

# 計算機システム学習支援環境「港」における OS 可視化ツールの開発

本橋 大樹<sup>(1)</sup> 西野 洋介<sup>(2)</sup> 早川 栄一<sup>(3)</sup>

拓殖大学大学院工学研究科<sup>(1)</sup>

東京都立府中工業高等学校<sup>(2)</sup> 拓殖大学工学部情報工学科<sup>(3)</sup>

## 1. 研究の背景と目的

現在、拓殖大学工学部情報工学科早川研究室で行われている研究としてオペレーティングシステム（以下 OS）の学習支援環境「港」というプロジェクトが行われている [1][2]。このプロジェクトはスケジューリングといった OS の基本的な動作の学習に用いるために開発されているもので、アニメーションを用いることにより学習を容易にすることを目的としている。

しかし、本プロジェクトにおける OS 可視化ツールには操作のしにくさや可視化画面レイアウトといった問題がある。これは学習者にとっては効率よく学習が行えないといったことが起こる可能性がある。

そこで、本研究では OS 可視化ツールの整備を行い、使い勝手を向上させることで学習の効率を向上させることを目的とする。

## 2. 問題分析

### (1) OS 可視化ツールの問題点

現在の OS 可視化ツールは、OS の基礎学習のためのツールと実際の OS の動作を可視化で再現するツールの 2 種類に分かれている。基礎学習のためのツールでは、タスクの生成などはすべて手動で操作するものである。OS の動作を可視化するツールは、OS の動作を、ログデータを用いて可視化するものである。

現状では、ツールが 2 つ存在するため OS の動作原理の学習の後に実際に OS を動作させて学習するという一貫した学習が行えない。1 つのツール内で一貫した学習が行えることで学習効率が上がると思われる

Development of Tool of Making to visible for Operating System on Educational Support Environment "Minato"

(1) Daiki Motohashi · Graduate School of Engineering, Takushoku University

(2) Eiichi Hayakawa · Faculty of Engineering, Takushoku University

(3) Yousuke Nishino · Tokyo Metropolitan Fuchuu Technical High School

## (2) OS 可視化ツールへの要求

従来の可視化環境では次のような要求が挙げられる。

- 操作が複雑であり、学習者はまず操作に慣れることが必要であること
- 可視化画面が学習のサポートをしきれていないこと
- 複数のアルゴリズムの連動が行えないこと

これらの要求を満たすような可視化ツールを開発することで学習者にとって利用しやすく学習しやすいシステムになると考えられる。

## 3. 特徴

本システムの特徴は次に挙げるものである。

- 操作を簡略化し、操作性を向上させること。
- 2 種類の可視化ツールを 1 つに統合させ、一貫した学習を可能にする。
- 表示方法を見直し、より学習しやすくする

## 4. 設計

### 4.1 全体構成

図 1 に全体構成を示す。2 種類あった可視化ツールを 1 つに統合した。可視化制御部で操作を監視することで、ログの読み込みと手動での操作の両方の操作を実現する。

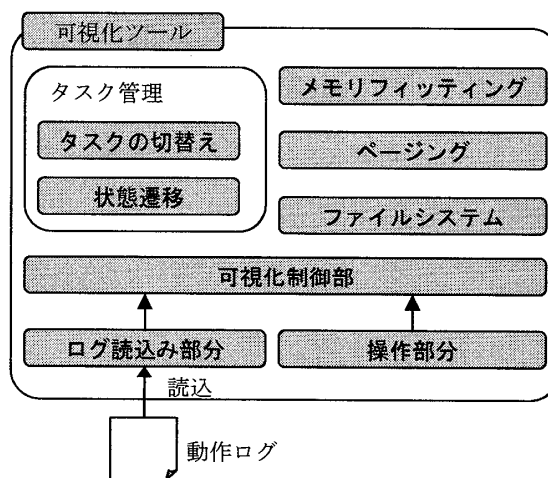


図 1 全体構成図

## 4.2 可視化ツールの統合

現在、「港」プロジェクトの可視化環境にある2種類の可視化環境にはOSの基礎概念の学習と実際のOSの動作の確認を行うための環境が用意されている。これらの学習は一貫して行うことでより理解が深まることが考えられる。そのため、これら2つの可視化環境を統合することで1つのツール内で一貫した学習を行うことができるようにする。

