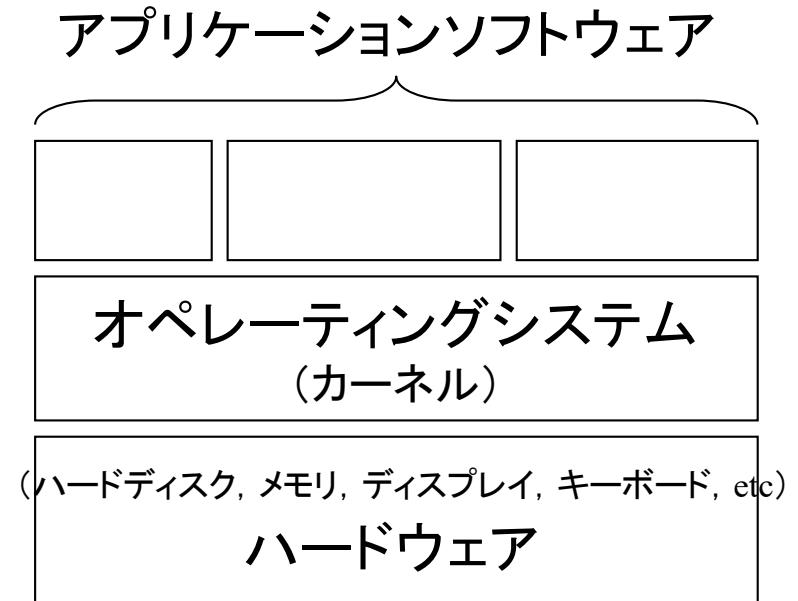

第12章「オペレーティングシステム」

- ・ オペレーティングシステムの概要と役割
- ・ プロセスとスレッド
- ・ 仮想記憶

オペレーティングシステムとは

.....

- ・ 最もハードウェアに近いところに位置し, コンピュータシステム全体を管理するソフトウェア
- ・ よく利用される基本機能をアプリケーションソフトウェアに提供
 - 画面表示, キーボード入出力
 - ハードディスク入出力, メモリ管理
- ・ アプリケーションソフトウェアの起動, 終了から, ハードウェア資源(CPU, メモリ等)の効率的な利用まで, 計算機システムを底辺で支える



