

7P-5

OSの階層モデルの考察

岩田憲和

NTT情報通信処理研究所

1. はじめに

オペレーティングシステム(OS)は大規模で複雑となりがちであるうえに常に機能拡張の要求に応じていかななければならない宿命を負っている。このため、OSの実現(インプリメント)にとって見通しの良い構成法を確立することは重要である。OSがハードウェアの仮想化層、資源間の関係の管理層、名前管理層に分類できるというモデルを示す。このモデルによるOS構成法の利点を述べる。

2. 従来のOS構成法

OSは大規模であるためモジュールに分割しモジュール間にルーチンの呼出関係に基づく階層を設けて実現することが常である。

通常、図-1のような構成を汎用計算機ではとる。しばしば階層間で制御表の共有が行われ、層間のインタフェースは明確ではない。また、各層は下の層を隠蔽していない(例えば、ジョブ管理はプロセスもメモリも確保する)ので層という意味が明確ではないだけでなく、モジュール間の参照が複雑である。

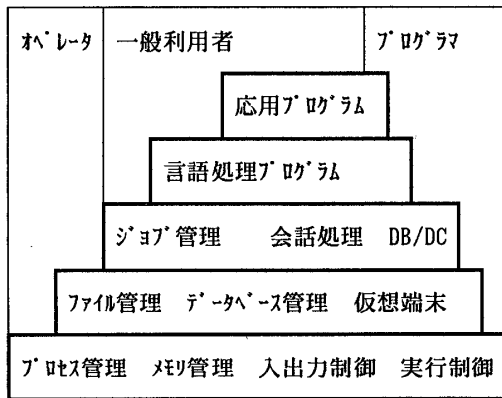


図-1 汎用OSの階層構成

3. モデル

OSは計算機内の資源を仮想化や多重化し利用者に提供することと、その上で実現するサービスに共通な機能を提供することや、システムで一元管理する名前の管理を行っている。これらを順に仮想化層、関係管理層、名前管理層と名付けることにする。これは以下のようにして階層をなす。

(0)ハードウェア

プロセッサによる情報の加工やメモリやファイルでの情報の

保存機能を上位に提供する。上位層(利用者)からの値の書込み要求と読出し要求を処理する。

(1)仮想化層

プロセッサの時間的分割(多重化)によるプロセスや、空間的分割によってメモリやディスク上にページやブロックを設けたりして仮想化を行う。この層には多重化のためのディスクリッチやページングルーチン、スワップ、バッファリングルーチンやディスクブロックのファイル化ルーチンが属す。上位に対して各種の資源の作成、削除、拡張、縮小という機能を提供する。即ち、仮想化層は「どのようなOS資源があるか」を規定すればその上位インタフェースは決まる。上位からある種類の資源の作成を依頼されると要求に応じて作成し、システム内一意の識別子を返す。

この層で提供される異種の仮想資源間には互いに関係をもたない。例えば、プロセスはリジョン(論理空間)を持たない。この層では資源の仮想化を行うだけにとどめたいからである。

(2)関係管理層

資源間の関係をグラフ的に管理する。作成した資源をノードで、資源間の種々の関係(所有者、ロードしている、オープン指定、等)をアクセス経路の向きのエッジで表す。例えば、ファイルをオープンするという操作はプロセスからカーサ(ファイルへのアクセス位置を管理するもの)へエッジを作成し、カーサからファイルへのエッジを作成することで表す。また、プロセス間でのカーサの共有は一つのカーサへ共有プロセスからエッジを張ることで表す。このようにしてシステム内に存在する資源間の関係を管理する。

エッジにアクセス制御情報を付与することによってこの層でアクセス制御も行う。

(3)名前管理層

利用者が指定する名前と関係管理層のノードとの対応辞書を管理し、利用者からの問い合わせに応じる。

(4)制御の流れの例

ファイルをオープンすることを例にしてどのように処理が行われるかを述べる。

```
fd = open("FILE", mode);
```

という利用者からの要求に対して、名前管理層で自己プロセスの名前で"FILE"が何かを得る。関係管理層で先ずカーサに対応するノードを作成する。作成されたカーサに対して仮想化層でのカーサ(modeとアクセス位置とが格納できるもの)の作成を依頼する。そして関係管理層で自己プロセスからエッジを設定する。関係管理層のカーサは仮想化層に対

