# WSクラスタにおける遠隔ディスクアクセスとその評価分析

1B－4

相場 雄一　青木 久幸
日本電気(株) C&C研究所

## 1　はじめに

近年、高性能なワークステーション(WS)が安価で入手可能となり、また、ATM-LANなど高速な接続網が汎用化した。これにより、複数のWSを接続したWSクラスタが容易に構築可能となった。WSクラスタは各WSが独立に管理運営される分散システムであるが、高速な接続網を活かし、1つのアプリケーション(AP)を複数のWSで並列処理することが可能である。

WSクラスタではディスクが各WSに分散接続されているので、ディスクI/Oを伴うAPの並列処理を想定すると、別のWSのディスクにアクセスできる機構がなくてはならない。そこで、APからWSクラスタ内の任意のディスクにアクセスするための遠隔ディスクアクセス機構を構築し、性能評価を行ったので報告する。

## 2　遠隔ディスクアクセスの問題

別のWSのディスクにアクセスする手段としてNFSがある。しかし、データベース検索などでは、単にrawのディスクにアクセスできれば良く、NFSのファイル管理操作は余計になる。また、NFSでは通信回数を減らすため、APの動くWS側でデータをキャッシングする。このため、データの一貫性保持が不完全であり、各WS上でAPが並列動作する状況には不適切と考えられる。

データー貫性保持のため、AP側のキャッシングを省いた場合、遠隔ディスクへのアクセスで必ずWS間でデータを転送する。この時の問題点を以下に挙げる。

・必要資源の増加 … WS間のデータ転送をCPUで処理するため、必要なプロセッサ資源が増加する。

・アクセス時間の増加 … ディスク物理I/Oの時間に加え、接続網中のデータ転送時間がかかり、APにはアクセス時間が増加して見える。更に、アクセスするデータが大きい程、接続網中データ転送時間は増加する。

## 3　パイプライン転送

前述の問題点の内、アクセス時間の増加を抑える手法としてパイプライン転送を提案する。この手法は、APがアクセスするデータを、ある小さなサイズ(ブロック)に細かく分け、遠隔ディスク←→AP間におけるデータ転送をパイプライン的に流す方式である(図1)。これにより、各ブロックの通信時間やCPU消費時間を別のブロックのディスクI/O時間と重ね、APからの遠隔ディスクへのアクセス時間を全体として短縮する。

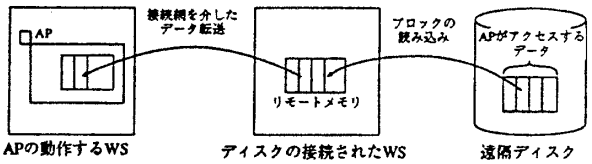


図1: パイプライン転送(リードの場合)

## 4　遠隔ディスクアクセス機構の実装

遠隔ディスクアクセス機構は、APにリンクされるファイルアクセスライブラリ(FAL)と、ディスクの接続されているWS上で動作するファイルサーバ(FS)というプロセスから構成される(図2)。

FALはrawディスクアクセスのインタフェースを提供する。APからの呼出しをFSに対する要求に変換し、該当FSに送信する。この要求に対し、FSはディスクに物理I/Oを行い、結果を要求元のFALに返す。

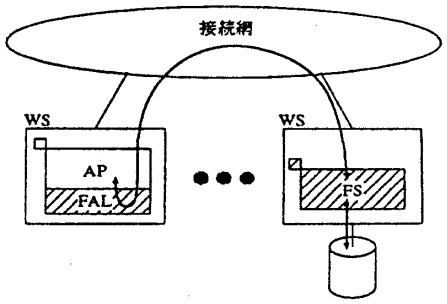


図2: 遠隔ディスクアクセス機構

### 4.1　遠隔ディスクへのアクセスパスの確定

各ディスクにはシステム内唯一の名前を付けて識別し、名前からアクセスパスを確定する必要がある。そこで、WS毎に管理しているディスク名と、そのWSの識別名を組み合わせて、システム内唯一の名前とする。

アクセスパス確定のために、マルチキャスト通信を用いる。APからディスク名を指定してFALが呼出されると、この名前を全てのFSに対しマルチキャスト通信で通知する。各FSは通知された名前のディスクを管理しているか調べ、そのディスクを管理しているFSだけが、最終的に元のAPに対し応答を通知する。これにより、ディスク名と要求先のFSとの対応付けができる。

### 4.2　パイプライン転送の実装

パイプライン転送は、APからのリード／ライト呼出しを以下のように処理することにより実現される。

APからの呼出しを、一定サイズのデータに対する要求に区切り、FSに対し一度に複数の要求を送信する。

---

Remote Disk Access Mechanism for Workstation Clusters and its Evaluation
Yuichi AIBA, Hisayuki AOKI
C&C Research Laboratories, NEC Corporation

