

オペレーティングシステムOMNI Iの  
UNIXスーパーバイザ

2N-9

竹内 律子 鈴木 茂 岸田 一

(沖電気工業(株) コンピュータ開発本部)

## 1. はじめに

我々は、密結合マルチプロセッサ用オペレーティングシステム OMNI I を試作中である。

OMNI I は、仮想マシンを上位層に対して提供するカーネル、カーネルの機能を利用し、アプリケーションに対してOSインタフェースを提供するサービスから構成される。<sup>[1]</sup>

本稿では、OMNI I カーネル上でUNIXインタフェースを実現するUNIXスーパーバイザについて述べる。

プログラム作成には、C++を用い、全ソースを新規にコーディングした。

## 2. サービス概要

OMNI I は、カーネル、サービスの2層に分けることができる。

カーネルは、仮想プロセッサ<sup>[2]</sup>、仮想メモリ<sup>[3]</sup>、仮想デバイス<sup>[4]</sup>を上位層であるサービスに提供する。

サービスは、共通スーパーバイザ、ファイルシステム、特定OS用のスーパーバイザから成る(図1)。

共通スーパーバイザは、各特定OS用スーパーバイザとファイルシステム間でのインタラクションにおいての各OS用スーパーバイザ間の差異を吸収する目的で設けられた。各OS間でファイルシステムを共有することも可能である。ファイルシステムがスーパーバイザに対してサービスを要求するときは、すべて共通スーパーバイザを介して行う。要求例としては、スリープ/ウェイクアップ、ユーザID、プロセスIDのセット、ゲット等がある。共通スーパーバイザは、ファイルシステムからの要求があると、該当プロセスがどのスーパーバイザに属しているかを調べ、属するスーパーバイザにサービスを要求する。この機能は、各スーパーバイザのプロセスクラスが共通スーパーバイザのプロセスクラスを継承すること、及びスリープキューを共通スーパーバイザ内に持つことによって実現されている。

ファイルシステムは、仮想デバイスを使って、実現される。前述したように、ファイルシステムは複数のOSスーパーバイザ間で共有することができるが、通信している相手のプロセスがどのスーパーバイザに属しているかを知る必要がない。

## 3. UNIXスーパーバイザ

UNIXスーパーバイザ(以下、OMNI/UNIX)の目的は、アプリケーションにUNIXインタフェースを提供することである。プロセスには、カーネルの提供する仮想プロセッサと論理空間が割り当てられている。

OMNI/UNIXはOMNI I カーネル上に構築するため、次の点が従来のUNIXと大きく異なる。

- ・サービスがメモリ上に常駐しない。

- ・スケジューリングは、カーネルが行うため、コンテキストスイッチが不用意に生じる。

- ・マルチプロセッサへの対応が必要である。

OMNI/UNIXは、いくつかの種類のオブジェクトから成っている。クラスは、UNIXの構造体を基本に作成した。しかし、カーネルがすでに提供している機能があるため、内容は、オリジナルとは異なったものとなっている。主な機能の概要及び属するクラスを示す。

## a. プロセス管理

プロセスの生成/実行終了、それに伴うプロセスの状態遷移を管理する。processクラス、u\_processクラス、process\_classクラスが属する。

## ・ u\_process

プロセスを構成するオブジェクト全体を管理しているクラスである。signal、interval\_timer、resourceの各オブジェクトは、本クラスのオブジェクトに埋め込まれている。本クラスのオブジェクトは、プロセス固有空間に置かれる。したがって、他のプロセスからの参照はできない。

## ・ process

他のプロセスから参照されるプロセス属性を管理しているクラスである。共通スーパーバイザのプロセスクラスを継承している。システム共有空間に置かれる。本クラスのオブジェクトはカーネルオブジェクトに於ける仮想プロセッサへのポインタを持つ。その他の属性は、サービス独自のものである。memoryオブジェクトが本クラスの内部に埋め込まれる。

プロセスのスケジューリングは、カーネルが行う。そのため、プライオリティの計算に必要な値は持たない。ただし、ナイス値のみは本クラスのオブジェクトが管理する。本クラスのオブジェクトへのアクセスは、排他制御を必要とする。

## ・ process\_class

前述のprocessオブジェクト全体の管理を行うクラスである。フリーキュー、アロケートキュー、ゾンビキューの3つのキューをもっている。それらのキューによって、processオブジェクトの割り当て管理を行っている。さらに、新規プロセスを作成するユニークなプロセスIDの割り当て管理を行っている。

本クラスのオブジェクトへのアクセスは、排他制御を必要とする。

## b. メモリ管理

OMNI I 実行カーネルが提供するメモリ管理機能を用いて、プロセスの論理空間を管理する。テキストの共有関係も管理する。memoryクラス、textクラス、text\_classクラスが属する。