してカーサにmode値を設定させる。また、カーサからファイルに対してIppjを設定する。その後Ippjの識別情報を返す。ここで注目すべきことは関係管理層のノドが仮想化層に対して処理を依頼する場合、自分に対応する仮想化層のノドに対してしか行わないことである。従来はジョブ管理からプロセス管理やメモリ管理といった複数のモジュールへの制御移行があるのに対して、このモデルでは関係管理層から仮想化層への移行は仮想化層内の自分に対応する資源への処理依頼に限ってよい。これは仮想化層で異種の資源間に関係が無いことによる。

4. 提案モデルの利点

従来のモデルと比べて提案したモデルは次の利点がある。

(1)ル-フjの設け方が示されている。

関係管理層で行う処理はすべてノド間のIppj操作である。従ってIppjの作成、削除、通過という基本的な操作を設ければよいことが分かる。

(2)モジュール化

関係管理層でIppjの種類毎に設けるル-フjをIppjの終端側に設けると規定すると、自分に対するアクセス方法を自分の内部で定義できることになる。即ち情報の隠蔽が図れるためモジュール性が向上する。

(3)制御の流れ

階層間での制御の流れが上述のように従来に比べて分かり易くなる。

5. おわりに

名前管理層、関係管理層、仮想化層という枠組がOSの構築にとって利点のあることを示した。今後、さらにこのモデルの有効性を検証していく。

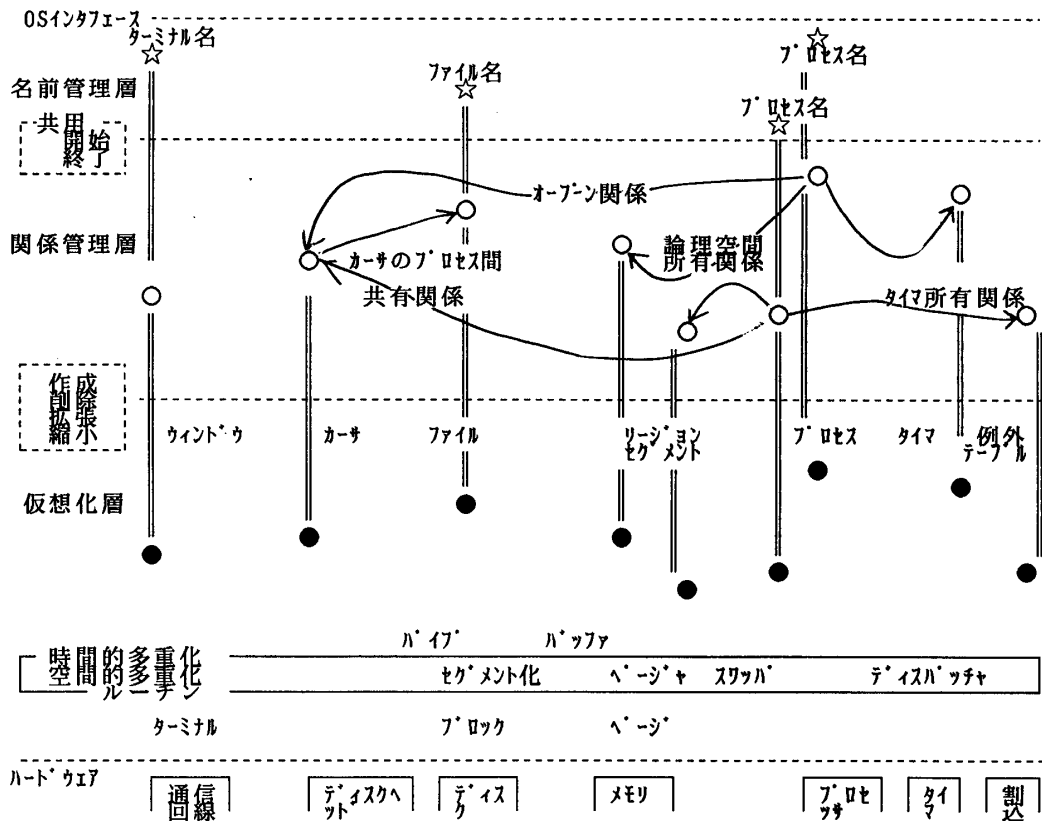


図-2 OSモデル

☆: 名前管理層に
●: 仮想化層に
→: 仮想化資源との対応関係 (アクセス経路)