

並列ファイルシステム PF3 の概要

5 Q-6

大和 純一 大谷 寛之 相場 雄一 青木 久幸

NEC C&C メディア研究所

1 はじめに

ビジネス計算分野で用いられるデータベースは、年々情報蓄積量が多くなる傾向にある。さらに、近年ではデータベースに蓄積された膨大なデータは様々な意思決定に利用されている。このような使い方では、検索時間の短縮のために、記憶容量の増大に見合ったアクセス性能の向上が必要である。また、VoD 等のマルチメディア応用では、動画を多数蓄積するため、サーバでは数百GB～数TBの記憶容量が必要である。更に複数の端末に動画を供給するためにサーバでは二次記憶に関して非常に高いスループットが要求される。一方で、MPP や WS クラスタを想定した大規模な数値計算向けのファイルシステムの研究も行われている[1]。

このような状況の中、我々も、WSクラスタ上で動作し、二次記憶容量の拡大・スループットの向上を実現する並列ファイルシステムの研究・開発を行っている[3][4]。

本稿ではディスク共有型クラスタシステムを想定した並列ファイルシステム PF3 の概要を述べる。

2 並列ファイルシステム

並列ファイルシステムは、大容量ファイル・高スループットなファイルアクセス・スケーラブルな規模拡大を目指して研究を行っているシステムである。

並列ファイルシステムでは、多数のディスクにより論理的なファイルシステムを構成する。このファイルシステムに対して複数のアプリケーションが並列にアクセスす

ることで、システム全体として高いスループットを提供する。また、ディスク数を増加させることにより、スループットを容易に向上させることが可能である。

個々のファイルに関しても図 1のようにデータブロック単位でデータをディスクに分散格納することで、ファイル内での多重アクセスを可能としている。

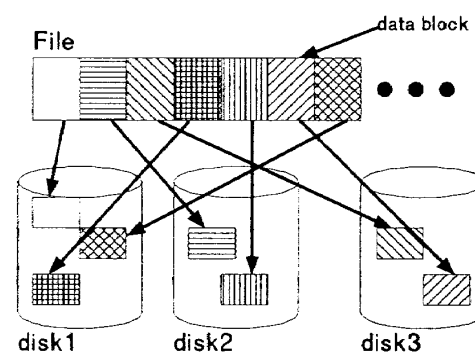


図 1: 分散格納

複数のクライアントから多数のディスク装置に並列アクセスを行い、スケーラブルな規模拡大を可能とするために、ファイル等の管理に複数の管理主体を用いる。これら複数の管理主体を効率よく管理するために、システム全体の一元管理を行う。また、各管理主体間の状態（起動・終了 mount 等）に関しても一貫性制御を行う[4]。

3 従来の並列ファイルシステム

MPP および WS クラスタにおけるファイルシステムの研究として Vesta[1]・Tiger Shark[2]・MFS[3]がある。これらは、ディスク非共有なクラスタシステムという構築が容易なプラットフォームを対象としている。

従来の並列ファイルシステムの例として、我々が開発した MFS の構成を図 2に示す。MFS は、システム全体を管理する SystemManager、ディスク管理ノードで動作し、ファイルの管理・データの転送を行う FileServer、ア

“An Overview of Parallel File System PF3”,

Jun-ichi YAMATO, Hiroyuki OHTANI, Yuichi AIBA,
Hisayuki AOKI

C&C Media Research Laboratories, NEC Corporation.

