

Real-Time Mach 3.0 における I/O サーバの構成と評価†

1H-2

持田 茂人^{1‡} 緒方 正暢^{2‡} 河内谷 清久仁^{2‡} 西尾 信彦³ 和田 英彦^{4‡} 徳田 英幸⁵²オムロン(株) システム総合研究所 ²日本 IBM(株) 東京基礎研究所 ³慶應義塾大学 環境情報研究所⁴横河電機(株) オープンシステム研究所 ⁵慶應義塾大学 環境情報学部/カーネギーメロン大学 計算機科学科

1 はじめに

マルチメディア環境下では、音声や動画像など時間的制約を持ったデータオブジェクトを扱うために、I/O デバイスのハンドリングに QOS を意識した細かな制御を行わなければならない。またさまざまな新しいデバイスもサポートしなければならない。これらの要求に柔軟に対応するため、従来のカーネル内に実現されたデバイスドライバが変わって、マイクロカーネル上のユーザレベルサーバが提案されている。本稿では Real-Time Mach 3.0(以下 RT-Mach) 上に実装された I/O サーバの性能評価を行ない、ユーザレベル I/O サーバ実装の際の問題点について検討する。

2 I/O サーバの構成

マイクロカーネルアーキテクチャは、これまでの OS の機能をできる限りユーザレベルサーバ化し、OS の構成に必要な最小限の機能のみをマイクロカーネルで実現している。マイクロカーネルを用いたシステムでの I/O ハンドリングには、いくつかの実装が考えられる。代表的なものとして以下のものがあげられる。

2.1 カーネル内ドライバ方式

伝統的なモノリシックカーネルアーキテクチャと同様に I/O ドライバをマイクロカーネル内に置き、I/O のハンドリングをカーネルモードで行う。この方式では、I/O 処理がカーネルアドレス空間で、データを要求する OS サーバやユーザタスクがユーザアドレス空間でそれぞれ実行されることになり、カーネル-ユーザ間のデータのコピーによるパフォーマンスの低下が指摘されている。その他にも、プリエンタビリティの低下、I/O の自由度の制限などが問題点としてあげられる。

2.2 OS サーバ内ユーザレベルドライバ方式

I/O ドライバを OS サーバ内に持ち、ユーザレベルで実行する。プリエンタビリティの向上、柔軟な I/O 構成が望める。しかし、I/O アドレス空間はマイクロカーネルが管理するため、データの転送によるパフォーマンスの低下は改善されない。この問題を解決するため、I/O 空間をユーザアドレス空間にマッピングしたユーザレベルタスクの OS サーバで I/O ドライバを実現する方法も試みられている [1, 2]。また、(i)I/O インターフェイスが OS で

規定される、(ii) マイクロカーネル上の複数 OS で単一デバイスをアクセスすることができないなどの問題もある。

2.3 I/O サーバ方式

基本的にはユーザレベルドライバと同じであるが、I/O サーバのみで独立したタスクとなっている点で異なる。これにより OS に依存せずデバイスに応じた制御が行なえる。問題点としては、頻繁に I/O 要求が発生した時のコンテキストスイッチによるオーバーヘッドが考えられる。

現在我々が構築中の RT-Mach をベースとしたマルチメディアシステム [3, 4, 5] においては、動画像や音声などの時間的制約を持った大量のデータを扱うために、プリエンタビリティや QOS を考慮した I/O ハンドリングを行わなければならない。同時に、関連した複数のデータストリーム間での同期を保証しなければならない。我々はこれらの要求を満足させるために、サウンドサーバを I/O サーバとして実現した。

