

仮想という考え方

総合科目「ネットワーク社会を支える情報技術入門 II」、2009年09月28日

筑波大学システム情報工学研究科/電子・情報工学系
新城 靖
<yas @ cs.tsukuba.ac.jp>

このページは、次の URL にあります。

<http://www.softlab.cs.tsukuba.ac.jp/~yas/gen/it-2009-09-28>

あるいは、次のページから手繰っていくこともできます。

<http://www.softlab.cs.tsukuba.ac.jp/~yas/gen/>

<http://www.cs.tsukuba.ac.jp/~yas/>

印刷配布資料 <http://www.softlab.cs.tsukuba.ac.jp/~yas/gen/it-2009-09-28/it-2009-09-28.pdf>

■今日の重要な話

- 「仮想(virtual)」という考え方は、コンピュータとネットワークの技術に伴い生じた考え方である。
- 「仮想」の一義的な意味は、「現物と置き換え可能なもの」の意味である。「実体がない」という意味ではない。日本語訳、漢字の語感に惑わされないこと。
- 重要な「仮想」がつく概念
 - 仮想記憶(virtual memory)
 - 仮想計算機(virtual machine)
- 仮想記憶の実現には、連想メモリが使われている。
- オペレーティング・システムの誕生と「プロセス」の考え方の発明で、仮想化が始まった。

◆仮想という考え方を身につける意義

- 経済的利益を得る。未熟な情報技術を活用する時には、情報技術固有の考え方を知っていた方が有利になる。
- 人生を楽しむ。

■仮想とは

◆コンピュータ由来の「仮想」の意味

英語では、virtual。

- ソフトウェアの働きで作られている。
- 現実のもの(ハードウェア)と置き換え可能である。仮に取り替えても、(ある範囲では)利用し

ている側には気がつかれない。

類語：抽象(abstract)、論理(logical) (←→物理(physical))

反語：実(real)

◆英語のvirtualの説明

virtual (adjective)

almost or nearly as described, but not completely or according to strict definition :
the virtual absence of border controls.

- Computing** not physically existing as such but made by software to appear to do so : a virtual computer. See also virtual reality .
- Optics** relating to the points at which rays would meet if produced backward.
- Physics** denoting particles or interactions with extremely short lifetimes and (owing to the uncertainty principle) indefinitely great energies, postulated as intermediates in some processes.

Thesaurus (類語)

- a virtual guarantee* effective, in effect, near, near enough, essential, practical, to all intents and purposes.
- a virtual shopping environment* simulated, artificial, imitation, make-believe; computer-generated, online, virtual reality.

MacOSX付属 New Oxford American Dictionary, New Oxford Writers Thesaurus より。

◆英和辞典

virtual (形容詞)

《表面的にはまたは名目上はそうでないが》事実上の, 実質上の, 実際(上)の; 仮想の, パーチャルな;

- 【光】 虚像の (opp. real)
- 【理】 仮想の《遷移の中間状態として現われ直接検出にかからない; opp. real》
- 【電算】 仮想記憶 (virtual memory) の[を用いる];
- 《古》 実効のある。

リーダーズ英和辞典第2版より。

◆日本語の辞書に現れる「仮想」

この講義で中心的に扱う「仮想の考え方」とは、適合しない。virtual の訳語に「仮想」を当てたのは、失敗だった。「事実上の」

仮想 (広辞苑)

仮に考えること。仮に想定すること。例：仮想敵国。

仮想敵国の英語は、{hypothetical、imagined、potential} enemy。virtual enemy とは言わない。

■仮想記憶 (virtual storage,仮想メモリ,virtual memory)

◆(物理的な)メモリ

メモリ(メインメモリ)、RAM(Random Access Memory)

実行中のプログラムを保持する。加工するデータを一時的に保持する。IC (Integrated Circuit、シリコンという元素による半導体で作られた電気回路) で作られているので、速い。容量は、ハードディスクよりは小さい。値段が高い。(揮発的(電源を切ると消えてしまう。))

