

オペレーティングシステム

3章 メモリ管理

3.2節 仮想メモリーメインメモリの隠ぺい（前半）



大阪大学大学院情報科学研究科
村田正幸

murata@ist.osaka-u.ac.jp

<http://www.anarg.jp/>



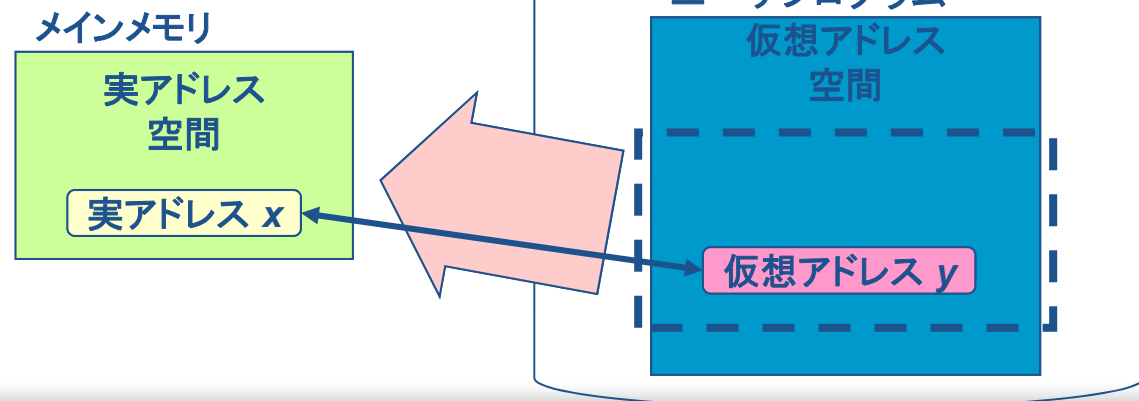
3.2 仮想メモリーメインメモリの隠ぺい

- 仮想メモリ機能

- ユーザプログラムやハードウェア装置(特に、メインメモリと対になる内部装置であるプロセッサ)から、物理的なメインメモリを「隠ぺい」する
- ユーザプログラムやプロセッサには、一定サイズの巨大な仮想のすなわち論理的なメモリを見せかける
- OSによるメモリ管理機能を支える原理

物理メモリの大きさを意識することなくプログラミングが可能になる

```
main() {float p; p=1000000000; printf("%d¥n", *p); }
```





3.2 仮想メモリーメインメモリの隠ぺいー

3.2.1 メモリの性質ーメモリ階層と参照局所性ー

- メモリ機能
 - ー ハードウェア:メモリ階層
 - ー ソフトウェア:参照局所性

[1] メモリ階層

定義3.1(メモリ階層)

容量とアクセス時間とのトレードオフすなわち両立しない関係によって識別できるメモリ機能、および、それを実現するメモリ装置の種類のそれぞれをメモリ階層という。

- ー メモリ階層間のトレードオフ(主として、容量とアクセス時間)によって、種々のメモリ装置の種類および機能や特性を表現
- ー コンピュータシステムにおけるメモリ装置の構成や設定時に、トレードオフとなる容量とアクセス時間のそれぞれの特性によってメモリ装置を使い分ける、すなわち、適材適所を図るための指標



3.2 仮想メモリーメインメモリの隠ぺいー

3.2.1 メモリの性質ーメモリ階層と参照局所性ー

[2] 主要なメモリ階層

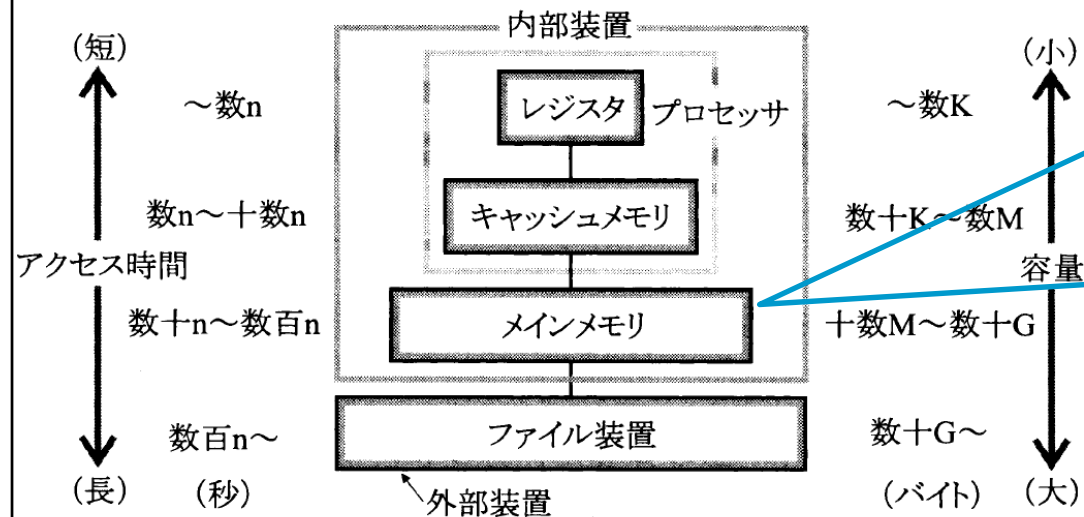
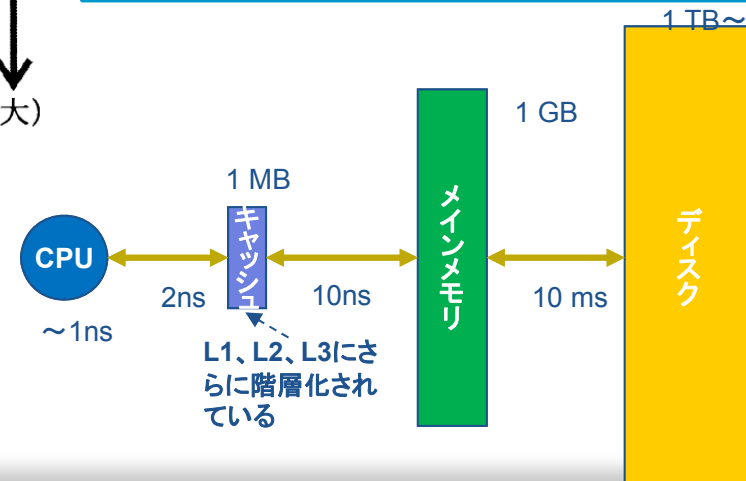


