

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2009-26117

(P2009-26117A)

(43) 公開日 平成21年2月5日(2009.2.5)

(51) Int.Cl.

G06F 9/46 (2006.01)

F I

G06F 9/46 350

テーマコード (参考)

審査請求 未請求 請求項の数 33 O L (全 45 頁)

(21) 出願番号 特願2007-189336 (P2007-189336)
(22) 出願日 平成19年7月20日 (2007.7.20)

(71) 出願人 000004237
日本電気株式会社
東京都港区芝五丁目7番1号
(74) 代理人 100088959
弁理士 境 廣巳
(72) 発明者 内田 智士
東京都港区芝五丁目7番1号 日本電気株式会社内

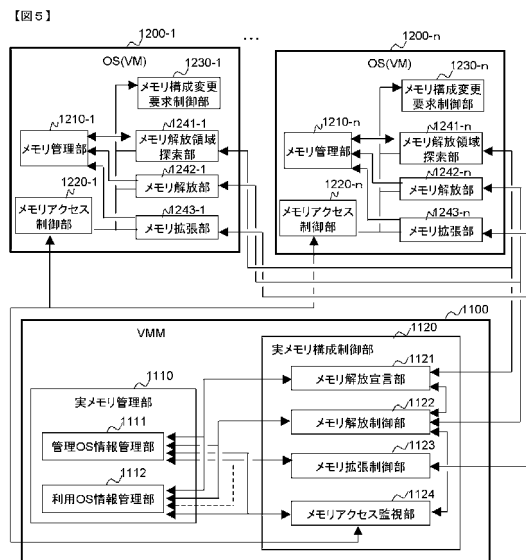
(54) 【発明の名称】 仮想計算機システムにおけるメモリ管理方式及びその方法、およびプログラム

(57) 【要約】

【課題】 仮想計算機システムのメモリ構成変更処理の速度を高速化する。

【解決手段】 メモリ領域に対する権限として管理権と利用権の2種類の権限を設け、実メモリ管理部1110で管理する。第1の仮想計算機のメモリ領域Xを第2の仮想計算機に割り当ててする場合、実メモリ構成制御部1120は、まず管理権だけを移譲し、利用権は第2の仮想計算機が実際にメモリ領域Xをアクセスした時点で移譲する。利用権の移譲に際しては、第1の仮想計算機がメモリ領域Xにデータを書き込んでいればページアウト/スワップ処理が必要となって時間がかかるが、管理権だけの移譲に際してはそのような処理は不要である。このため、管理権の移譲は高速に行える。管理権を持つ第2の仮想計算機は、何時でも利用権を獲得してメモリ領域Xに対してアクセスすることができるので、実質的にメモリ領域の割り当てを受けた状態と同等である。

【選択図】 図5



【特許請求の範囲】**【請求項 1】**

実計算機上に複数の仮想計算機を実現し、各仮想計算機のメモリに実計算機のメモリの一部を割り当てる仮想計算機システムにおけるメモリ管理方式において、

前記複数の仮想計算機を制御する仮想計算機制御部に実メモリ管理部と実メモリ構成制御部とを備え、

前記実メモリ管理部は、実計算機の所定サイズのメモリ領域毎に、そのメモリ領域の管理権を持つ仮想計算機とそのメモリ領域の利用権を持つ仮想計算機とを管理するものであり、

前記実メモリ構成制御部は、メモリ領域の管理権を持つ仮想計算機から当該メモリ領域の管理権を剥奪するメモリ解放宣言手段と、仮想計算機に対してメモリ領域の管理権を付与した時点で当該メモリ領域の増加処理の完了を仮想計算機に通知するメモリ拡張制御手段と、メモリ領域の利用権を持つ仮想計算機からメモリ領域を解放させると共に利用権を剥奪するメモリ解放制御手段と、メモリ領域へのアクセスを利用権を持つ仮想計算機に制限する手段であって、利用権を持たないが管理権を持つ仮想計算機からのメモリ領域に対するメモリアクセス時、そのメモリ領域の利用権を持つ他の仮想計算機が存在していれば、前記メモリ解放制御手段によって当該他の仮想計算機から当該メモリ領域を解放させると共に利用権を剥奪した後に、要求元の仮想計算機に利用権を付与するメモリアクセス監視手段とを有することを特徴とする仮想計算機システムにおけるメモリ管理方式。

【請求項 2】

前記メモリ解放宣言手段は、管理権を剥奪したメモリ領域の利用権が管理権を剥奪した仮想計算機以外の他の仮想計算機に付与されている場合、前記メモリ解放制御手段によって、管理権を剥奪した仮想計算機から当該メモリ領域を解放させることを特徴とする請求項 1 記載の仮想計算機システムにおけるメモリ管理方式。

【請求項 3】

前記メモリ解放宣言手段は、管理権を剥奪したメモリ領域の利用権が管理権を剥奪した仮想計算機以外の他の仮想計算機に付与されている場合、前記メモリ解放制御手段によって当該他の仮想計算機から当該メモリ領域を解放させると共に利用権を剥奪した後に、管理権を剥奪した仮想計算機に利用権を付与することを特徴とする請求項 1 記載の仮想計算機システムにおけるメモリ管理方式。

【請求項 4】

前記メモリアクセス監視手段は、管理権を持たず利用権を持つ仮想計算機のメモリ領域に対するメモリアクセスの統計値が利用権を剥奪する条件を満たした場合に、前記メモリ解放制御手段によって当該仮想計算機から当該メモリ領域を解放させると共に利用権を剥奪することを特徴とする請求項 1 ないし 3 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方式。

【請求項 5】

前記メモリアクセス監視手段は、前記条件を満たすか否かの判定を仮想計算機からメモリ領域に対するメモリアクセス毎に行い、前記条件を満たしたことによってメモリ領域の解放を行った場合には、ページアウト/スワップアウトされた領域へのアクセスに変更するためにメモリ再アクセス要求を起こさせる結果を生成することを特徴とする請求項 4 記載の仮想計算機システムにおけるメモリ管理方式。

【請求項 6】

前記仮想計算機は、自仮想計算機のメモリ領域の管理権が剥奪された時点で当該メモリ領域に関して自仮想計算機で保有するメモリ情報の一部を当該メモリ領域が解放された後の状態に合うように変更し、当該メモリ領域の利用権が剥奪された時点でメモリ情報の残りの部分を当該メモリ領域が解放された後の状態に合うように変更することを特徴とする請求項 1 ないし 5 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方式。

【請求項 7】

前記仮想計算機は、自仮想計算機のメモリ領域の管理権が剥奪された時点で当該メモリ

領域に対応するメモリ利用形態を変更することを特徴とする請求項 1 ないし 6 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方式。

【請求項 8】

前記仮想計算機に、自仮想計算機に対するメモリ構成変更要求を制御するメモリ構成変更要求制御手段を備え、

前記仮想計算機制御部に、仮想計算機システムに対するメモリ再構成要求を解析し、前記実メモリ管理部で管理される情報によって把握される現在のメモリ構成と比較して、個々の仮想計算機に対するメモリ構成変更要求を生成するメモリ再構成要求解析手段を備えることを特徴とする請求項 1 ないし 7 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方式。

10

【請求項 9】

前記メモリ拡張制御手段は、仮想計算機に対して管理権を付与するメモリ領域を、前記実メモリ管理部を参照して、管理権が付与されていないメモリ領域の中から選択することを特徴とする請求項 1 ないし 8 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方式。

【請求項 10】

前記メモリ拡張制御手段は、仮想計算機に対して管理権を付与するメモリ領域を、前記実メモリ管理部を参照して、管理権および利用権が付与されていないメモリ領域の中から選択することを特徴とする請求項 1 ないし 8 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方式。

20

【請求項 11】

前記メモリ拡張制御手段は、仮想計算機に対して管理権を付与するメモリ領域を、前記実メモリ管理部を参照して、利用権が付与されていないメモリ領域から優先的に選択することを特徴とする請求項 1 ないし 8 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方式。

【請求項 12】

前記仮想計算機は、自仮想計算機のメモリに関する情報を管理するメモリ管理手段と、実計算機のメモリとの対応付けが必要なメモリアクセスに関して利用権の有無を前記メモリアクセス監視部へ問い合わせるメモリアクセス制御手段と、メモリ削減要求に応じて解放に適するメモリ領域を決定し、該決定したメモリ領域を前記メモリ解放宣言部に通知するメモリ解放領域探索手段と、前記メモリ解放制御手段からの要求に従って解放するメモリ領域に関するメモリ情報を前記メモリ管理手段から削除するメモリ解放手段と、拡張するメモリ領域の取得を前記メモリ拡張制御手段に対して要求し、取得したメモリ領域に関するメモリ情報を前記メモリ管理手段に追加するメモリ拡張手段とを備えることを特徴とする請求項 1 ないし 11 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方式。

30

【請求項 13】

実計算機上に複数の仮想計算機を実現し、各仮想計算機のメモリに実計算機のメモリの一部を割り当てる仮想計算機システムであって、実計算機の所定サイズのメモリ領域毎に、そのメモリ領域の管理権を持つ仮想計算機とそのメモリ領域の利用権を持つ仮想計算機とを管理する実メモリ管理部を有する仮想計算機システムにおけるメモリ管理方法であって、

40

a) メモリ領域の管理権を持つ仮想計算機からメモリ領域の管理権を剥奪するステップと、

b) 仮想計算機から管理権を剥奪したメモリ領域の管理権を別の仮想計算機に付与し、メモリ領域の増加処理の完了を当該別の仮想計算機に通知するステップと、

c) 利用権を持たないが管理権を持つ仮想計算機からのメモリ領域に対するメモリアクセス時、当該メモリ領域の利用権を持つ他の仮想計算機が存在していれば、当該他の仮想計算機から当該メモリ領域を解放させると共に利用権を剥奪した後に、要求元の仮想計算機に利用権を付与してメモリアクセスを許可するステップとを含むことを特徴とする仮想

50

計算機システムにおけるメモリ管理方法。

【請求項 14】

前記ステップ a)において、管理権を剥奪したメモリ領域の利用権が管理権を剥奪した仮想計算機以外の他の仮想計算機に付与されている場合、管理権を剥奪した仮想計算機から当該メモリ領域を解放させることを特徴とする請求項 13 記載の仮想計算機システムにおけるメモリ管理方法。

【請求項 15】

前記ステップ a)において、管理権を剥奪したメモリ領域の利用権が管理権を剥奪した仮想計算機以外の他の仮想計算機に付与されている場合、当該他の仮想計算機から当該メモリ領域を解放させると共に利用権を剥奪した後に、管理権を剥奪した仮想計算機に利用権を付与することを特徴とする請求項 13 記載の仮想計算機システムにおけるメモリ管理方法。

10

【請求項 16】

d) 管理権を持たず利用権を持つ仮想計算機のメモリ領域に対するメモリアクセス時、当該メモリ領域に対する当該仮想計算機からのメモリアクセスの統計値が利用権を剥奪する条件を満たした場合に、当該仮想計算機から当該メモリ領域を解放させると共に利用権を剥奪することを特徴とする請求項 13 ないし 15 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方法。

【請求項 17】

前記ステップ d)において、前記条件を満たしたことによってメモリ領域の解放を行った場合には、ページアウト/スワップアウトされた領域へのアクセスに変更するためにメモリ再アクセス要求を起こさせる結果を生成することを特徴とする請求項 16 記載の仮想計算機システムにおけるメモリ管理方法。

20

【請求項 18】

前記仮想計算機は、自仮想計算機のメモリ領域の管理権が剥奪された時点で当該メモリ領域に関して自仮想計算機で保有するメモリ情報の一部を当該メモリ領域が解放された後の状態に合うように変更し、当該メモリ領域の利用権が剥奪された時点でメモリ情報の残りの部分を当該メモリ領域が解放された後の状態に合うように変更することを特徴とする請求項 13 ないし 17 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方法。

30

【請求項 19】

前記仮想計算機は、自仮想計算機のメモリ領域の管理権が剥奪された時点で当該メモリ領域に対応するメモリ利用形態を変更することを特徴とする請求項 13 ないし 18 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方法。

【請求項 20】

e) 前記複数の仮想計算機を制御する仮想計算機制御部が、仮想計算機システムに対するメモリ再構成要求を解析し、前記実メモリ管理部で管理される情報によって把握される現在のメモリ構成と比較して、個々の仮想計算機に対するメモリ構成変更要求を生成するステップと、

f) 前記仮想計算機が、自仮想計算機に対するメモリ構成変更要求を制御するステップを含むことを特徴とする請求項 13 ないし 19 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方法。

40

【請求項 21】

仮想計算機に対して管理権を付与するメモリ領域を、前記実メモリ管理部を参照して、管理権が付与されていないメモリ領域の中から選択することを特徴とする請求項 13 ないし 20 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方法。

【請求項 22】

仮想計算機に対して管理権を付与するメモリ領域を、前記実メモリ管理部を参照して、管理権および利用権が付与されていないメモリ領域の中から選択することを特徴とする請求項 13 ないし 20 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方法。

50

【請求項 2 3】

仮想計算機に対して管理権を付与するメモリ領域を、前記実メモリ管理部を参照して、利用権が付与されていないメモリ領域から優先的に選択することを特徴とする請求項 1 3 ないし 2 0 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方法。

【請求項 2 4】

前記仮想計算機は、自仮想計算機のメモリに関する情報を管理するメモリ管理手段と、実計算機のメモリとの対応付けが必要なメモリアクセスに関して利用権の有無を前記メモリアクセス監視部へ問い合わせるメモリアクセス制御手段と、メモリ削減要求に応じて解放に適するメモリ領域を決定し、該決定したメモリ領域を前記メモリ解放宣言部に通知するメモリ解放領域探索手段と、前記メモリ解放制御手段からの要求に従って解放するメモリ領域に関するメモリ情報を前記メモリ管理手段から削除するメモリ解放手段と、拡張するメモリ領域の取得を前記メモリ拡張制御手段に対して要求し、取得したメモリ領域に関するメモリ情報を前記メモリ管理手段に追加するメモリ拡張手段とを備えることを特徴とする請求項 1 3 ないし 2 3 の何れか 1 項に記載の仮想計算機システムにおけるメモリ管理方法。

10

【請求項 2 5】

実計算機上に複数の仮想計算機を実現し、各仮想計算機のメモリに実計算機のメモリの一部を割り当てる仮想計算機システムであって、実計算機の所定サイズのメモリ領域毎に、そのメモリ領域の管理権を持つ仮想計算機とそのメモリ領域の利用権を持つ仮想計算機とを管理する実メモリ管理部を備えた仮想計算機システムにおける前記複数の仮想計算機を制御する仮想計算機制御部を構成するコンピュータを、

20

メモリ領域の管理権を持つ仮想計算機から当該メモリ領域の管理権を剥奪するメモリ解放宣言手段と、仮想計算機に対してメモリ領域の管理権を付与した時点で当該メモリ領域の増加処理の完了を仮想計算機に通知するメモリ拡張制御手段と、メモリ領域の利用権を持つ仮想計算機からメモリ領域を解放させると共に利用権を剥奪するメモリ解放制御手段と、メモリ領域へのアクセスを利用権を持つ仮想計算機に制限する手段であって、利用権を持たないが管理権を持つ仮想計算機からのメモリ領域に対するメモリアクセス時、そのメモリ領域の利用権を持つ他の仮想計算機が存在していれば、前記メモリ解放制御手段によって当該他の仮想計算機から当該メモリ領域を解放させると共に利用権を剥奪した後に、要求元の仮想計算機に利用権を付与するメモリアクセス監視手段として機能させるためのプログラム。

30

【請求項 2 6】

前記メモリ解放宣言手段は、管理権を剥奪したメモリ領域の利用権が管理権を剥奪した仮想計算機以外の他の仮想計算機に付与されている場合、前記メモリ解放制御手段によって、管理権を剥奪した仮想計算機から当該メモリ領域を解放させることを特徴とする請求項 2 5 記載のプログラム。

【請求項 2 7】

前記メモリ解放宣言手段は、管理権を剥奪したメモリ領域の利用権が管理権を剥奪した仮想計算機以外の他の仮想計算機に付与されている場合、前記メモリ解放制御手段によって当該他の仮想計算機から当該メモリ領域を解放させると共に利用権を剥奪した後に、管理権を剥奪した仮想計算機に利用権を付与することを特徴とする請求項 2 5 記載のプログラム。

40

【請求項 2 8】

前記メモリアクセス監視手段は、管理権を持たず利用権を持つ仮想計算機のメモリ領域に対するメモリアクセスの統計値が利用権を剥奪する条件を満たした場合に、前記メモリ解放制御手段によって当該仮想計算機から当該メモリ領域を解放させると共に利用権を剥奪することを特徴とする請求項 2 5 ないし 2 7 の何れか 1 項に記載のプログラム。

【請求項 2 9】

前記メモリアクセス監視手段は、前記条件を満たすか否かの判定を仮想計算機からメモリ領域に対するメモリアクセス毎に行い、前記条件を満たしたことによってメモリ領域の

50

解放を行った場合には、ページアウト / スワップアウトされた領域へのアクセスに変更するためにメモリ再アクセス要求を起こさせる結果を生成することを特徴とする請求項 28 記載のプログラム。

【請求項 30】

前記コンピュータをさらに、仮想計算機システムに対するメモリ再構成要求を解析し、前記実メモリ管理部で管理される情報によって把握される現在のメモリ構成と比較して、個々の仮想計算機に対するメモリ構成変更要求を生成して、前記仮想計算機に出力するメモリ再構成要求解析手段として機能させることを特徴とする請求項 25 ないし 29 の何れか 1 項に記載のプログラム。

【請求項 31】

前記メモリ拡張制御手段は、仮想計算機に対して管理権を付与するメモリ領域を、前記実メモリ管理部を参照して、管理権が付与されていないメモリ領域の中から選択することを特徴とする請求項 25 ないし 30 の何れか 1 項に記載のプログラム。