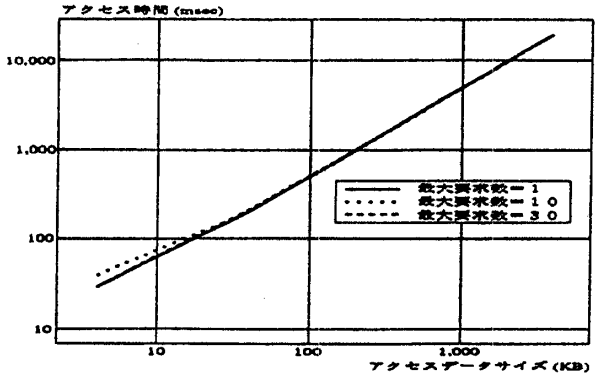


図 3: ブロックサイズ4KB の時の結果

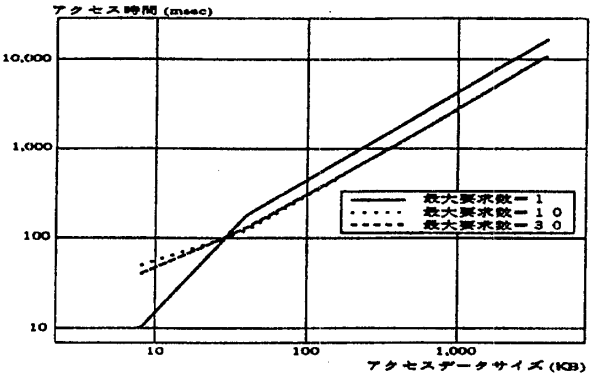


図 4: ブロックサイズ8KB の時の結果

一方、FS では、1つの要求に関する物理I/O を非同期で発行しては、次の要求の処理に切替えるなどして複数の要求を多重処理する。すなわち、複数の要求の多重処理によって、物理I/O 時間と CPU 消費時間／通信時間が重なり、AP からの呼出しの処理時間が短縮される。

FAL から FS に一度に送信する要求の数を最大要求数と呼び、FAL で複数の要求に区切る際のデータサイズをブロックサイズと呼ぶこととする。

## 5 パイプライン転送の評価

パイプライン転送によるアクセス時間の短縮効果を見るため、2台の NEC 製 EWS4800 を Ethernet で接続した。ディスクを接続した EWS 上で FS を動作させ、もう片方の EWS 上で FAL をリンクした AP を動かし、このディスクにリードを行なう。AP から FAL のリード呼出1回にかかる時間 (アクセス時間) を測定した。

ブロックサイズ4KB, 8KB について、FAL のリード呼出1回のサイズを1ブロックから4MB まで変化させた。更に、FAL から FS に対する最大要求数を変化させて測定した結果、図3, 4に示すグラフとなった。

## 6 考察

図3, 4とも、リード呼出サイズが1ブロックの時のバラツキが大きい。これは、ディスクのシーク＋回転待ちの時間に確率的要素があるためである。

40KB 以上では、図3を見ると最大要求数による差がほとんどないが、図4では、最大要求数が1の時だけ他よりもアクセス時間が55% 程度長い。最大要求数1では、FS に対して1つずつの要求しか送信せずパイプライン転送とならない点が他と異なる。つまり、最大要求数1とのアクセス時間の差がパイプライン転送の効果と見られる。図3の場合は、物理I/O や接続網を介したデータ転送の時間が短く、重なり時間が短いため効果が小さいことになる。転送レートは、ブロックサイズ4KB で 210KB/sec, 8KB で 360KB/sec に収束する。

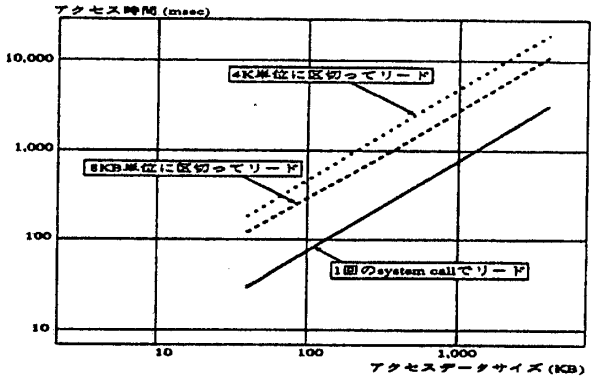


図 5: ローカルディスクにアクセスした時の結果

また、ローカルディスクに対するアクセスと比較するため、4KB 及び8KB の raw のリードシステムコールをループで呼ぶ場合と、1回のシステムコールを呼ぶ場合について図5に示す。例えば、図4と図5の8KB 単位のグラフを比較すると、最大要求数10, 30 ではほぼ同等のアクセス時間となっている。これは、パイプライン転送により通信の時間が隠れていることを示す。

ブロックサイズを大きくするとアクセス時間が短くなるが、これは、1回の物理I/O のサイズが大きい方が、ディスクのシーク＋回転待ちの回数が減るためである。ローカルディスクに対し1回のシステムコールでリードした場合、1回の物理I/O のサイズが最大となり、アクセス時間が最短となる。今回の評価では、遠隔ディスクアクセス機構のブロックサイズは最大8KB であったが、更に大きなブロックサイズを実現できれば、更に短いアクセス時間を達成できる。

## 7 まとめ

本稿では、WS クラスタにおいて AP から遠隔ディスクにアクセスするための機構について、方式と実装を説明した。更に、遠隔ディスクへのアクセス時間を短縮するパイプライン転送機構について簡単な実験を行い、その効果を確認した。

## 参考文献

[1] A. L. Cheung and A. P. Reeves, "High Performance Computing on a Cluster of Workstations", *Proc. of 1st Int. Symp. on High-performance Distributed Computing*, Sep. 1992.