ハード・ディスク (HD(Hard Disk), HDD(Hard Disk Drive))

プログラムやデータをデータを保持する。容量は、メモリより大きい。値段が安い。(永続的である(電源を切っても残っている。))

メモリには、数字で番地 (アドレス) が付いている。番地を指定して、データを保存する。番地を指定すると、データが取り出せる。

◆仮想記憶(仮想メモリ)

仮想記憶(仮想メモリ、virtual memory)とは、[オペレーティングシステム\(後述\)](#)の働きにより、実際に備えているメモリ容量よりも大きなプログラムを動かすための仕組みである。

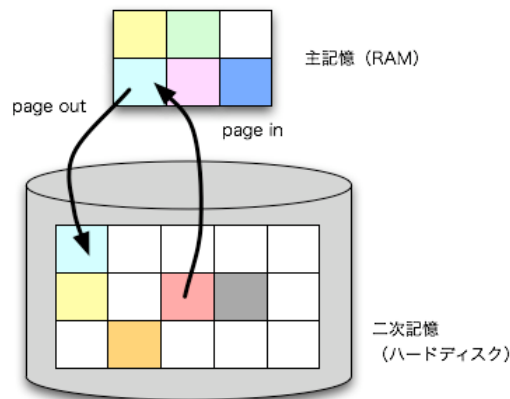


図 仮想記憶による主記憶容量の拡大。主記憶よりも、「事実上」多くのメモリがあるように見せかける。

主記憶 (main memory, primary storage)

RAM で作る。小容量、高価。

二次記憶(secondary storage)

ハードディスクで作る。大容量、高価。

ページ

主記憶と二次記憶 (ハードディスク) を固定長のブロックに分けたもの。大きさは、4Kバイト-8Kバイトが一般的。

ページアウト

主記憶にある (当分使いそうにない) ページを、二次記憶へコピーする。

ページイン

二次記憶にある (現在必要とされている) ページを、主記憶へコピーする。

仮想記憶の基本的なアイデアは、高価な RAM の容量を、安価なハードディスクを使って、「事実上」拡大する。すぐに使うところだけを速いメモリ (IC) に、当分使わない所を、遅いハードディスクに置き、ディスクとメモリの内容を入れ替えながら仕事を進める。この時、速いメモリを主記憶、遅いハードディスクを二次記憶とう。

仮想記憶を使うと、メモリが100Mバイトしかないコンピュータで、200Mバイトのメモリを使うプログラムを実行することができるようになる。

もともとは、1つのプログラムで実際のメモリ容量以上のものを使うための仕組みである。最近では、複数のプログラムが使うメモリの総量で考えることもある。

◆ページ置き換えアルゴリズム

主記憶が足りなくなった時に、どのページを犠牲にして、空きを確保するか。

- OPT(Optimal)アルゴリズム。最も遠い将来まで利用されない部分を二次記憶へ。
- FIFO (First-In First-Out) アルゴリズム。最も早い時刻に読み込まれた部分を二次記憶へ。
- Randomアルゴリズム。ランダムに選んで二次記憶へ。
- LRU((Least Recently Used))アルゴリズム。最も長い時間参照されていない部分を二次記憶へ。

一番よいアルゴリズムは、OPTアルゴリズムだが、実現不可能である。未来の情報を使うので。LRU は、過去の情報を使って未来を予測する方法。

野口悠紀雄(のぐち ゆきお)の「超整理法」の押し出しファイルは、LRUアルゴリズム。

アルゴリズム(手法、算法)とは、問題を解くためのあらかじめ記述された定義の明確な規則、あるいは、指示の集まり。日本語などで書くこともできる。直接的にはコンピュータでは実行できない。なんらかの方法で実行できるようになったものは、プログラムである。