【請求項 32】

前記メモリ拡張制御手段は、仮想計算機に対して管理権を付与するメモリ領域を、前記実メモリ管理部を参照して、管理権および利用権が付与されていないメモリ領域の中から選択することを特徴とする請求項 25 ないし 30 の何れか 1 項に記載のプログラム。

【請求項 33】

前記メモリ拡張制御手段は、仮想計算機に対して管理権を付与するメモリ領域を、前記実メモリ管理部を参照して、利用権が付与されていないメモリ領域から優先的に選択することを特徴とする請求項 25 ないし 30 の何れか 1 項に記載のプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、1台の計算機上で複数のOSを同時に動作させる仮想計算機システムにおけるメモリ管理方式に関し、特に、仮想計算機に割り当てるメモリを動的に変更可能なメモリ管理方式に関する。

【背景技術】

【0002】

1台の計算機上で複数のオペレーティングシステム(OS)を動作させる仮想計算機技術が知られている。

【0003】

仮想計算機技術は、CPU、メモリ、ストレージ、ネットワークなど様々な計算機資源の利用を時間や資源の区切りで分割し、分割された資源利用区画を元に、複数の仮想的な計算機環境 (VM) を構築し、複数のOSを動作させる。

【0004】

仮想計算機に対して利用期間の割当などを制御し、それぞれの仮想的な計算機資源への要求と実際の物理資源への処理との対応付けなどを行って、全体の仮想計算機環境を制御している処理部を仮想計算機モニタ(VMM)やハイパバイザ(Hypervisor)などと呼ぶ。

【0005】

VMは、一般に、VMMにより実現された仮想計算機空間のことを指す。

【0006】

実際に仮想的な計算機環境を実現するために制御しているのはVMMである。

【0007】

VMMのうち、個々の仮想的なハードウェアアーキテクチャをエミュレーションする部分をVMと呼ぶこともある。

【0008】

またVMは、機能がOSと密接に関わりを持つ場合もある。

【0009】

例えば、後述において、本件の主題となるメモリ再構成処理において、メモリを増減さ

10

20

30

40

50

せるという処理は、VM単体のメモリエミュレーションの管理部分を修正するという機能とOSのHot Add、Hot RemoveなどのOSメモリ管理における増減機能を組み合わせる方法と、Xenでのballoon機能などのようにOSと結びつけてOSと一体化した機能として実現される方法などが存在する。

【0010】

本件では、これらを踏まえ、OSとVMに含まれる機能を明確に分類せず、特に明記しない場合は、OSの機能として記載することにする。

【0011】

ただし、前述のように必要に応じてVMとして機能を提供することが可能である。

【0012】

メモリ管理における仮想計算機技術による仮想化の例を図1に示す。

【0013】

VMMは、実装された物理メモリ全体を把握し、管理する。

【0014】

この物理メモリのアドレス空間を実メモリ空間と呼ぶこととする。

【0015】

VMMは、実メモリ空間を分割し、各OSへ割当ててゐる。

【0016】

OSに割当てられたメモリ空間は、OSからしてみると一連の実メモリ空間を認識しているときと同様に動作する。

【0017】

すなわち、OSは、メモリ空間が0番地から割当てられているように認識する。

【0018】

VMMにより分割され、OSから見える物理アドレス空間をマシンアドレス空間と呼ぶこととする。

【0019】

さらに、OSはプログラム(AP)ごとにアドレス空間を変換して提供している。これをプロセス空間と呼ぶこととする。

【0020】

すなわち、VMMが実メモリ空間とマシンメモリ空間の変換テーブルを管理し、複数のマシンメモリ空間を作成できるため、複数のOSのメモリ空間が実現できる。

【0021】

仮想計算機技術と同様の技術は、物理的な装置単位に計算資源を予め区画分割する物理パーティション、物理パーティションの概念を論理的な計算資源に適用した論理パーティション技術としても知られている。

【0022】

パーティション技術により、分けられた計算機環境を論理区画と呼び、論理区画を作成するための制御部をハイパバイザと呼ぶ。

【0023】

これらの技術も仮想的な計算機環境を複数作成する技術であり、仮想計算機技術として同一視することができる。

【0024】

以後、VMと区画、VMMとハイパバイザをそれぞれ同一視し、主に、VMとVMMとして説明する。

【0025】

ただし、従来技術の紹介においては、引用参考文献で利用している例・用語を用いて説明する。

【0026】

前記仮想計算機環境では、時間や区画に対して配分や割当を変更するだけで仮想計算機環境を変更できるという利点を活かして、計算機資源を有効活用することが行われる。

10

20

30

40

50

【 0 0 2 7 】

すなわち、OS上で動作しているアプリケーションの要求に合わせて動的に仮想計算機資源の配分や区画を変更し、仮想計算機環境を動的に再構成することが行われる。

【 0 0 2 8 】

本発明は、この仮想計算機環境の動的な再構成において、メモリ資源の再構成処理に関する。

【 0 0 2 9 】

仮想計算機環境において、メモリ資源の動的な再構成は、通常、次の手順で行われる。

【 0 0 3 0 】

割当てられているメモリ容量を削減するOSにおいて、データの一貫性を保つために、削減するメモリ領域における使用中メモリ領域を外部記憶装置へスワップアウト/ページアウトにより退避させて、削減するメモリ領域全体を未使用状態に移行させる。

10

【 0 0 3 1 】

メモリの未使用状態への移行後、メモリ領域をOS内部の管理から除外するというメモリ解放処理が行われ、正式にVMM上の管理からメモリ領域の使用状態が未使用状態へ変更される。

【 0 0 3 2 】

割当てられているメモリ容量を削減するOSにおけるメモリ解放後、割当てられているメモリ容量を増加させるOSにおいて、VMMから未使用状態のメモリ領域を必要な容量だけ確保し、確保したメモリ領域をOS内部の管理に追加させるというメモリ拡張処理が行われる。

20

【 0 0 3 3 】

また、仮想計算機として割当てるためのメモリ空間に連続性を必要とするメモリ管理手法では、メモリ容量を増加させる前に、増加させるOSのメモリの使用領域と追加領域を連続させるために、メモリの移動処理が行われる。

【 0 0 3 4 】

メモリの移動処理は、メモリアドレスの連続性を必要とするメモリ管理手法に特化して付随するものであり、メモリ再構成の補助的要素となるので、本件では省略することとする。

【 0 0 3 5 】

メモリ再構成の1手段が、特許文献1に記載されている。

30

【 0 0 3 6 】

特許文献1では、VMと記載されているため、OSでなくVMと記載して説明する。

【 0 0 3 7 】

特許文献1によると、動的メモリ再構成処理におけるメモリ内の各VM領域の移り変わりは、図2に示されるようになる。

【 0 0 3 8 】

ここでは、VM1にアドレス0から2GB、VM2に2GBから1GBのメモリ領域が割当てられているときに、VM1を1GBに削減し、VM2を2GBに増加させる場合を例に説明する。

【 0 0 3 9 】

特許文献1によると、メモリ構成の変更要求は、VMMに設定され、VMMから各VMへ対応した変更要求を発行する。

40

【 0 0 4 0 】

図2の例では、VMMが全体の構成変更要求イベントを受取ると、まずVM1において2GBから1GBへ削減する要求が発行される。

【 0 0 4 1 】

VM1は、前記削減要求に従い、自分が管理しているメモリ領域をアドレス値の高いメモリ（高位アドレス）空間から、削除要求で要求されているメモリ容量分だけ、管理から排除、すなわち、未接続状態に変更する。

【 0 0 4 2 】

50

このとき、当該メモリ領域が使用中である場合、スワップアウト／ページアウトによりメモリ内容を外部記憶装置などへ退避させる必要がある。

【 0 0 4 3 】

前記の動作により、当該メモリ領域はVMから使用されない未使用状態になり、図2(a)の初期状態から図2(b)の状態と変更される。

【 0 0 4 4 】

次に、VMMは、VM2の高位アドレス側に未使用領域が存在しないため、VM2のメモリ領域を1GBからの未使用領域に移動した後、VM2に2GBから1GB分のメモリ領域の増加を要求する。ここで、メモリ移動を要したのは、例えば、実メモリ空間とマシンアドレス空間の対応付けが1つのレジスタの加減算でのみで計算されるような、メモリ空間の連続性が必要となるメモリ管理機構を用いているためである。

10

【 0 0 4 5 】

VM2は、前記増加要求に従い、自分が管理しているメモリ領域の高位アドレスに1GBのメモリ空間を接続状態に変更し拡張する。

【 0 0 4 6 】

最後に、VMMは、それぞれのVMをCPUにディスパッチすることで、メモリ構成変更が終了し、図2(c)の状態となる。

【 0 0 4 7 】

このとき、スワップアウト／ページアウトのような外部記憶への退避処理は、通常プログラムと比べて、一般的に、高負荷、かつ、時間を要する処理である。

20

【 0 0 4 8 】

そのため、メモリ領域解放処理はこの退避処理に依存して遅延し、その結果、メモリ再構成処理に時間を要するという問題がある。

【 0 0 4 9 】

この問題を解決する一手段が、特許文献2に開示されている。

【 0 0 5 0 】

図3に特許文献2の構成図を示す。

【 0 0 5 1 】

特許文献2は、論理パーティショニング技術であり、上記のVMの機能に相当する機能（アドレス管理機能）はOSの機能として提供されている。

30

【 0 0 5 2 】

特許文献2の説明は、OSと記載して説明する。

【 0 0 5 3 】

特許文献2によると、資源一時切り離し制御部114により、あるOSからメモリ領域の切り離しが宣言されると、メモリ領域を直接ユーザが使用できない一時使用可能領域に変更する。一時使用可能領域とは、予約された特殊な領域であり、通常、OSにメモリ割当てができない。これに対して、OSにメモリ割当てが可能で、活用できる通常の領域をユーザ使用可能領域という。一時使用可能領域を、データを完全に解放するまで使用中状態であり、新規割り当て不可であるが、割当て領域数に数えない領域として用いている。また、資源切り離し制御部115により、OS内部のメモリ管理情報から削除されたときに真にメモリ領域の解放を行いハイパバイザから情報が未使用状態に変更される。資源接続制御部116により、切り離し宣言され、実際に切り離し中であるメモリ領域を含め、空きメモリ領域を別のOSへ割り当てることが可能になる。

40

【 0 0 5 4 】

一時使用可能領域を設け、ユーザ使用可能領域と分けて管理し、ユーザ使用可能領域において領域解放時に一時使用可能領域の未使用領域と切り替えて割当てを制御できる機構を導入することにより、スワップアウト／ページアウトなど切り離しに要する時間を無視して、別のOSへメモリ資源を割り当てることが可能とし、メモリ再構成の速度を向上させることが可能となる。

【 0 0 5 5 】

50

また、スワップアウト / ページアウトなどの処理は、切り離しているOSのシステムからは参照可能状態にあるため、負荷を下げながら他に負荷影響を与えないように実行することが可能になる。

【 0 0 5 6 】

図4に特許文献2の計算機資源動的割り当て変更処理の流れをシーケンス図で示す。

【 0 0 5 7 】

特許文献2によると、メモリ構成の変更要求は、コマンドという形で、各OSへ発行される。

【 0 0 5 8 】

論理区画 1 上でのメモリ削減要求のコマンドが投入されると(201)、論理区画1からハイパバイザへメモリ領域の切り離し開始通知が通知される(202)。

【 0 0 5 9 】

ハイパバイザは、当該メモリ領域を一時使用可能領域として扱うように管理情報を変更する(203)。

【 0 0 6 0 】

このとき、当該メモリ領域分が一時使用可能領域になった分だけ、一時使用可能領域からユーザ使用可能領域に未使用領域が移されることになる。

【 0 0 6 1 】

ハイパバイザは、論理区画1へメモリ領域の切り離しを受け付けたことを応答し(204)、論理区画1は、コマンド受付結果をオペレータに表示する(205)。

【 0 0 6 2 】

次に、論理区画2にて、メモリ増加要求のコマンドが投入されると(206)、論理区画2からハイパバイザへメモリ領域の接続要求が通知される(207)。

【 0 0 6 3 】

ハイパバイザは、ユーザ使用可能領域の未接続状態にある領域から必要分を確保し、管理情報を接続状態、すなわち、増加させるOSの利用状態へ変更する(208)。

【 0 0 6 4 】

ハイパバイザは、論理区画2へメモリ領域の接続が完了したことを応答し(209)、論理区画2は、コマンド完了結果をオペレータに表示する(210)。

【 0 0 6 5 】

論理区画2のメモリ増加の間、平行して、論理区画1はページアウト処理を実行することができる。また、このとき、ページアウト処理の速度を規制することで、負荷の上昇も抑えることが可能になる。

【 0 0 6 6 】

論理区画1でページアウト処理により解放メモリ領域のデータ退避が完了したとき、論理区画1はメモリ領域の切り離し要求をハイパバイザへ通知する(211)。

【 0 0 6 7 】

ハイパバイザは、管理情報の当該メモリ領域を使用状態から未使用状態へ変更し、管理情報を初期化し(212)、切り離しが完了したことを論理区画1へ通知する(213)。

【 0 0 6 8 】

論理区画 1 は、メモリ削減要求のコマンドが完了したことをオペレータへ表示する。

【 0 0 6 9 】

以上の流れにより、解放中のメモリ領域を使用領域と切り離して制御することができ、メモリ再構成時の速度を向上させることができる。

【 0 0 7 0 】

【特許文献 1】特開平6-110715号公報

【特許文献 2】特開2005-148875号公報

【発明の開示】

【発明が解決しようとする課題】

【 0 0 7 1 】

10

20

30

40

50

特許文献2に示されるメモリ動的変更方法には以下の問題がある。

【0072】

動的なメモリ再構成処理の処理時間短縮が図れるものの、メモリ変更量に相当する一時使用可能領域を必要とするため、メモリ領域の常時活用がユーザ使用可能領域分に制限されてしまうことがある。

【0073】

一時使用可能領域分は、活用したい要求があっても使用することができない。そのため、メモリ再構成要求が頻繁に起きない場合、発生するメモリ再構成要求のために、ほとんどの時間、一時使用可能領域分を未使用状態として待機させておくことになり有効的に活用されない。

【0074】

また、1回のメモリ変更量に相当する一時使用可能領域がないと特許文献2の手法は有効に機能しない。

【0075】

例えば、一時使用可能領域が512MB分確保しており、他の領域はユーザ使用可能領域として利用している状態で1GBのメモリ容量変更を使用とした場合、メモリ未使用領域容量が足りないため、増加側の論理区画はコマンド失敗になる。

【0076】

増加要求が成功するためには、削減側の論理区画で512MB分の解放完了を待つ必要があるが、メモリの解放は削減側のOSの負荷に応じて自由に作業速度を決定できるため、いつ終わるか明示的でない。

【0077】

また、一度に512MB分のメモリ解放処理を実行したとしても、メモリ領域解放処理はこの回避処理に依存して遅延し、その結果、メモリ再構成処理に時間を要するという問題を復元させる。

【0078】

このように、一時使用可能領域の容量設定がメモリ変更時の動作に大きな影響を持つようになる。

【0079】

[発明の目的]

本発明の目的は、仮想計算機に対するメモリの動的割付処理において、メモリ領域を有効活用しつつ、メモリ再構成処理に要する時間を短縮することにある。

【課題を解決するための手段】

【0080】

本発明の仮想計算機システムにおけるメモリ管理方式は、実計算機上に複数の仮想計算機を実現し、各仮想計算機のメモリに実計算機のメモリの一部を割り当てる仮想計算機システムにおけるメモリ管理方式において、前記複数の仮想計算機を制御する仮想計算機制御部に実メモリ管理部と実メモリ構成制御部とを備え、前記実メモリ管理部は、実計算機の所定サイズのメモリ領域毎に、そのメモリ領域の管理権を持つ仮想計算機とそのメモリ領域の利用権を持つ仮想計算機とを管理するものであり、前記実メモリ構成制御部は、メモリ領域の管理権を持つ仮想計算機から当該メモリ領域の管理権を剥奪するメモリ解放宣言手段と、仮想計算機に対してメモリ領域の管理権を付与した時点で当該メモリ領域の増加処理の完了を仮想計算機に通知するメモリ拡張制御手段と、メモリ領域の利用権を持つ仮想計算機からメモリ領域を解放させると共に利用権を剥奪するメモリ解放制御手段と、メモリ領域へのアクセスを利用権を持つ仮想計算機に制限する手段であって、利用権を持たないが管理権を持つ仮想計算機からのメモリ領域に対するメモリアクセス時、そのメモリ領域の利用権を持つ他の仮想計算機が存在していれば、前記メモリ解放制御手段によって当該他の仮想計算機から当該メモリ領域を解放させると共に利用権を剥奪した後に、要求元の仮想計算機に利用権を付与するメモリアクセス監視手段とを有する。

【0081】

本発明の仮想計算機システムにおけるメモリ管理方法は、実計算機上に複数の仮想計算機を実現し、各仮想計算機のメモリに実計算機のメモリの一部を割り当てる仮想計算機システムであって、実計算機の所定サイズのメモリ領域毎に、そのメモリ領域の管理権を持つ仮想計算機とそのメモリ領域の利用権を持つ仮想計算機とを管理する実メモリ管理部を有する仮想計算機システムにおけるメモリ管理方法であって、a)メモリ領域の管理権を持つ仮想計算機からメモリ領域の管理権を剥奪するステップと、b)仮想計算機から管理権を剥奪したメモリ領域の管理権を別の仮想計算機に付与し、メモリ領域の増加処理の完了を当該別の仮想計算機に通知するステップと、c)利用権を持たないが管理権を持つ仮想計算機からのメモリ領域に対するメモリアクセス時、当該メモリ領域の利用権を持つ他の仮想計算機が存在していれば、当該他の仮想計算機から当該メモリ領域を解放させると共に利用権を剥奪した後に、要求元の仮想計算機に利用権を付与してメモリアクセスを許可するステップとを含む。

