	使用率50%以上のときに取得
ディスク使用量	1000 IOPS以上のとき取得
ネットワーク使用量	500Mbps以上のとき取得

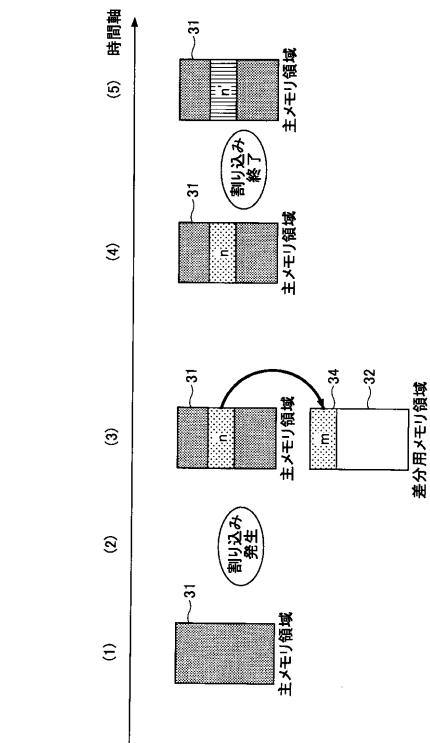
【図5】

メモリ変更テーブルを示す図

メモリ領域のアドレス	変更フラグ	保存先のアドレス
0x00000000	1	0x10003000
0x00001000	1	0x0000f000
0x00002000	1	0x10006000
0x00003000	0	0xffffffff
0x00004000	1	0x30000000
.	.	.
.	.	.
.	.	.