手動で操作を行う場合、設定画面を開きタスクを生成していき、アニメーションによる可視化を行う。ログを読み込む場合は、ログに書かれているデータを元に可視化を行っている。

ツールを統合させる上での問題はログの読み込みと手動での操作という2種類の操作がある点である。この問題を解決するために、それぞれの操作を管理する可視化制御部を実装する。

可視化制御部では、それぞれの操作を監視するものである。ファイルが読み込まれた場合は、読み込んだログデータを可視化部分に渡す。手動で操作された場合は、生成されたタスクなどのデータを制御部が可視化部分に渡す。このような方法をとることで、ログの読み込みと手動での操作の両方の操作が行えるようになる。

## 4.4 表示方法の見直し

可視化のアニメーションで共通している操作は次のものである。

- スタート
- ストップ
- リセット
- アニメーションの表示速度の変更

これらは全ての可視化画面で行われる操作なので、常に画面に表示させておく。これ以外の操作はタスク管理部分でタスクの生成・削除、優先度の設定などそれぞれの可視化部分での固有の操作である。これらの操作は、可視化画面の動作の前に事前に設定するものであり、頻繁に操作されることはない。そのため常に画面に表示させている必要はなく設定を変更するときだけポップアップウィンドウなどで表示させる。これによって可視化部分が見やすくなるため学習者はより学習に集中できるようになると思われる。

## 4.5 機能の連動

機能の連動の対象としては、OSで主な機能であり現在の可視化環境で特に関連があるタスク管理とメモリ管理の部分とする。

タスクが生成されたときに可視化制御部に渡

されたデータからそれぞれの可視化画面に表示する。スタートのボタンが押された時点で制御部からそれぞれ実行されるようにすることで連動することができる。動作時には実行しているタスクとアクセスされているメモリ領域を同じ色で表すことで、状態変化によるメモリアクセスの様子を知ることができる。

また、それぞれの可視化画面は図2のようにあらかじめそれぞれを表示させておくことでそれぞれが連動して動作することをしめす。

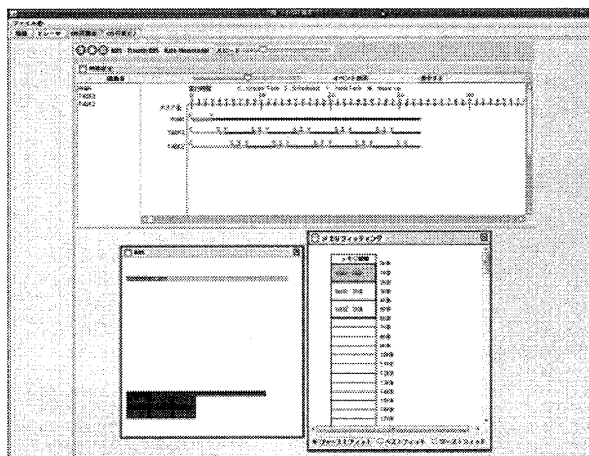


図2 実行画面

## 5. おわりに

本研究では、「港」プロジェクトの可視化環境にメモリ管理の可視化としてメモリフィッティングの可視化画面の追加を行った。また、タスク管理の可視化画面とともにメモリ管理の画面を表示させ、同期して表示することで機能を連動させた。

今後の課題は、操作の簡略化である。画面上にあるボタンなどの操作をできる限り少なくすることでより学習しやすい環境を構築する。

## 参考文献

- [1] 大角圭吾ほか, システムソフトウェア教育支援環境「港」における実装レベルのOS学習支援システム, Vol.2005, No.15, pp. 9-14, 2005-CE-78-(2)
- [2] 坂本祐紀ほか, OS学習支援環境「港」の学習教材におけるユーザインタフェースに関する検討, 第70回情報処理学会全国大会, 2008
- [3] 久保秀士, OS概論, 共立出版, 1988
- [4] ジョゼフ・オニール, 独習Java第3版, 翔泳社, 2005