【0082】

本発明のプログラムは、実計算機上に複数の仮想計算機を実現し、各仮想計算機のメモリに実計算機のメモリの一部を割り当てる仮想計算機システムであって、実計算機の所定サイズのメモリ領域毎に、そのメモリ領域の管理権を持つ仮想計算機とそのメモリ領域の利用権を持つ仮想計算機とを管理する実メモリ管理部を備えた仮想計算機システムにおける前記複数の仮想計算機を制御する仮想計算機制御部を構成するコンピュータを、メモリ領域の管理権を持つ仮想計算機から当該メモリ領域の管理権を剥奪するメモリ解放宣言手段と、仮想計算機に対してメモリ領域の管理権を付与した時点で当該メモリ領域の増加処理の完了を仮想計算機に通知するメモリ拡張制御手段と、メモリ領域の利用権を持つ仮想計算機からメモリ領域を解放させると共に利用権を剥奪するメモリ解放制御手段と、メモリ領域へのアクセスを利用権を持つ仮想計算機に制限する手段であって、利用権を持たないが管理権を持つ仮想計算機からのメモリ領域に対するメモリアクセス時、そのメモリ領域の利用権を持つ他の仮想計算機が存在していれば、前記メモリ解放制御手段によって当該他の仮想計算機から当該メモリ領域を解放させると共に利用権を剥奪した後に、要求元の仮想計算機に利用権を付与するメモリアクセス監視手段として機能させる。

【発明の効果】

【0083】

本発明によれば、メモリ再構成処理に要する時間を短縮することができる。その理由は、或る仮想計算機に割り当てられているメモリ領域を別の仮想計算機に割り当てる場合、当該メモリ領域の管理権を移譲するだけでメモリ再構成処理が見かけ上完了し、メモリ解放処理を必要とするメモリ領域の利用権の移譲は、管理権の移譲を受けた仮想計算機が当該メモリ領域をアクセスした際に行われるためである。

【0084】

さらに、特許文献2におけるような予約された予備領域を必要としないため、全メモリ領域を有効に活用することができる。

【0085】

さらに、メモリ解放処理に伴うページアウト/スワップアウト処理などのデータ退避処理は、再構成するメモリ全体の容量と関係なく、ページなどの単位で実施されるため、メモリ再構成のための処理における負荷分散が行える。そのため、通常の業部プロセスへの高負荷処理による影響も削減することができる。

【発明を実施するための最良の形態】

【0086】

本発明の実施の形態を詳細に説明するのに先立って、その概要を説明する。

【0087】

本発明は、実計算機のメモリ領域に対する仮想計算機の権限として、管理権と利用権という2種類の権限を創設する。

【0088】

利用権はメモリ領域に対してアクセスすることができる権利であり、或るメモリ領域の

利用権を持つ仮想計算機は、そのメモリ領域に対してアクセスすることが許される。但し、当該メモリ領域の管理権を他の仮想計算機が持っている場合、当該他の仮想計算機が当該メモリ領域に対してアクセスした時点で、利用権を手放さなければならない。これに対して、管理権は利用権よりも強い権利であり、管理権を持つ仮想計算機は、何時でも利用権を獲得してメモリ領域に対してアクセスすることができる。管理権を持つ仮想計算機が利用権を獲得するために特別な手続きは不要であり、メモリ領域を単にアクセスするだけで良い。

【0089】

本発明は、メモリ領域に対する権限である管理権と利用権とをそれぞれ独立に移譲することによって、メモリ領域の再割り当てを実現する。具体的には、或る仮想計算機 A に割り当てられているメモリ領域 X を、別の仮想計算機 B に割り当てる場合、メモリ領域 X の管理権および利用権を一度に移譲するのではなく、まず管理権だけを移譲し、利用権は仮想計算機 B が実際にメモリ領域 X をアクセスした時点で移譲する。利用権の移譲に際しては、仮想計算機 A がメモリ領域 X にデータを書き込んでいればページアウト/スワップ処理が必要となって時間がかかるが、管理権だけの移譲に際してはそのような処理は不要である。このため、管理権の移譲は高速に行える。そして、管理権を持つ仮想計算機は、何時でも利用権を獲得してメモリ領域に対してアクセスすることができるので、実質的にメモリ領域の割り当てを受けた状態と同等である。これによって、メモリ再構成処理に要する時間を短縮することができる。

【0090】

仮想計算機 B がメモリ領域 X を最初にアクセスするまでの間、メモリ領域 X の利用権は仮想計算機 A が保有しているため、仮想計算機 A はなおも継続してメモリ領域 X をアクセスすることが許される。ここで、メモリ領域 X の管理権を得た仮想計算機 B がメモリ領域 X を一度もアクセスすることなくメモリ領域 X の管理権を別の仮想計算機 C に移譲した場合、メモリ領域 X の利用権をどの仮想計算機に設定すべきかについて、以下の 2 通りの方式が考えられる。

【0091】

第 1 の方式は、メモリ領域 X を現在使用している仮想計算機 A に以後も利用権を残して利用を継続させる方式である。この場合、管理権を放棄した仮想計算機 B からメモリ領域 X へのアクセスが生じるのを防ぐために、仮想計算機 B の管理下からメモリ領域 X が解放される。この方式は第 1 の実施の形態で採用している。

【0092】

第 2 の方式は、仮想計算機 B がメモリ領域 X の管理権を得たのはそれを利用するためであり、たまたまメモリ領域 X を一度もアクセスしなかったのは時間的な関係によるものと考え、仮想計算機 C がメモリ領域 X をアクセスするまでの間、仮想計算機 B に利用の機会を残しておく方式である。この場合、仮想計算機 A の管理下からメモリ領域 X が解放されると共にその利用権が仮想計算機 B に移譲される。この方式については第 2 の実施の形態で詳述する。

【0093】

ところで、第 1 および第 2 の方式の何れの方式を採用するにしても、管理権を持つ仮想計算機 C からメモリ領域 X へのアクセスが行われるのを待って、利用権を管理権を持つ仮想計算機 C に移譲する構成では、管理権を持つ仮想計算機 C がメモリ領域 X をアクセスした際にメモリ領域の解放処理が行われるためにオーバーヘッドが大きくなる。そこで、利用権を持つ仮想計算機 A や B のメモリ領域 X に対するアクセスの統計値（例えば、管理権を放棄してからメモリアクセス時間など）を採取し、統計値が所定の条件を満たした場合に、管理権を持つ仮想計算機 C からのメモリ領域 X へのアクセスを待たずに、仮想計算機 A や B からメモリ領域 X を解放させると共にその利用権を剥奪しておく方法が考えられる。この方法については第 3 の実施の形態で詳述する。

【0094】

上述したように、メモリ領域 X に対して管理権を放棄しても利用権を保有している仮想

計算機 A または B は、メモリ領域 X に対してアクセスを続けることができるが、利用権も放棄した場合には、もはやメモリ領域 X に対してアクセスすることはできない。従って、メモリ領域 X の管理権および利用権を放棄した時点で、仮想計算機 A または B におけるメモリ領域 X に関するメモリ情報を更新するのが基本である。しかし、メモリ領域 X を放棄した仮想計算機 A または B は、近い将来、その利用権も放棄しなければならないため、それに備えて、一部のメモリ情報（例えばメモリ総容量）を、管理権を放棄した時点で更新しておいても良い。また、読書可能であったメモリ領域 X を読込専用領域として設定するといったようなメモリ利用形態の変更を、管理権の放棄時に行うようにしても良い。これらについては第 4 の実施の形態で説明する。

【 0 0 9 5 】

10

メモリ構成の変更要求は、オペレータ等から各仮想計算機に個別に与えるようにしても良いし、VMMなどの仮想計算機制御部を通じて各仮想計算機に与えるようにしても良い。前者の方法は第 1 の実施の形態で、後者の方法は第 5 の実施の形態で、それぞれ説明する。

【 0 0 9 6 】

次に本発明の実施の形態について図面を参考にして詳細に説明する。

【 0 0 9 7 】

『第 1 の実施の形態』

図 5 を参照すると、本発明の第 1 の実施の形態は、VM を構築し、複数の OS の動作を制御する VMM1100 と、構築された VM 上で各々動作する OS1200-i (i = 1 ~ n) とから構成される。

20

【 0 0 9 8 】

< VMM1100 >

VMM1100 は、実メモリ空間とマシンアドレス空間との対応関係を制御する実メモリ管理部 1110 と、VM のメモリ再構成を制御する実メモリ構成制御部 1120 とを含む。

【 0 0 9 9 】

< 実メモリ管理部 1110 >

実メモリ管理部 1110 は、メモリ領域に対するメモリ再構成処理後にメモリ領域を割当られ優先権を持つ OS (管理 OS) を判別するための情報を管理する管理 OS 情報管理部 1111 と、メモリ再構成前に利用しており解放処理完了前で、メモリ再構成要求受付後にまだメモリ領域を利用できる OS (利用 OS) を判別するための情報を管理する利用 OS 情報管理部 1112 とを含む。

30

【 0 1 0 0 】

今後、管理 OS はメモリ領域に対して優先権をもつため「管理権を持つ」と言うことにする。管理権とは本来の所有者であるべきことを表し、必要に応じて利用している OS へ解放を要求できる立場にあることを示す。

【 0 1 0 1 】

また、利用 OS はメモリ領域に対して利用することが可能なため「利用権を持つ」と言うことにする。利用権とは、メモリを実際に利用していることを表し、アクセスしたらデータを読込、保存できることを示す。

【 0 1 0 2 】

40

さらに、実メモリ管理部 1110 は、実メモリ空間とマシンアドレス空間の対応関係 (図示せず) を保有し、メモリアクセス時にアドレス変換を行う。

【 0 1 0 3 】

本発明においては、実メモリ管理部 1110 がメモリ管理の最小単位を決定する要素となる。実メモリ管理部 1110 は、ページ単位、一定ページを 1 ブロックとしたブロック単位、モジュール単位などの単位を最小単位としてメモリ管理を行う。

【 0 1 0 4 】

これら、メモリ空間の対応関係、管理 OS 情報、利用 OS 情報は、同時に同じ機構で記憶してもよい。

【 0 1 0 5 】

50

例えば、後述する実計算機システムの実施例を示す図19のように、メモリ領域の対応関係、管理OS情報、利用OS情報をテーブル10により、一括して管理することができる。

【0106】

メモリ空間の対応関係は、メモリの領域を示す識別子で表現できる。例えば、アドレス値で表現することもできるし、特許文献2のようにモジュール番号で表現することもできる。

【0107】

管理OS情報、利用OS情報は、OSを特定できる識別子で表現できる。例えば、マシン名、NetBIOS名、ホスト名、UUID (Universally Unique Identifier)、GUID (Globally Unique Identifier)、IPアドレスやMACアドレス、VMMが管理しているOSのIDやOS名などが挙げられる。 10

【0108】

このとき、未使用や未設定の表現法としては、情報に空情報 (例えば、ヌル(null)) や、OS管理情報として利用しないことを前提に特定の識別子 (例えば、OSのIDとして0など) を利用することができる。

【0109】

メモリ空間の対応関係、利用OS情報、管理OS情報を管理する別の例としては、メモリ管理構造体などにパラメータとして付随することができる。

【0110】

今後、前者のテーブル管理の例を主体にして説明を続ける。 20

【0111】

< 実メモリ構成制御部1120 >

実メモリ構成制御部1120は、メモリ削減要求時に解放メモリ領域における管理権を放棄し一時接続状態を生成するメモリ解放宣言部1121と、真のメモリ解放として実メモリとの接続関係を削除するメモリ解放制御部1122と、メモリ増加要求時に管理権を保有していないメモリ領域を先行してOSにメモリ割当を行うメモリ拡張制御部1123と、メモリ領域の利用時に他の利用状況がないかを判断し、メモリ割当ての許可を判定するメモリアクセス監視部1124とを含む。

【0112】

< メモリ解放宣言部1121 >

メモリ解放宣言部1121は、メモリ領域の管理権放棄を制御する。 30

【0113】

具体的には、メモリ解放宣言部1121は、メモリ解放領域探索部1231- i ($i=1 \sim n$) からメモリ削減により解放させる領域が渡されたなら、管理OS情報管理部1111を介して管理OS情報を未使用状態へ変更させる。

【0114】

解放させるメモリ領域を渡す方法、すなわち、解放領域を宣言する方法としては、メッセージパッシング、システムコール (VMMとの間ではハイパーコールなどと呼ぶ)、遠隔関数呼び出し (RPC) などが挙げられる。

【0115】

今後、その他の情報伝達方法としても同様の表記を用いることとする。 40

【0116】

< メモリ解放制御部1122 >

メモリ解放制御部1122は、メモリの利用権放棄を制御する。

【0117】

具体的には、メモリ解放制御部1122は、メモリアクセス監視部1124からメモリ領域のメモリ解放が要求されたときに、メモリ解放部1232- i ($i=1 \sim n$) へ当該メモリ領域のメモリ解放処理を要求する。

【0118】

メモリ解放制御部1122は、メモリ解放部1232- i ($i=1 \sim n$) から解放したメモリ領域を渡さ 50

れたなら、利用OS情報管理部1112を介して利用OS情報を未使用状態へ変更させる。

【0119】

<メモリ拡張制御部1123>

メモリ拡張制御部1123は、メモリ領域の管理権確保を制御する。

【0120】

具体的には、メモリ拡張制御部1123は、メモリ拡張部1233-i (i=1~n)により、メモリ拡張する領域の追加を要求されたなら、管理OS情報管理部1111を参照して、管理OS情報が未使用状態であるメモリ領域を探索し、適切な領域を拡張メモリ領域として決定し、当該拡張メモリ領域の管理OS情報を拡張OSの識別子に変更し、当該拡張メモリ領域を応答する。

【0121】

拡張メモリ領域の選択基準として、例えば、連続未使用領域の容量を用いる方法がある。例えば、連続未使用選択領域容量が大きい領域を優先して選択する方法、連続未使用領域容量が要求メモリ容量とほぼ等しい領域を優先して選択する方法、連続未使用選択領域容量が小さい領域を優先して選択する方法などある。

【0122】

連続未使用選択領域容量が大きい領域を優先して選択する方法は、割当てる際の割当て領域の数を減らすことが可能となる。

【0123】

連続未使用領域容量が要求メモリ容量とほぼ等しい領域を優先して選択する方法は、割当てる際の割当て領域の数を減らす一方、より大きな連続未使用領域を残すことで、さらに後に起こるメモリ領域変更に同様の領域を割当てられる可能性を残すことができる。

【0124】

連続未使用選択領域容量が小さい領域を優先して選択する方法は、連続未使用領域をできるだけ残すようにすることで、限られた範囲内でのメモリ利用を行うと同時に、後に起こるメモリ領域変更や新規OSの割当てにおいて一度に割当てられる可能性を残すことができる。

【0125】

拡張メモリ領域の選択基準の別の例として、拡張OSの使用メモリ領域情報を用いる方法がある。例えば、拡張OSの使用しているメモリ領域に近い領域、すなわち、拡張OSが使用中のメモリ領域のすぐ側に続く高位アドレス領域を選択する方法である。

【0126】

これは、単一OSにおけるメモリ領域の連続性を保つことが可能になる。

【0127】

拡張メモリ領域の選択基準の別の例として、物理メモリモジュール構成情報を用いる方法がある。例えば、できるだけ複数のメモリモジュールから領域を選択する方法や、できるだけ同じメモリモジュールに割当てる方法がある。

【0128】

できるだけ複数のメモリモジュールから領域を選択する方法は、複数のメモリモジュールと平行にアクセス可能にするため、メモリアクセス速度を向上させるような技術を用いることが可能になったり、RAID5技術のようにパリティ技術を用いることで、対故障性を向上させることが可能になったりする。

【0129】

できるだけ同じメモリモジュールに割当てる方法は、あるOSが割当てられるメモリをできるだけ最小にすることができ、そのメモリモジュールが壊れたときに、動作不良を起こすOSを最小限に食い止めることを可能にできる。

【0130】

また、拡張メモリ領域の選択基準の別の例として、利用OS情報を用いる方法がある。利用OS情報は、利用OS情報管理部1112を介して得ることができる。例えば、利用OSがない領域、すなわち、管理OS情報だけでなく利用OS情報においても未使用状態であるメモリ領域を選択する方法がある。

10

20

30

40

50

【0131】

利用OS情報においても未使用状態であるメモリ領域を選択する方法は、利用OSがメモリ領域を解放するタイミングを作るきっかけを減らすことになり、結果として、利用OSがメモリ領域を確保し続け有効利用することが可能になる。

【0132】

また、前記選択基準、もしくは、前記選択方法を少なくとも2つ以上併用し、組み合わせることによって実現される選択基準、もしくは、選択方法を採用することもできる。

【0133】

メモリ拡張制御部1123は、また、適切な領域を選択する際に、未使用領域が要求する拡張メモリ領域に対して不足する場合、以下のような処置をとることができる。

10

【0134】

未使用領域が不足する際の処置の例として、メモリ拡張制御部1123は要求を受け入れることができないこととして、要求拒否として応答する。

【0135】

未使用領域が不足する際の処置の例として、メモリ拡張制御部1123は要求をベストエフォートの処理することとして、確保できた領域分だけ割当て可能だと応答する。

【0136】

<メモリアクセス監視部1124>

メモリアクセス監視部1124は、メモリアクセス権の監視を行い、メモリアクセスの可否、および、メモリアクセスを可能にするためのメモリ構成変更指示、および、利用権確保を制御する。

20

【0137】

具体的には、メモリアクセス監視部1124は、メモリアクセス制御部1220- i ($i=1 \sim n$) から実メモリ対応付けが必要なメモリアクセスが発生し、アクセスメモリ領域を渡されたなら、利用OS情報管理部1112を介して、当該アクセスメモリ領域の利用OS情報とメモリアクセスしてきたOS情報とを比較し、利用可能であるかどうか、すなわち、利用権を保有しているかを識別子が等しいかで判定する。