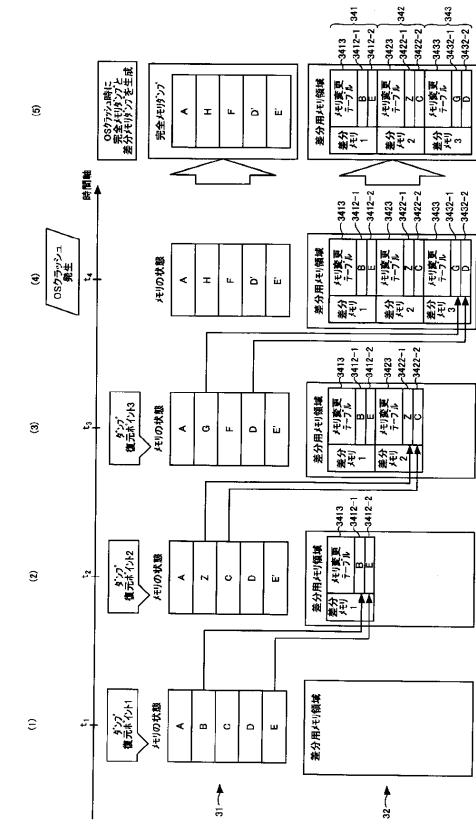
【図7】

書き込みトラップとメモリデータの保存を示す図



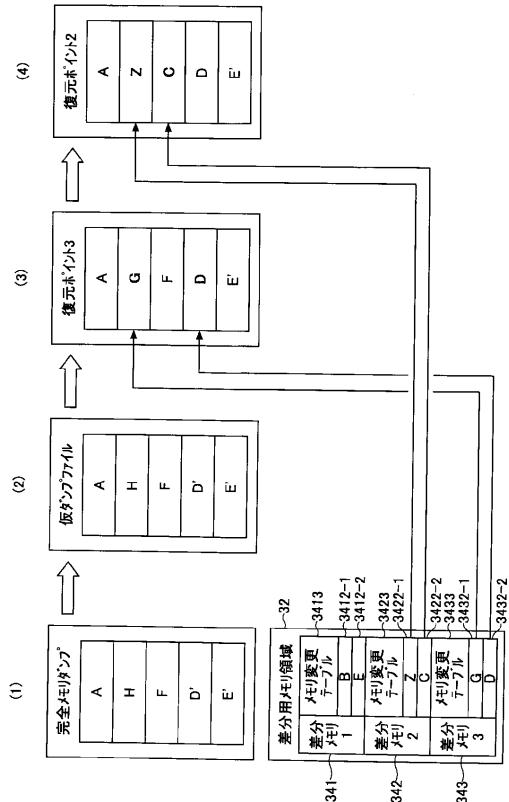
【図8】

差分メモリの確保とメモリデータの保存を示す図



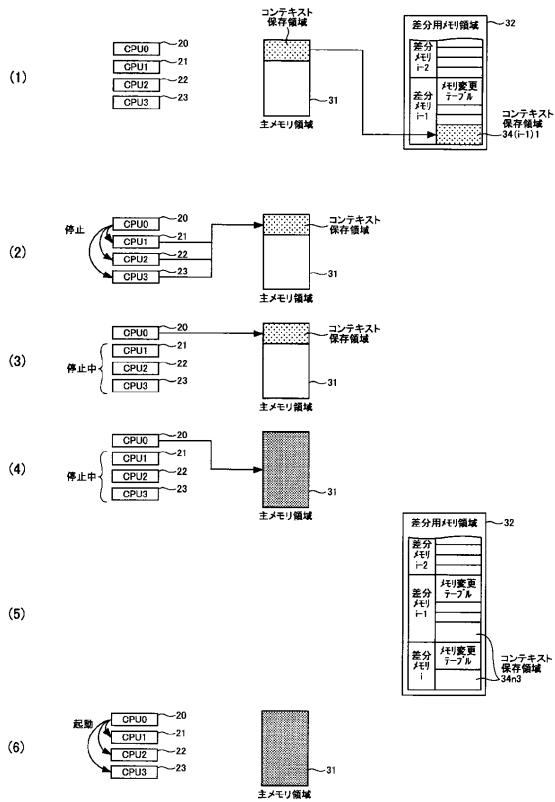
【図9】

メモリダンプの復元を示す図



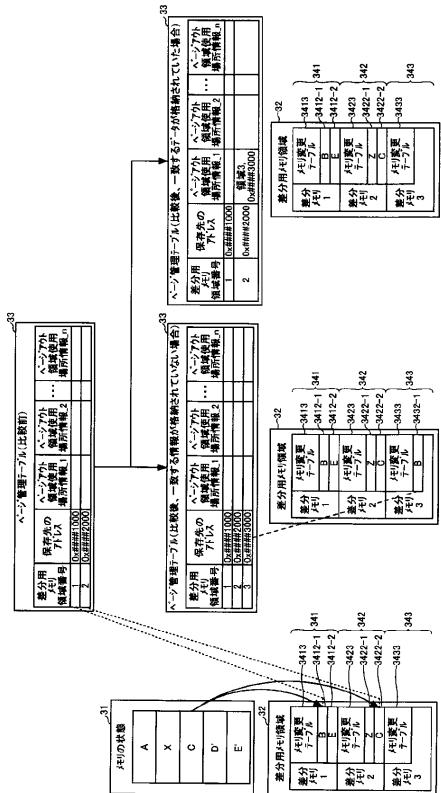
【図10】

差分メモリとコンテキスト情報との対応を示す図



【図11】

ページング発生時のメモリデータの保存を示す図



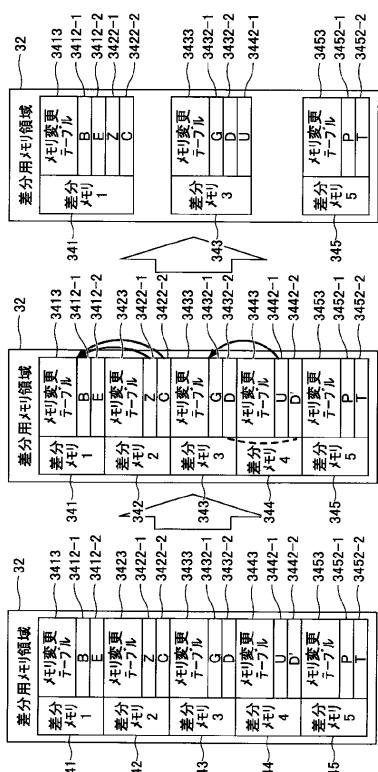
【図12】

差分メモリの削除方法の設定を示す図

設定項目(複数選択可)	パラメタ(変更可能)
現在時刻より一定時間古いデータを削除する	1時間
差分メモリデータを一定数ごとに削除する(間引く)	2時点ごと
CPUコンテキストに特定の命令アドレスが含まれる場合、差分メモリデータを残す	特定の命令アドレス