※オペレーティングシステムの中核となるソフトウェアをカーネルと呼ぶ

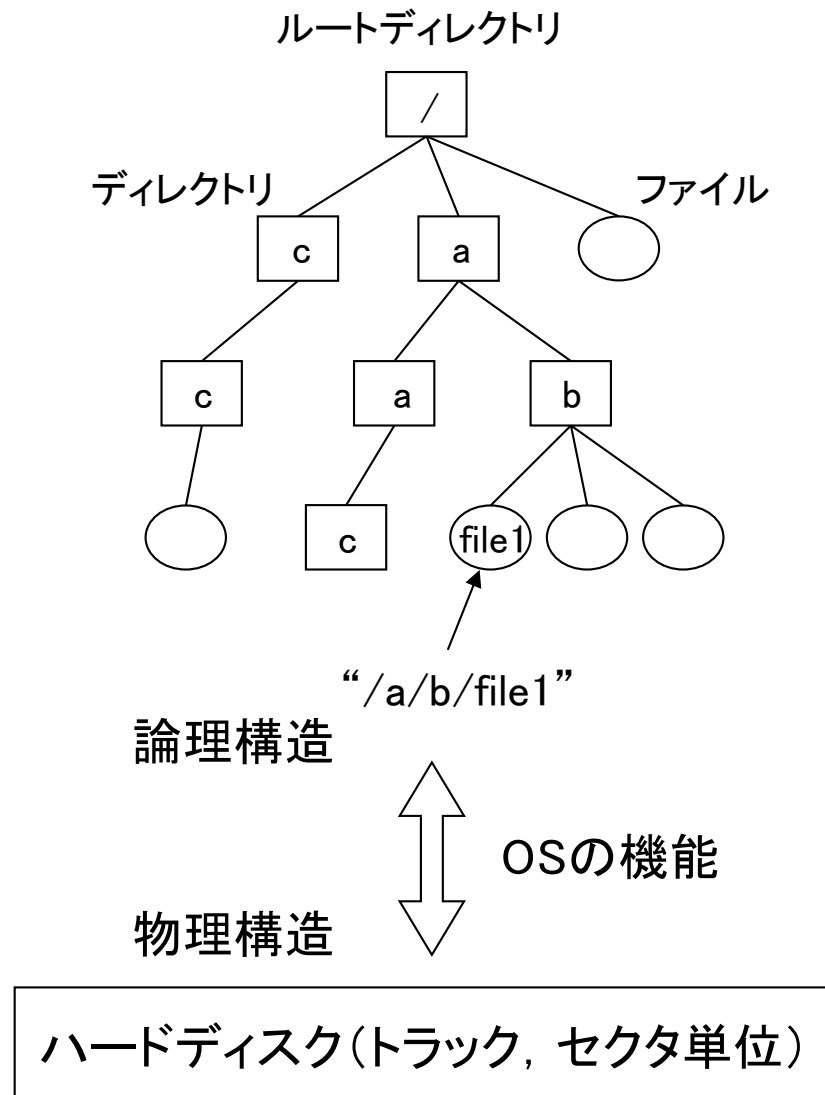
オペレーティングシステムの役割

.....

- ・ ファイルシステムの提供
 - 2次記憶装置（ハードディスク, DVD, CDROM）の抽象化
- ・ プログラム実行管理
- ・ プロセス管理とスケジューリング
 - マルチタスク
- ・ 記憶管理
 - 主メモリの管理, 仮想記憶の提供

ファイルシステム

- ・ データやプログラムを1つの「ファイル」という単位で扱う
- ・ 2次記憶装置を抽象化
 - － ハードディスク, DVD, CDROM
- ・ 木構造(ディレクトリ=ノード, ファイル=葉)
- ・ ファイル操作
 - － 読出し, 書き込み, 生成, 削除, コピー等
- ・ 保護機能

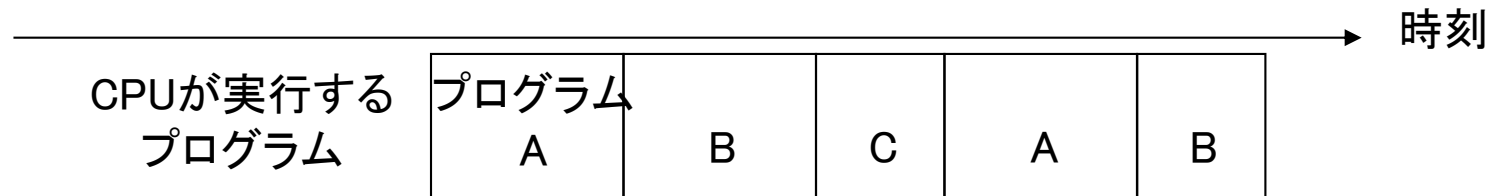


プログラムの実行

- ・ プログラム(コンパイル済み, 機械語)を2次記憶より主メモリにロードし, 実行を開始する
 1. プロセス生成
 2. メモリ上に領域確保
 3. プログラムをメモリにコピー
 4. 実行開始
 5. 終了後の後始末

マルチタスク

- 1つのCPUしか持たない計算機では1度に一つのプログラムしか実行できない
- 複数のプログラムを逐次的に順番に処理することで同時に複数のプログラムを実行しているように見せかける＝マルチタスク