【0138】

メモリ領域が利用可能であれば、メモリアクセス監視部1124は、当該メモリアクセス領域の対応付けを行い、メモリアクセス制御部1220- i ($i=1 \sim n$) へ応答する。

30

【0139】

メモリ領域が利用可能でないならば、メモリアクセス監視部1124は、管理OS情報管理部1111を介して、当該アクセスメモリ領域の管理OS情報とメモリアクセスしてきたOS情報とを比較し、管理権を保有しているかどうか、すなわち、識別子が等しいかどうかを判定する。

【0140】

メモリ領域の管理権を保有しているのならば、メモリアクセス監視部1124は、メモリ解放制御部1122へ当該メモリアクセス領域のメモリ解放を要求し、メモリ解放を応答された後、利用OS情報管理部1112を介して、当該アクセスメモリ領域の利用OS情報をアクセスOSの識別子に変更し、当該メモリアクセス領域の対応付けを行い、メモリアクセス制御部1220- i ($i=1 \sim n$) へ応答する。

40

【0141】

メモリ領域の管理権を保有していないのであれば、メモリアクセス制御部1220- i ($i=1 \sim n$) からのアクセス要求はアクセス権限がないOSからのアクセス要求であるため、メモリアクセス監視部1124は、メモリアクセス制御部1220- i ($i=1 \sim n$) へメモリアクセスエラーとして応答する。

【0142】

<OS1200- i >

各OS1200- i ($i=1 \sim n$) は、メモリ領域やその属性を制御するメモリ管理部1210- i ($i=1 \sim n$) と、実メモリとの対応付けを必要とするメモリアクセスを検知するメモリアクセス制御手

50

段1220- i ($i=1 \sim n$)と、メモリ削減対象のときに解放すべき候補を探索するメモリ解放領域探索手段1231- i ($i=1 \sim n$)と、メモリ解放を行うメモリ解放手段1232- i ($i=1 \sim n$)と、メモリ拡張を行うメモリ拡張手段1233- i ($i=1 \sim n$)とから構成されている。

【0143】

<メモリ管理部1210- i >

メモリ管理部1210- i ($i=1 \sim n$)は、各OS内のメモリに関する情報を管理する。

【0144】

メモリに関する情報としては、特性値や統計値などのメモリ全体の特性、統計を表現するメモリ内容情報と、ページの状態やそれらの情報を管理するための情報であるメモリ制御情報とがある。

10

【0145】

メモリ内容情報の例としては、総メモリ容量、未使用容量、バッファ容量、キャッシュ容量、スワップキャッシュ容量、プログラムとして利用中である容量、キャッシュ上に保存されており未使用状態である情報、拡張メモリ領域にあるメモリ容量、拡張メモリ領域にある未使用メモリ容量、通常メモリ領域にあるメモリ容量、通常メモリ領域にある未使用メモリ容量、情報の更新があるメモリ容量、スワップデバイスに書き出し中のメモリ容量、ページテーブルからマップされているメモリ容量、スラブアロケータとして利用中のメモリ容量、ページテーブルとして利用中のメモリ容量、プログラムが割当てられているメモリ容量などが挙げられる。

【0146】

メモリ制御情報の例としては、ページの状態を表すためのフラグ、ページの参照カウンタ、未使用ページのリスト、使用中ページのリスト、過去使用しており再利用可能なページのリスト、ページ領域の特性などが挙げられる。

20

【0147】

ここで、ページの状態を表すためのフラグとは、例えば、ロックされているかどうか、データ転送中にエラーが発生したか、ページが参照されたか、読込完了したか、更新されたか、LRUリストにつながっているか、スラブで利用されているか、リザーブされているか、スワップキャッシュとして利用されているかなどの情報を表すために使用される。

【0148】

また、ページ領域の特性とは、通常使用の領域か、DMA転送などのために利用する領域か拡張メモリ領域であるかなどを表現している。

30

【0149】

メモリ内容情報とメモリ制御情報を併せて、メモリ情報と呼ぶ。

【0150】

<メモリアクセス制御部1220- i >

メモリアクセス制御部1220- i ($i=1 \sim n$)は、OSのメモリアクセスを制御する。

【0151】

本発明において、メモリアクセス制御部1220- i ($i=1 \sim n$)は、少なくとも、実メモリとの対応付けが必要なアクセスを検知する機能を持つ。

【0152】

例えば、プログラム記憶域やそのデータ入出力のためのキャッシュ確保のためや、メモリ領域への初回の入出力など、マシンアドレス空間を通して実メモリ空間への変換を必要とするメモリアクセスを検知することができる。

40

【0153】

メモリアクセス制御部1220- i ($i=1 \sim n$)は、実メモリとの対応付けが必要になったときに、メモリアクセス監視部1124へメモリの利用権があるかを問い合わせる。

【0154】

<メモリ構成変更要求制御部1230- i >

メモリ構成変更要求制御部1230- i ($i=1 \sim n$)は、メモリ構成変更の要求を入力された場合、メモリ構成変更がメモリを削減する要求の場合はメモリ解放領域探索部1241- i ($i=1 \sim n$)

50

へ、メモリ構成変更がメモリを増加させる要求の場合は、メモリ拡張部1243- i ($i=1 \sim n$)へ、各要求を発行する。

【0155】

メモリ構成変更要求を入力する元は、例えば、オペレータであったり、資源配分を制御する資源管理システムであったり、OS内部に取り込ませ、OS内の資源使用状況により構成変更を指示する資源監視エージェントプログラムであったりする。

【0156】

今後、説明の便宜上、オペレータからの入力を前提に説明を続けることにするが、要求入力元はオペレータとは限らない場合もあることは明白である。

【0157】

10

また、オペレータなどがOSからコマンドを入力する際、メモリ削減用のコマンドとメモリ増加用のコマンドやシステムコールを明示的に分けることもできるが、これはコマンド解釈部やシステムコール管理プログラムなどにより、コマンドやシステムコールが判定され、識別されていると捉えることができるので、メモリ構成変更要求制御部1230- i ($i=1 \sim n$)相当の機能であると捉えることができる。

【0158】

ここで、メモリ構成変更要求の表現としては、マシンアドレス空間の範囲を指定する方法がある。

【0159】

例えば、「1～2ギガバイト(GB)の領域を削減する」などである。

20

【0160】

メモリ構成変更要求の別の表現としては、メモリの増減量で指定する方法がある。

【0161】

例えば、「1GBの容量を削減する」などである。

【0162】

<メモリ解放領域探索部1241- i >

メモリ解放領域探索部1241- i ($i=1 \sim n$)は、メモリ削減要求時に、解放に適するメモリ領域を決定する。

【0163】

具体的には、メモリ解放領域探索部1241- i ($i=1 \sim n$)は、メモリ構成変更要求制御部1230- i ($i=1 \sim n$)からメモリ解放領域の探索を要求された場合、メモリ管理部1210- i ($i=1 \sim n$)のメモリ制御情報を参照して、メモリ解放に適した領域を選択する。

30

【0164】

解放メモリ領域の選定方法として、例えば、メモリ領域のアドレス位置を利用することができる。例えば、高位アドレスを優先して解放メモリ領域を選択することができる。また、前記マシンアドレス空間の範囲を指定された場合は、この方法で探索する必要がある。

【0165】

別の解放メモリ領域の選定方法として、メモリの使用状態を利用することができる。例えば、未使用ページを優先して選択することができる。また、別の例として、キャッシュ領域として利用しているページを優先して選択することができる。

40

【0166】

また、もう1つの解放メモリ領域の選定方法として、メモリアクセス情報を利用することができる。例えば、使用された時刻が最も過去であるページを優先することができる。また、アプリケーションの動作が決まっており、将来のページアクセスパターンが予測できる環境では、アクセス開始までの時刻が最も長いページを優先することができる。

【0167】

メモリ解放領域探索部1241- i ($i=1 \sim n$)は、解放メモリ領域を決定したなら、メモリ解放宣言部1121に当該解放メモリ領域の解放宣言を伝え、管理権を放棄する。

【0168】

50

ただし、OS内部としては当該メモリ領域を解放していないため、OSの管理からは外れていないことに注意してもらいたい。

【0169】

<メモリ解放部1242-i (i=1~n)>

メモリ解放部1242-i (i=1~n)は、OSからメモリ領域に関するメモリ情報を削除し、メモリの解放を行う。

【0170】

具体的には、メモリ解放制御部1122から解放メモリ領域が渡され、解放命令を受取ったなら、メモリ管理部1210-i (i=1~n)から当該解放メモリ領域に関するメモリ情報を削除する。

10

【0171】

ここで、メモリ情報を削除するとは、メモリ内容情報から当該解放メモリ領域に対する情報分を差分して変更することと、メモリ制御情報から当該解放メモリ領域に対応する情報を削除することをいう。

【0172】

本処理にて、OS内部の情報からメモリに関する情報が無くなるために、OSの管理からは外れることになる。

【0173】

<メモリ拡張部1243-i>

メモリ拡張部1243-i (i=1~n)は、メモリの増加処理を行う。

20

【0174】

具体的には、メモリ拡張部1243-i (i=1~n)は、メモリ構成変更要求制御部1230-i (i=1~n)からメモリ領域の拡張を要求された場合、メモリ拡張制御部1123へ拡張するメモリ領域の取得を要求する。拡張するメモリ領域が返答されたなら、当該拡張メモリ領域に対するメモリ情報をメモリ管理部1210-i (i=1~n)へ追加する。

【0175】

ここで、メモリ領域に対するメモリ情報を追加するとは、メモリ内容情報から当該拡張メモリ領域に対する情報分を増分して変更することと、メモリ制御情報へ当該拡張メモリ領域に対応する情報を生成し、追加させることをいう。

【0176】

30

<第1の実施の形態の動作>

次に、本実施の形態の動作について、図5の構成図、および、図6乃至図9のフローチャート図を参照して説明する。

【0177】

<メモリ構成変更要求応答処理>

まず、図5の構成図、および、図6のフローチャートを参照して、メモリ構成変更要求応答処理の詳細な動作を説明する。

【0178】

メモリ構成変更要求制御部1230-i (i=1~n)が、オペレータからメモリ構成変更要求を受信したら、メモリ構成変更要求応答処理が開始する。

40

【0179】

メモリ構成変更要求制御部1230-i (i=1~n)は、オペレータからメモリ構成変更要求を受信した場合、当該メモリ構成変更要求がメモリ削減要求かメモリ増加要求かを判定する(ステップS101)。

【0180】

メモリ構成変更要求制御部1230-i (i=1~n)は、メモリ構成変更要求がメモリ削減要求の場合、当該メモリ削減要求をメモリ解放領域探索部1241-i (i=1~n)へ発行する。メモリ解放領域探索部1241-i (i=1~n)はメモリ管理部1210-i (i=1~n)を参照して、削減要求に適した容量分の解放に適するメモリ領域を選択する(ステップS102)。

【0181】

50

メモリ解放領域探索部1241-i (i=1~n)は、選択した解放対象のメモリ領域の管理権放棄をメモリ解放宣言部1121へ要求する(ステップS103)。

【0182】

メモリ解放宣言部1121は、利用OS選択処理に入り、当該解放メモリ領域の利用OSを決定する。これは、後で詳細に記載する(ステップS104)。

【0183】

メモリ解放宣言部1121は、管理OS情報管理部1111を介して、当該解放メモリ領域の管理権を未使用状態に変更する(ステップS105)。

【0184】

メモリ解放宣言部1121は管理権放棄の結果をメモリ解放領域探索部1241-i (i=1~n)へ応答し、メモリ解放領域探索部1241-i (i=1~n)は削減要求に対して結果をメモリ構成変更要求制御部1230-i (i=1~n)へ応答し、メモリ構成変更要求制御部1230-i (i=1~n)はオペレータへ削減要求の結果を応答する(ステップS106)。

10

【0185】

メモリ構成変更要求制御部1230-i (i=1~n)は、メモリ構成変更要求がメモリ増加要求の場合、当該メモリ増加要求をメモリ拡張部1243-i (i=1~n)へ発行する。メモリ拡張部1243-i (i=1~n)は増加要求を満たす容量のメモリ領域確保をメモリ拡張制御部1123へ要求する(ステップS107)。

【0186】

メモリ拡張制御部1123は、管理OS情報管理部1111を介して管理OS情報を参照し、必要であれば、利用OS情報管理部1112を介して利用OS情報を参照し、確保要求に適したメモリ領域を選択する(ステップS108)。

20

【0187】

メモリ拡張制御部1123は、選択した追加メモリ領域に対して、管理OS情報管理部1111を介して管理OSを要求元OSとして登録する(ステップS109)。

【0188】

メモリ拡張制御部1123は、当該追加メモリ領域を確保したことをメモリ拡張部1243-i (i=1~n)へ応答する。メモリ拡張部1243-i (i=1~n)は、当該追加メモリ領域に対するメモリ情報をメモリ管理部1210-i (i=1~n)に登録し、メモリの拡張を行う(ステップS110)。

【0189】

30

メモリ拡張部1243-i (i=1~n)は、メモリ構成変更要求制御部1230-i (i=1~n)へメモリ拡張の結果を応答し、メモリ構成変更要求制御部1230-i (i=1~n)はオペレータへ増加要求の結果を応答する(ステップS111)。

【0190】

これら一連の処理により、メモリ構成変更要求応答処理が行われる。

【0191】

< 利用OS選択処理 >

図5の構成図、および、図7のフローチャートを参照して、ステップS104の利用OS選択処理の詳細な動作を説明する。

【0192】

40

メモリ解放宣言部1121は、前記解放メモリ領域に対して、利用OS情報管理部1111を介して、利用OS情報を参照して、利用OSが異なるメモリ領域へ領域分割する(ステップS201)。

【0193】

メモリ解放宣言部1121は、前記分割されたメモリ領域に対して、ステップS203から始まる利用OS情報判定が終了していない分割されたメモリ領域がないかをチェックする(ステップS202)。

【0194】

ステップS202にて未判定メモリ領域が存在した場合、当該メモリ領域に対して、メモリ解放宣言手段1121は解放要求元のOSの識別子と利用OS情報を比較し、解放要求元が利用OSであるかどうかを判定する(ステップS203)。

50

【 0 1 9 5 】

ステップS203において、解放要求元が利用OSと判定した場合は、既に利用OSとして接続し、利用を継続している状態であるため、当該メモリ領域の利用OS選択は完了する。

【 0 1 9 6 】

ステップS203において、解放要求元が利用OSでないと判定した場合は、メモリ解放宣言手段1121は、今後、解放を要求しているOSから当該メモリ領域へのメモリアクセスを発生させないために、メモリ解放制御部1122へ前記解放を要求しているOSに対して当該メモリ領域の解放を要求する(ステップS204)。

【 0 1 9 7 】

ステップS204においては、メモリ解放制御部1122を起点にメモリ解放処理がなされる。ここで、解放要求元は当該メモリ領域を一度も利用していないため、メモリ解放処理においてページアウト/スワップ処理は行われない。

10

【 0 1 9 8 】

ステップS204において、メモリ解放の応答が返答されれば、利用OS情報判定は完了する。

【 0 1 9 9 】

ステップS202において、利用OS情報判定が前記分割した全メモリ領域に対して完了したなら、利用OS選択処理が完了する。

【 0 2 0 0 】

これら一連の処理により、利用OS選択処理が行われる。

20

【 0 2 0 1 】

<メモリ解放処理>

図5の構成図、および、図8のフローチャートを参照して、メモリ解放処理の詳細な動作を説明する。

【 0 2 0 2 】

メモリ解放制御部1122は、メモリ解放を要求された場合、要求されているメモリ解放領域を要求されているOSのメモリ解放部1242-i(i=1~n)へ、OSのメモリ管理から解放するように要求する(ステップS301)。

【 0 2 0 3 】

メモリ解放部1242-i(i=1~n)は、メモリ解放を要求された場合、要求されているメモリ解放領域のメモリ情報をメモリ管理部1210-i(i=1~n)から取得する(ステップS302)。

30

【 0 2 0 4 】

メモリ解放部1242-i(i=1~n)は前記取得したメモリ情報においてメモリ領域の利用状況を参照し、該当メモリ領域が使用中かどうかを判定する(ステップS303)。

【 0 2 0 5 】

ステップS303においてメモリ解放領域が使用中であると判定した場合、メモリ解放部1242-i(i=1~n)はページアウト/スワップ処理を行いメモリ上のデータの退避を行う(ステップS304)。

【 0 2 0 6 】

ステップS303においてメモリ解放領域が未使用であると判定した場合、もしくは、S504にてページアウト/スワップ処理が終了した場合、メモリ解放部1242-i(i=1~n)は、当該メモリ解放領域に対するメモリ情報をメモリ管理部1210-i(i=1~n)から削除する(ステップS305)。

40

【 0 2 0 7 】

メモリ解放部1242-i(i=1~n)は、メモリ解放処理の完了を返答し、メモリ解放制御部1122は、メモリ解放要求に対して結果を返答する(ステップS306)。

【 0 2 0 8 】

これら一連の処理により、メモリ解放処理が実行される。

【 0 2 0 9 】

<実メモリ領域の融通処理>

50

図5の構成図、および、図9のフローチャートを参照して、メモリアクセスによる実メモリ領域を融通する処理の詳細な動作を説明する。

【0210】

実メモリ領域を融通する処理は、実メモリ空間へのアクセスが生じるときに行う。

【0211】

メモリアクセス制御手段1220-i ($i=1 \sim n$) は、実メモリとの対応付けが必要になったことを検知する。

【0212】

これを契機に図9のフローチャートであるメモリアクセス判定処理が開始される。

【0213】

メモリアクセス制御手段1220-i ($i=1 \sim n$) は、メモリアクセス監視部1124へメモリ領域の利用権があるかを問い合わせる（ステップS401）。

【0214】

メモリアクセス監視部1124は、利用OS情報管理部1112を介して利用OS情報を参照し、メモリアクセスを要求しているOSが当該アクセスメモリ領域の利用OSであるかどうかを判定する（ステップS402）。

