体験学習モデルを利用したOSの学習システムの開発

米田 孝文　　西野 洋介　　早川 栄一
拓殖大学大学院 工学研究科 電子情報工学専攻

1　背景

情報工学の学習者にとって、オペレーティングシステム (以下 OS とする) 学習は重要である。なぜなら OS は、多くのアルゴリズムと結びついて、その中で頻繁な制御の移行を繰り返す複雑なシステムだからである。そのため情報工学では、コアカリキュラムの一つとなっている。

ところが学習者は、その学習を「難しい」と認識している。その要因は三つである。(1)OS 機能のブラックボックス化によって、普段 OS を利用していながら、そのアルゴリズムの理解まで到達できない。(2)OS が巨大で動的な実装構造であるため、その制御の移行を追跡することが困難である。(3)OS 学習の概念アルゴリズムと実装プログラミングにギャップがある。この中で学習者は、これらを支援してくれる教材を持たない。そして学習者は、現状の学習環境だけで OS を学ぶことが困難になっている。

2　研究の目的

本研究は、OS 学習におけるアルゴリズムの概念と実装構造を可視化して結びつけた、OS の制御の移行を静的に追跡しながらその全体像を学べる教材の実現を目的とする。特徴は、学習者が OS の役割を担って、資源管理方式を体験しながら学習できることである。これにより学習者は、資源管理の一連の仕事を引き受けることになる。また、この中で学習者は、学習する資源管理アルゴリズムが特定される。このため学習者は、OS 構造を把握しやすくなる。可視化する対象の OS は、Linux[1] である。

我々は、この目的を元に、ファイルシステムをベースにしたプロトタイプを実現した [2]。そのシステムの簡単なテストも行った。その中で明らかになったことは「プロセス管理とメモリ管理の実装の重要性」と「学習内容の分かりづらさ」である。本研究は、これらの解決を試みる。

3　システムの設計方針

本システムの対象者は、OS の概念もしくは実装のいずれかの学習経験のある学習者とする。設計方針は、次の二つである。

1. 学習者を OS 側の視点に位置付ける
   資源管理方式を可視化し、それを資源管理モジュールとして学習者に提供する。実際のユーザの立場は仮想化し、仮想ユーザとしてシステムに組み込む。これらによって学習者は、OS 側の視点から資源管理方式に触れることができるようになる。
2. OS の振舞いを体験しながら学習させる
   設計方針の (1) だけでは、正確な学習環境を提供することは困難である。そこでシステムに、課題の発生とそれを解釈して対処する仕組みを実装し、学習者が個別学習できる環境を構築する。その中で学習者は、何も分からないで操作に行き詰まることはなく、試行錯誤で必ず解決できる仕組みが提供される。これらと設計方針の (1) が連携する仕組みを、本システムの中で、OS 学習のための体験学習モデルと定義する。

4　システムの全体構成

学習者は、仮想ユーザと資源管理モジュールの間に位置付けられる。その中で学習者は、課題プールから、仮想ユーザが蓄積した課題を参照する。そして、それを解釈して資源管理インタフェースを操作する。学習者は、このインタフェースを通して、資源管理モジュールを操作することが可能になっている。また資源管理インタフェースは、資源管理モジュールの連携を管理している。