4-1-1 Miyazaki, Miyamae, Kawasaki 216-8555, Japan

アプリケーションプログラムにリンクされ、モジュール間の通信を行う FileAccessLibrary(FAL)から構成される。

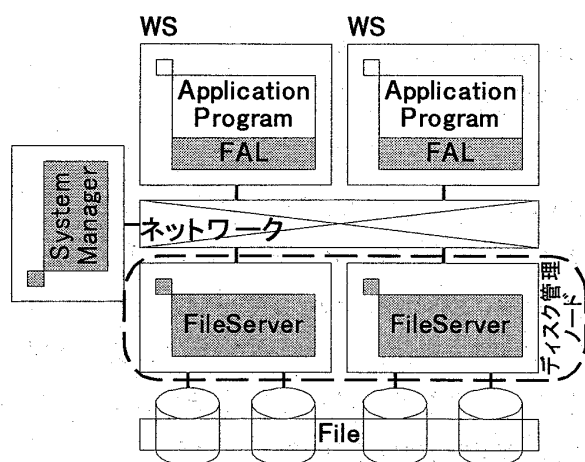


図 2: 並列ファイルシステム MFS の構成

これらの並列ファイルシステムでは、ディスク管理ノードがクライアント・ディスク間のデータ転送を仲介するため、レスポンスタイムが長くなる。

また、一つのノードに多数のディスクを接続するとノード内での転送がネックとなる。従って、大規模なシステムを構築するためには、ディスクを接続するノードが多数必要となり、ディスク管理ノードがシステム全体のコストを引き上げる要因となる。

4 ディスク共有型並列ファイルシステム

前述のディスク非共有型クラスタシステムでの問題を解決する構成として、FC(Fibre Channel)等のディスク共有網を用いて全てのノードからディスクを直接アクセス可能なディスク共有型のクラスタシステムが考えられる。我々は、ディスク共有クラスタシステムに対応した並列ファイルシステムである PF3 の研究・開発を開始した。

PF3 の構成を図 3 に、各モジュールの役割を以下に示す。

- SystemController: システム全体の管理
- FileController: ファイルの管理(複数動作)
- FAL: アプリケーションプログラムにリンクし、モジュール間の通信・ディスク I/O を行う

PF3 では、ディスク共有網によりアプリケーションが直接ディスクにアクセスを行うことで、ノード間データ転送によるレスポンスタイムの悪化を解決する。さらに、ディ

スク管理ノードを必要としないため大規模システムでのディスク管理ノードによるコスト問題も発生しない。

なお、PF3 では並列ファイルシステムとしてディスク共有網を有効に用いるため、ファイル管理方式に改良を加えている[5]。

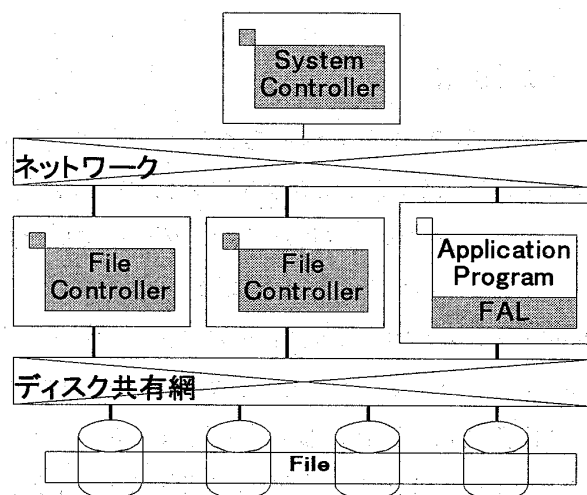


図 3: PF3 の構成

5 おわりに

本稿では、従来型の並列ファイルシステムの問題点を述べ、これらの問題を解決するディスク共有型の並列ファイルシステムである PF3 の構成方法を述べた。

PF3 は現在、実機上に実装中であり、実装完了後、評価を行う予定である。

参考文献

- [1] Peter Corbett, et al., "The Vesta Parallel File System", ACM Transactions on Computer Systems, Vol. 14, No.3, pp. 225-264, 1996
- [2] Roger Haskin, et al., "The Tiger Shark File System", Proceedings of COMPCON, 1996.
- [3] 青木, 他, "並列ファイルシステムMFS", 情処研報, Vol.96, No79, pp31-36, 1996
- [4] 相場, 他, "並列ファイルシステムにおける運用管理機能", コンピュータシステムシンポジウム論文集, Vol. 97, No.8, pp125-132, 1997
- [5] 大谷, 他, "並列ファイルシステム PF3 におけるファイル管理方式", 第57回情処全大 5Q-7, 1998