3 サウンドサーバ

3.1 サウンドサーバの概要

RT-Mach 上で I/O サーバ方式で実現された音声ボード(Media Vision 社製 Pro Audio Spectrum 16) 用のサーバを使い実験を行なった。サウンドサーバは、要求待ちスレッド、割り込みスレッドなどの複数のリアルタイムスレッドから構成される。サウンドデータは、マイクロカーネルとサウンドサーバで共有されたメモリ空間に置かれ、コマンドはマイクロカーネルで提供されるデバイスインターフェイスを使って実行される。デバイスの割り込みは、マイクロカーネル内のデバイスドライバからイベントを通じてサウンドサーバの割り込みスレッドに通知される。

サウンドボードを使うクライアントタスクは、その要求を実時間拡張された Mach IPC[6] を通じて行なう。サウンドボードは、DMA 機能を有しており、サーバは再生要求を発行するだけでブロックされることなく、処理が継続される。

3.2 サウンドサーバの性能評価

今回は音声データの再生を異なるデータサイズで行ない、IPC、データコピー、I/O ハンドリングなどの各処理時間を高分解能タイマを用いて実測した。実験には RT-Mach MK83, Intel i486DX2/66MHz¹ を搭載した IBM PC AT 互換機² を用いた。データサイズを指定できるクライアントを実行させ、サーバ、クライアント内の実行時にタイマボード³ を参照し、各処理にかかる時間を計測した。データ計測は、図 1 で示す各時点で行なった。I/O 要求サイズを DMA バッファの 16Kbyte から 512byte まで変化させた。測定には、データ長 16bit, サンプリング周波数 8KHz の音声データを使用した。図 1 中の (1-9) は、次の各点を示す。

¹米インテル社の商標です。²米 IBM Corp. の商標です。³精度は 250 ナノ sec.

"Evaluation of I/O Server on Real-Time Mach 3.0,"

Shigeto Mochida¹, Masanobu Ogata², Kiyokuni Kawachiya², Nobuhiko Nishio³, Hidehiko Wada⁴, Hideyuki Tokuda⁵¹OMRON Co., Advanced Systems Research Center, Shimokaiinji, Nagaokakyo, Kyoto 617, ²IBM Research, Tokyo Research Laboratory, ³Keio University, ⁴Yokogawa Electric Co., ⁵Keio University / Carnegie Mellon UniversityE-Mail: ¹moti@nff.ncl.omron.co.jp

†この研究は、情報処理振興事業協会(IPA)が実施している開放型基盤ソフトウェア研究開発評価事業「マルチメディア統合環境基盤ソフトウェア」プロジェクトのもとに行なわれた。‡開放型基盤ソフトウェア 湘南藤沢キャンパス研究室の研究員として IPA に登録されている。

| data size | 1 要求毎 (usec) | | | | | | | | | IPC(%) | 200KB 換算 (usec) time |
|-----------|--------------|------|---|------|------|-----|-----|---|-------|--------|-------------------------|
| | a | b | c | d | e | f | g | h | total | | |
| 16KB | 2350 | 2760 | 8 | 2570 | 1220 | 120 | 185 | 8 | 9221 | 58.9 | 115263 |
| 8KB | 1190 | 1340 | 8 | 1340 | 1160 | 115 | 180 | 8 | 5341 | 53.2 | 133525 |
| 4KB | 595 | 780 | 8 | 718 | 1100 | 110 | 175 | 7 | 3493 | 48.0 | 174650 |
| 2KB | 310 | 490 | 7 | 400 | 1050 | 114 | 165 | 7 | 2543 | 43.0 | 254300 |
| 1KB | 165 | 360 | 7 | 250 | 1015 | 100 | 165 | 7 | 2069 | 38.9 | 413800 |
| 512B | 90 | 280 | 7 | 165 | 915 | 100 | 165 | 7 | 1729 | 37.5 | 691600 |

