WS向け高速分散共有メモリシステムの試作と評価
1B－6 ―― ソフトウェア性能評価 ――

小林　伸治　　陣崎　明
(株) 富士通研究所

1　はじめに

既存のUNIXワークステーション(WS)に接続可能な分散共有メモリPUMA-IIを試作した。PUMA-IIはWSのVMEバスに搭載したメモリ(分散メモリ)を100Mb/sのトークンリングネットワークで結合し、最高11.5MB/sのメモリ間ページ転送速度を実現する。分散共有メモリ制御にネットワーク仮想記憶(NET-VMS)方式[1]を採用することにより、高速転送性能とWSソフトウェアの簡略化を実現している[2](図1)。本稿では、PUMA-IIを共有RAMディスクとして利用するデバイスドライバを用い、PUMA-IIをUNIXユーザプロセスから使用した場合の性能評価を行う。

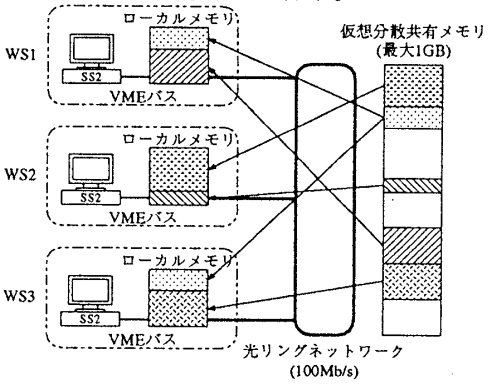

図1: PUMA-IIシステム

2　共有RAMディスク

共有RAMディスクは、メモリ(RAM)をディスクと同じ操作で扱えるようにするRAMディスクの仕組みを分散共有メモリを利用して複数のWSが共有できるように拡張したものである。

共有RAMディスクに対する読み出しアクセス時の動作は次のようになる(図2)。本デバイスドライバはユーザプロセスからreadシステムコールを受けとり、PUMA-IIの仮想共有記憶空間にページ単位でアクセスする。必要とするページが自WSのローカルメモリに存在しない場合はネットワーク経由でページを取得する必要があるが、ページ単位の複写といったネットワーク操作はPUMA-IIのハードウェアが受け持つため、デバイスドライバは必要なページの仮想アドレスをPUMA-IIのハードウェアに渡すだけでよい。複数ページを必要とする場合でも、PUMA-IIのコマンドFIFOにまとめて書き込める。PUMA-IIは必要なネットワーク操作を行った後、操作の完了を応答FIFO経由でデバイスドライバに通知する[2]。

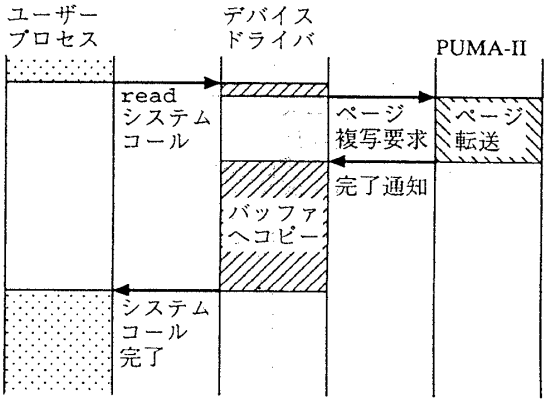

図2: readアクセス時の処理のながれ

共有RAMディスクへの書き込み操作では、複数WS間でのコンシステンシ制御を行う必要がある。共有RAMディスクを構成するPUMA-IIの分散共有メモリはハードウェアによるコンシステンシ制御機能を持っているので、各WSが持つローカルメモリ間ではコンシステンシ制御が行える。しかし、UNIXはカーネル内にディスクキャッシュを持つため、ディスクキャッシュも含めたコンシステンシ制御が必要である。すなわち、コンシステンシ制御によりローカルメモリ上のデータの書き換えもしくは無効化が行われた場合、それをディスクキャッシュにも反映する必要がある。しかし、ディスクキャッシュの操作はデバイスドライバからは行えないため、コンシステンシ制御を行うにはカーネルを書き

Implementation and Evaluation of a High-Speed Distributed Shared Memory System for Workstations (Software Performance Evaluation)
Shinji Kobayashi, Akira Jinzaki
Fujitsu Laboratories Ltd.
1015 Kamikodanaka, Nakahara-ku, Kawasaki 211, Japan