図 3.5 主要なメモリ階層

- ・ プロセッサと対になる主要な“内部装置”であり、内部メモリあるいは1次メモリともいう
- ・ 「生きているプロセス」すなわち「プロセッサが実行中の命令や使用中のデータ」を保持する
- ・ 仮想メモリ機構の直接の対象となる物理的または実際のメモリ(実メモリ)であり、OSがプロセッサやユーザプログラムなどから隠ぺいする。
- ・ **プロセッサが直接アクセスできるので、ファイル装置と相対的に比較すると、格納機能よりもアクセス機能を重視する、すなわち、小容量でも高速性を必要とするメモリ階層**
- ・ 現代の代表的なメインメモリは**DRAM**





3.2 仮想メモリーメインメモリの隠ぺいー

3.2.1 メモリの性質ーメモリ階層と参照局所性ー

[2] 主要なメモリ階層

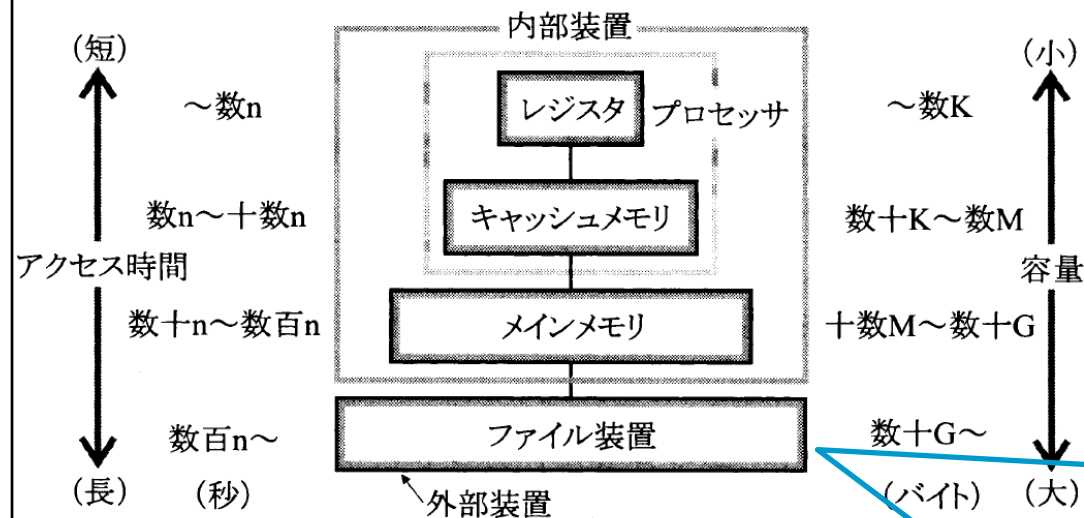


図 3.5 主要なメモリ階層

- ユーザーが直接に操作(生成、消去、併合、分割、編集など)するファイルを格納する外部装置であり、補助メモリ(外部メモリ、あるいは2次メモリ)という
- プロセッサが使用中でない、すなわち、まだプロセスになっていない、プログラムやデータも格納する
- **仮想メモリ機構では、メインメモリのバックアップ(退避)用メモリとして機能し、論理的また仮想のメモリ(仮想メモリ)を構成するのに用いる**
- プロセッサから見ると外部装置あるいは周辺装置という位置付け。プロセッサからはメインメモリをいったん経由する間接アクセスとなる。
- **メインメモリと比較すると、アクセス機能よりも格納機能を重視する、すなわち、低速でも大容量を必要とするメモリ階層**
- 現代の代表的なファイル装置はハードディスクドライブ装置(HDD)



3.2 仮想メモリーメインメモリの隠ぺいー

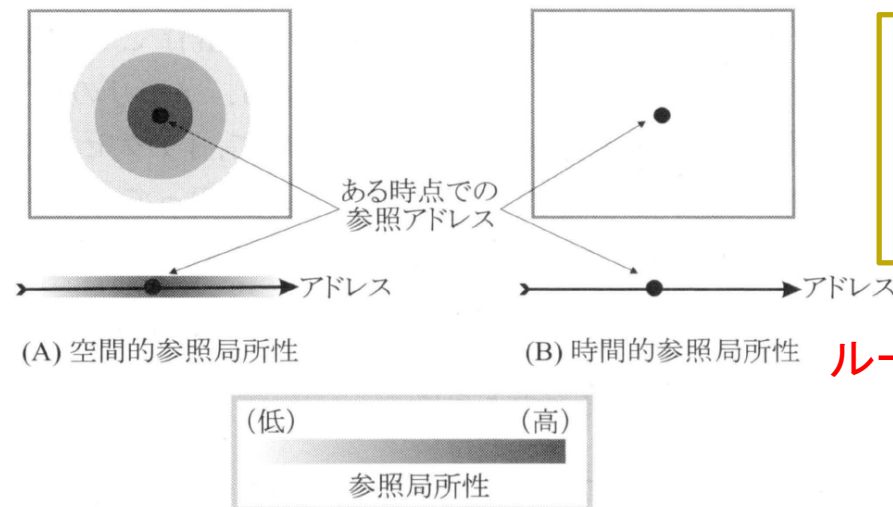
3.2.1 メモリの性質ーメモリ階層と参照局所性ー

[3] 参照局所性

定義3.2 (参照局所性)

「あるプログラム(実際には、マシン命令列)がアクセスあるいは参照する命令やデータのアドレス(address;番地、格納場所)は一部あるいは特定の個所に集中する」ことを参照局所性あるいは「参照局所性がある」または「参照局所性が高い」という。

一度アクセスしたアドレスに近接する、すなわち、格納場所が近いアドレスは近いうちにアクセスする可能性が高い。



一度アクセスしたアドレスそのものは近いうちに、たとえば、同じプログラムの実行中にまたアクセスする可能性が高い

ループ、配列

ループ、関数呼び出し

OSは、隣接するメモリ階層の特性を利用すれば、参照局所性が高いプログラムやプロセスを、当該メモリ階層において空間的または時間的に効率良く管理し処理できるようになり、結果として、当該メモリ階層そのものの機能を改善



