

OS/omicon 第4版マイクロカーネルの外部ページインタフェースの設計

1F-9

森永智之, 佐藤元信, 早川栄一, 並木美太郎, 高橋 延匡
(東京農工大学工学部)

1. はじめに

マイクロカーネルの外でページを定義するユーザ定義ページは、分散共有メモリなど多様な用途に有効であり、多くのマイクロカーネルでサポートされている。

我々の開発している OS/omicon 第4版[3](V4)は、手書き文字、音声や画像といったパターンデータ処理を指向している。パターンデータは種類が多様であり、動画は時間制約を持ちシーケンシャルにアクセスされるなど、メモリのアクセスパターンもデータの属性によって異なることが予想されるので、データの属性に応じたページングを行うことが必要である。

そこで我々は、データの属性に応じたページングポリシーを提供できるユーザ定義ページのためのインタフェースを V4 マイクロカーネルでサポートすることにした。本報告では、このインタフェースの設計について述べる。

2. OS/omicon 第4版マイクロカーネル

V4 はパターンデータ処理を指向しており、多様なデータを扱う必要がある。そこで、V4 では資源管理の拡張性に優れたマイクロカーネル構成を採用する。システムの全体構成を図1に示す。

V4 マイクロカーネルは次の特徴を持つ。

(1) 単一2次元アドレス空間の提供

データ間の関係をポインタで表現するため、全システムでアドレス空間を共有し、メモリをオブジェクト

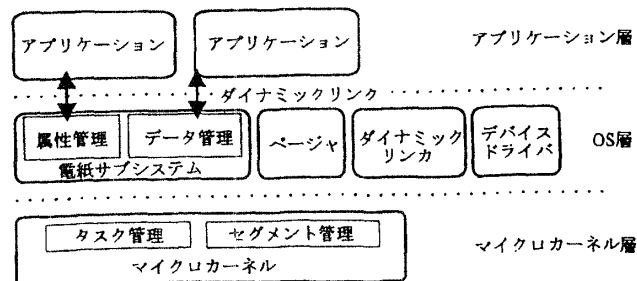


図1 OS/omicon 第4版の全体構成

Design of the External Pager Interface of the Microkernel for OS/omicon Version 4

Tomoyuki Morinaga, Motonobu Satoh, Eiichi Hayakawa, Mitarou Namiki and Nobumasa Takahashi

Tokyo University of Agriculture and Technology

id とオフセットでアクセスする単一2次元アドレス空間を採用する。

(2) 関数呼出しによるシステム構成

メッセージ通信におけるコンテキスト切替えのオーバーヘッドなどを解決するため、関数呼出しでシステムを構成する throw と呼ぶ構成モデルを採用する。本システムでは SVC も関数呼出しで実装される。

(3) 柔軟な保護機構の提供

資源管理の拡張において性能と信頼性とのトレードオフを柔軟に図るため、ユーザが保護のポリシーを設定できる保護空間と呼ぶ保護機構を提供する。

3. ページインタフェースの設計方針

本マイクロカーネルのページインタフェースの設計では、次のような設計方針をとった。

(1) アクセスパターンに応じたページングポリシーを実現可能にする

Mach[1]におけるメモリオブジェクトのようなインタフェースでは、ページアウト候補の選定など、アクセスパターンを考慮したポリシーを実装することは難しい。そこで、本システムではページテーブルの書込みや TLB のフラッシュといった、メモリのアクセスパターンに応じたページングポリシーを実現できるようなインタフェースを提供する。また、ハードウェアの機能を最大限に利用するため PTE の構成などはそのままユーザに提供する。SPIN[2]や Exo-kernel では MMU 管理をユーザが定義できるが、我々のシステムはアドレス空間や保護機構はマイクロカーネルで提供し、ページングをユーザ定義可能にする。

(2) マイクロカーネルをページャから保護する

マイクロカーネルでは、ユーザ定義ページャによってマイクロカーネルが破壊されないように保護を行う。なお、ユーザ定義ページャ同士の保護は、ページャ間でのページ共有、コピーオンライトなどを実現した後で検討することにし、今回の設計では見送ることにした。

