

コンピュータシステムB -ソフトウェアを中心に -

#5 マルチプログラミング

Yutaka Yasuda

システムソフトウェアの価値

- システムソフトウェアは何のためにあるか？
 - ハードウェア資源の利用効率を上げるため
 - 一度使った資源を迅速に、効率よく再利用したい
- なぜそれが必要だったのか？
 - 高価だったから
 - 性能が低かったから
- 「限られたハードウェア能力を限界まで使いたい」

ハードウェアの性能向上

- ハードウェアの高性能化に伴い

より多くの資源を積みながら、かつより高効率に利用する方向へ進化

- マルチプログラミング

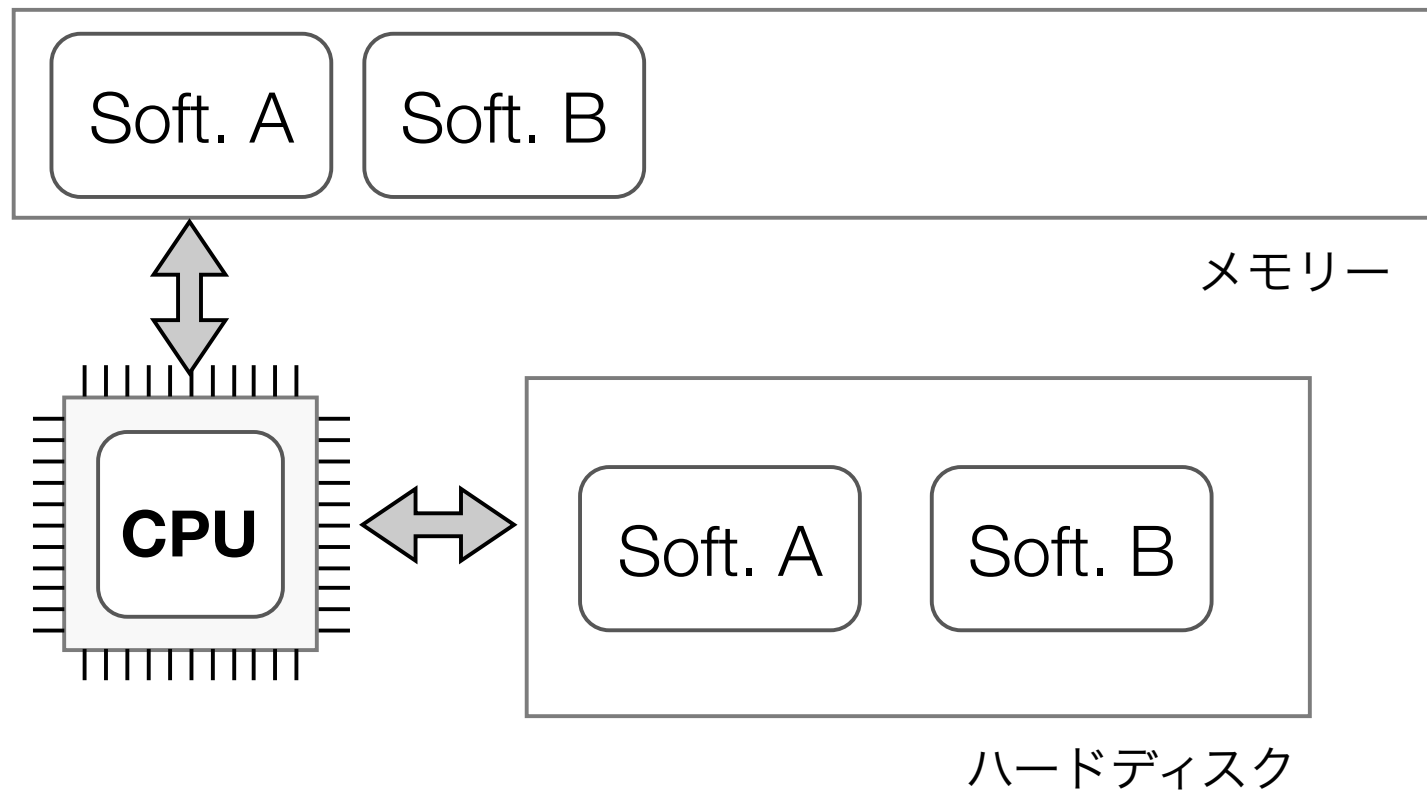
複数のソフトウェアを並行に実行したい

- CPU 処理能力やメモリ量などの向上による

向上した資源をよりよく使うためでもある

ニワトリタマゴな状態

マルチプログラミング



バッチ（一括）処理

- メインフレーム
- 利用時間を決めて利用者を入れ替える
- 単一の処理プログラムを順繰りに実行する
- 単一のプログラムでメモリもCPUも使い切るような状況では有効（現在でもスパコンの運用はこれに類する）

TSS (Time Sharing System)

- 複数のプログラムがメモリに載る
- CPU の遊休時間ができる (I/O待ちやオペレータ待ち)
- マルチプログラミング：複数のプログラムを並行に処理

プロセス：実行中のプログラム

Round Robin による順繰り実行

- 複数の「端末」によって複数ユーザからの同時利用

利用者の反応速度の方がコンピュータの動作時間より充分遅いので、見かけ上「同時並行」に見える

CPU 切り替え方式

- preemptive

タイマーによる切り替え

- non-preemptive

タイマーによらずプログラムが自主的に譲る

- 現在の Windows XP/Vista, MacOSX, Linux などはずべて TSS かつ preemptive (個人向けなので「端末」はない)

マルチプログラミングによる問題

- スケジューリング

優先順位づけ

- 排他制御

メモリやデバイス、ファイルに対するロック

デッドロック

- メモリ管理＝再利用