参照の局所性の例

- type 1: write (赤)
- type 2: read (青)
- type 3: execute (緑)

空間的局所性の例：
時間的局所性があるかどうか
はわからない。

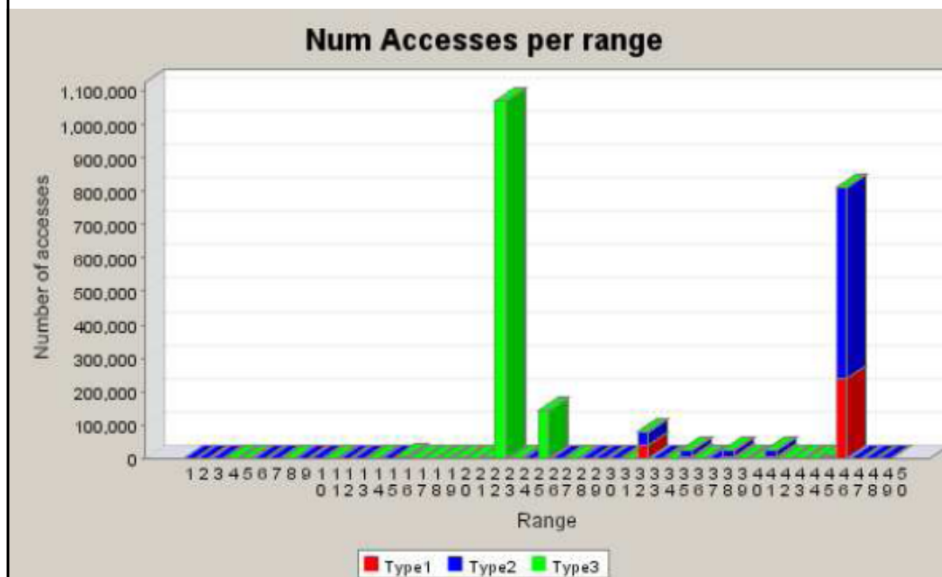


Fig. 2. Access pattern of m-fft

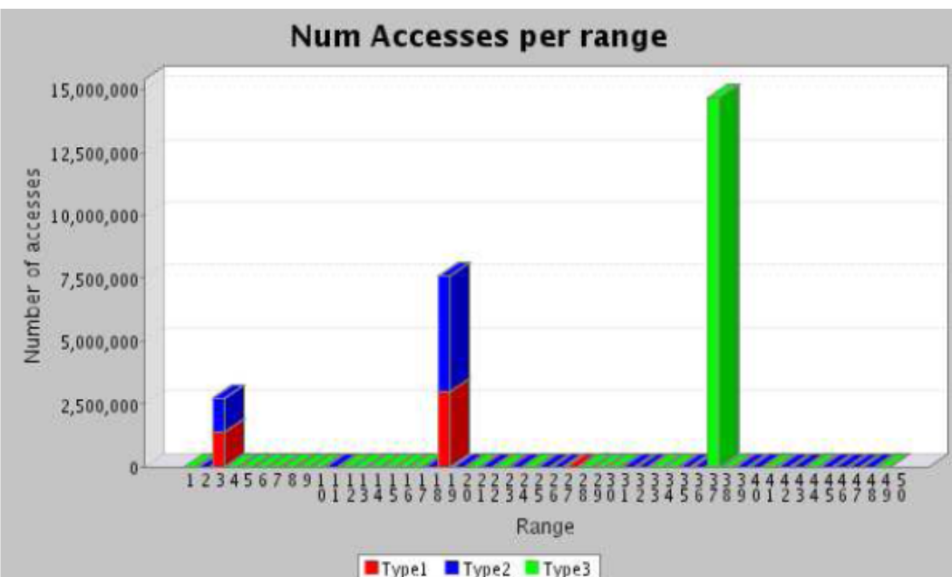


Fig. 1. Access pattern of tower of Hanoi program



3.2 仮想メモリーメインメモリの隠ぺいー

3.2.1 メモリの性質ーメモリ階層と参照局所性ー

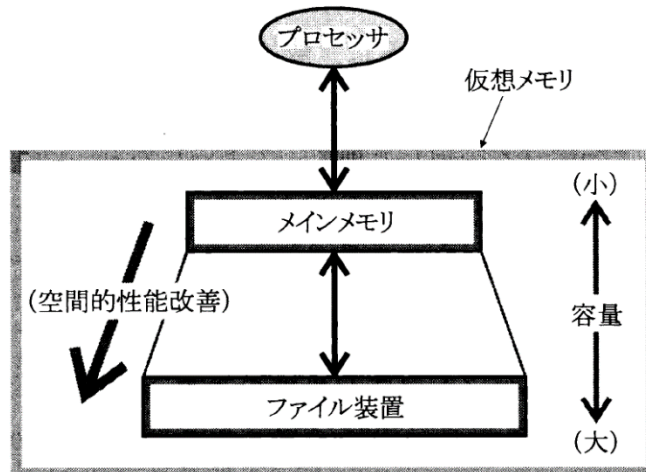
[4] メモリ階層によるメインメモリ性能の改善ー代表例: 仮想メモリー

・ 目標

- ー メインメモリとファイル装置の隣接する2種類のメモリ階層を利用してメインメモリ性能を空間的に改善する
- ー プロセッサから見えるメインメモリのアドレス空間という空間的制限を撤廃

定義3.3 (アドレス空間)

- ・ メモリには、バイトごとに振られているアドレスを指定することによってアクセスする。
- ・ メモリのアドレスは重複なし欠番なしの1次元で振る。
- ・ メモリにおいて、アクセス可能な(アクセス対象となる)アドレスを、「アドレスが振ってある領域あるいは空間」という意味で、アドレス空間という。
- ・ アドレス空間の大きさはメモリ領域のサイズすなわち容量も示している。



1. ファイル装置をメインメモリのバックアップメモリとする
2. マシン命令中のオペランドで指定するメインメモリのアドレス空間を、実際のすなわち物理的なメインメモリ容量とは独立にする



3.2 仮想メモリーメインメモリの隠ぺいー

3.2.2 仮想メモリとは？

[1] 仮想メモリの原理

- 仮想メモリは、実メモリと仮想メモリの2種類のメモリすなわちアドレス空間をそれぞれ独立に構成し、相互に対応付ける(マッピング)ことによって実現
- マシン命令中のオペランドで指定できるメモリのアドレス空間を仮想的に拡大しまた一定サイズに固定できる