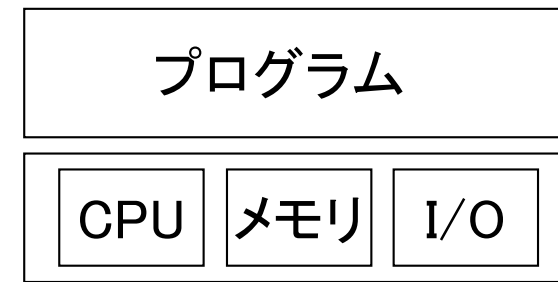


- 資源の有効利用や複数ユーザが同一の計算機を利用できることなど、効率性や便利さを高める

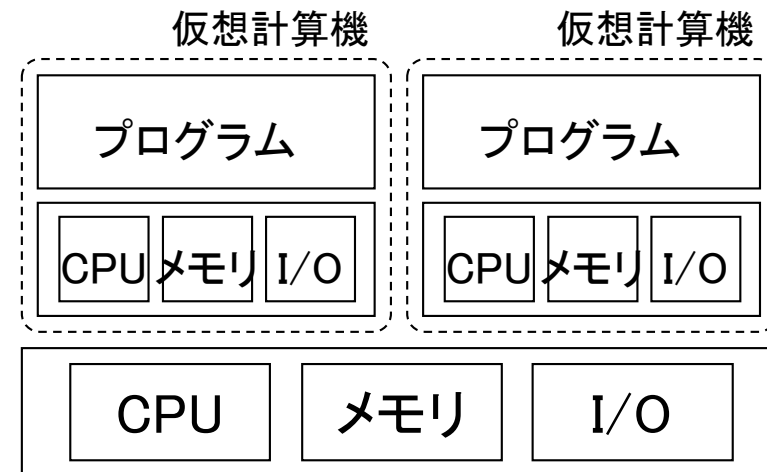
プロセス

- マルチタスクでは, プログラムは途中で実行が中断され(他のプログラムの実行に移る), 後に再開される
- 1つのプログラムには仮想的な1つの計算機が割り当てられる
- 1つのプログラムを実行するための論理的な実体を「プロセス」と呼ぶ
 - 「タスク」や「ジョブ」とも呼ばれる. 「スレッド」とは異なる

