

GearsOS の Paging と Segmentation

桃原 優¹ 東恩納 琢偉¹ 河野 真治^{1,†1}

概要: 現代の OS では、User Space で Page Table Entry によるメモリ管理を行える OS は少ない。本研究室ではメタレベルの処理を記述できる CbC と CbC を用いて実装する GearsOS の開発を行っている。CbC を用いることでメモリ管理などの資源管理を行えるようになるが、現在の GearsOS でのメモリ管理では単に Page Table Entry をコピーする Fork で実装している。さらに、資源管理を行える CbC で軽量のハードウェアでも動かせるように Arm のバイナリを出力する Xv6 という OS を CbC で書き直して GearsOS を開発する手法も行なっている。本論文では Xv6 を参考に GearsOS にメモリ管理を行う API を考察する。

Implement Paging and Segmentation on GearsOS

1. User Space でのメモリ管理

基本、メモリやスレッド、CPU の管理は OS が行なっている。時代とともに急速に進歩するハードウェアやソフトウェアに対して、OS も

2. Gears OS

本研究室では並列実行のサポートと、信頼性を保証する Gears OS の開発を行っている。従来の OS が行うメモリ管理や並列実行などは Meta レベルで処理される。メタレベルの処理を行える CbC という言語で Gears OS を実装する事で、ノーマルレベルから並列実行環境に合わせた記述ができるように設計や実装を行う。

2.1 Continuation based C

本研究室では、Code Gear と Data Gear という単位でプログラムを記述する CbC と CbC を用いて実装する Gears OS の開発を行っている。Code Gear は CbC における最も基本的な処理の単位である。入力と出力を持ち、goto によって Code Gear から次の Code Gear へ遷移する事で継続的に処理を行う事によって並列処理を行うことができる。Data Gear は Gears OS におけるデータの基本的な単位である。Input Data Gear と Output Data Gear があ

り、Code Gear の遷移の際に Input Data Gear を受け取り、Output Data Gear を書き出す。

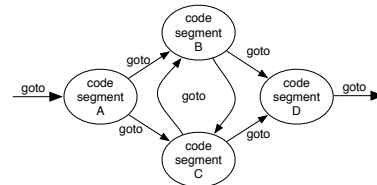


図 1 Code Gear 間の継続

2.2 Meta Code Gear と Meta Data Gear

Gears OS ではメタ計算を Meta Code Gear, Meta Data Gear で表現する。CbC でもノーマルレベルとメタ計算を行うメタレベルの記述の 2 種類がある。この 2 つのレベルはプログラミング言語レベルでの変換として実現される。メタレベルへの変換は、Perl による変換スクリプトで実装している。Gears OS では、Meta Code Gear は通常の Code Gear の直前、直後に挿入され、メタ計算を実行する。Code Gear 間の継続はノーマルレベルでは図 2 の上のように見えるが、メタレベルでは Code Gear は図 2 の下のよう

に継続を行っている。

図 2 に Gears OS の構成図を示す。

3. Paging と Segmentation

メモリ管理の手法に、Paging と Segmentation がある。

¹ 情報処理学会
IPJS, Chiyoda, Tokyo 101-0062, Japan
^{†1} 現在、情報処理大学
Presently with Johoshori University

- [6] Higham N.J.: *Handbook of Writing for the Mathematical Sciences*, SIAM (1998).
- [7] 情報処理学会論文誌ジャーナル編集委員会：投稿者マニュアル (online), 入手先 http://www.ipsj.or.jp/journal/submit/manual/j_manual.html (2007.04.05).
- [8] 情報処理学会論文誌ジャーナル編集委員会：べからず集 (online), 入手先 <http://www.ipsj.or.jp/journal/manual/bekarazu.html> (2011.09.15).

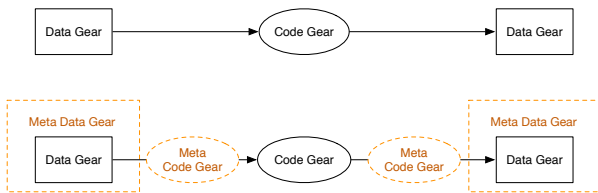


図 2 ノーマルレベルとメタレベルの継続の見え方

Paging ではメモリを Page と呼ばれる固定長の単位に分割し、メモリとスワップ領域で Page を入れ替えて管理を行う。Segmentation では可変長の単位に分割するので、Paging と比べてメモリを無駄なく扱うことができる。近年では、Paging を採用している OS が多い。

4. Xv6

5. Paging と Segmentation の GearsOS への実装

ファイルは次のようになる。下線部は投稿時に省略可能なもの。

5.1 表題・著者名等

謝辞 A4 横型に対するガイドを基に、本稿を作成した。クラスファイルの作成においては、京都大学の中島 浩氏にさまざまなご教示を頂き、さらに Bi T_E X 関連ファイルの利用についても快諾頂いたことを深謝する。また、A4 横型に対するガイドを作成された当時の編集委員会の担当者に深謝する。

参考文献

- [1] 奥村晴彦：改訂第 5 版 $\text{L}^{\text{A}}\text{T}_\text{E}\text{X} 2_\epsilon$ 美文書作成入門，技術評論社 (2010).
- [2] Goossens, M., Mittelbach, F. and Samarin, A.: *The LaTeX Companion*, Addison Wesley, Reading, Massachusetts (1993).
- [3] 木下是雄：理科系の作文技術，中公新書 (1981).
- [4] Strunk W. J. and White E.B.: *The Elements of Style, Forth Edition*, Longman (2000).
- [5] Blake G. and Bly R.W.: *The Elements of Technical Writing*, Longman (1993).