並列ファイルシステムにおける並列検査機能の実装*

1F－4

大谷 寛之, 相場 雄一, 大和 純一, 青木 久幸†
NEC C&C 研究所‡
e-mail: {ohtani, aiba, yamato, aoki}@csl.cl.nec.co.jp

1 はじめに

近年、データベースの大規模化、マルチメディアサーバの発展に伴い、大容量、高スループットアクセスが可能なファイルシステムの需要が高まっている。そこで我々は、多数の記憶装置が接続されたワークステーションクラスタシステム上で動作する並列ファイルシステム (MFS)[1] を開発した。しかし、クラスタシステムでは、個々の装置の障害によって全体がその影響を受けるために、フォールトトレラントや、障害リカバリーが重要な課題である。

本稿ではMFS における障害リカバリー後の整合性検査機能について記述する。一般にファイルシステムにおいて障害が発生した後は、ファイルシステムの整合性検査が必須である。これは、障害でシステムダウンしたことにより、メモリ中に存在していたファイルキャッシュの内容と、記憶装置上に残された内容に矛盾が生じている可能性があるためである。この処理は、すべてのファイルの管理情報を調査するため、比較的コストを要する処理である。特に、多数の記憶装置から構成される大規模なファイルシステムにおいては、膨大なコストを要する。そこで、検査処理の高速化を目的とし、並列的に検査処理を実行する並列検査機能の実装を行なった。

2 従来のファイルシステムにおける検査機能

ファイルシステムの整合性検査は、ファイルのデータ管理構造の依存関係に矛盾が存在しないことを確認する検査である。例えば、UNIX ファイルシステムにおける整合性検査では、主に以下に挙げられる項目の検査を実施する。

- ファイルシステム管理構造 (スーパーブロック)
- 参照ブロック管理構造
- フリーブロック管理構造
- 名前空間 (ディレクトリ) 管理構造

また、検査の結果、矛盾が検出されると以下の修復処理を実施する。

- ファイルシステム管理情報の修正
- 不正な状態のファイル消去
- ブロック管理構造から浮いたブロックの救済
- 名前空間から浮いたファイルの救済

*Implementation of Multispindle File System Parallel Check

†Hiroyuki Ohtani, Yuichi Aiba, Junichi Yamato, Hisayuki Aoki

‡C&C Research Laboratories, NEC Corporation

3 並列ファイルシステム MFS の概要

3.1 MFS の構成と機能

MFS は、図1に示されるように複数の計算機ノードが、高速ネットワークで接続されたクラスタシステム上で動作するファイルシステムである。各ノード上の記憶装置に分散格納されたデータを一つの論理的なファイルとして扱うノード間ストライピング格納が可能であり、大容量ファイル、高スループットアクセスを提供する。

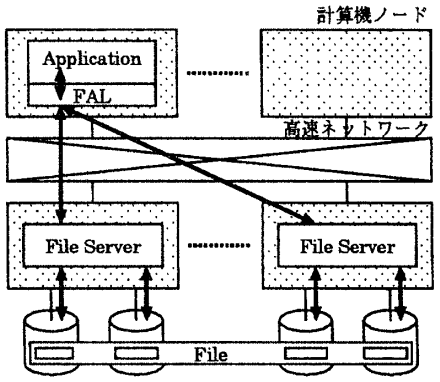

図 1: MFS の構成

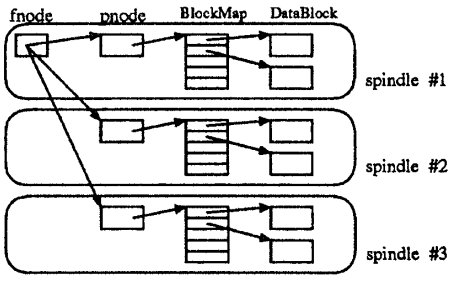

図 2: MFS のデータ管理構造

3.2 MFS のデータ管理構造

一つのファイルのデータは、ブロック毎に分割され、各記憶装置 (スピンドル) にストライピング格納される。特に、分割されたブロックの中で、同一のスピンドルに格納されたブロックの集合をそのファイルの部分ファイルと呼ぶ。MFS のデータ管理構造は、図2のように、ファイル