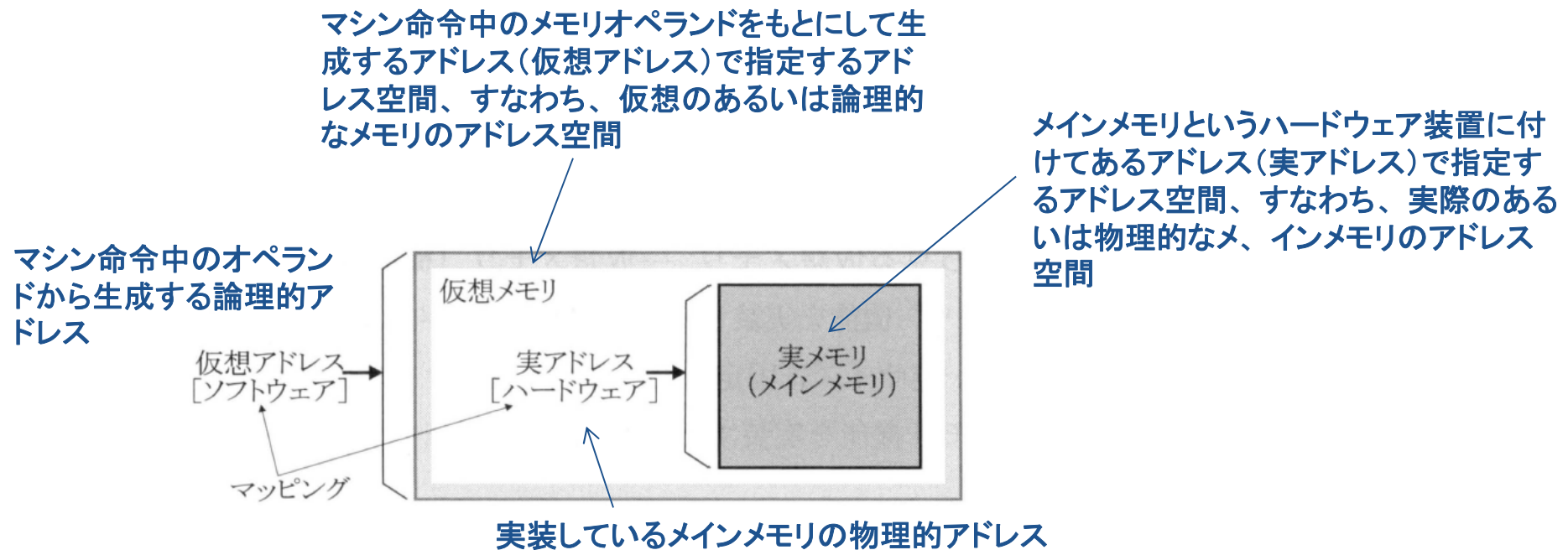


図 3.8 仮想メモリの原理



3.2 仮想メモリーメインメモリの隠ぺいー

3.2.2 仮想メモリとは？

[2] 仮想メモリの効果

- マシン命令の使用者からは、実メモリや実アドレス空間ではなく、一定で巨大なサイズの仮想メモリや仮想アドレス空間が見える
 - (a) プロセッサ: プログラムやプロセスを実行する
 - (b) OS: プロセスを管理する
 - (c) コンパイラ: マシン命令を生成する
- 実メモリであるメインメモリの多種多様な性能仕様(特に、容量)の相違を隠ぺいし、統一した性能仕様のメモリ領域やアドレス空間をハードウェア(プロセッサ)およびソフトウェア(OS、コンパイラ、ユーザプログラムなど)のそれぞれに、提示あるいは提供する
- 効果
 1. 実メモリであるメインメモリの利用効率が良くなる
 2. 実メモリであるメインメモリのサイズによる種々の制約を事実上撤廃できる