- 論理構造
シングルタスク



マルチタスク



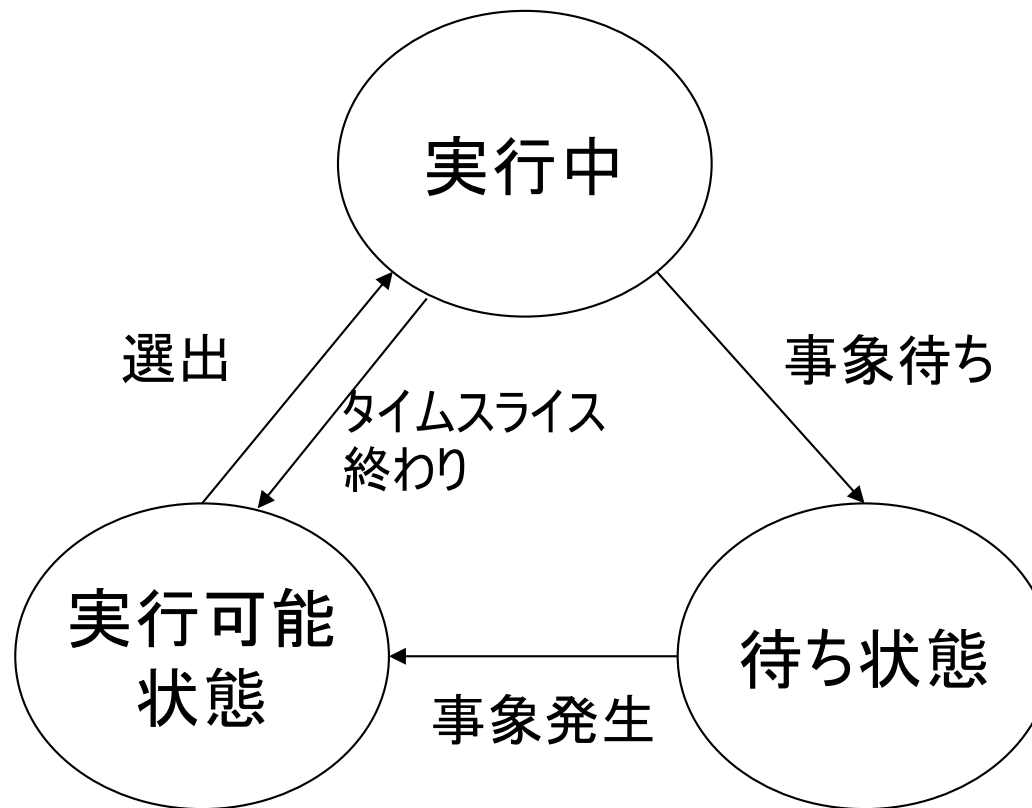
プロセスの切り替え

.....

- ・ 次のタイミングで実行するプロセスを切り替える
 - プロセスの実行終了
 - 入出力装置や他のプロセスとの同期で継続できないとき
 - 割り当てられた時間(タイムスライス)が終了したとき
- ・ 「プロセスにCPUを割り当てる(ディスパッチ)」と表現
- ・ 切り替えられるプロセスは次の実行に備えて必要なすべての情報を保存しておく
 - CPU内のレジスタの保存, メモリのマッピング
 - コンテキストの切り替えと呼ぶ