## ・ memory

プロセスクラス内に埋め込まれているクラスである。カーネルオブジェクトであるデータ、スタック、ヒープの各仮想リージョン、セグメントへのポインタを持つ。

## ・ text

テキストの共有関係を管理する。テキストセグメント、及びテキストセグメントの属する仮想リージョンを管理する。オブジェクトは、システム共有空間におかれる。

## ・ text\_class

UNIX Supervisor in the Operating System OMNI

Ritsuko TAKEUCHI, Sigeru SUZUKI, Hajime KISHIDA  
(OKI Electric Industry Co., Ltd)

- textオブジェクト管理する。使用されていないテキストオブジェクトを管理するためのフリーキューを持つ。
- c. シグナル  
アプリケーションプログラムの定義したシグナルハンドラを管理する。signalクラスが属する。signalクラスは、アプリケーションプログラムで定義したシグナルの処理方法（シグナルハンドラ、マスク、無視するか否か）を管理する。
- d. タイマ管理  
プロセスごとに設定するタイマの管理を行う。timeval、interval\_timerが属する。interval\_timerは、指定した時間が経過したら、シグナルを発生するインタバルタイマを管理するクラスである。timevalは、時間の計算をするためのクラスである。クラスclockとクラスinterval\_timerによって使用され、システムコール関数を持たない。
- e. 資源管理  
資源の利用状況、及び資源利用制限をプロセスごとに管理する。resourceクラスが属する。
- f. 仮想インタラプト管理  
実行カーネルの提供する機能である仮想インタラプトを処理するハンドラを管理する。システムコール用トラップの関数への割り当ても行。virtual\_interrupt、system\_callクラスが属する。

- g. システム管理  
システムに唯一であるホスト名、ホストID、及び時間を管理する。hostクラス、clockクラスが属する。
- h. 排他制御  
UNIXのsleep-wakeup機構を持ったロックメカニズムを提供する。semaphoreクラスが属する。共有空間にあるオブジェクトの排他制御に使用する。

4. おわりに

本システムは、コーディングを終え、現在デバッグ中である。

参考文献

- [1] 飯間、他 "オペレーティングシステムOMNIの概要"、情報処理学会第37回全国大会
- [2] 長谷部、他 "オペレーティングシステムOMNIのプロセッサ管理"、情報処理学会第37回全国大会
- [3] 矢野、他 "オペレーティングシステムOMNIのメモリ管理"、情報処理学会第37回全国大会
- [4] 竹元、他 "オペレーティングシステムOMNIの入出力管理"、情報処理学会第37回全国大会

\* UNIXは、AT&Tが開発したOSです。

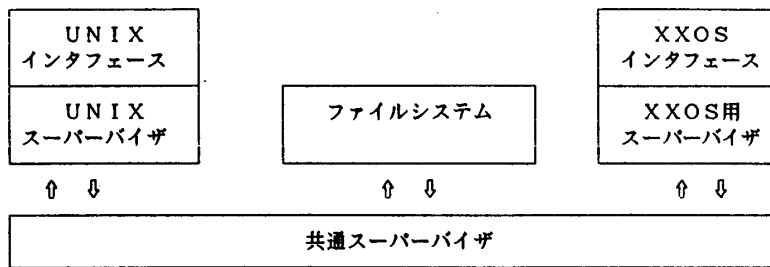


図1. サービス概要

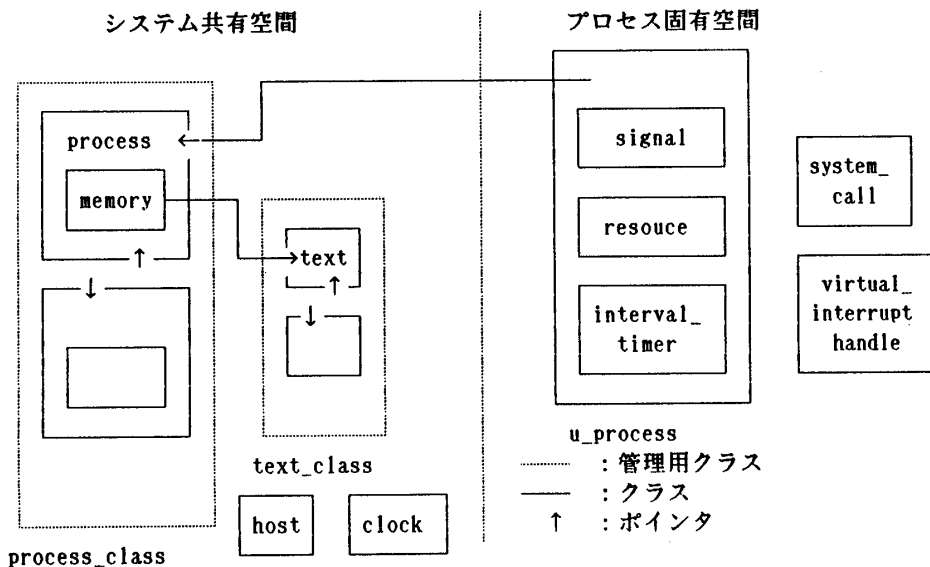


図2. オブジェクト構成図