5J-2

DIMMnet-2向けWindowsドライバの設計と実現

金井 遵† 森 拓郎†† 荒木 健志†† 中條 拓伯‡‡ 並木 美太郎‡‡

†東京農工大学工学部情報コミュニケーション工学科

††東京農工大学工学教育部情報コミュニケーション工学専攻 ‡‡東京農工大学大学院共生科学技術研究部

1 はじめに

近年，HPC の分野において，多数の PC を相互に接続した PC クラスタシステムの躍進は目覚しい．今後，分散処理環境も多様性を求められていくと思われるが，HPC 用超高速ネットワークインタフェース DIMMnet-2 においては Windows をはじめとしたコモディティOS 環境下での分散処理環境が全く整っていない状況にある．そこで本稿では，DIMMnet-2 を用い，Windows 向けに DIMMnet-2 用デバイスドライバを設計および実装することで，DIMMnet-2 の有用性の提示および，OS に手を加えることなく，コモディティOS と単一仮想記憶管理によってクラスタシステムを構築する方法を提案する．

2 DIMMnet-2

DIMMnet-2 は安価にシステムを構築できる DIMM スロットへハードウェアを接続することにより，従来の汎用バスに接続するタイプの HPC 用 NIC に比べ 1/10 程度のアクセスレイテンシ，最上位レベルの帯域幅を実現しているネットワークインタフェースである．

DIMMnet-2 は，SO-DIMM による数百 MB〜数 GB の大容量バッファを持ち，通信用バッファやデータ待避用領域として利用することが可能である．ホストの主記憶へはウィンドウと呼ばれる小容量のバッファおよび，各種レジスタがマップされ，ウィンドウに読み書きするデータを，レジスタに命令であるプリミティブを読み書きすることにより大容量バッファやリモートノードの DIMMnet-2 へのアクセスを行う．

また，ローカルおよびリモートの大容量バッファへのアクセス方法として，通常の連続的なロードストアに加え，等間隔のデータをロードストアするストライドアクセス命令等の多様な拡張命令も用意されている．

3 RAM ディスクドライバ ”AT-dRAM”

3.1 目的

DIMMnet-2 の大容量バッファをディスク領域とする RAM ディスクドライバを開発し，DIMMnet-2 の煩雑な間接アクセス機構をプログラマに対して隠蔽するとともに，クラスタシステムで多く利用されるデータ共有の機構である分散共有メモリ，分散ファイルシステムを実現する．

3.2 AT-dRAM ドライバの概要

AT-dRAM では，複数の DIMMnet-2 を一つのディスクとして仮想化し，リモートへのアクセスおよび，ウィンドウを使った間接アクセス機構を隠蔽する．これにより，図 1 のように通常のファイルアクセス方法により，ローカルおよびリモートの DIMMnet-2 大容量バッファへのアクセスが可能になる．

さらに併せてメモリマップドファイル機能を利用することで，仮想的にローカルおよびリモートの DIMMnet-2 大容量バッファを主記憶にマップすることも可能であり，従来の主記憶を利用するアルゴリズムがそのまま適用可能になるため，DIMMnet-2 プログラミングが容易となる．

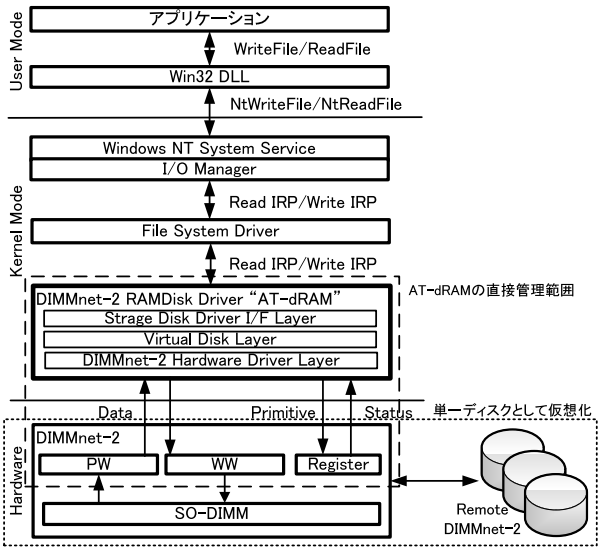

図 1: AT-dRAM 概要図

Windows におけるファイルシステムは階層化しており，主にディスクを管理するストレージディスクドライバ，ファイルシステムを管理するファイルシステムドライバからなる．AT-dRAM はストレージドライバとして実装され，パフォーマンスを最重要視し，ストレージドライバが通信部分も含めて処理を行うことで分散ファイルシステムとしての機能を持たせる．

複数のノードにおいて AT-dRAM をロードすることで，分散ファイルシステムとしての利用が可能である．

4 DIMMnet-2 操作ドライバ ”MT-dNET”

4.1 目的

AT-dRAM では DIMMnet-2 の存在を極力隠蔽したが，メッセージパッシングシステムの実装，ストライ

Design and Implementation DIMMnet-2 Device Drivers for Microsoft Windows

Jun Kanai†, Takuro Mori††, Takeshi Araki††, Hironori Nakajo‡‡ and Mitaro Namiki‡‡

Dept. of Computer and Communication Science, Tokyo Univ. of Agri. and Tech. (†)

Graduated School of Computer and Communication Science, Tokyo Univ. of Agri. and Tech. (††)

Graduated School of Computer and Communication Science, Tokyo Univ. of Agri. and Tech. (‡‡)