プロセスの状態

- ・ 個々のプロセスは次の3状態をそれぞれとる
 - CPUが1つなら実行中のプロセスは各瞬間で1つだけ

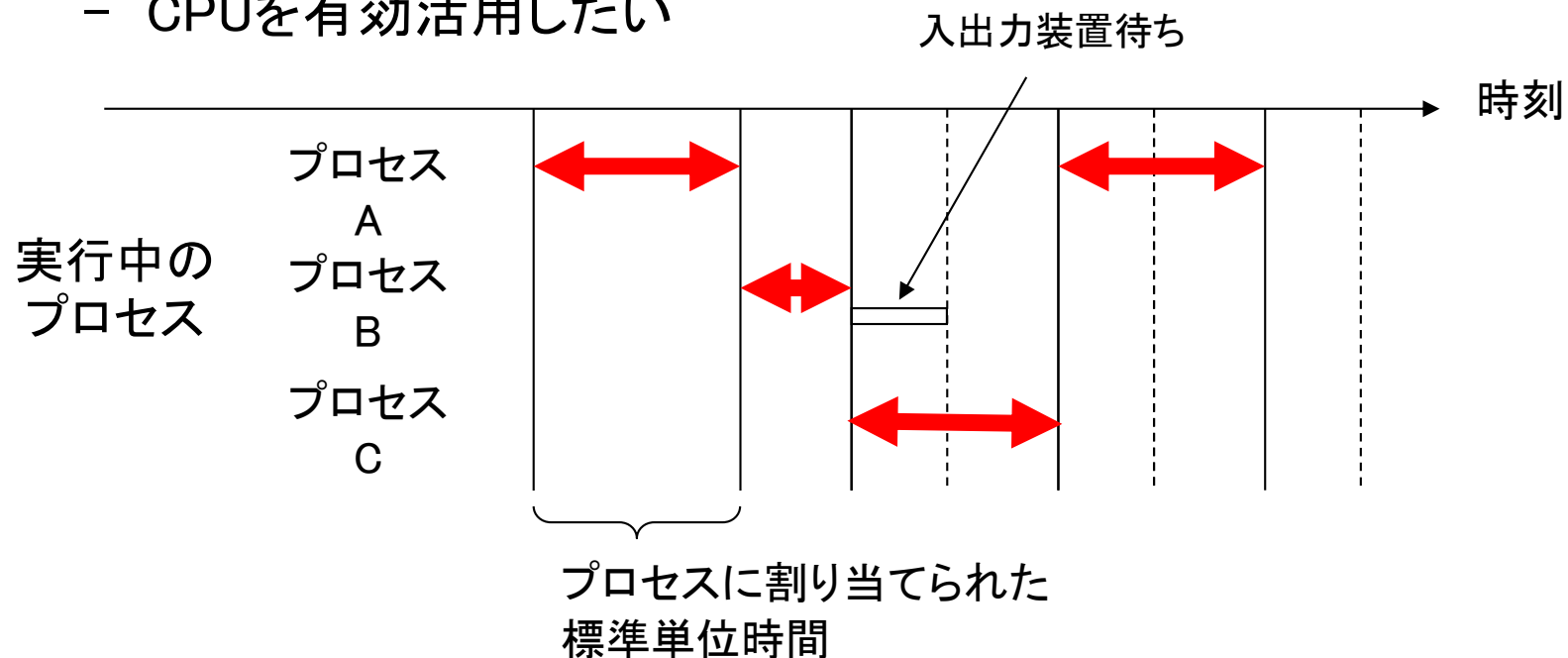


プロセスとスレッド

- ・ アプリケーションプログラム1つにプロセス1つが対応
- ・ 1つのアプリケーションプログラム中でも並行処理を行いたい場合があり, これのための仕組みがスレッド
 - スレッドはプロセス中の並行処理
- ・ 同じプロセス中の複数のスレッドはメモリ空間を初め多くの情報を共有するので, コンテキストの切り替えにかかるオーバーヘッドが小さい
 - 効率的な処理が可能に

スケジューリング

- 入出力装置とのやり取りにかかる時間は不定
 - CPUを有効活用したい



- 効果的なプロセスの実行順序や切り替えタイミングを求めることをスケジューリングと呼ぶ
 - 到着順(First Come First Service), ラウンドロビン, etc...

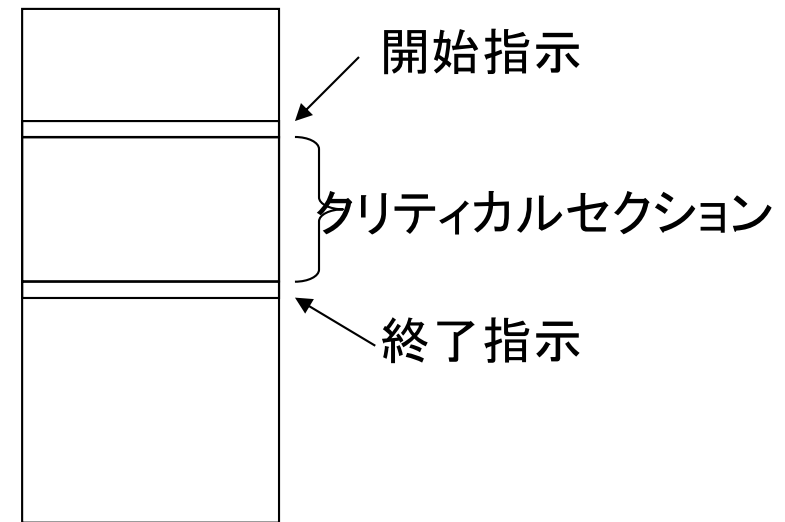
共有資源と排他処理

.....

- ・ プロセスは仮想計算機上で実行されるが、実体としてのハードウェアは一つしかないので、複数のプロセスが一つのハードウェア要素(資源)を共有することがある
- ・ 「複数人が異なるATMを使ってほぼ同時に同じ銀行口座から現金を引き出す」イメージ
- ・ プロセス間で情報交換を行う仕組みが必要