表 1: 要求データサイズと処理時間

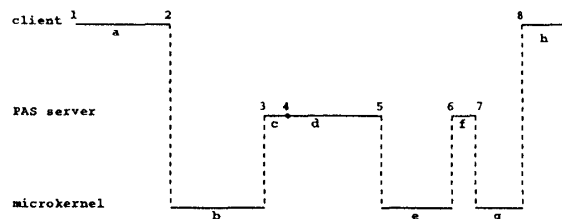


図 1: I/O 要求処理の流れと計測点

1. I/O 要求 (クライアントタスク)
2. Mach IPC(クライアントタスク)
3. Mach IPC(サウンドサーバ)
4. server function(サウンドサーバ)
5. device_write[7] 発行 (サウンドサーバ)
6. device_write 復帰 (サウンドサーバ)
7. Mach IPC(サウンドサーバ)
8. Mach IPC(クライアントタスク)
9. I/O 要求完了 (クライアントタスク)

図 1 中の (a-h) は、次の各処理を示す。

- a. クライアント: サウンドサーバへの I/O リクエストメッセージ作成
- b. マイクロカーネル: IPC 処理
- c. サウンドサーバ: クライアントからのメッセージのアンマージアル
- d. サウンドサーバ: データのコピー
- e. マイクロカーネル: device_write(I/O 空間のアクセス)
- f. サウンドサーバ: IPC エラーチェック
- g. マイクロカーネル: IPC 処理
- h. クライアント: IPC エラーチェック

測定結果を表 1 に示す。表中の 200KB 換算の処理時間は、各データサイズによる処理時間を 200KB 処理に必要な時間に換算したもので、コンテキストスイッチによるオーバーヘッドなどは含まれない。

この結果から、データのコピーが発生する a,b,d が、データサイズに比例して増加していることが確認できる。一方、e の device_write システムコールではデータサイズが変化してもその処理時間はほぼ一定である。これは、サウンドサーバがカーネルとの共有メモリ上に DMA バッファを置いてデバイスをアクセスしているためである。同じ量のデータを再生する場合、I/O 要求サイズが大きいほど有利である。

4 まとめと課題

現在のサウンドサーバでは、データサイズの増加にともない IPC の発生回数が減少するため、I/O 要求サイズを大きくした方が良いパフォーマンスが得られた。しかし、我々が対象としている動画像や音声などの時間的制約を

持ったデータオブジェクトでは、1 要求当たりのデータサイズを大きくするとジッターの原因ともなる。要求データサイズを大きくしようとする、I/O 要求の際に実 I/O の時刻を指定するような timed I/O の機構も必要である。IPC のコストの削減に関しては、クライアントタスクが IPC メッセージバッファに直接データを用意することによってメッセージマーシャリングの際のコピーを減らすことも考えられる。スケジューリングを考慮したデータの測定も今後の課題である。

謝辞

本研究を行なうにあたり協力/助言していただいている慶応大学「マルチメディア統合環境基盤ソフトウェア」プロジェクトの皆様へ感謝いたします。さらに、御指導いただいている慶応大学環境情報学部の斎藤信男教授に感謝いたします。

参考文献

- [1] A. Forin, et al.: "An I/O System for Mach 3.0," *Proc. of USENIX Mach Workshop*, pp.163-176 (1991).
- [2] 加藤, 他: "マイクロ・カーネル・アーキテクチャにおける I/O driver 実装方法に関する一考察," 情報処理学会 OS 研究会, 93-OS-58(1993).
- [3] 西尾, 他: "マイクロカーネルによる連続メディア処理の基盤技術," 第 5 回コンピュータシステム・シンポジウム論文集, pp.17-24 (1993).
- [4] 平林, 他: "実時間メディアサーバの設計," 第 5 回コンピュータシステム・シンポジウム論文集, pp.25-32 (1993).
- [5] 多田, 他: "実時間カーネルを用いた連続メディアベースの設計," 第 5 回コンピュータシステム・シンポジウム論文集, pp.33-40 (1993).
- [6] T. Kitayama, et al.: "RT-IPC: An IPC Extension for Real-Time Mach," *Proceedings of USENIX 2nd Microkernel and Other Kernel Architecture Symposium*, (1993)
- [7] OSF and CMU: "Mach 3 Kernel Interfaces,"