ド命令をはじめとした各種拡張命令の利用時など，明示的に DIMMnet-2 を操作したい場合も存在する．そこで，DIMMnet-2 の機構をプログラマに見せ，応用プログラムからの明示的な操作を可能にするドライバを開発する．

4.2 MT-dNET ドライバの概要

MT-dNET では各種ウィンドウやレジスタをプロセスの仮想アドレス空間へマップする機能を提供する．

AT-dRAM においては DIMMnet-2 の大容量バッファへデータをコピーする場合，必ずドライバを介する必要があったが，パフォーマンスを重視する場合はドライバを介さず，ユーザモードのみでプリミティブ発行を行った方が有利である．MT-dNET では大容量バッファへのアクセスや，リモートノードの DIMMnet-2 へのアクセスをユーザランドで完結することが可能になり，オーバヘッドの小さい通信を可能にしている．

また，クラスタコンピューティングにおけるデータ共有の方法として，分散共有メモリが利用される一方で，明示的なメッセージパッシングが用いられることも多く，多くの場合において，メッセージパッシングは分散共有メモリよりもパフォーマンスに優れることが多い．DIMMnet-2 においてメッセージパッシングを実装する場合，MT-dNET と DIMMnet-2 の間接アクセス命令を使うことにより容易に構築可能である．

5 評価

5.1 行列乗算におけるプログラミング例

AT-dRAM と，MT-dNET およびユーザモードで動作するライブラリを使い，行列の乗算を行う場合のプログラミング例を図 2 に示す．

```
//AT-dRAM
mat_a = (CAST)dn_calloc(MATSIZE);
mat_b = (CAST)dn_calloc(MATSIZE);
mat_c = (CAST)dn_calloc(MATSIZE);
for (i = 0; i < SIZE; i++)
  for (j = 0; j < SIZE; j++)
    for (k = 0; k < SIZE; k++)
      (*mat_c)[i][j] += (*mat_a)[i][k]
                        * (*mat_b)[k][j];

//MT-dNET
for (i = 0; i < SIZE; i++){
  VL(mat_a, i * step, step);
  for (j = 0; j < SIZE; j++){
    VLS(mat_b, MATSIZE + j * szdbl, 3, SIZE, step);
    for (k = 0; k < SIZE; k++)
      mat_c[k] += mat_a[k] * mat_b[k];
  }
  VS(mat_c, MATSIZE * 2 + i * step, step);
}
```

図 2: 行列乗算例

このように，AT-dRAM では従来とほぼ同じアルゴリズムが適用可，MT-dNET とストライド命令ではキャッシュを有効に使ったプログラミングが可能である．

5.2 単一ノードによる評価

作成したドライバに関して効果を実証するため性能計測を行った．AT-dRAM と MT-dNET を用い，単体のノードで行列乗算を行った場合の性能比較結果を図 3 に示す．

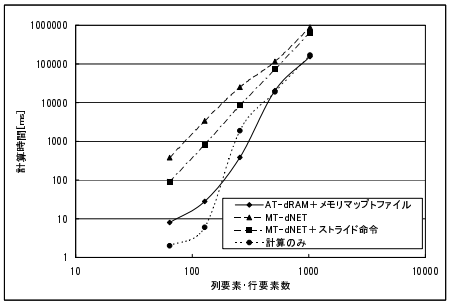

図 3: 単一ノードによる行列乗算

結果より，AT-dRAM は標準的なページアウト先の HDD より高速であるため，要素数が多くなるに従い AT-dRAM が有用に働く．また，ストライド命令を使用した場合は，使用しない場合に比べ，性能が向上しており，ストライド命令が有用であることが分かる．

5.3 分散処理による評価

AT-dRAM とメモリマップドファイルを用い，複数ノードで共有メモリを使用して行列乗算の分散処理を行った場合の結果を図 4 に示す．この実験では，分散共有メモリの管理，バリア同期等を行うユーザモードで動作するライブラリを開発し使用している．

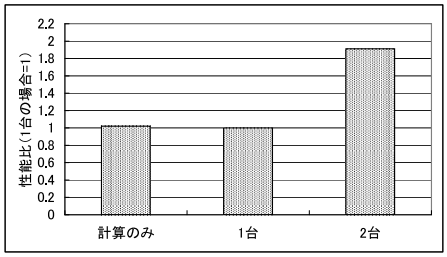

図 4: 分散処理による行列乗算 (要素数 1024*1024)

2 台での分散処理の場合，性能は 1.9 倍以上になっており，今回開発した AT-dRAM および DIMMnet-2 がコモディティ OS 下の分散処理において有用であることが示された．

6 考察および今後の課題

以上より，コモディティ OS と単一仮想記憶管理のデバイスを用いて分散共有メモリや分散ファイルシステムが実現可能であるとともに，DIMMnet-2 の各種ベクトルアクセス命令や大容量バッファが各種アルゴリズムにおいて有用であること，AT-dRAM とメモリマップトファイルを利用し容易に DIMMnet-2 プログラミングが可能であることを示すことができた．今後はファイルシステムの一貫性制御の徹底，高速化などが課題である．

参考文献

[1] 北村 聡 他: DIMMnet-2 ネットワークインタフェースコントローラの設計と実装, 情報処理学会論文誌, Vol.46, No.SIG 12, pp.13–26 (2005).

[2] David Solomon, Mark Russinovich, ”Microsoft Windows Internals”: Microsoft Press(2005).