3.2 仮想メモリーメインメモリーの隠ぺいー

3.2.2 仮想メモリとは？

[3] OSから見る仮想メモリー仮想メモリ(概要)ー

- OSから見たメモリ:一定サイズに固定した、また、巨大サイズの(仮想)メモリ

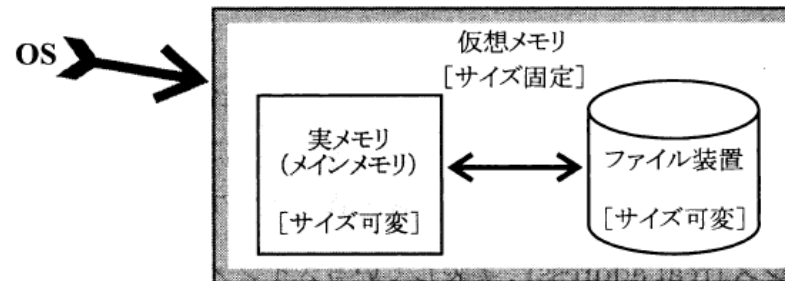


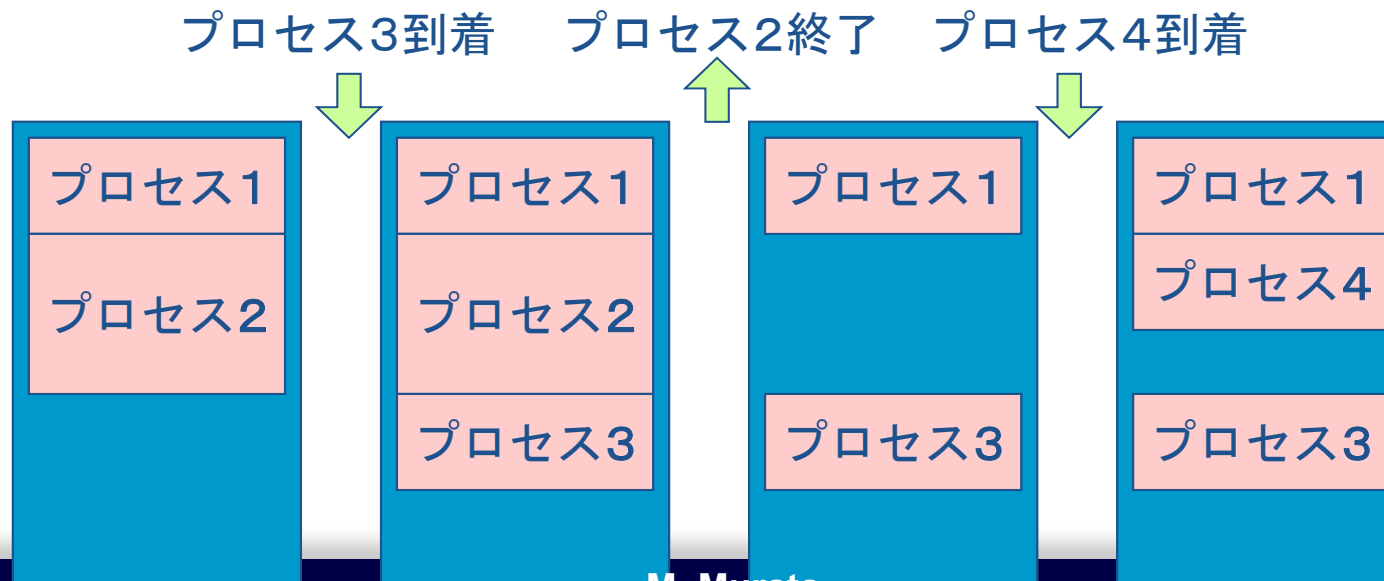
図 3.9 OS から見る仮想メモリ

- 効果
 - 一定サイズで広い仮想メモリの連続領域をプロセスに割り付けられるので、実アドレス空間の外部フラグメンテーションを防止できる。また、プロセス割り付けにおいて、実アドレス空間のサイズへの配慮が不要となる(実際には仮想アドレスにもサイズ(制約)はある)
 - プログラムやデータの論理的な意味に配慮したプロセス割り付けやプロセス管理およびファイル管理が可能となる(後述)
 - プログラムを動的にメインメモリ(実メモリ)上でリロケーションできるので、プロセス割り付けが容易になる



可変長領域割り付けの例(再掲)

- 区画サイズを動的に変更
- プロセスが到着すると
 - 格納できる空き区画を探す
 - 空き区画にプロセスを割り当てる
 - 残りは空き区画



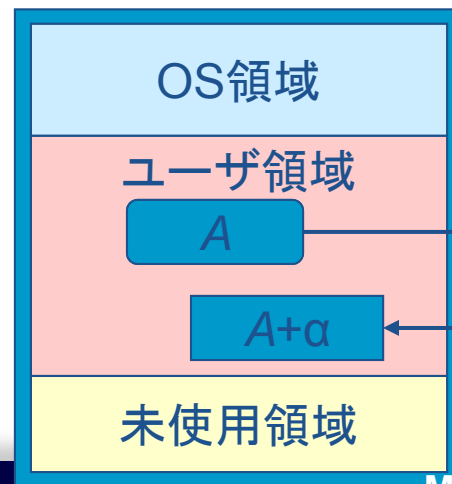


リロケーション

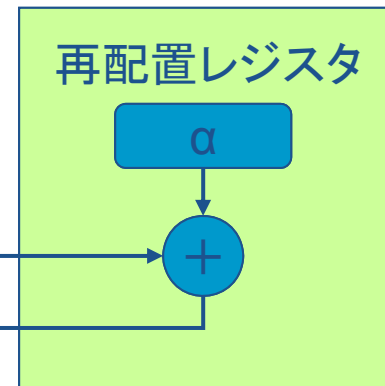
- 絶対番地形式と相対番地形式
 - プログラムをコンパイルしたあとのオブジェクトプログラムの形式
- 絶対番地形式
 - メモリ上のアドレスが固定→ロードされるアドレスも決まっている
- 相対番地形式
 - 先頭を0として、そこからの相対的なアドレス(オフセット)が埋め込まれている:「再配置可能(リロケートブル)」
 - 静的再配置(プロセスの実行前)と動的再配置(プロセスの実行中)
 - 静的再配置: OS領域とユーザ領域の境界アドレスが変更されるとプロセスを再ロードしなければならない
 - 動的再配置: 再配置レジスタが必要になる
 - 実アドレス = 論理アドレス + (再配置レジスタ)

メインメモリ

$\alpha + 0$ 番地



再配置レジスタ



アドレス変換機構



3.2 仮想メモリーメインメモリの隠ぺいー

3.2.2 仮想メモリとは？

[4] 仮想メモリの実現ー仮想メモリ機構の概要ー

仮想メモリ機構の役割

- 参照局所性を活用するために、仮想メモリと実メモリとのマッピングは一定サイズのブロックで行う
→参照局所性が高いブロックを優先して、実アドレス空間にマッピングする。
- 実メモリ(実アドレス空間)は仮想メモリ(仮想アドレス空間)よりも小さい
→ある時刻に実アドレス空間にマッピングできる仮想アドレス空間はそのごく一部。
- 仮想メモリにあるアクセス対象の命令やデータはファイル装置に必ずある
→仮想メモリ機構において、ファイル装置はメインメモリのバックアップメモリとして機能する。
- 仮想メモリ機構は、マッピングを統合して管理する
 - 仮想メモリ(仮想アドレス空間)と実メモリ(実アドレス空間)とのマッピング
 - 仮想メモリ(仮想アドレス空間)とその実体がバックアップしてあるファイル装置のアドレス空間とのマッピング

プロセッサから見たら(ユーザから見たら)

- マシン命令中のオペランドによって、実メモリとは独立した仮想メモリという一定サイズに固定した、また、巨大なメモリのアドレス(仮想アドレス)を指定できる
- 仮想メモリ機構によって仮想アドレス空間と実アドレス空間とはマッピングしてあるので、ある1個の仮想アドレスから対応する1個の実アドレスを生成すればよい
- 仮想メモリ機構が生成する実アドレスによって、実際のメインメモリにアクセスする

(これらはOSが実現する)