【0215】

ステップS402において、利用OSであると判定された場合、メモリ利用権を既に保有しているため、成功結果作成（ステップS408）へと移る。

【0216】

ステップS402において、利用OSでないと判定された場合、続いて、未使用領域でないかを判定する（ステップS403）。

【0217】

ステップS403において、未使用領域であると判定された場合、当該OSにメモリ利用権を登録（ステップS407）し、成功結果作成（ステップS408）へと移る。

【0218】

ステップS403において、未使用領域でないと判定された場合、メモリアクセス監視部1124は、管理OS情報管理部1111を介して管理OS情報を参照し、メモリアクセスを要求しているOSが当該アクセスメモリ領域の管理OSであるかどうかを判定する（ステップS404）。

【0219】

ステップS404において、管理OSでないと判定された場合、メモリの利用権、および、管理権がないOSからのアクセスなので、メモリアクセス監視部1124は、メモリアクセスのエラー結果を作成する（ステップS409）。

【0220】

ステップS404において、管理OSであると判定された場合、つまりメモリアクセスを要求しているOSが利用権を保有せず管理権を保有するメモリ領域へアクセスを要求してきた場合、これがメモリ融通のためのタイミングであるため、メモリアクセス監視部1124は、メモリ解放制御部1122へ当該アクセスメモリ領域の利用OSに対するメモリ解放を要求する（ステップS405）。

【0221】

メモリ解放制御部1122は、メモリ解放要求に従い、前述したメモリ解放処理を行う（ステップS406）。

【0222】

メモリ解放制御部1122は、利用OS情報管理部1112を介して、利用OS情報を未使用状態に更新し、メモリ解放が終了したことをメモリアクセス監視部1124へ応答する。これにより、メモリアクセス監視部1124は、メモリ領域にアクセスが可能な状態、すなわち、データ退避等が終了し、利用してもデータの不整合を起こさない状態（以後、アクセス可能状態）に変更されたことを知ることができ、利用OS情報管理部1112を介して、利用OS情報にメモリアクセスを要求しているOSを登録する（ステップS407）。

【0223】

10

20

30

40

50

メモリアクセス監視部1124は、ステップS406までの処理により、メモリアクセスを要求しているOSがメモリ利用権を保有した状態に移行するため、成功結果を作成する（ステップS408）。

【0224】

メモリアクセス監視部1124は、ステップS408もしくは、ステップS409のメモリ領域の利用権結果をメモリアクセス制御手段1220-i (i=1~n)へ応答する（ステップS410）。

【0225】

これら一連の処理により、メモリアクセス判定処理が実行される。

【0226】

メモリアクセス制御手段1220-i (i=1~n)は、この結果を参照して、メモリアクセス処理を継続し、成功結果ならメモリアクセスを実現する。

【0227】

<動作シーケンス>

次に、図5の構成図、図6乃至図9のフローチャート図、図10のシーケンス図、および、図11のメモリ情報の概念図を参照して、本実施の形態の動作シーケンスを説明する。

【0228】

ここで、構成変更条件は、図2のときと同様であり、実メモリ管理部でのメモリ領域の管理単位としては、256MB単位として説明する。

【0229】

オペレータは、メモリを削減するOS(OS1)へ1GBのメモリ削減要求を発行する（図10の2001）。

【0230】

これにより、図6のフローチャートに示される処理が開始される。

【0231】

メモリ構成変更要求制御部1230-1は、メモリ削減要求であるので、メモリ解放領域探索部1241-1へメモリ削減要求を発行する（図6のS101で削除選択）。

【0232】

メモリ解放領域探索部1241-1は、メモリ管理部1210-1を介して適切な解放メモリ領域を探索する（図6のS102）。

【0233】

ここでは、1GB~2GBのメモリ領域が解放領域として選択されたものとする。

【0234】

メモリ解放領域探索部1241-1は、決定した解放メモリ領域を基に、メモリ解放宣言部1121に当該解放メモリ領域の解放宣言を伝え、管理権を放棄するように要求する（図6のS103、図10の2002）。

【0235】

メモリ解放宣言部1121は、メモリ解放領域探索部1241-1からの解放メモリ宣言要求に基づき、実メモリ管理部1110のメモリ空間対応表、および、管理OS情報管理部1111を介して、当該解放メモリ領域の管理OS情報を未使用状態へ変更する（図6のS104~S105、図10の2003）。

【0236】

メモリ解放宣言部1121は、管理OS情報変更後、すなわち、解放対象OSに対する管理権剥奪後、メモリ解放領域探索部1241-1へ結果を返答する（図6のS106、図10のS2004）

【0237】

メモリ解放領域探索部1241-1は、メモリ構成変更要求制御部1230-1へメモリ削減要求により、管理権を失ったことをオペレータに報告する（図10のS2005）。

【0238】

これにより、VMMからは、当該メモリ領域（1GB~2GB）がOS1から解放する意思があると認識し、未使用領域候補として扱うことができ、OS1からは現在も利用している領域として扱うことになる。

10

20

30

40

50

【0239】

オペレータは、メモリを増加するOS(OS2)へ1GBのメモリ増加要求を発行する(図10の2006)。

【0240】

これにより、再び、図6のフローチャートに示す処理が開始される。

【0241】

メモリ構成変更要求制御部1230-1は、メモリ増加要求であるので、メモリ拡張部1243-1へメモリ増加要求を発行する(図6のS101で増加選択)。

【0242】

メモリ拡張部1243-1は、メモリ拡張制御部1123へ必要な1GB分の拡張メモリ領域の確保を要求する(図6のS107, 図10の2007)。 10

【0243】

メモリ拡張制御部1123は、メモリ拡張部1243-1から拡張メモリ領域の確保を要求されたので、管理OS情報管理部1111と必要に応じて、利用OS情報管理1123を介して、未使用領域から1GB分の領域を選択し、管理OS情報管理部1111を介して管理OS情報を拡張要求OSであるOS2として登録し、さらに、実メモリ管理部1110のメモリ空間対応表へ、実メモリ空間とマシンアドレス空間の対応付けを登録することで、拡張メモリ領域として確保する(図6のS108～S109, 図10の2008)。

【0244】

メモリ拡張制御部1123は、確保した拡張メモリ領域をメモリ拡張部1243-1へ応答する(図10の2009)。 20

【0245】

メモリ拡張部1243-1は、メモリ管理部1210-1へ拡張メモリ領域に関するメモリ情報を追加する(図6のS110)。

【0246】

メモリ拡張部1243-1は、メモリ構成変更要求制御部1230-1へメモリ増加要求により、管理権を得たことをオペレータに報告する(図6のS111, 図10のS2010)。

【0247】

これにより、OS2からは新たな1GBのメモリ領域が管理下に入り、当該メモリ領域が拡張されたように扱わせることができる。 30

【0248】

ここまでで、図11(a)の初期状態から図11(b)の状態へ移行したことになる。

【0249】

すなわち、1GB～2GBのメモリ領域は、管理OSとしてOS2が登録され、OS2からは利用可能に見えるが、利用OSとしてOS1が登録されており、OS1からは解放前であるためまだ利用可能領域として扱い、実際にOS1がメモリ領域を利用している。

【0250】

OS2は、OS上でプログラムが動作していく過程において、拡張された領域へのアクセスが発生する。

【0251】

ここでは、最初に1.5GB～1.5256GBのメモリ領域へアクセスされたとする。 40

【0252】

これを契機に、メモリアクセス制御部1220-2は、実メモリ空間の取得を検知し、メモリアクセス監視部1124へメモリアクセスを問合せる(図9のS401, 図10の2011)。

【0253】

メモリアクセス監視部1124は、利用OS情報管理部1112を参照して、当該アクセスメモリ領域がメモリアクセスするOS2の利用領域かどうかを判定する。

【0254】

メモリ領域が利用領域ならメモリアクセスを許可するが、今の場合、利用OS情報がOS1であるため、利用権がないと判断する(図9のS402でNoを選択)。 50

【 0 2 5 5 】

また、利用OS情報がOS1であるため、未使用状態でないと判断する(図9のS403でNoを選択)。

【 0 2 5 6 】

メモリアクセス監視部1124は、次に、管理OS情報管理部1111を参照して、当該アクセスメモリ領域がOS2の管理領域かどうかを判定する(図9のS404)。

【 0 2 5 7 】

メモリ領域が管理領域でないのであれば、管理権も利用権も持たないためメモリアクセスエラーとして返答するが(図9のS409へ)、今回は、図10の2007～図10の2009のシーケンスにより該当メモリアクセス領域の管理権はOS2と設定されているため、管理領域として判断される(図9のS404でYesを選択)。

10

【 0 2 5 8 】

メモリアクセス監視部1124は、管理権を持つOS、すなわち、メモリ変更要求によりメモリ再構成後にメモリ領域を割られるOSからのアクセスであるため、利用OSからメモリ領域を解放し、アクセス可能にする処理に移る。

【 0 2 5 9 】

メモリアクセス監視部1124は、メモリ解放制御部1122へ当該アクセスメモリ領域のメモリ解放を要求し、メモリ解放させる(図9のS405～S406)。

【 0 2 6 0 】

メモリ解放制御部1122は、利用OSであるOS1のメモリ解放部1242-1へ当該アクセスメモリ領域のメモリ解放を要求する(図8のS301, 図10の2012)。

20

【 0 2 6 1 】

メモリ解放部1242-1は、メモリ管理部1210-1を介して要求された解放するメモリ領域のメモリ情報を参照し、使用状態にないかを判断し、使用状態であれば、ページアウト/スワップアウト処理によるデータ退避を行うことで未使用状態に変更する(図8のS302～304, 図10の2013)。

【 0 2 6 2 】

メモリ解放部1242-1は、当該解放メモリ領域が未使用状態になったなら、メモリ管理部1210-1からメモリ情報を削除する(図8のS305)。

【 0 2 6 3 】

これにより、OS1からは、該当メモリ領域が管理下から外れたことになる。

30

【 0 2 6 4 】

続いて、メモリ解放部1242-1は、当該解放メモリ領域のメモリ解放が完了したことをメモリ解放制御部1122へ応答する(図8のS306, 図10の2014)。

【 0 2 6 5 】

メモリ解放制御部1122は、メモリ解放部1242-1からメモリ解放完了の応答を受取ったなら、利用OS情報管理部1112を参照して、利用OS情報を未使用状態に変更する(図8のS407, 図10の2015)。

【 0 2 6 6 】

これにより、VMMからも当該解放メモリ領域の利用からOS1が外れることになる。

40

【 0 2 6 7 】

メモリ解放制御部1122は、メモリ解放部1242-1へ利用OS情報から削除したこと、すなわちメモリの切り離しが完了したことを通知する(図9のS407, 図10の2016)。

【 0 2 6 8 】

メモリ解放部1242-1は、メモリ構成変更要求制御部1230-1へ当該解放メモリ領域の解放が完了したことを伝える。

【 0 2 6 9 】

メモリ構成変更要求制御部1230-1は、必要に応じて、当該解放メモリ領域の解放、すなわち、最初のメモリ削減要求の一部が完了したことを表示する(図10の2017)。

【 0 2 7 0 】

50

メモリ解放制御部1122は、メモリ解放が終了したことをメモリアクセス監視部1124へ応答する(図9のS407)。

【0271】

メモリアクセス監視部1124は、要求したアクセスメモリ領域の解放が完了し、当該アクセスメモリ領域が使用可能状態になったため、アクセスしたOS2へメモリアクセスを許可するために、利用OS情報管理部1112を参照して、当該アクセスメモリ領域の利用OS情報にアクセスしたOS2を登録する(図9のS407)。

【0272】

これにより、OS2へメモリ利用権限が融通されたことになる。

【0273】

メモリアクセス監視部1124は、メモリアクセスを許可し、メモリアクセス制御部1220-2へ返答する(図9のS408, S410, 図10の2018)。

【0274】

これにより、OS2からは、自分がメモリアクセスした際に融通されていることを知らずに、すなわち、通常割当てられたメモリと同様にメモリアクセスを実行することができる。

【0275】

メモリアクセス制御部1220-2は、メモリ構成変更要求制御部1230-2へ当該アクセスメモリ領域の割当てが完了したことを伝える。

【0276】

メモリ構成変更要求制御部1230-2は、必要に応じて、当該アクセスメモリ領域の割当て、すなわち、最初のメモリ増加要求の一部が完了したことを表示する(図10の2019)。

【0277】

これにより、1.5GB~1.75GBのメモリ領域は融通される。

【0278】

そのため、利用者(利用OS)がOS1からOS2へと変更され、図11(b)の状態から図11(c)の状態になる。

【0279】

図10の2011~2019と同様の処理が繰り返され、メモリ融通が進み、最終的に、メモリ再構成処理が行われる。

【0280】

最終的に、図11(d)に示されるように、1GB~2GBの領域はOS2へ利用権が移行される。

【0281】

メモリ構成変更要求制御部1230-1、および、1230-2は、それぞれ、最後のメモリ解放通知2100、メモリ割当て通知2102が完了したら、削減要求、増加要求によるメモリ構成変更が完了したことになり、完了をオペレータへ表示する(図10の2101、および、2103)。ここで、最後のメモリ解放通知か否か、最後のメモリ割当て通知か否かの判断は、メモリ構成変更要求制御部において、メモリ構成変更要求を出した時点で、どこの領域が解放予定か、割当て予定かを自身が要求したメモリ構成変更要求を覚えておき、実際にアクセスが生じて解放されたメモリ領域、割り当てられたメモリ領域の結果と照らし合わせて、全ての領域において解放/割当てが完了したかどうかを検知することで可能である。

【0282】

次に、本実施の形態の効果について説明する。

【0283】

本実施の形態は、メモリ領域に対して、メモリ再構成後にメモリ領域を割当てられメモリの管理権をもつべきであるOSとなる管理OSと、メモリ再構成前にメモリ領域を使用しており、メモリ領域の解放前であるためにメモリ領域にアクセスする権利を持つ利用OSとの情報を保持し、メモリ構成変更要求に対して、要求受信時に管理権を変更させ、メモリアクセスを契機に利用権を変更させる2段階構造に分け、利用権変更時にはメモリ解放を保証することにより、メモリ再構成に対して、要求への見かけ上の対応を早めると同時に、

10

20

30

40

50

実際のメモリ融通を遅延させて効果的なタイミングで行え、予約領域を必要としないため、全メモリ領域を有効活用することができる。

【0284】

これにより、以下のような効果も得られる。

【0285】

追加の効果の1つは、メモリ再構成後にメモリが増加されるOSからは、メモリ領域の拡張によるメモリ情報の更新を早められることにより、メモリ容量などによるプログラム起動規制などをいち早く対応可能にすることができる。

【0286】

追加の効果のもう1つは、メモリアクセスを契機にアクセスするメモリ領域に対して、ページアプト/スワップアウトのデータ退避処理を行うため、負荷が大きく時間を要するデータ退避処理を分散させることができ、OS上で動作している通常プログラムの影響を和らげることができる。

【0287】

また別の追加の効果は、メモリの利用効率を上げることができる。

【0288】

メモリ効率向上の理由の1つとしては、メモリの融通に関する処理は、メモリ解放してからメモリ拡張させるという2段階を必要とし、その間の時間、該当メモリ領域は使用不可能状態になるが、メモリ領域の融通に関する単位がメモリ再構成要求に依存しなく、メモリアクセスする際のメモリ容量に分割されるため、使用不可能状態になるメモリ領域の相対的な累計時間を短くすることができるためである。

【0289】

メモリ効率向上の理由のもう1つとしては、メモリを減少させるOSからは、メモリ融通先のOSが利用するまでの間、該当メモリ領域を利用することが可能であり、再構成変更後のメモリ容量より大きな容量を使用できるほか、解放すべきメモリ領域に確保されているキャッシュやデータの再利用ができるためである。

【0290】

『第2の実施の形態』

次に、本発明の第2の発明を実施するための最良の形態について説明する。

【0291】

本実施の形態では、利用OS決定の別の例として、メモリ領域を権利が強いOSが使用継続を優先するために、その時点での管理OSを優先する方式になる。

【0292】

本実施の形態では、第1の実施の形態と比べて、利用OS選択処理だけが異なるので、この処理だけを説明する。

【0293】

これは、図7のフローチャートが変形する、すなわち、図6のステップS104の処理だけが異なることに相当する。

【0294】

図5、および、図12のフローチャートを参照して、ステップS104の利用OS選択処理において、管理OSを優先する方式の詳細な動作を説明する。

【0295】

メモリ解放宣言手段1121は、前記解放メモリ領域に対して、利用OS情報管理部1111を介して、利用OS情報を参照して、利用OSが異なるメモリ領域へ領域分割する(ステップS501)。

【0296】

メモリ解放宣言手段1121は、前記分割されたメモリ領域に対して、ステップS503から始まる利用OS情報判定が終了していない分割されたメモリ領域がないかをチェックする(ステップS502)。

【0297】

10

20

30

40

50

ステップS502にて未判定メモリ領域が存在した場合、当該メモリ領域に対して、メモリ解放宣言手段1121は解放宣言したOSの識別子と利用OS情報を比較し、解放要求元が利用OSであるかどうかを判定する(ステップS503)。

【0298】

ステップS503において、解放要求元が利用OSと判定した場合は、既に利用OSとして接続し、利用を継続している状態であるため、当該メモリ領域の利用OS選択は完了する。

【0299】

ステップS503において、解放要求元が利用OSでないと判定した場合は、管理権を保有していたOSからの解放宣言であるので、メモリ領域の権利が、利用していたOSより強いと判断し、当該メモリ領域の利用権を解放宣言したOSに移行する処理に入る。

10

【0300】

メモリ解放宣言部1121は、メモリ解放制御部1122へ利用OSに対して当該メモリ領域の解放を要求する(ステップS504)。

【0301】

ステップS504においては、図8のフローチャートに従って、メモリ解放制御部1122を起点にメモリ解放処理がなされる。

【0302】

メモリ解放宣言手段1121は、利用OS情報管理部1112を介して利用OS情報を解放宣言したOS、すなわち、管理OSに更新し、利用OS情報判定を完了する(ステップS504)。