◆仮想記憶の実現

仮想記憶は、メモリ管理ユニット(Memory Management Unit) というハードウェアの働きにより実現されている。

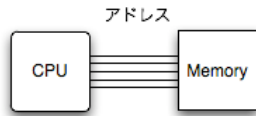


図 CPUによるメモリアクセス

MMUがないコンピュータの場合

- メモリは、アドレス(番地、2進数)という数値を指定すると、値(2進数)を返すハードウェアである。
- CPUは、読み書きしたいメモリのアドレスを指定を指定する。値を送受信する。

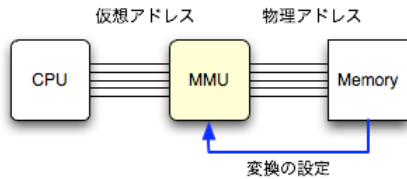


図 MMUによる仮想アドレスと物理アドレスの変換

MMUがあるコンピュータの場合

- CPUは、読み書きしたいメモリの「仮想アドレス」を指定する。
- MMUは、仮想アドレスを物理アドレスに変換する。
- MMUは、変換の設定をメモリから読み出す。変換の設定は、後述するオペレーティング・システムが作成する。

メモリアクセスの度に、毎回毎回、メモリから設定を読み出していると遅い。(2倍遅い。) MMUの中には、変換を高速に行うための小容量(8~数1000)の連想メモリがある。

通常のメモリ

アドレスを指定して値を取り出す

連想メモリ

値(仮想アドレス)を指定して値(物理アドレス)を取り出す

MMUに含まれる連想メモリは、Translation Lookaside Buffer (TLB) と呼ばれる。

◆配列と連想配列

配列は、プログラミング言語で、メインメモリを抽象化したもの。連続した番地におかれた様なデータ。数の添字でアクセスする。連想配列では、数の添字ではなくて、文字列等でアクセスする。

もの	指すもの	内容
メインメモリ	番地(数)	1バイト(8ビット)のデータ
(通常の)配列	添字(index)(数)	任意のデータ
連想配列	値(数以外のもの、文字列等)	任意のデータ

連想配列のことを、プログラミング言語によっては、ハッシュ表(Hash table)ということもある。

◆配列の例

(大量の)一連のデータの処理が得意。

[[ruby-array.rb](#)]

```
#!/usr/bin/ruby
a = Array.new(12);
a[0] = 31; a[1] = 28; a[2] = 31; a[3] = 30; a[4] = 31; a[5] = 30;
a[6] = 31; a[7] = 31; a[8] = 30; a[9] = 31; a[10] = 30; a[11] = 31;

i=0; sum=0;
while( i<=11 )
  sum += a[i]
  i += 1
end
printf("sum == %d\n", sum );
```

実行例:

```
% ./ruby-array.rb ↵
sum == 365
% █
```

◆連想配列の例

[[ruby-assoc.rb](#)]

```
#!/usr/bin/ruby
a = Hash.new();

a["January"] = 31; a["February"] = 28; a["March"] = 31; a["April"] = 30;
a["May"] = 31; a["June"] = 30; a["July"] = 31; a["August"] = 31;
a["September"] = 30; a["October"] = 31; a["November"] = 30; a["December"] = 31;

month="September";
printf("%s has %d days.\n", month,a[month])

month="October";
printf("%s has %d days.\n", month,a[month])
```

実行例:

```
% ./ruby-assoc.rb ↵
September has 30 days.
October has 31 days.
% █
```

ハッシュ表は、連想メモリの高速な実装方法の1つ。うまく実装すると、要素数が増えても検索速度は一定で変わらない。ハッシュ表では、決して先頭から順に探しているのではない。

MMU で使われている連想メモリは、ハードウェアで複数同時に探す。あまり大 容量のものは作れない。

◆仮換地

「仮換地」は、不動産用語。区画整理中の土地を管理するための考え方。土地 の区画整理には時間がかかるので、途中段階で使う。区画整理終了時には、無 くなる。

従前地

物理アドレスで指すような土地。

仮換地

仮想アドレスで指すような土地。

建物の位置を把握したり、税金を課す時に区画整理で国土（メモリ）が増えたり減ったりすることはない。事務手続きで、従前値と仮換地の対応表（仮換地 証明）が必要になることがある。