・ クリティカルセクション

プログラム



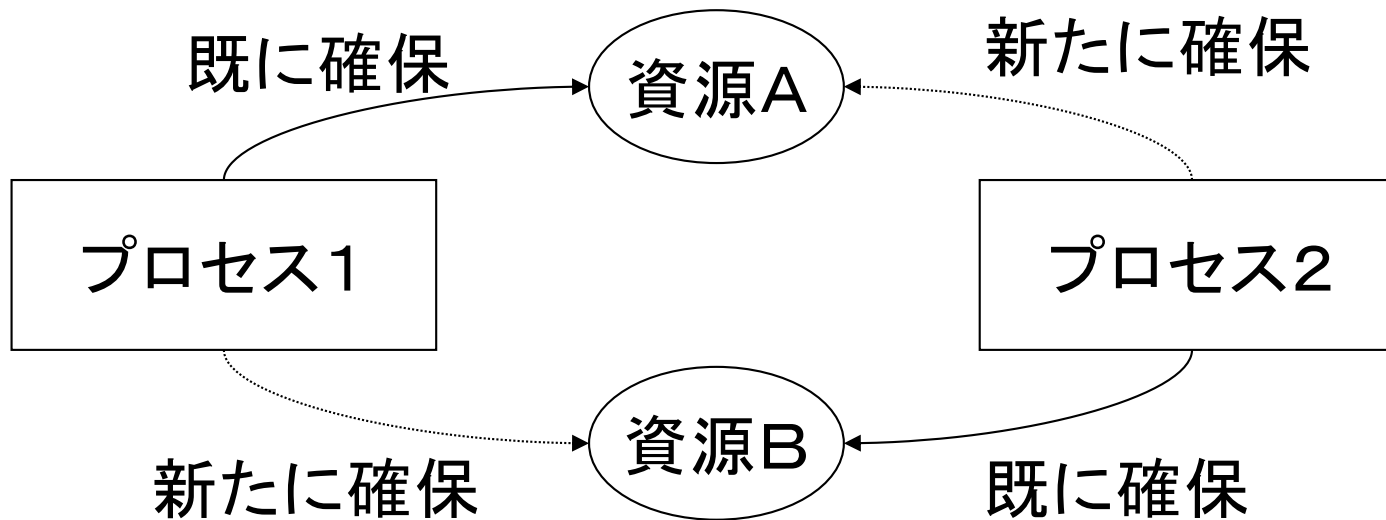
・ セマフォ

- プロセス間で共有するフラグを使って資源の所有権を管理

資源の競合

.....

- 異なるプロセスが資源を取り合う結果、処理がそれ以上進まないことが起こりうる(デッドロックと呼ぶ)
 - いくつかの回避アルゴリズムが提案されているが多くのOSでは放置されている→デッドロックを起こす可能性をなくすようにプログラムを書く