【0303】

20

ステップS502において、利用OS情報判定が前記分割した全メモリ領域に対して完了したなら、利用OS選択処理が完了する。

【0304】

これら一連の処理により、メモリ領域を、権利が強いOSが使用継続を優先する利用OS選択処理が行われる。

【0305】

本発明の第1の実施の形態と本発明の第2の実施の形態の違いを図13、および、図14の管理OS情報、利用OS情報の情報管理遷移図を参照し、具体例を挙げて説明する。

【0306】

具体例としては、OS1が2GBを利用する。その後、高位1GBを融通させてOS2を起動させる。さらにその後にOS2のうちの高位512MBを融通させてOS3を起動させる場合を考える。

30

【0307】

また、OS2を起動させるタイミングはOS1が全領域をアクセスした後、OS3を起動させるタイミングは、OS2が低位、および、高位256MBを利用した後とする。

【0308】

図13は、第1の実施の形態の管理OS情報、利用OS情報の情報管理遷移図を示す。

【0309】

管理OS情報、利用OS情報の情報管理表の初期状態は、図13(a)に示すように全部未使用状態である。

【0310】

40

まず、2GBの容量がOS1に割当てられる。すなわち、管理OS情報としてOS1が登録され、管理OS情報、利用OS情報の情報管理表は図13(b)のように更新される。

【0311】

OS1は、割当てられた各領域にアクセスを開始し、実際の利用権を獲得する。すなわち、利用OS情報をOS1へと変更していき、管理OS情報、利用OS情報の情報管理表は図13(c)のように更新される。

【0312】

ここで、OS2が高位1GB、すなわち、1024MB～2048MBを融通させて利用しようとする。

【0313】

OS1は、高位1GBを解放宣言し、管理権を放棄する。すなわち、管理OS情報を未使用状態

50

に変更し、管理OS情報、利用OS情報の情報管理表は図13(d)へと更新される。

【0314】

OS2は、未使用領域から1GBを確保する。すなわち、1GB～2GBの領域を確保し、管理OSとして登録される。

【0315】

この間、利用OS選択機構が動作するが、ここでは、すべてOS1が利用OSであり、OS1が解放宣言しているため、メモリ解放処理は一切動作せず、その結果、管理OS情報、利用OS情報の情報管理表は図13(e)へと更新される。

【0316】

OS2は、メモリアクセスを始め、低位256MB(1024MB～1280MB)と高位256MB(1792MB～2048MB)をアクセスし、メモリ融通を経て、利用権を融通し、利用権を得る。すなわち、各領域に対する利用OS情報は、OS1からOS2へと変更され、管理OS情報、利用OS情報の情報管理表は図13(f)となる。

【0317】

ここで、OS3がOS2の高位512MB、すなわち、1536MB～2048MBを融通させて利用しようとする。

【0318】

OS2は、高位512MBを解放宣言し、管理権を放棄する。すなわち、管理OS情報を未使用状態に変更する。

【0319】

このとき、1536MB～1792MBと、1792MB～2048MBで利用OSが異なることになるので、各領域に対して利用OS選択機構が動作する。

【0320】

1536MB～1792MBのメモリ領域は、OS1が利用権を持ち利用しているため、OS2から今後、アクセスが生じないようにするために、OS2から1536MB～1792MBのメモリ領域を解放するメモリ解放処理がなされる。

【0321】

1792MB～2048MBのメモリ領域は、OS2が利用権を持ち利用しており、メモリ融通の際にOS1の管理下からは外れているために、メモリ解放処理を必要としない。

【0322】

すなわち、図7のフローチャートにて、1536MB～1792MBのOS2に対するメモリ解放処理がなされる。

【0323】

図7のフローチャートの結果、管理OS情報、利用OS情報の情報管理表は図13(g)へと更新される。

【0324】

OS3は、未使用領域から512MBを確保する。すなわち、1536MB～2048MBの領域を確保し、管理OSとして登録され、管理OS情報、利用OS情報の情報管理表は図13(h)へと更新される。

【0325】

次に、第2の実施の形態における動作を、図14の管理OS情報、利用OS情報の情報管理遷移を参照して説明する。

【0326】

図14f)までは図13f)までと動作に変わりがないので、説明を省略することとする。

【0327】

ここで、OS2は、高位512MBを解放宣言し、管理権を放棄する。すなわち、管理OS情報を未使用状態に変更することになる。

【0328】

このとき、1536MB～1792MBと、1792MB～2048MBで利用OSが異なることになるので、各領域に対して利用OS選択機構が動作する。

【0329】

1536MB～1792MBのメモリ領域は、OS1が利用権を持ち利用していたが、管理権を持つOS2が管理権を外れることから、今後、直前に管理権を持っているOS2がアクセス可能になるように、利用権をOS2へ移行することになる。

【0330】

そのため、OS1に対して1536MB～1792MBのメモリ領域を解放させた後、利用OS情報をOS2として登録する。

【0331】

1792MB～2048MBのメモリ領域は、OS2が利用権を持ち利用しており、メモリ融通の際にOS1の管理下からは外れているために、メモリ解放処理を必要としない。

10

【0332】

すなわち、図12のフローチャートにて、1536MB～1792MBのOS1に対するメモリ解放処理がなされる。

【0333】

図12のフローチャートの結果、管理OS情報、利用OS情報の情報管理表は図14(g)へと更新される。

【0334】

OS3は、未使用領域から512MBを確保する。すなわち、1536MB～2048MBの領域を確保し、管理OSとして登録され、管理OS情報、利用OS情報の情報管理表は図14(h)へと更新される。

20

【0335】

このように、第1の実施の形態と第2の実施の形態は、利用OSとして設定するOSの選択方式が異なる。

【0336】

本実施の形態の効果は、直前の設定されたメモリ構成を保てることにある。

【0337】

第1の実施の形態では、既にメモリを利用しているOSを利用OSとして継続させることでデータの再利用性を向上させることができることに對し、本実施の形態では、直前のOS(アプリケーション)に適した利用構成の設定を維持することで、今後のメモリ配分効果を高めることにある。

30

【0338】

すなわち、たまたまメモリアクセスが生じなかったために利用OSにできなかった領域は、その後の処理を考えた場合に直前の管理構成を持続させた方がメモリの利用効率が向上し、アプリケーションの動作を向上させることが考えられる。

【0339】

また、第1の実施の形態で挙げた効果を併せ持つ。

【0340】

『第3の実施の形態』

次に、本発明の第3の実施の形態について説明する。

【0341】

本発明の第3の実施の形態は、第1、および、第2の実施の形態に比べて、メモリアクセス判定処理だけが異なるので、この処理だけを説明する。

40

【0342】

図5の構成図、および、図15のフローチャートを参照して、第3の発明のメモリアクセス判定処理の詳細な動作について説明する。

【0343】

第1、および、第2の実施の形態と同様、実メモリ領域を融通する処理は、実メモリ空間へのアクセスが生じるときに行う。

【0344】

メモリアクセス制御手段1220-i ($i=1 \sim n$)は、実メモリとの対応付けが必要になったこと

50

を検知する。

【 0 3 4 5 】

これを契機に図15のフローチャートであるメモリアクセス判定処理が開始される。

【 0 3 4 6 】

メモリアクセス制御手段1220-i (i=1~n)は、メモリアクセス監視部1124へメモリ領域の利用権があるかを問い合わせる(ステップS601)。

【 0 3 4 7 】

メモリアクセス監視部1124は、利用OS情報管理部1112を介して利用OS情報を参照し、当該アクセスメモリ領域が利用OSであるかどうかを判定する(ステップS602)。

【 0 3 4 8 】

ステップS602において、利用OSでないと判定された場合、続いて、未使用領域でないか判定する(ステップS603)。

【 0 3 4 9 】

ステップS603において、未使用領域であると判定された場合、メモリ利用権を登録(ステップS607)し、成功結果作成(ステップS608)へと移る。

【 0 3 5 0 】

ステップS603において、未使用領域でないと判定された場合、メモリアクセス監視部1124は、管理OS情報管理部1111を介して管理OS情報を参照し、当該アクセスメモリ領域が管理OSであるかどうかを判定する(ステップS604)。

【 0 3 5 1 】

ステップS604において、管理OSでないと判定された場合、メモリの利用権、および、管理権がないOSからのアクセスなので、メモリアクセス監視部1124は、メモリアクセスのエラー結果を作成する(ステップS609)。

【 0 3 5 2 】

ステップS604において管理OSであると判定された場合、メモリ融通のためのタイミングであるため、メモリアクセス監視部1124は、メモリ解放制御部1122へ当該アクセスメモリ領域の利用OSに対するメモリ解放を要求する(ステップS605)。

【 0 3 5 3 】

メモリ解放制御部1122は、メモリ解放要求に従い、前述したメモリ解放処理を行う(ステップS606)。

【 0 3 5 4 】

メモリ解放制御部1122は、利用OS情報管理部1112を介して、利用OS情報を未使用状態に更新し、メモリ解放が終了したことをメモリアクセス監視部1124へ応答する。これにより、メモリアクセス監視部1124は、メモリ状態がアクセス可能状態に変更されたことを知ることができ、利用OS情報管理部1112を介して、利用OS情報にメモリアクセスを要求しているOSを登録する(ステップS607)。

【 0 3 5 5 】

メモリアクセス監視部1124は、ステップS606までの処理により、メモリアクセスを要求しているOSがメモリ利用権を保有した状態に移行するため、成功結果を作成する(ステップS608)。

【 0 3 5 6 】

ここまでの処理は図9に示した第1の実施の形態における処理と同じである。

【 0 3 5 7 】

ステップS602において、利用OSであると判定された場合、メモリアクセス監視部1124は、メモリアクセスを要求したOSが管理OSかどうかを判定する(ステップS617)。管理OSであれば、ステップS608へ処理を進める。

【 0 3 5 8 】

メモリアクセスを要求したOSが利用OSであるが管理OSでなければ(ステップS617でNo)、メモリアクセス監視部1124は、メモリアクセスを要求したOSからの当該メモリ領域に愛するメモリアクセスの統計値を計算する(ステップS611)。

10

20

30

40

50

【0359】

メモリアクセスの統計値としては、例えば、メモリアクセス回数、メモリ管理権を放棄してからのメモリアクセス回数、一定時間あたりのメモリアクセス回数、メモリアクセスデータ量、総メモリアクセスデータ量、メモリ管理権を放棄してからの総メモリアクセスデータ量、メモリ管理権を放棄してからの平均メモリアクセスデータ量、一定時間あたりの総メモリアクセスデータ量、一定時間あたりの平均メモリアクセスデータ量、管理権を放棄してからのメモリアクセス時間が挙げられる。

【0360】

メモリアクセス監視部1124は、前記計算した統計値を基にメモリ領域を解放すべきかどうかを判定する（ステップS612）。例えば、一定時間あたりのメモリアクセス回数が少なくなっている場合、そのメモリ領域に対する利用はないと考えられる。その場合、利用OS側がメモリ領域の利用権を持ち続けるよりは、これからそのメモリ領域を必要とするであろう管理OS側にメモリ領域の利用権を直ぐに渡せる状態にした方が、ページアウト処理等の処理時間を省くことができ有効である。逆に、メモリアクセスが頻繁に起こっているようであれば、データ利用効率が良いので解放するのは好ましくない。また、管理権を放棄してからのメモリアクセス時間では、利用OS側に一定時間の猶予を与えているのと同じ状態になる。このような状況を考慮して、予めどのような条件が成立した場合に、メモリ領域を解放すべきかが事前に設定されている。

【0361】

具体的な判定方法としては、例えば、統計値に一定値との比較演算（以上、以下、越える、超えない、等しい、等しくないなど）により求める方法、統計値を一定の評価関数に当てはめた後に一定値との比較演算により求める方法、また、前記比較する数値も一定値でなく、状況に合わせて変化する値として扱う方法が挙げられる。

【0362】

もし解放すべき領域ではないと判定した場合、メモリ利用権を既に保有し、解放条件にも当てはまらないため、利用してよい領域になるため、成功結果作成（ステップS608）へと移る。

【0363】

もし解放すべき領域であると判定した場合、メモリ融通のためのタイミングが到来したものと判断して、メモリアクセス監視部1124は、メモリ解放制御部1122へ当該アクセスメモリ領域の利用OSに対するメモリ解放を要求する（ステップS613）。

【0364】

メモリ解放制御部1122は、メモリ解放要求に従い、前述したメモリ解放処理を行う（ステップS614）。

【0365】

メモリ解放制御部1122は、利用OS情報管理部1112を介して、利用OS情報を未使用状態に更新し、メモリ解放が終了したことをメモリアクセス監視部1124へ応答する（ステップS615）。

【0366】

メモリアクセス領域は前記メモリ解放処理において、ページアウト/スワップアウトされた領域へのアクセスに変更されるため、メモリアクセス監視部1124はメモリ再アクセス要求を起させる結果を生成する（ステップS616）。

【0367】

メモリアクセス監視部1124は、ステップS608、S609もしくは、ステップS616のメモリ領域の利用権結果をメモリアクセス制御手段1220-i (i=1~n)へ応答する（ステップS610）。

【0368】

これら一連の処理により、メモリアクセス判定処理が実行される。

【0369】

メモリアクセス制御手段1220-i (i=1~n)は、この結果を参照して、メモリアクセス処理を継続し、メモリアクセスを実現する。

10

20

30

40

50

【0370】

次に、第3の実施の形態の効果について説明する。

【0371】

本実施の形態は、第1、または、第2の実施の形態の効果の有すると共に、追加の効果として、管理OSにおけるアクセス速度を向上させることができる。

【0372】

その理由は、利用OSのメモリアクセスの統計結果に基づきメモリ解放処理が行われ、未使用領域へ移行するためである。すなわち、管理OSがメモリアクセスを要求するときに、既にメモリ解放処理が終了しており、既に未使用領域となっている場合が生じ、すぐに利用権を確保できてメモリアクセスに移ることができるためである。

10

【0373】

これにより、管理OS側から見ると、利用権を保有していない領域へのメモリアクセス速度は平均して向上することができる。

【0374】

『第4の実施の形態』

次に、本発明の第4の実施の形態について図面を参照して詳細に説明する。

【0375】

本発明の第4の実施の形態は、第1乃至第3の発明を実施するための最良の形態に比べて、メモリ構成変更要求応答処理だけが異なるので、この処理だけを説明する。

【0376】

図5の構成図、および、図16のフローチャートを参照して、メモリ構成変更要求応答処理の詳細な動作を説明する。

20

【0377】

メモリ構成変更要求制御部1230-i (i=1~n)が、オペレータからメモリ構成変更要求を受信したら、メモリ構成変更要求応答処理を開始する。

【0378】

メモリ構成変更要求制御部1230-i (i=1~n)は、オペレータからメモリ構成変更要求を受信した場合、当該メモリ構成変更要求がメモリ削減要求かメモリ増加要求かを判定する(ステップS701)。

【0379】

メモリ構成変更要求制御部1230-i (i=1~n)は、メモリ構成変更要求がメモリ削減要求の場合、当該メモリ削減要求をメモリ解放領域探索部1241-i (i=1~n)へ発行する。メモリ解放領域探索部1241-i (i=1~n)はメモリ管理部1210-i (i=1~n)を参照して、削減要求に適した容量分の解放に適するメモリ領域を選択する(ステップS702)。

30

【0380】

メモリ解放領域探索部1241-i (i=1~n)は、メモリ解放領域に対して、メモリ管理部1210-i (i=1~n)を通じて、メモリ情報を変更する(ステップS703)。

【0381】

ここで、メモリ情報の変更として、メモリ内容情報を変更することができる。このとき、メモリ解放処理におけるメモリ情報からの削除(図8のフローチャートのステップS305)時には、該当メモリ内容情報を変更しない。例えば、メモリ総容量情報を変更することができる。本ステップにて、メモリ総容量情報を変更するため、メモリ解放処理時にはメモリ総容量の変更は行なう必要がなくなる。

40

【0382】

また、メモリ情報の別の変更として、メモリ制御情報を変更することができる。例えば、メモリのページ利用形態を一時退避領域などに設定することができる。

【0383】

メモリ解放領域探索部1241-i (i=1~n)は、選択した解放メモリ領域の管理権放棄をメモリ解放宣言部1121へ要求する(ステップS704)。

【0384】

50

メモリ解放宣言部1121は、利用OS選択処理に入り、当該解放メモリ領域の利用OSを決定する。これは、後で詳細に記載する(ステップS705)。

【0385】

メモリ解放宣言部1121は、管理OS情報管理部1111を介して、当該解放メモリ領域の管理権を未使用状態に変更する(ステップS706)。

【0386】

メモリ解放宣言部1121は管理権放棄の結果をメモリ解放領域探索部1241- i ($i=1 \sim n$)へ応答し、メモリ解放領域探索部1241- i ($i=1 \sim n$)は削減要求に対して結果をメモリ構成変更要求制御部1230- i ($i=1 \sim n$)へ応答し、メモリ構成変更要求制御部1230- i ($i=1 \sim n$)はオペレータへ削減要求の結果を応答する(ステップS707)。

10

【0387】

メモリ構成変更要求制御部1230- i ($i=1 \sim n$)は、メモリ構成変更要求がメモリ増加要求の場合、当該メモリ増加要求をメモリ拡張部1243- i ($i=1 \sim n$)へ発行する。メモリ拡張部1243- i ($i=1 \sim n$)はメモリ拡張制御部1123へ増加要求を満たす容量のメモリ領域確保をメモリ拡張制御部1123へ要求する(ステップS708)。

【0388】

メモリ拡張制御部1123は、管理OS情報管理部1111を介して管理OS情報を参照し、必要であれば、利用OS情報管理部1112を介して利用OS情報を参照し、確保要求に適したメモリ領域を選択する(ステップS709)。