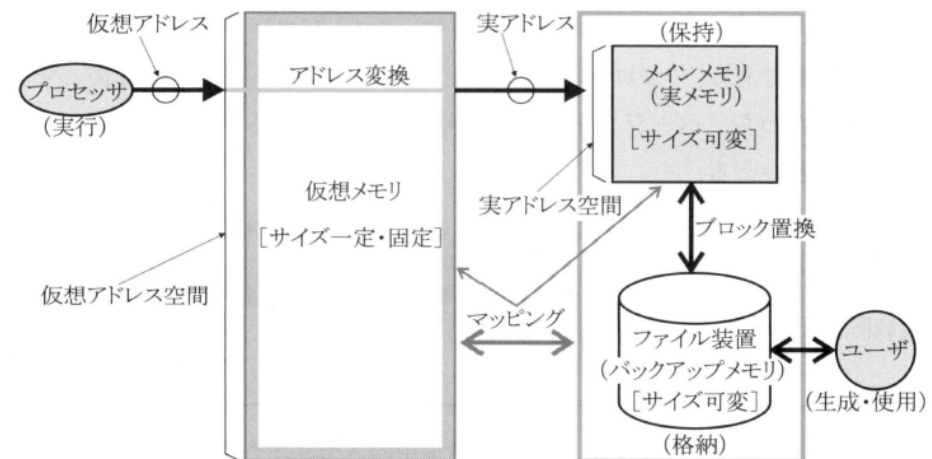


図 3.10 仮想メモリ —ハードウェア装置との関係—



3.2 仮想メモリーメインメモリの隠ぺいー

3.2.2 仮想メモリとは？

[5] 仮想メモリ機構の動作(OSの機能)

1. メモリオペランドから実効アドレスとして生成する仮想アドレスを含むブロックが実メモリ上に(マッピングして)あるかないかをチェックする

2(a) ある場合

- － アドレス変換: マッピングにしたがって、仮想アドレスから対応する実アドレスへ変換する

2(b) ない場合(ページフォールト)

① ブロック置換(ページ置換)

実メモリ上にある不要なブロックをパックアップメモリであるファイル装置へ追い出し、代わりに仮想アドレスで指定したアクセス対象の命令やデータを含むブロックを、パックアップメモリであるファイル装置から実メモリ上に取りってくる

② 入れ替え後のマッピングにしたがって、仮想アドレスから対応する実アドレスへ変換する

- 不要なブロックを決定する戦略: ブロック置換アルゴリズムあるいはページ置換アルゴリズム



3.2 仮想メモリーメインメモリの隠ぺいー

3.2.2 仮想メモリとは？

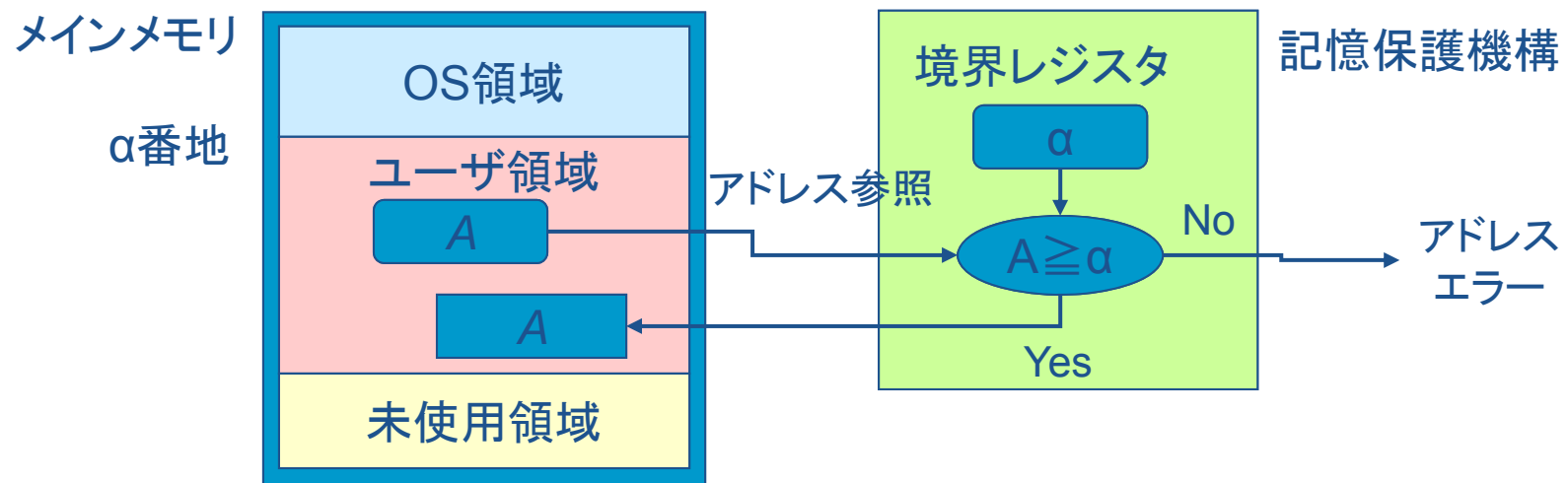
[6] 多重仮想アドレス空間

- (A)単一仮想アドレス空間
 - 単一の仮想アドレス空間を複数プロセスで共有利用する。
OSが管理する仮想アドレス空間は唯一
- 長所
 - 機能の実現が簡単
- 短所
 - 仮想アドレス空間に存在する複数プロセス間相互のメモリ保護機能や動的リロケーション機能が必要となる