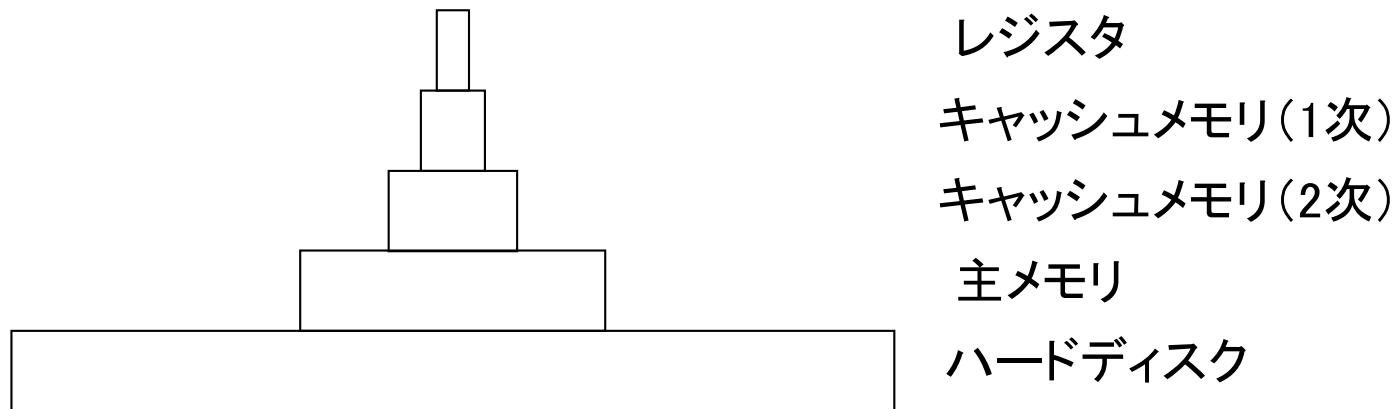
リアルタイムOS

- ・ 物理的なハードウェアを制御するシステムでは、処理にはリアルタイム性が求められる
 - ロボットの制御など
- ・ ある事象に即座に反応してそれに対応した処理を行う
 - 数 10μ ～ 数10m秒程度の決められた時間以内に処理することを保障する
- ・ 組み込みシステム (embedded system)

記憶階層

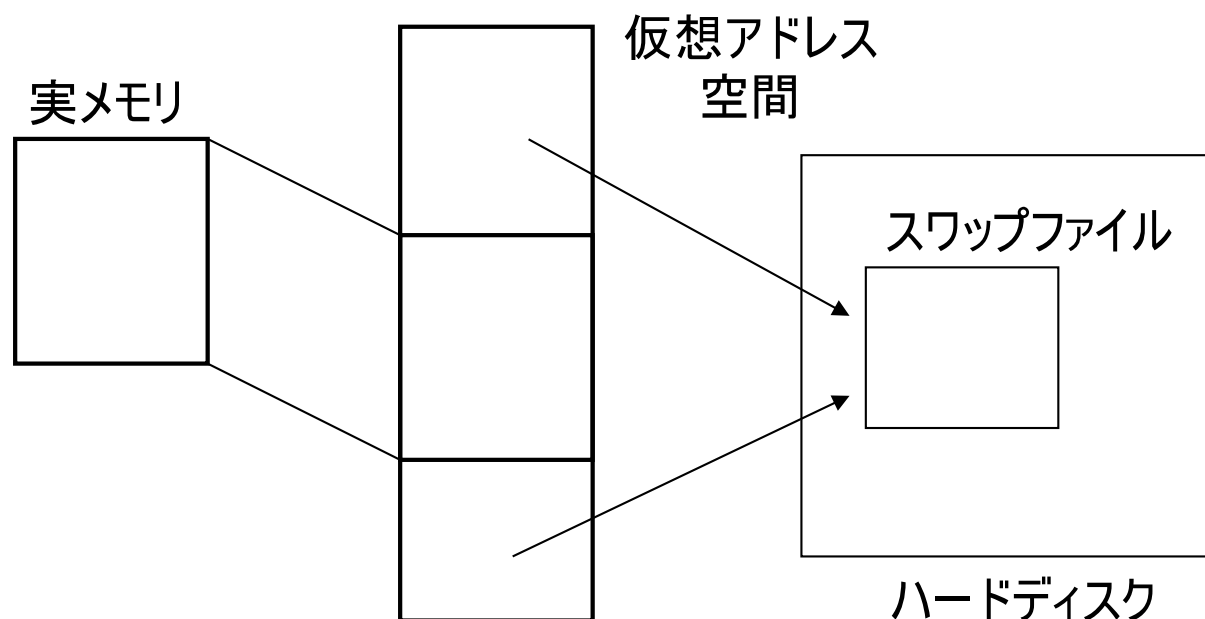
.....

- ・ 高速な記憶装置は小容量であり, 大容量の記憶装置は低速である
- ・ 階層構造を作ることによってコストと性能のバランスを取る
 - 空間的局所性と時間的局所性



仮想記憶

- ・ プログラムの配置や主記憶の領域確保の自由度を高めるため、実在するアドレス空間とは直接無関係な仮想的なアドレス空間を用いてプログラムを実行
 - 仮想アドレス(あるいは論理アドレス)と実アドレス(あるいは物理アドレス)と呼ぶ



※実メモリが少ないと
パフォーマンスが大幅に
低下する(スラッシング)