【0389】

20

メモリ拡張制御部1123は、選択した追加メモリ領域に対して、管理OS情報管理部1111を介して管理OSを要求元OSとして登録する(ステップS709)。

【0390】

メモリ拡張制御部1123は、当該追加メモリ領域を確保したことをメモリ拡張部1243- i ($i=1 \sim n$)へ応答する。メモリ拡張部1243- i ($i=1 \sim n$)は、当該追加メモリ領域に対するメモリ情報をメモリ管理部1210- i ($i=1 \sim n$)に登録し、メモリの拡張を行う(ステップS711)。

【0391】

メモリ拡張部1243- i ($i=1 \sim n$)は、メモリ構成変更要求制御部1230- i ($i=1 \sim n$)へメモリ拡張の結果を応答し、メモリ構成変更要求制御部1230- i ($i=1 \sim n$)はオペレータへ増加要求の結果を応答する(ステップS712)。

30

【0392】

これら一連の処理により、メモリ構成変更要求応答処理が行われる。

【0393】

次に、第4の実施の形態の効果について説明する。

【0394】

本実施の形態は、第1乃至第3の実施の形態の効果を有すると共に、追加の効果として、メモリを削減するOSのメモリ利用形態を変更することにより、メモリを削減するOSでのプログラムへの影響、および、ページアウト/スワップ処理の量を削減する効果が現れる。

【0395】

例えば、メモリ内容情報の変更の例として、メモリ総容量を変更することが挙げられる。これにより、実際に解放処理した以後の仮想計算機環境を想定でき、必要以上のプログラム起動などを抑えることができる。

40

【0396】

別の例として、メモリ制御情報を変更する場合では、メモリ領域を一時退避領域に設定することが挙げられる。これにより、利用OS内メモリアクセスにおいて、メモリアクセスがページイン/スワップイン処理に変更され、管理権を持つメモリ領域内へのページコピーと一時退避メモリ領域の解放処理になる。その結果、管理OSからメモリアクセスが発生しているメモリ領域のページアウト/スワップ処理を低減することができる。

【0397】

また、別の例として、メモリ領域を読込専用領域として設定し、データの更新を抑える

50

ことにより、管理OSがメモリアクセスした際のページアウト/スワップアウトする処理量を抑えることができ、管理権放棄後に解放処理の負荷を高めるような動作を抑えることができる。

【0398】

『第5の実施の形態』

次に、本発明の第5の実施の形態について図面を参照して詳細に説明する。

【0399】

本発明の第5の実施の形態は、第1乃至第4の実施の形態に比べて、メモリ再構成における要求受付部だけが異なる。

【0400】

図17は、第5の実施の形態の構成を表すブロック図である。第1乃至第4の実施の形態に比べて、メモリ再構成要求解析部1130だけが加わる点で相違する。したがって、メモリ再構成要求解析部1130に関してだけ説明し、それ以外の構成および動作は、第1乃至第4の実施の形態と等しいため説明を省略する。

【0401】

メモリ再構成要求解析部1130は、メモリ再構成要求を受け付け、システムのメモリ再構成を開始させる。

【0402】

具体的には、まず、メモリ再構成要求解析部1130は、メモリ再構成要求を受け付ける。

【0403】

メモリ再構成要求を入力する元は、例えば、オペレータであったり、資源配分を制御する資源管理システムであったり、システム管理用のOSからのAPIだったりする。

【0404】

今後、説明の便宜上、オペレータからの入力を前提に説明を続けることにするが、要求入力元はオペレータとは限らないことは当然であるので注意されたい。

【0405】

メモリ再構成要求解析部1130は、メモリ再構成要求を受け付けたなら、各OSへ発行するメモリ構成変更要求を生成する。

【0406】

ここで、メモリ再構成要求の表現としては、全体、もしくは、特定の実アドレス空間の範囲を指定する方法がある。

【0407】

例えば、「0～1GBの領域はOS1、1～2GBの領域はOS2に割当て」や「0～1GBの領域はOS1に割当て」などである。

【0408】

これらは、実メモリ管理部1110、または、管理OS情報管理部1111を介して管理OS情報を参照して現在のメモリ構成を把握し、メモリ再構成要求と比較することにより差異を求め、さらに、アドレス変換などを計算することで、それぞれのOSに対するメモリ構成変更要求を生成することができる。

【0409】

メモリ再構成要求の別の表現としては、メモリの増減量や、マシンアドレス空間の範囲で指定する方法がある。

【0410】

例えば、「OS1は1GBの容量を削減し、OS2は1GBの容量を増加させる」や「OS1は高位1GBを削減し、OS2は1GBの容量を増加させる」「OS1はOS内の1GB～2GBの容量を削減し、OS2は1GBの容量を増加させる」などである。

【0411】

これらは、OS毎に命令を分割し、それぞれのOSに対するメモリ構成変更要求を生成することができる。

【0412】

10

20

30

40

50

さらに、これらの要求は、複数のより小さな要求を表すメモリ構成変更要求に分割して生成することができる。

【0413】

例えば、「OS1は1GBの容量を削減する」という要求は、「OS1は256MBの容量を削減する」という要求4つに分割し、生成することもできる。

【0414】

メモリ再構成要求解析部1130は、生成したメモリ構成変更要求を、要求するOS1200- i ($i=1 \sim n$) に対応するメモリ構成変更要求制御部1230- i ($i=1 \sim n$) に発行する。

【0415】

これにより、個々のOSのメモリ変更を促し、メモリ再構成要求に対する構成変更処理の一連を行うことができる。

【0416】

生成したメモリ構成変更要求を発行する方法には、同時に複数のメモリ構成変更要求を発行する方法、1つずつ順序だてて生成したメモリ構成変更要求を発行する方法がある。

【0417】

さらに、生成したメモリ構成変更要求を発行する方法には、メモリを削減するか、増加させるかの情報を利用して、発行させる順序を決定することができる。

【0418】

例えば、削減する要求を先に発行し、増加させる要求を削減する要求の処理終了後に発行させるといったことができる。また、別の例として、一部の削減する要求を先に発行し削減させた後、できた未使用領域の容量に対応する増加要求を発行する。これを1サイクルとして、サイクルを繰り返すことで全部の要求を発行する。といったことができる。さらに、別の例としては、前記の1サイクルで行う例において、あるサイクルの増加要求を発行するときに、別のサイクルの削減要求を発行するという、サイクル間の発行タイミングを重ねるといったことができる。

【0419】

さらに、生成したメモリ構成変更要求を発行する方法には、発行するOSの情報を利用して、発行させる順序を決定することができる。

【0420】

例えば、特定のOSに対する要求は同時に発行しないといったことができる。

【0421】

メモリ再構成要求解析部1130は、メモリ構成変更要求制御部1230- i ($i=1 \sim n$) を通してメモリ構成変更要求の処理結果が報告された場合、必要に応じて、オペレータに結果を伝えることができる。

【0422】

メモリ再構成要求解析部1130は、すべての生成したメモリ構成変更要求に対して結果が返答された場合、オペレータへ最終結果を報告する。

【0423】

次に、本実施の形態の動作について詳細に説明する。

【0424】

図17の構成図、および、図18のフローチャート図を参照して、メモリ再構成要求を受信したときの処理について説明する。

【0425】

メモリ再構成要求解析部1130が、メモリ再構成要求を受信することで、メモリ再構成処理が開始される。

【0426】

メモリ再構成要求解析部1130は、個々のOSへメモリ構成変更を要求できる量に、メモリ再構成要求からメモリ構成変更要求を生成する（ステップS801）。

【0427】

メモリ再構成要求解析部1130は、生成したメモリ構成変更要求のうち、メモリ構成変更

10

20

30

40

50

要求制御部1230- i ($i=1 \sim n$)に発行していない要求があるかどうかをチェックする(ステップS802)。

【0428】

チェックの方法は、例えば、要求にチェックフラグをつけておいて管理する方法がある。また別の例として、リスト構造として記憶しておき、発行済み箇所を覚えておくことで、発行済み箇所からのリストの前後によりチェックする方法がある。また、別の例として、発行済みリストと発行前リストを作成し、管理する方法がある。

【0429】

メモリ再構成要求解析部1130は、未発行のメモリ構成変更要求が存在した場合、未発行のメモリ構成変更要求のうち、前述のさまざまな基準で選択されるメモリ構成変更要求を、メモリ構成変更要求を要求するOS1200- i ($i=1 \sim n$)に対応するメモリ構成変更要求制御部1230- i ($i=1 \sim n$)に発行し、結果を得る(ステップS803)。

10

【0430】

ステップS803において、結果を途中結果としてオペレータに報告してもよい。

【0431】

メモリ再構成要求解析部1130は、未発行のメモリ構成変更要求が存在しない場合、最終結果をまとめて、オペレータへ報告する(ステップS803)。

【0432】

これにより、メモリ再構成要求を受信したときの処理が完了する。

【0433】

20

次に、第5の実施の形態の効果について説明する。

【0434】

第1乃至第4の実施の形態は、OSに直接要求が届いている。しかし、実メモリ管理、さらに、仮想計算機環境管理においては、仮想計算機環境の制御用プログラムであるVMMが一元制御しているのが一般的である。すなわち、制御するオペレータは通常、VMMに関連付けられるべきで、VMMに制御情報を発行する。第5の実施の形態は、上記のような状況に対応するためのものである。例えば、特許文献1の実施例で説明されている構成テーブルによるシステム変更要求などに対応する方式になる。

【0435】

本実施の形態の構成の利点としては、VMMのオペレータしかメモリの構成を変更する要求を発行できない点である。すなわち、仮想計算機環境全体を管理するVMMとしてのオペレータと個々のOSを管理するOSとしてのオペレータ(管理者)を明確に分けることができる点である。

30

【0436】

次に、上述した第1乃至第5の実施の形態に係る仮想計算機システムを実現する実計算機システムの実施例について説明する。

【0437】

図19は実計算機システムの第1の実施例のブロック図であり、メモリ装置1、および、メモリ制御装置2と、演算装置3と、入出力装置4、および、補助記憶装置5とそれらを接続するバス6とから構成される。

40

【0438】

前述した各実施の形態におけるVMM1000と各OS1200- i ($i=1 \sim n$)は、補助記憶装置5に保存され、実行時に入出力装置4、および、メモリ制御装置2を通してメモリ装置1に展開され、演算装置3により動作する。すなわち、通常のコンピュータプログラムとして動作する。メモリ空間対応関係テーブル/管理OS情報テーブル/利用OS情報テーブル10は、管理OS情報、利用OS情報、および、メモリ対応表を管理している。これは前述の通り、3つの表を一まとめとして管理している場合の一例を表している。実メモリアドレスは、実メモリ空間でのアドレス値を示す。管理OS、利用OSは、実メモリ空間において、それぞれ、管理権、利用権を持っているOSのIDを示す。これらのメモリ空間に対して、実メモリ管理部で、管理OS、利用OSへ対応付けし割り当てたマシンアドレスが、それぞれ、管理OSアド

50

レス、利用OSアドレスである。nullは、管理OSや利用OSが存在しないことや、アドレス対応関係が無いことを示している。

【0439】

まず、VMM1100が先行して展開され動作する。次に、VMM1100上に、管理用のOS(ここでは、OS1000-1とする)を起動する。このときに、VMM1100、及び、実メモリ管理部1110、実メモリ構成制御部1120、および、それらに含まれる各機能を用いて、管理用OS1000-1の管理OS情報を変更し、メモリ領域を割当てする。

【0440】

管理用OS1000-1がメモリアクセスする度に、利用OS情報を参考にしながら、VMM1100、及び、実メモリ管理部1110、実メモリ構成制御部1120はメモリ領域のアクセスチェックにより、実割当てを行う。

10

【0441】

また、その後、ゲストOSと呼ばれる、実際に利用環境として用いるOS1000-i(i=2~n)が起動要求、停止要求が繰り返され、複数のVMが構築される。このときに、VMM1100、及び、実メモリ管理部1110、実メモリ構成制御部1120の各処理部が連動し、実メモリ領域の管理をし、各OS1000-i(i=2~n)に対して管理OS情報、利用OS情報によるメモリアクセスを制御する。

【0442】

これにより、各仮想計算機環境に対して効率よく、かつ、整合性の取れたメモリ割当て、メモリ再構成が実現できる。

20

【0443】

また、別の例として、VMMはホストOSと呼ばれるプログラムが起動した後、ホストOS上、もしくは、同列に起動し、各VMが起動することもある。

【0444】

次に図20は、実計算機システムの第2の実施例のブロック図である。この実施例では、VMM1100はコ・プロセッサ7により実行される。VMM1100のプログラム本体は、ROMやFLASH RAMなどのコ・プロセッサ用の記憶装置8に記憶されている。すなわち、VMM1000がファームウェアとして提供される。

【0445】

計算機システム起動後、ファームウェアが起動し、バス制御などを開始し、論理区画を作成する。論理区画により分割された環境1つがVMに相当する。このときに、論理区画は設定として記憶している場合もあれば、コマンド入力により作成することもある。論理区画ごとにOSの起動、終了が制御可能になり、複数のOSが起動できる。

30

【0446】

コ・プロセッサ7上のVMM1100を通して、メモリ領域を論理区画へ割り当て、再構成、解放するときに、第1乃至第5の実施の形態を通し、各論理区画に対して効率よく、かつ、整合性の取れたメモリ割当て、メモリ再構成が実現できる。

【0447】

次に図21は、実計算機システムの第3の実施例のブロック図である。この実施例では、VMM1100はメモリ装置1、および、メモリ制御装置2に展開され実行される。VMM1100の多くの機能はメモリ装置1に展開され、実メモリ管理部1110、実メモリ構成制御部1120がメモリ制御装置2に展開される。これは、Memory Management Unit(MMU)と呼ばれるメモリ制御用プロセッサに、本発明のメモリ区画制御機能を実現することに相当する。VMM1100はMMUによるサポートを利用しながら動作する。それ以外の動作は、第1の実施例と同じである。

40

【産業上の利用可能性】

【0448】

本発明は、仮想計算機技術を用いたサーバ統合やシンククライアント・サーバに、より効率的にメモリを活用するために適用できる。また、仮想計算機技術によるセキュリティを向上させる組込み端末に、より効率的にメモリを活用するために適用できる。また、開発

50

環境と事務環境などを混在させたり、セキュリティ目的として導入したりする仮想計算機技術製品に、より効率的にメモリを活用するために適用できる。

【図面の簡単な説明】

【0449】

【図1】仮想計算機環境におけるメモリ区画分割と仮想化空間への対応付けを示す概念図である。

【図2】VMの動的メモリ構成変更の過程を示すメモリ情報の概念図である。

【図3】特許文献1のメモリ再構成手法の構成を示すブロック図である。

【図4】特許文献1のメモリ再構成手法の動作過程を示すシーケンス図である。

【図5】本発明の第1の実施の形態の構成を示すブロック図である。

10

【図6】本発明の第1の実施形態におけるメモリ構成変更処理の動作を示すフローチャートである。

【図7】本発明の第1の実施の形態における利用OSの選択処理の動作を示すフローチャートである。

【図8】本発明の第1の実施の形態におけるOSのメモリ解放処理の動作を示すフローチャートである。

【図9】本発明の第1の実施の形態におけるOSのメモリアクセス判定の動作を示すフローチャートである。

【図10】本発明の第1の実施の形態におけるメモリ再構成の動作過程を示すシーケンス図である。

20

【図11】本発明の第1の実施の形態におけるメモリ再構成の過程を示すメモリ情報の概念図である。

【図12】本発明の第2の実施の形態における利用OSの選択処理の動作を示すフローチャートである。

【図13】本発明の第1の実施の形態における管理OS、利用OSの変化の過程を示す管理テーブルの概念図である。

【図14】本発明の第2の実施の形態における管理OS、利用OSの変化の過程を示す管理テーブルの概念図である。

【図15】本発明の第3の実施の形態におけるOSのメモリアクセス判定の動作を示すフローチャートである。

30

【図16】本発明の第4の実施の形態におけるメモリ構成変更処理の動作を示すフローチャートである。

【図17】本発明の第5の実施の形態の構成を示すブロック図である。

【図18】本発明の第5の実施の形態におけるVMMからのメモリ再構成要求を処理する動作を示すフローチャートである。

【図19】本発明の仮想計算機システムを実現する実計算機システムの第1の実施例のブロック図である。

【図20】本発明の仮想計算機システムを実現する実計算機システムの第2の実施例のブロック図である。

【図21】本発明の仮想計算機システムを実現する実計算機システムの第3の実施例のブロック図である。

40

【符号の説明】

【0450】

1100 ... 仮想計算機モニタ (VMM、hypervisor)

1110 ... 実メモリ管理部

1111 ... 管理OS判定部

1112 ... 利用OS判定部

1120 ... 実メモリ構成制御部

1121 ... メモリ解放宣言部

1122 ... メモリ解放制御部

50

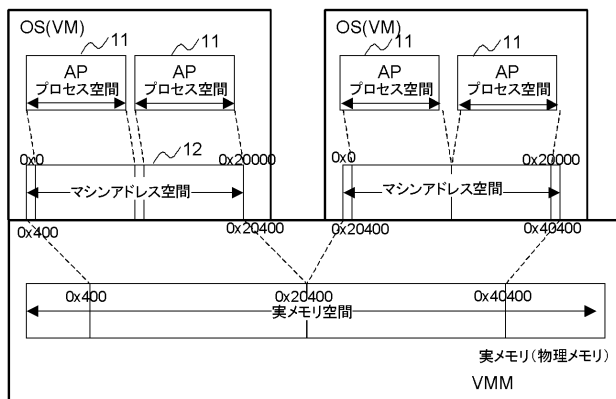
- 1 1 2 3 ... メモリ拡張制御部
 1 1 2 4 ... メモリアクセス監視部
 1 1 3 0 ... メモリ再構成要求解析部
 1 2 0 0 - i ($i = 1 \sim n$) ... オペレーティングシステム (OS)、または、仮想計算機 (VM)
 1 2 1 0 - i ($i = 1 \sim n$) ... OS(VM) 内のメモリ管理部
 1 2 2 0 - i ($i = 1 \sim n$) ... OS(VM) 内のメモリアクセス制御部
 1 2 3 0 - i ($i = 1 \sim n$) ... OS(VM) 内のメモリ構成変更要求制御部
 1 2 4 1 - i ($i = 1 \sim n$) ... OS(VM) 内のメモリ解放領域探索部
 1 2 4 2 - i ($i = 1 \sim n$) ... OS(VM) 内のメモリ解放部
 1 2 4 3 - i ($i = 1 \sim n$) ... OS(VM) 内のメモリ拡張部
 1 ... 主記憶装置
 2 ... 主記憶制御装置
 3 ... 演算装置
 4 ... 入出力制御装置
 5 ... 補助記憶装置
 6 ... バス
 7 ... コ・プロセッサ
 8 ... VMM 記憶装置
 9 ... 仮想計算機対応メモリ管理機能付主記憶制御装置
 1 0 ... 管理OS情報テーブル、および、利用OS情報テーブル