メモリ保護

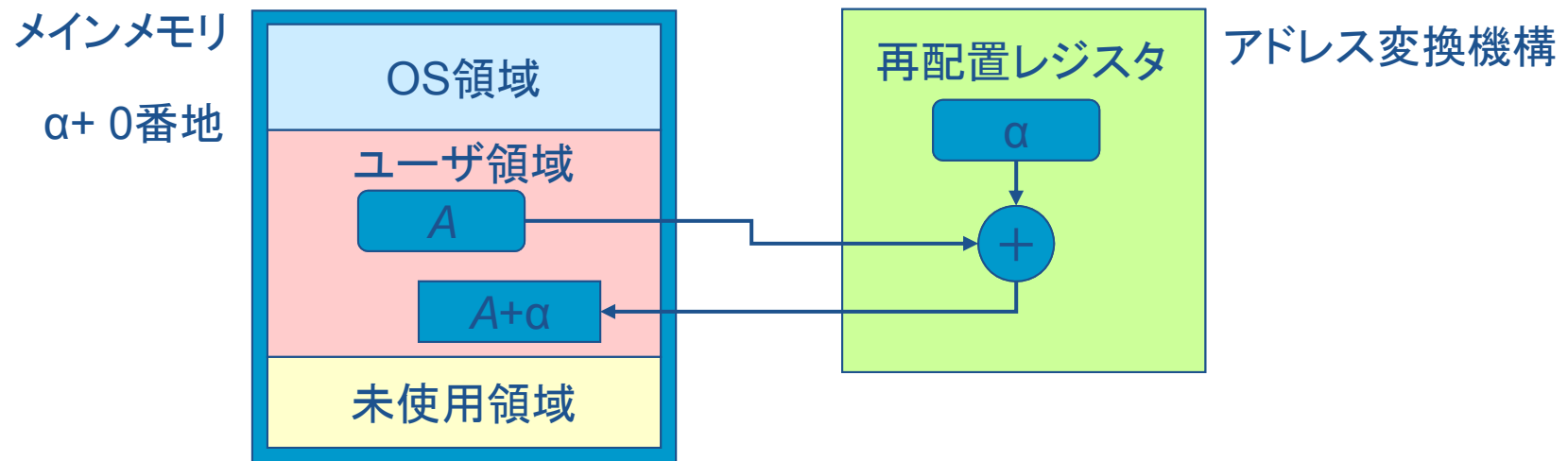
- 不当なアクセス
 - ユーザプログラムからOS領域へのアクセス
 - 複数のユーザプログラム間のアクセス
- 境界レジスタ方式
 - 単一ユーザシステムの場合
 - マルチプログラミングシステムの場合
 - ユーザ区画の最小アドレスと大きさを保持する2つの境界レジスタ





リロケーション(再配置)

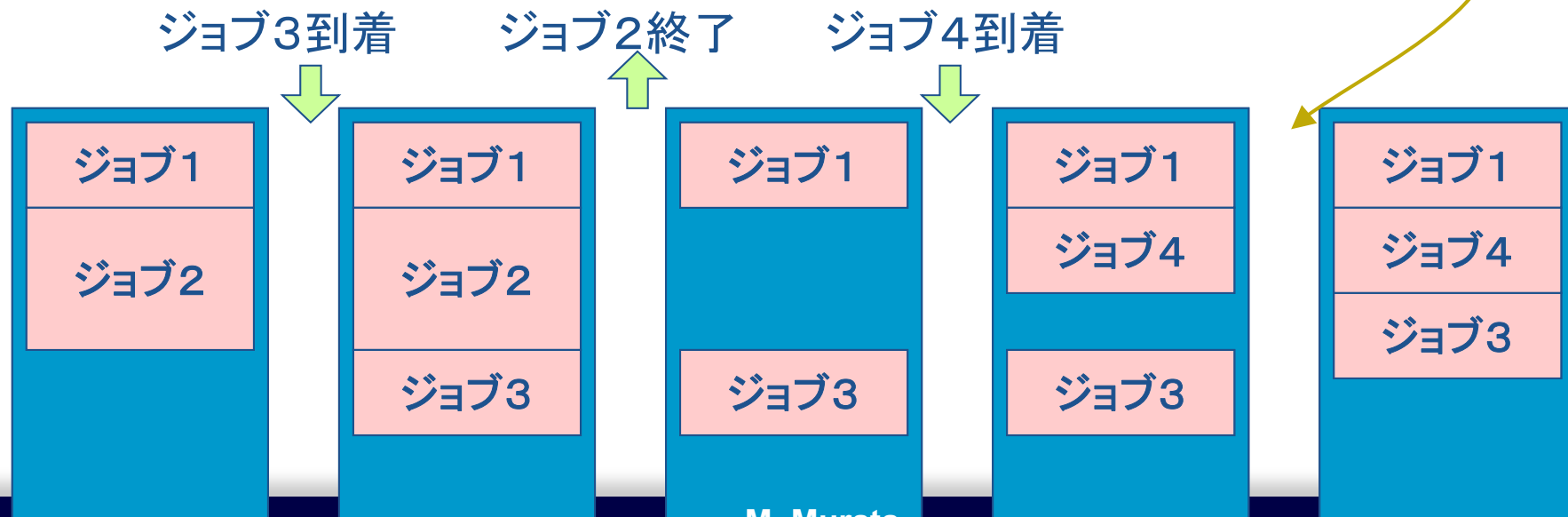
- 相対番地形式
 - 動的再配置(プロセスの実行中)
 - 動的再配置: 再配置レジスタが必要になる
 - 実アドレス = 論理アドレス + (再配置レジスタ)





可変区画割付けにおけるデフラグ(再掲)

- デフラグ(ガベージコレクション)
 - 使用中の区画を移動して、空き領域を連結
 - 動的再配置前提
- メモリの利用率向上
 - 内部断片化が避けられる、ただし、外部断片化は起こる
- ただし、CPUを浪費するため、実際には使われない
 - 256MBメモリのマシンで4バイトのコピーに40nsecかかる場合→2.7秒





3.2 仮想メモリーメインメモリの隠ぺいー

3.2.2 仮想メモリとは？

[6] 多重仮想アドレス空間

- (B) 多重仮想アドレス空間
 - プロセスなどの「論理的なプログラムブロック」ごとに仮想アドレス空間を割り付ける
 - OSが管理する仮想アドレス空間は複数個で多数ある
- 長所
 1. プロセスやタスクごとに独立した個別の論理アドレス空間を設定できる
 2. 仮想アドレス空間を実アドレス空間から独立して設定できる。マルチタスキングでさらに活用できる
- OSに対する効果
 - プロセスごとに論理アドレス空間を割り付け管理できる
 - 個々のプロセスのアドレス空間が独立しているので、複数プロセス間相互のメモリ保護が容易になる