■仮想計算機(virtual machine)

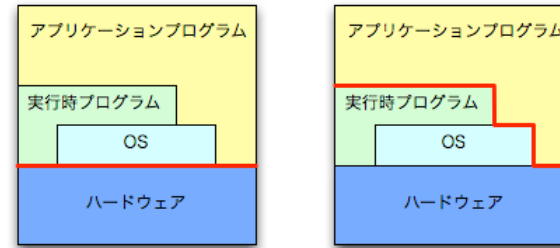
あるいは、仮想マシン、仮想コンピュータ。 計算機(専門用語)とは、コンピュータのこと。電卓ではない。

◆計算機とは

計算機(machine)とは、インタプリタ(interpreter)の一種。 プログラミング言語で記述されたソース・プログラムを解釈しながら実行する もの。普通は、それ自体もまたプログラムの一種。

- **ハードウェア**
 - CPU (Central Processing Unit)。「機械語命令」と呼ばれる二進数の並び (01の並び) を、解釈・実行する。
 - 入出力装置。ハードディスク、キーボード、画面、マウス、ネットワークなど。
- **ソフトウェア**
 - 高級言語で記述されたプログラムのための実行時ソフトウェア。

両者ともインタフェースが明確。



図? 計算機とは

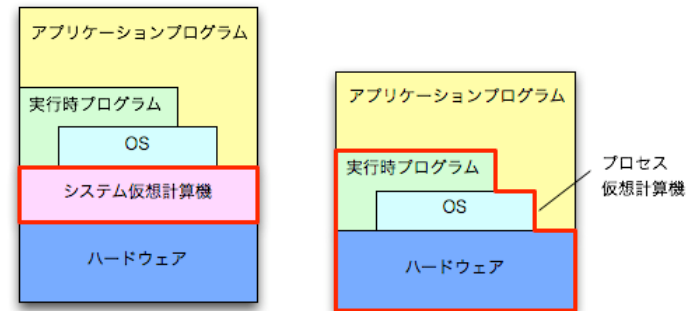
◆システム仮想計算機とプロセス仮想計算機

システム仮想計算機

ハードウェアの代わりに、オペレーティング・システムが動作可能な 完全なシステム環境を提供する。

プロセス仮想計算機

1つの [プロセス](#) (後述) を実行するためのプラットフォーム。



図? 2種類の仮想計算機

仮想計算機モニタ(virtual machine monitor, VMM)

システム仮想計算機を実現するための小さなソフトウェア。仮想計算機 モニタが動作することで、仮想計算機が作られる。仮想計算機モニタは1個でも、複数の仮想計算機を作ることができる。

ハイパーバイザ(hypervisor)

仮想計算機モニタと同義。

スーパーバイザ(supervisor)

オペレーティング・システムと同義。

◆システム仮想計算機の利用方法

- もともと複数のコンピュータで行っていた仕事を1台の物理的なコンピュータでこなすために集約する。
- 1台のコンピュータで、複数のオペレーティング・システム用の **アプリケーション・プログラム**を同時に利用する。
- ハードウェアが故障した時に、簡単に交換できるようにする。
- データのバックアップや、ソフトウェアのインストールを簡単にする。
- クラッシュに強くする。(書き込みを取り消すことができるハードディスクを使う。)
- オペレーティング・システムの開発環境として利用する。
- セキュリティを高める。

◆コンピュータの集約

コンピュータの高速化で、性能が余ってきた。もともとハードウェア n 台でやっていた仕事を、ハードウェアとしては1台に集約する。並列処理の逆。



図? 仮想計算機によるサーバの集約(集約前)



図? 仮想計算機によるサーバの集約(集約後)