10

20

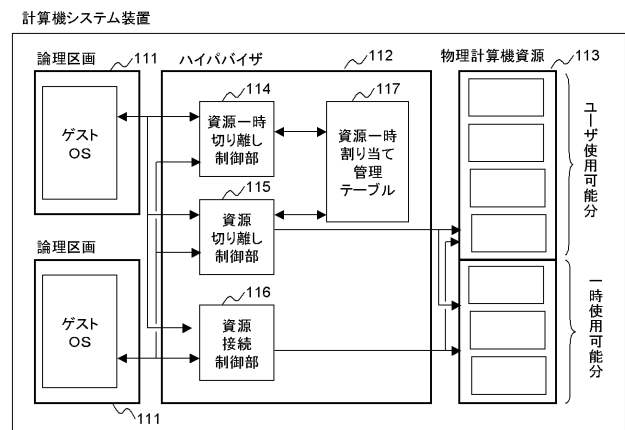
【図 1】

【図 1】



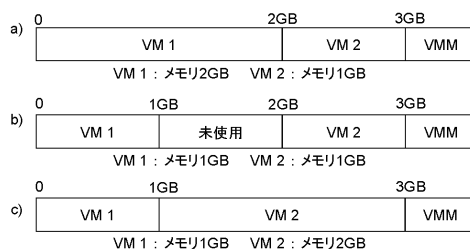
【図 3】

【図 3】



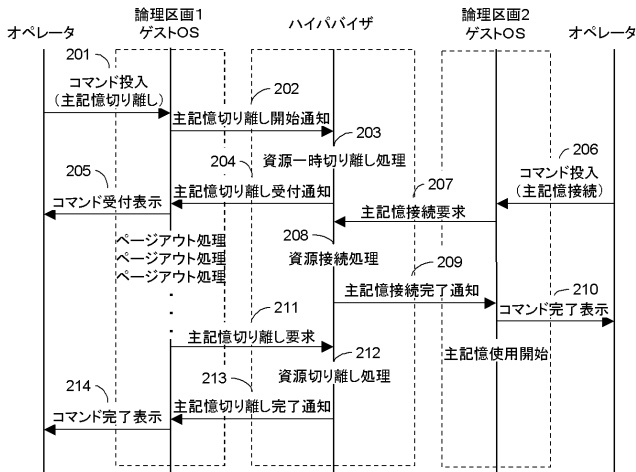
【図 2】

【図 2】



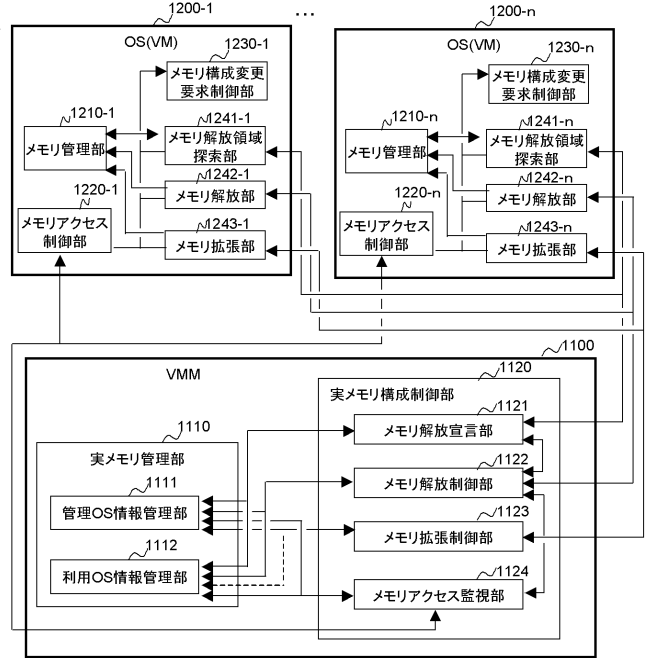
【図4】

【図4】



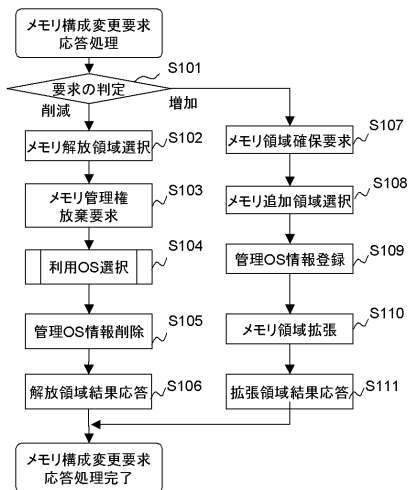
【図5】

【図5】



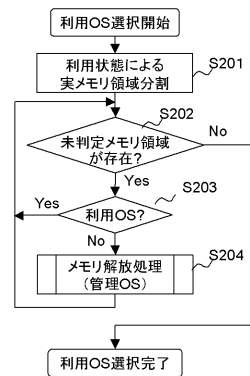
【図6】

【図6】



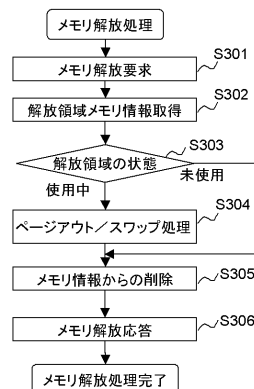
【図7】

【図7】



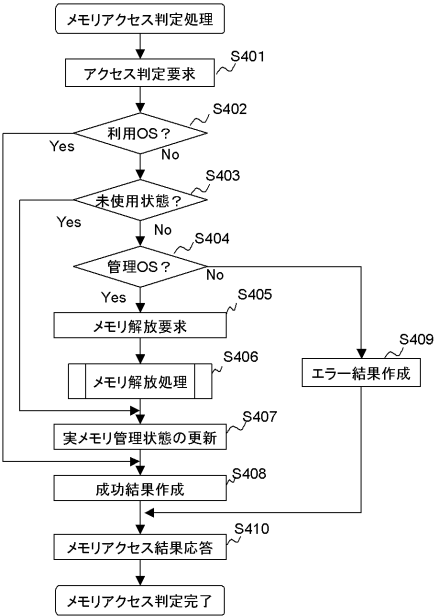
【図8】

【図8】



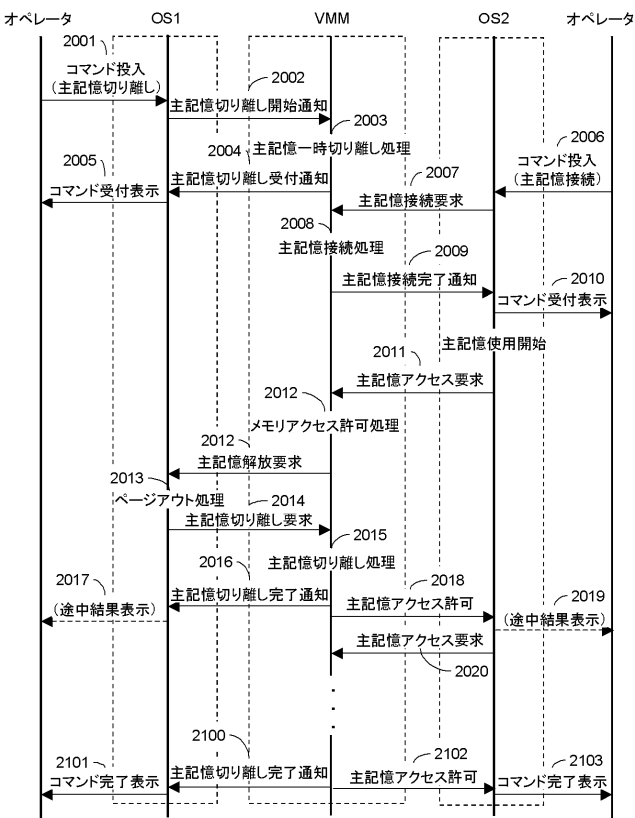
【 図 9 】

【図9】



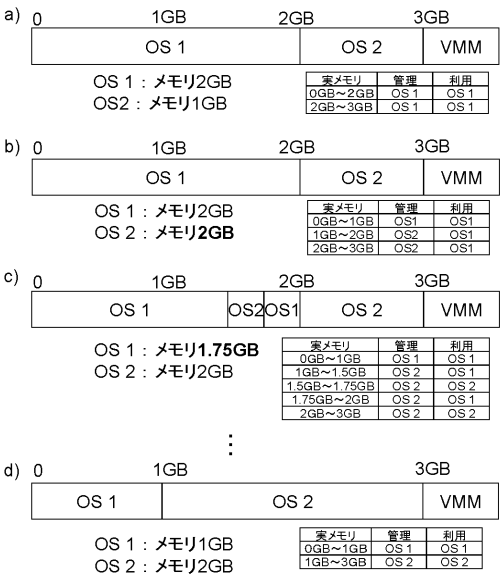
【 図 1 0 】

【図10】



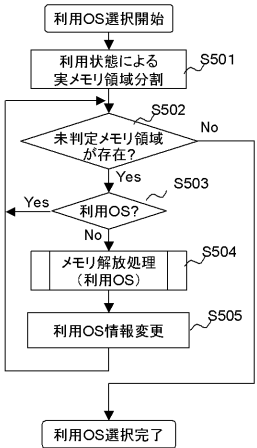
【 図 1 1 】

【図11】



【 図 1 2 】

【図12】



【図 1 3】

【図 1 3】

a)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	未	未	未	未	未	未	未	未
	管理OS情報	未	未	未	未	未	未	未	未
b)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	未	未	未	未	未	未	未	未
	管理OS情報	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1
c)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1
	管理OS情報	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1
d)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1
	管理OS情報	OS 1	OS 1	OS 1	OS 1	未	未	未	未
e)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1
	管理OS情報	OS 1	OS 1	OS 1	OS 1	OS 2	OS 2	OS 2	OS 2
f)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	OS 1	OS 1	OS 1	OS 1	OS 2	OS 1	OS 1	OS 2
	管理OS情報	OS 1	OS 1	OS 1	OS 1	OS 2	OS 2	OS 2	OS 2
g)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	OS 1	OS 1	OS 1	OS 1	OS 2	OS 1	OS 1	OS 2
	管理OS情報	OS 1	OS 1	OS 1	OS 1	OS 2	OS 2	未	未
h)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	OS 1	OS 1	OS 1	OS 1	OS 2	OS 1	OS 1	OS 2
	管理OS情報	OS 1	OS 1	OS 1	OS 1	OS 2	OS 2	OS 3	OS 3

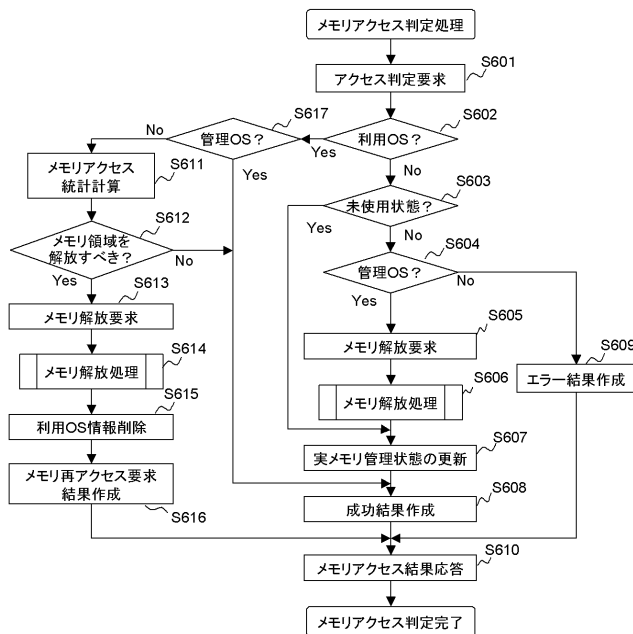
【図 1 4】

【図 1 4】

a)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	未	未	未	未	未	未	未	未
	管理OS情報	未	未	未	未	未	未	未	未
b)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	未	未	未	未	未	未	未	未
	管理OS情報	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1
c)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1
	管理OS情報	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1
d)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1
	管理OS情報	OS 1	OS 1	OS 1	OS 1	未	未	未	未
e)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1	OS 1
	管理OS情報	OS 1	OS 1	OS 1	OS 1	OS 2	OS 2	OS 2	OS 2
f)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	OS 1	OS 1	OS 1	OS 1	OS 2	OS 1	OS 1	OS 2
	管理OS情報	OS 1	OS 1	OS 1	OS 1	OS 2	OS 2	OS 2	OS 2
g)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	OS 1	OS 1	OS 1	OS 1	OS 2	OS 1	OS 2	OS 2
	管理OS情報	OS 1	OS 1	OS 1	OS 1	OS 2	OS 2	未	未
h)	メモリ領域(MB)	0~256	~512	~768	~1024	~1280	~1536	~1792	~2048
	利用OS情報	OS 1	OS 1	OS 1	OS 1	OS 2	OS 1	OS 2	OS 2
	管理OS情報	OS 1	OS 1	OS 1	OS 1	OS 2	OS 2	OS 3	OS 3

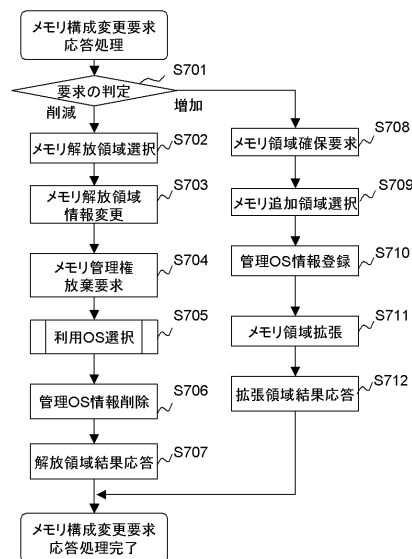
【図 1 5】

【図 1 5】



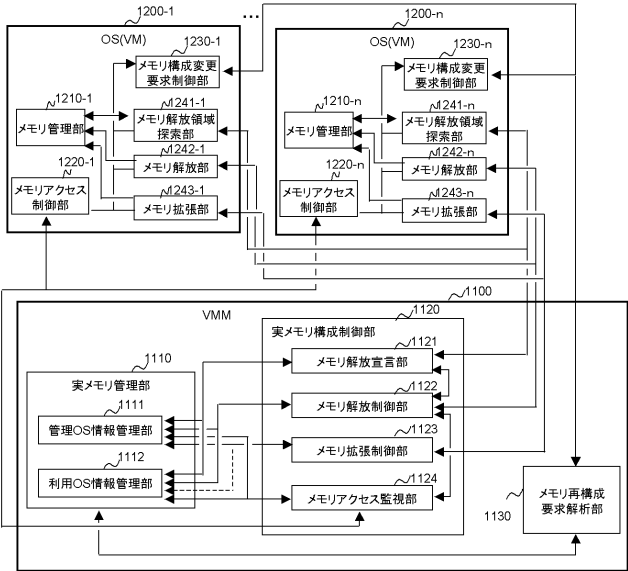
【図 1 6】

【図 1 6】



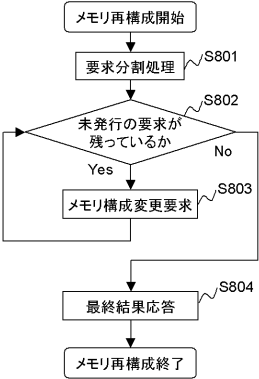
【図 17】

【図 17】



【図 18】

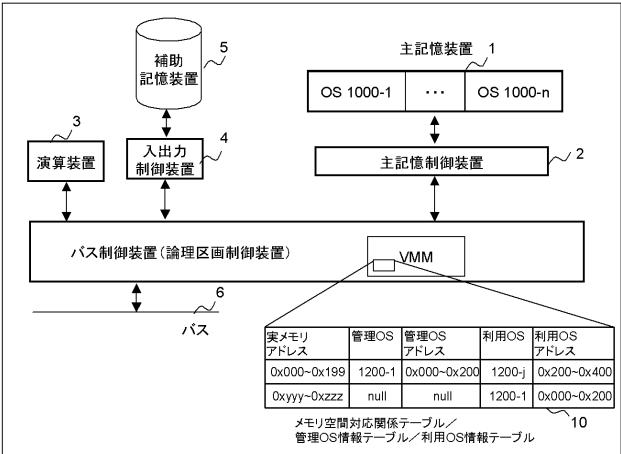
【図 18】



【図 19】

【図 19】

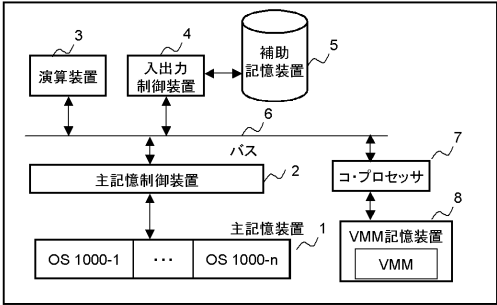
計算機システム装置



【図 20】

【図 20】

計算機システム装置



【図 21】

【図 21】

計算機システム装置