システムの構成図を図 1 に示す。

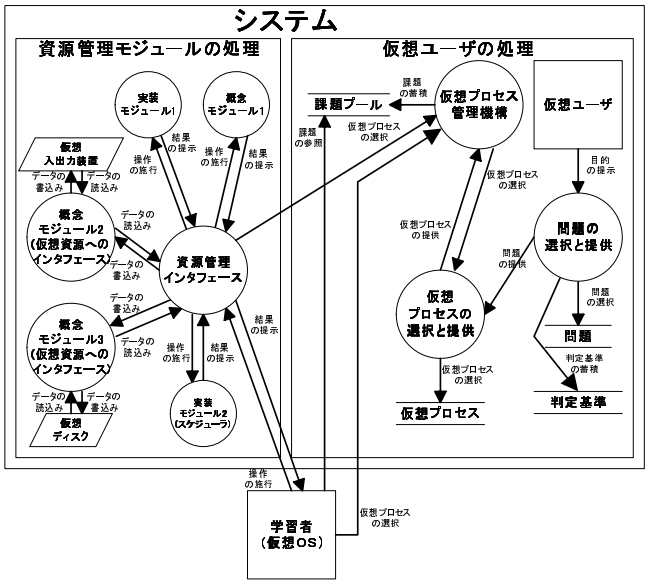

図 1: システムの構成図

5　システムの設計と実装

本システムの設計と実装を次に示す。資源管理インタフェースのイメージを図 2 に示す。

5.1　仮想ユーザ

仮想ユーザの提供する学習内容は、上位から目的、問題、課題の順に木構造を構成している。目的は、一般的なアプリケーション利用における OS 操作を基準に学習内容を分けている。例えば、エディタを使った文書作成である。問題は、目的の中で起こりうる OS への要求やトラブルの事例のことであり、複数が集まって一つの目的を構成している。また、その解決のために仮想プロセスが利用される。仮想プロセスとは、本来のプロセスの役割を担うものであり、C 言語で記述して提供される。これは、仮想プロセス管理機構と連携して課題を生成する。これにより課題は、仮想プロセスから切り出したコードの一部分 (システムコールなど) となる。学習者は、この課題だけを参照して学習を進めていく。また学習者にとっては、仮想プロセスを切り換えることが、課題を選択することと同じになる。

5.2　資源管理モジュール

資源管理モジュールは、OS の資源管理方式を可視化し、学習者にその仕組みを学習できる形で提供するモジュールである。これは、OS のデータ構造の違いで区分けされ、資源管理インタフェースを介して連携している。本システムは、モジュールの追加、除去を容易にしている。これによって学習者は、OS のアルゴリズムやデータ構造の有無による、OS の資源管理の変化を学ぶことができる。

モジュールは、次の三種類が存在する。

1. 概念モジュール
   これは、OS の資源管理アルゴリズムを可視化している。その表現内容は、参考書などで使われる概念図を参考にしている。
2. 実装モジュール
   これは、OS の実装構造を GUI コンポネントを用いて可視化している。OS の実装構造とは、既存の OS のソースコードで使われる変数名や関数名のことである。
3. 移動モジュール
   これは、ポインタにより多くのデータ構造とリンクする構造体を可視化している。この構造体とは、例えば、プロセスやページ構造体のことである。移動モジュールは、異なるモジュール間での受渡しを可能にしている。

また資源管理モジュールは、仮想ディスクなどの仮想装置にアクセスするインタフェースの役割も担う。仮想装置とは、ハードウェアが扱うデータ部分だけを仮想化したものである。

5.3　資源管理インタフェース

資源管理インタフェースは、そのウィンドウ内に資源管理モジュールを表示するための二か所の表示領域を設けている。学習者は、仮想ユーザが提供する課題を選択するとカーネル関数が提供される。カーネル関数は本インタフェース上で選択できる。学習者は、それを本システムの操作の制御に利用する。

学習者の操作内容は、実際のカーネルの仕事に準拠する。その中で概念モジュールはグラフィクスをマウスのドラッグ&ドロップなどで、実装モジュールは GUI コンポーネントを使った値の閲覧やボタンの押下などで、操作が実現される。概念モジュールと実装モジュールはリンクしていて、一方を操作すると、他方にも影響を与える。学習者は、本システムの GUI を操作していく上で、この二つの間を遷移する。これによって学習者は、これらの二つの関連性を学習することができる。

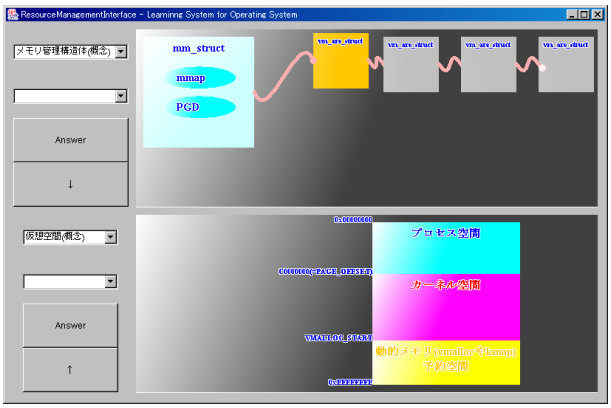

図 2: 資源管理インタフェース

6　おわりに

本報告は、体験学習モデルを利用した OS 学習システムの開発について述べた。本研究は、プロトタイプで明らかになった課題を解決するために、システムの再設計を行い、プロセスとメモリ管理に特化したシステムを実現した。今後、本システムが OS 学習を支援できたのかどうかを検証する。また、プロトタイプで実現した、ファイルシステムをベースにしたシステムを取り込んでいく。

参考文献

[1] 高橋浩和、三好和人：Linux カーネル 2.4 の設計と実装, 株式会社五橋研究所、Linux Japan 編集部, 2000

[2] 米田孝文、早川栄一、高橋延匡：体験学習モデルによる OS の学習システムの実現, 情報処理学会システムソフトウェアとオペレーティングシステム研究会, 2002/06