◆さまざまな仮想計算機

IBM メインフレーム用 (1970年代)。

1台のハードウェアで数種類のOSを動かす。

Virtual PC, Real PC, VMware, Parallels Workstation

Windows PC のハードウェアと同じような動きをする。Macintosh や Linux で Windows 用のプログラムを実行するために使う。

vMac (Virtual Macintosh)

Unix, Windows 等で(古い)Macintosh 用のプログラムを動かす。

VirtualGameStation

Macintosh 上で Playstation のプログラムを動かす。

◆x86の仮想化

x86 は、Intel 社の CPU 製品で、8086 に由来するものの総称。

- 8086, 80186, 80286
- 80386 (i386), i486
- Pentium, Pentium II, Pentium III, Pentium 4, Celeron, Xeon
- Core Solo, Core Duo, Core 2 Duo
- AMD社 Am386, K5, K6, Athlon, Duron

当初の x86 は、仮想計算機を動作させることに適していない。最近、仮想計算機を動作させるために適した機能や命令を備えたものが出荷されるようになってきた。

- Intel VT (Virtualization Technology)
- AMD virtualization (Pacifica)

◆エミュレータとシミュレータ

エミュレータ: 別のシステムで、本物のシステムと同じ動きを真似するソフトウェア。本物は、必ずしもコンピュータ全体(計算機)でなくてもよい。

- Wine: Unix 上で Windows 用のプログラムを実行するためのプログラム。Wine は、Windows のソフトウェア部分(システム・コールと実行時プログラム)をエミュレートする。
- CrossOver Mac: MacOSX 上で Windows 用のプログラムを実行するためのプログラム。
- 端末エミュレータ: [文字端末](#) の機能を全部、または、部分的にエミュレートする。

シミュレータ: 現実世界の予測に使うもの。数式でモデル化して計算で予測する。実際にやるとコストがかかるもの、取り返しがつかないものをコンピュータの中で実行する。現実の代わりにはならない。(エミュレータは、本物の代わりに使える。シミュレータは、本物の代わりには使えない。)

- フライト・シミュレータ
- 地球シミュレータ

(文字)端末

1台の(高価な)中央のコンピュータを、複数人で TSS (Time Sharing System) で使う時に使う(安価な)コンピュータ。キーボードから打ち込まれた文字を中央のコンピュータに送る機能と、中央のコンピュータから送られてきた文字を画面に表示する機能がある。

◆セキュアVM/BitVisor

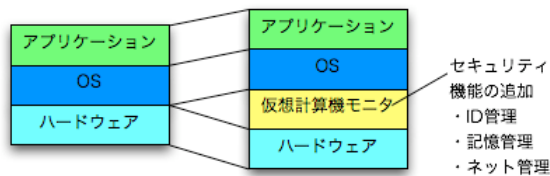
日本政府が決定した「セキュア・ジャパン2006」の項目の1つ。内閣官房情報セキュリティセンターが推進する。文部科学省平成18年度科学技術振興調整費「高セキュリティ機能を実現する次世代OS環境の開発」による支援を受ける。筑波大学がとりまとめ。電気通信大学、東京工業大学、慶應義塾大学、奈良先端科学技術大学院大学、豊田高専、富士通、NEC、日立製作所、NTT、N

TTデータ、ソフトイサの技術者が参加。インテルが技術協力。2009年3月開発プロジェクト終了。

目標

- オペレーティング・システムやアプリケーションに依存しない形でセキュリティ機能を付加する。
 - ICカード等による認証と起動制御
 - ハードディスクの暗号化
 - 仮想プライベート・ネットワークによる経路の暗号化
- Windows, Linux 等を変更しないで実行する。