換える必要が生じる。本デバイスドライバ開発の主目的はPUMA-IIの性能評価にあるため、今回はコンシステンシ制御の実装を見送り複数WSからの書き込みをサポートしないこととした。共有RAMディスクのデータはあらかじめ1つのWS(サーバ)で設定しておき、他のWSは読み出しアクセスのみを行う。他のWSが共有RAMディスクをマウントした後は、サーバからも書き込みは行えない。

3 性能評価

SPARCstation2を用いた測定結果を図3に示す。共有RAMディスク、カーネル内メモリにディスク領域を確保する方式のRAMディスク[3]、SCSIハードディスク、Ethernetを用いたNFSについて、1MBのファイルをシーケンシャル読み出しするときの時間を測定した。ディスクキャッシュの影響を排除するため、マウント直後に計測を行っている。

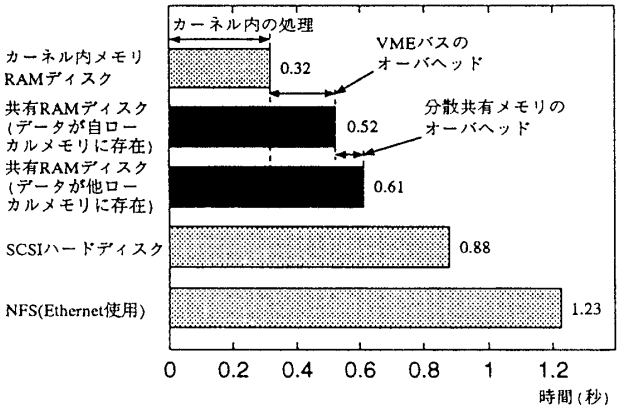

図 3: シーケンシャルアクセス時間

その結果、共有RAMディスクの性能は、データが自WSのローカルメモリに存在した場合でNFSの2.4倍、自WSのローカルメモリに存在せず、ネットワーク経由でデータを転送する場合にはNFSの2.0倍であった。SCSIハードディスクに対しては、同1.7倍、1.4倍である。本試作の性能は実用的であることを確認した。

データが他WSに存在する場合の共有RAMディスク処理時間の内訳は表1の通りである。

すなわち、PUMA-IIによる共有RAMディスクではカーネル内処理とVMEバス転送がボトルネックであり、ネットワーク制御・転送は相対的に小さいことがわかる。

表 1: 共有RAMディスク処理時間の内訳

処理内容	内訳
カーネル内処理	52%
VMEバスオーバヘッド	33%
ネットワーク転送	15%

ボトルネックとなっているVMEバス性能は、Sbus等を用いれば高速化可能である。ネットワーク転送は現状でも十分高速であるが、必要であればさらに高速なネットワークを用いることもできる。NET-VMS方式によりPUMA-IIのハードウェアがネットワーク制御を行いソフトウェアは関与しないため、ネットワークを高速化すればネットワーク転送の実効速度もほぼリニアに向上できる。

処理時間の大半を占めるカーネル内処理はカーネル内のバッファからユーザプロセス空間へのデータコピーなどであり、UNIXのディスクデバイスとして機能させるためには不可欠なものである。したがって、分散共有メモリをUNIX環境下で有効利用するためには、ディスクデバイスのようにカーネル経由で使うのではなく、分散共有メモリをユーザプロセス空間に直接マップして使う必要がある。

4 まとめ

今回の評価により、PUMA-IIの分散共有メモリはUNIXの枠組から利用しても十分に高速であることを確認した。また、ディスクデバイスとして利用した場合にはカーネル内部での処理やVMEバスのオーバヘッドが大きいことが明らかになった。今後は、バスオーバヘッド削減などのハードウェア改良とともに、分散共有メモリをユーザプロセス空間に直接マップして利用する情報共有方式を開発していく予定である。

参考文献

[1] 陣崎他: ネットワーク仮想記憶方式によるマルチプロセッサの試作について、信学技報 CPSY87-26(1987-11)

[2] 新家他:WS向け高速分散共有メモリシステムの試作と評価 — ハードウェアアーキテクチャ —、本大会予稿

[3] Writing Device Drivers、SunOS 4.1 Manual