4. ページインタフェースの設計

ここでは、V4 マイクロカーネルにおけるページャ

表1 提供するプリミティブ

INT define_pager(INT (*pf_handler)())	ページャ、ページフォルト ハンドラを定義する
INT allocate_vir(UDEBLWORD size, UDEBLWORD *top, UDEBLWORD *siz)	仮想空間を size 分確保する (確保されたサイズは siz に入る)
INT allocate_page(UDEBLWORD pages, UBYTE *bitmap)	物理ページを pages 分確保する 結果は bitmap に格納される
INT flush_TLB(VOID)	TLB をフラッシュする
INT write_PTE(UDEBLWORD index, UDEBLWORD pte)	ページテーブルの index に pte を書き込む

インタフェースの設計について述べる。また、プリミティブを表1に示す。

本システムでは、アドレス空間がオブジェクト id とオフセットからなる2次元アドレス空間を採用する。そこで、オブジェクトの属性としてページングポリシーを設定することにした。ユーザ定義ページャはそれぞれ自分がページングを担当するオブジェクトの集合、物理ページの集合、ページテーブルを管理し、独自のポリシーでページングを行う(図2)。物理ページと仮想空間はページャの初期化時にマイクロカーネルに SVC(関数呼出し)を発行することで確保する。

本マイクロカーネルで提供するプリミティブは次のように分類される。

(1) 初期化

各ページャは、初期化時に自分が管理する仮想アドレス空間と物理ページをマイクロカーネルに要求する。また、ページフォルトの通知先も登録する。これ以降は、自分が管理するオブジェクトに関するページフォルトがマイクロカーネルから通知され、物理ページやページテーブルを操作してマッピングを行う。ページアウト候補の選定アルゴリズムや、前もって複数ページをマップしておくといった、独自のポリシーでページングすることができる。

(2) ページテーブル操作

ユーザ定義ページャからマイクロカーネルを保護するためには、ページテーブルに設定される物理ペー

ジをマイクロカーネルが監視する必要がある。そこで、本マイクロカーネルではページテーブルへの書き込みを SVC 化して、マイクロカーネルがマップされている物理ページがセットされないかチェックする。なお、ページテーブルの読出しについては SVC とせず、直接アクセスすることを許す。また、TLB をフラッシュするプリミティブも提供する。

(3) その他

マイクロカーネルを保護するためには、ページテーブルのアクセスだけでなく、DMA やカーネルのスワップ領域へのアクセスを監視することも必要である。そこで、本マイクロカーネルでは、DMA や二次記憶デバイスへのアクセスも SVC とし、マイクロカーネルでチェックする設計とした。

5. おわりに

本報告では、データのアクセスパターンに応じたページングポリシーを持つユーザ定義ページャを実装できるインタフェースの設計について述べた。

今後はこのインタフェースを実装し、アクセスパターンの異なる複数のデータ形式をページングした場合の性能評価などを行う予定である。

謝辞

本研究は、文部省科学研究費補助金(基盤研究(B)(2)課題番号08458064)により行われた。

参考文献

- [1] Mike Accetta et al.: Mach: A New Kernel Foundation For UNIX Development, USENIX Summer '86, pp.93-112
- [2] Brian Bershad et al.: Safety and Performance in the SPIN Operating System, Proc. 15th SIGOPS, pp.267-284
- [3] Eiichi Hayakawa, et al.: Basic Design of SHOSHI Operating System that Supports Handwriting Interfaces, IPSJ, Vol35, No.12, pp.2590-2601

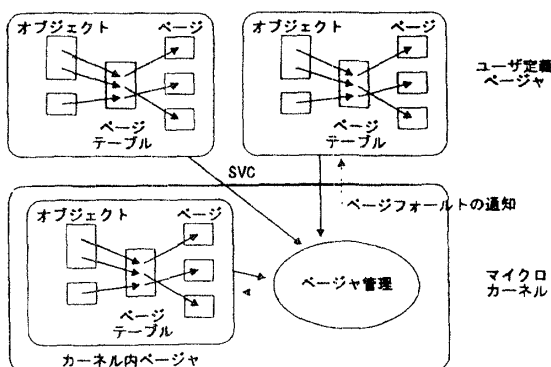


図2 ユーザ定義ページャのモデル