このために、仮想計算機（モニタ）で、セキュリティ機能を追加する。



図? BitVisor の機能



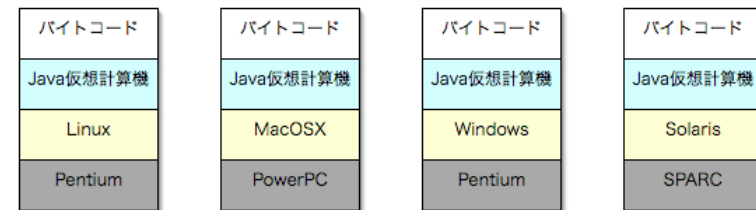
図? BitVisor の起動画面

<http://www.securevm.org/>

◆Java仮想計算機

分類としては、プロセス仮想計算機。Java 言語で書かれたプログラム（バイトコードと呼ばれる機械語に変換されたもの）を実行する。

Java仮想計算機を使うと、さまざまな種類ハードウェア(CPU)やオペレーティング・システムで共通にバイトコードという機械語が使えるようになる。バイト・コードに変換されたアプリケーションは、どの CPUやオペレーティング・システムでも動作する。



図? 共通のバイトコードが使える

■仮想記憶、仮想計算機以外の仮想

◆仮想回線(virtual circuit)

インターネットで通信しあう2つのプログラムは、それらが動作している2台のコンピュータがあたかも物理的な回線で直結されているかのように見える。

実際には、間に何台ものルータと呼ばれる機器がある。

◆VPN(Virtual Private Network)

本当は、インターネットという、誰もがアクセスできるネットワークを通っているが、あたかも専用線(private network) で結ばれているように見せかける。

例:

- PPTP
- L2PT over IPsec
- OpenVPN
- Packetix VPN (SoftEther VPN)
- SSL-VPN

◆仮想LAN(Virtual LAN, vLAN)

1本の物理的な媒体上に、独立した複数のLAN (Local Area Network) を構築する。物理的に切れているLANとLANを接続する。

通常のLANで使うアドレスの前にvLANタグという、vLANを区別するアドレスを付ける。

VPN と VLAN の違いは、本質的にはあまりないが、実際問題として、VLAN は、「VLANタグ」を使うもの、間にルータが入らないものの意味で使われる。

◆仮想ディスク(virtual disk)

1つの大きなファイルを、1つのハードディスクと同じように扱えるようにしたものを。仮想ディスクを使うと、次のようなことができる。

- データを暗号化して保存する。
- 複数のファイルからなるソフトウェアを1まとめにする。
- CD-R などに書き込むためのファイルを一時的に蓄える。

◆ユーザ・レベル・オペレーティング・システム

[オペレーティング・システム](#)は、本来ならハードウェア上で動作するものである。それを、一般のアプリケーションと同じように、他のオペレーティング・システム上で実行したものを、ユーザ・レベル・オペレーティング・システムという。

◆仮想ホスト(virtual host)

ネットワークに接続されたコンピュータをホストという。ホストは、サービスを提供するプログラム(サーバ)を動作とせるという意味もある。

仮想ホストは、本来なら複数のコンピュータが必要であったものを、1台に集約する。ただし、ネットワークを超えて向こう側で見た時には、1台に集約されていることには気がつかせない。

仮想ホストの実現方法

- 仮想計算機を使う
- を使う
- 個々のネットワーク・サービスで対応する
 - Apache Web サーバ
 - sendmail 等のメール・サーバ

◆仮想現実感(virtual reality)

高度なコンピュータ・グラフィックスやサウンドの技術を使って、実在しないものを(コンピュータの中に)実在しているかのごとく見せかける。

人間が働き掛けると、現実と同じように**応答**する。感覚のフィードバックがある。

◆世俗的な「仮想」

最近の日本語では、コンピュータによる原義を離れて、「現実には存在しない」という意味が使われ

ることがある。以前は、空間(と時間)の制約で実現できなかったことが、コンピュータとネットワークの発達により、現実に近いレベルにまで達してきた(部分的に現実のもの置き換え可能になってきた)という意味も生じてきている。

- 仮想都市、仮想社会
- 仮想大学、仮想教官、仮想学生
- . . .