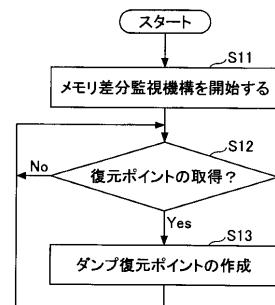
【図13】

差分メモリの削除を示す図



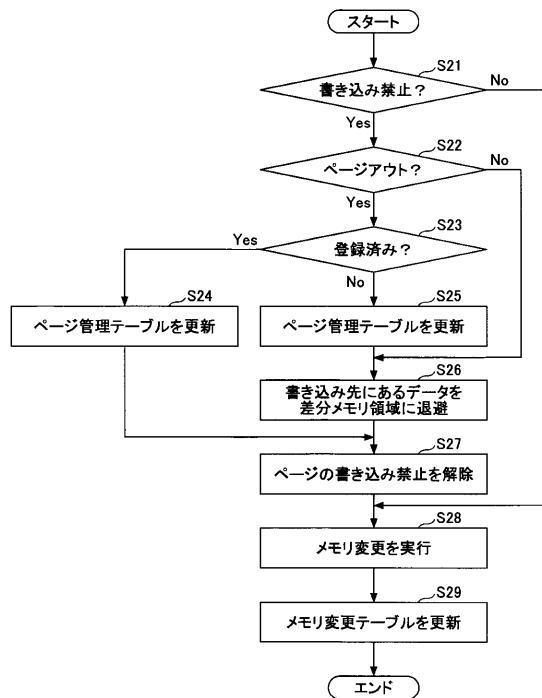
【図14】

メモリ管理動作を示すフローチャート



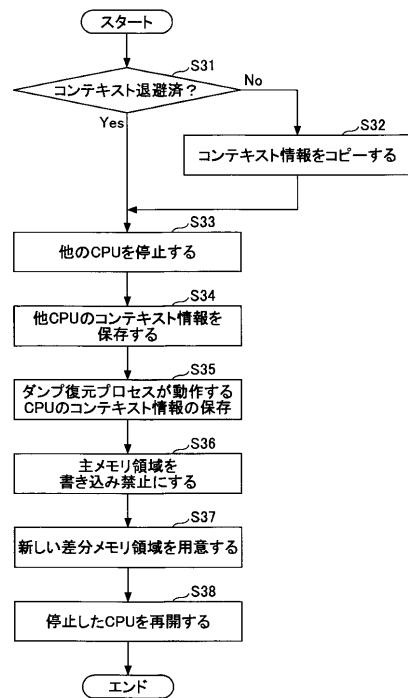
【図15】

メモリデータ取得の動作を示すフローチャート



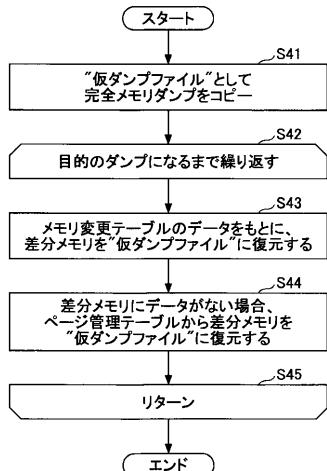
【図16】

メモリダンプ復元ポイントの作成動作を示すフローチャート



【図17】

メモリダンプの復元動作を示すフローチャート



フロントページの続き

(72)発明者 西原 健太郎
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

(72)発明者 長谷川 裕毅
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

(72)発明者 渡邊 和也
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

(72)発明者 今枝 一英
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

(72)発明者 山本 博之
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

(72)発明者 鈴木 保利
神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

審査官 大塚 俊範

(56)参考文献 特開平10-078884(JP,A)
特開2009-245216(JP,A)
特開平09-081405(JP,A)
特開2013-120524(JP,A)

(58)調査した分野(Int.Cl., DB名)

G 06 F 11/07
G 06 F 11/30 - 11/34