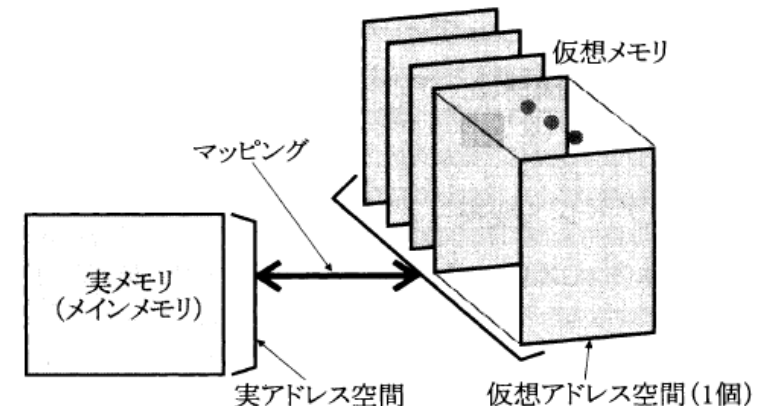
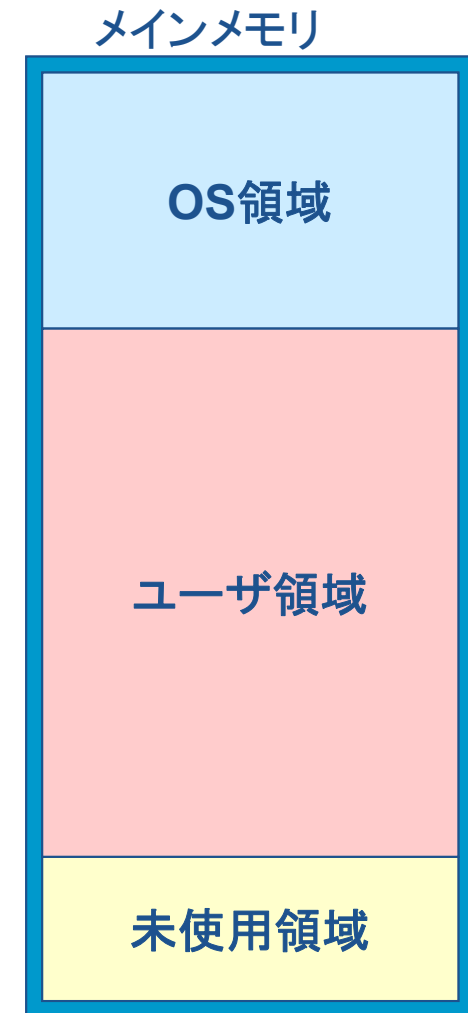


図 3.11 多重仮想アドレス空間



仮想メモリがなかったら (単一連続割付け)

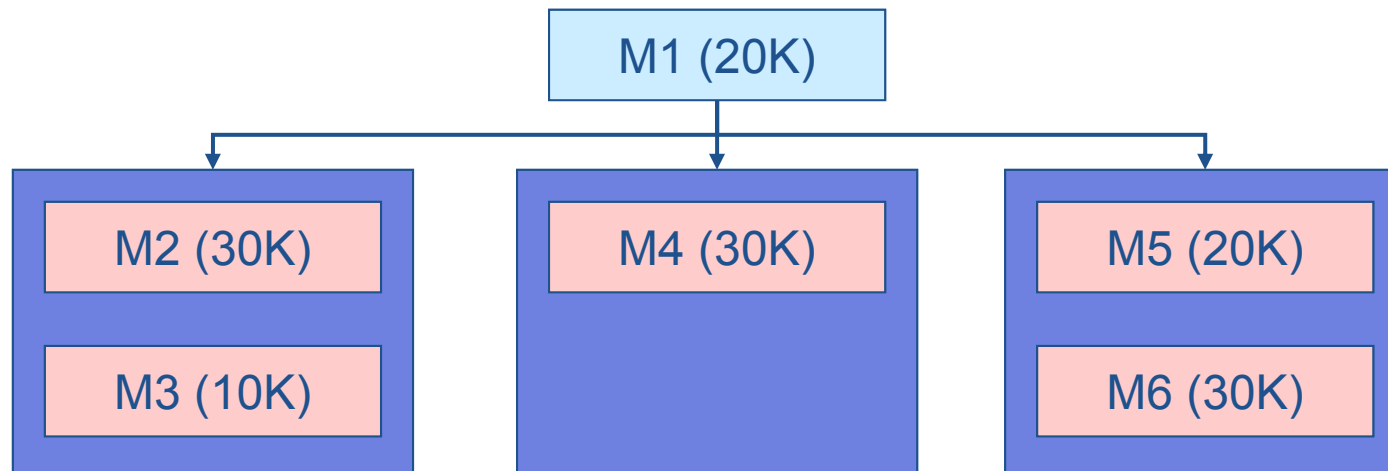
- システム領域とユーザ領域に割付け
- 同時に1プロセスだけ
 - MS-DOSなど
- ユーザ領域の管理技法
 - 再配置 (relocation)
 - OS領域の大きさが変わったときどうするか？
 - スワッピング (swapping)
 - I/O待ちに対する対応
 - オーバレイ (overlay)
 - メインメモリ上にプロセスが全部入りきらない時どうするか？
 - 仮想記憶の原型





仮想メモリがなかったら (オーバレイ)

- プログラムがメインメモリより大きい場合どうするか？
 - プログラムをモジュールに分割し、必要なモジュールだけをメインメモリにおく
 - 管理モジュール(メインルーチン)は必要
- プログラムが、メインメモリの大きさとモジュールの大きさから、同時に利用するモジュールの組み合わせを決定
- 仮想記憶に発展、ただし、ゲーム専用機では今でも使われている
- シングルプロセス前提





3.2 仮想メモリーメインメモリの隠ぺいー

3.2.2 仮想メモリとは？

[7] [まとめ]仮想メモリ機構でのOSの役割

- 仮想メモリ機構は、アクセス対象の仮想アドレスを含むブロックが実メモリ上に
 - (A) ある場合
 - アドレス変換だけを行う
 - (B) ない場合
 - まずブロック置換を行い、その後、アドレス変換を行う
- アドレス変換もブロック置換も、OSとハードウェアとの機能分担、すなわち、OSの管理・制御下で動作するハードウェア機構によって実現する
- アドレス変換時に発生するページフォールト割り込みは、仮想メモリ機構におけるハードウェア装置→OS通信
 - ページフォールト割り込みはハードウェアのアドレス変換機構からOSへのブロック置換の依頼
 - OSがこのページフォールト割り込みを受け付けると、その割り込み処理において、ブロック置換アルゴリズムによる不要ブロックの選択やブロック置換そのものの管理や制御を行う
- 仮想メモリはメインメモリ性能の空間的改善を図るメモリアーキテクチャであり、その主要な機能をOSによって実現する
 - OSは、仮想メモリ機構というハードウェアの性能を活用することによって、メインメモリの時間的性能の維持も図る