■オペレーティング・システム

◆オペレーティング・システム (Operating System,OS) とは

- 基本ソフトウェア(プログラム)。
- 電源を入れて、使えるようになった時に動いているプログラム。(オペレーティング・システム以外のプログラムは、すべてオペレーティング・システムの働きで実行される。)
- コンピュータの「顔」を作る。基本的な考え方と操作方法とを決める。同じハードウェアでも、違うオペレーティング・システムが動くと、考え方も操作方法も違う。
- ソフトウェアにより、基本的なもの(幻影、オブジェクト)を提供する。
 - ファイルとディレクトリ
 - プロセス
 - ウィンドウ、アイコン
- 管理職。ハードウェアを管理する。メモリ管理、CPU管理。

◆オペレーティング・システムのメモリへの読み込みと実行

徒然草(つれづれぐさ)。八になりし年。最初の仏問題。

OSは、他のプログラムを実行するプログラムである。他のプログラムは、OSの力を借りてはじめて実行される。

ではOSは、誰が実行するか？

ブート・ストラップ(Boot strap,靴ひも)。ほら吹き男爵は、底なし沼に落ちた時に、自力で脱出するために、自分の靴ひもにつかまって脱出した。

BIOSがブートローダをメモリに読み込み、ブートローダを実行する。

ブートローダがOSをメモリに読み込み、OSを実行する。

BIOS(Basic Input Output System)、ROM Monitor。OSをハードディスクからメモリにコピーして、OSに制御を渡すプログラム。ROM(Read-Only Memory)という、少容量の不揮発的なメモリに入っている。古いシステムでは、(フロッピーディスクへの)基本的な入出力の機能が含まれていた。

◆オペレーティング・システムの種類

- Unix, Linux
- MacOS
- Microsoft Windows 95/98/2000/NT/XP/Vista/7
- MS-DOS

ソフトウェアを買う時、自分のパソコンのOSで動作するものを買う。(CPUの種類も大事。同じOSでもCPUの種類がプログラムは違うと動作しない。)

自分が使いたいソフトウェアが動作するようなOSとCPUを積んだパソコンを買う。

◆歴史

- Loader
- Monitor
- バッチ処理
- TSS (Time Sharing System)
- ワークステーション用OS
- パソコン用OS
- ネットワークOS, 分散OS, 実時間OS

「パーソナル・コンピュータ」の考え方は、比較的新しい。昔は、コンピュータが高価だったので、1台のコンピュータを複数人(時には数百人)で同時に使うことが一般的だった。

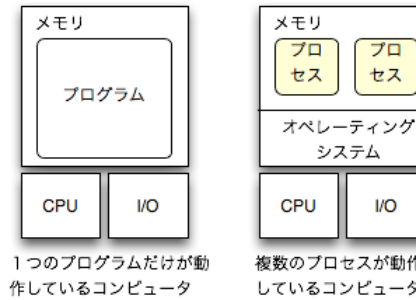
TSS (Time Sharing System)では、1台のコンピュータを大勢の人で同時に使う、一人ひとり、自分で占有しているかのように見える。

◆プロセス

プログラムとプロセス。

プロセスとは、オペレーティング・システムの働きにより、二次記憶に保存されていたプログラムがメモリ(主記憶)に読み込まれ、実行可能になったもの。

単にファイル(ハードディスク)に入っても、プログラムはプログラムである。1つのプログラムを同時に2個動かすことを考えると、プログラムとプロセスの違いがわかる。



図? オペレーティング・システムの構成要素

1つのプロセスから見ると、普段は自分一人で全部のコンピュータのハードウェアを利用しているように見える。コンピュータの仮想化の最初。

プロセスは、時々、他のプロセスからデータを受け取る(プロセス間通信)。

Last updated: 2009/09/23 17:19:21

[Yasushi Shinjo](mailto:yasushi@cs.tsukuba.ac.jp) / <yas@cs.tsukuba.ac.jp>