の属性情報を管理するfnode、部分ファイルを管理するpnode、ブロックのアドレス変換情報を管理するBlockMapによって構成される。

4 並列ファイルシステムの検査における課題

4.1 ノード間に分散した管理データの整合性

一般に、ファイルを各ノードの記憶装置にストライピング格納するようなファイルシステムでは、各記憶装置内のファイル管理データの依存関係の他に、ノード間に分散した管理データの依存関係が発生する。これは、各記憶装置にストライピング格納されたデータを論理的なファイルとして扱うためのファイル管理データが存在するためである。

つまり、このようなファイルシステムにおける整合性検査機能には、従来のファイルシステムのような各記憶装置毎の整合性検査に加え、ノード間に分散した管理データの整合性検査機能が必要となる。

4.2 検査処理の高速化

多数の記憶装置から構成される大規模なファイルシステムでは、ファイル管理データが増大し、それらの検査処理のコストも増大する。しかし、従来のファイルシステムにおける検査処理のように、個々の記憶装置単位で逐次的に実行するのでは、膨大なコストを要することになるため、高速化が必須である。ところが、ファイル管理データが各記憶装置に分散しているファイルシステムでは、他の記憶装置の状態に影響を受けず、各記憶装置内部の情報を基に検査が可能な処理が存在する。つまり、各検査処理において、並列化可能な部分と不可能な部分を分離し、並列化可能な検査処理を各ノードにおいてそれぞれ実行することにより高速化が期待できる。

これらの機能を実現するためには、各ノード上において各記憶装置の検査を単独に実行する機能、および各検査処理のフェーズを管理する機能が必要となる。

5 MFSにおけるファイルシステム検査機能

MFSにおけるファイルシステム検査機能mfsckの構成を図3に示す。mfsckは、各検査処理のフェーズ管理およびスピンドルにまたがるデータの整合性検査を担当するMfsck Managerと、Mfsck Managerからの要求により各スピンドルで独立な検査を実行するSpck Serverによって構成される。検査処理は、図4の通信シーケンスに示されるように各フェーズに従って実行される。また、1および3のフェーズは、各スピンドルを担当するSpck Serverにおいて並列に処理される。

1. スーパーブロック検査フェーズ
   各スピンドルのスーパーブロックの整合性を検査
2. スーパーブロック全体検査フェーズ
   各スピンドルのスーパーブロック間の整合性を検査
3. スピンドル検査フェーズ
   各スピンドル内部の管理データの整合性を検査
4. ファイルシステム全体検査フェーズ
   スピンドルにまたがる管理データの整合性を検査
5. スピンドル修復フェーズ
   修復の必要性があるスピンドルを修復する

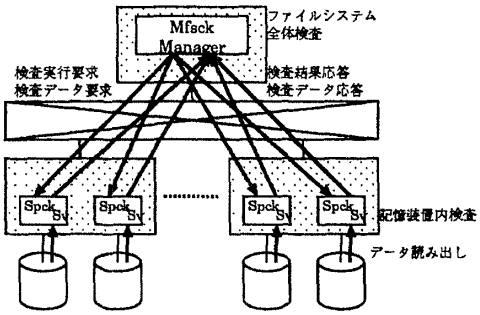

図3: mfsckの構成

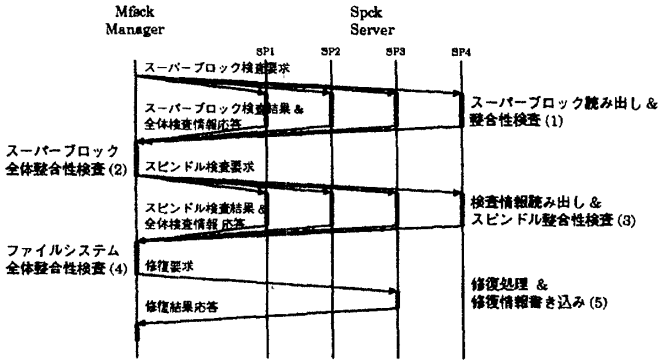

図4: mfsckの検査処理シーケンス

6 おわりに

本稿では、ファイルのデータを分割し、各ノード上の記憶装置にストライピング格納するファイルシステムにおいて、障害リカバリー時に必要となる整合性検査機能の課題を挙げ、MFSにおける実装方式を記述した。

今後は、今回実装した整合性検査機能の性能評価、また、障害発生時のMFSにおけるフォールトトレラント問題に関する検討を行なう予定である。

参考文献

[1] 青木 久幸, 大和 純一, 大谷 寛之, 相場 雄一, “並列ファイルシステム MFS”, SWoPP'96, pp.31-36, 1996

[2] “bit別冊 UNIX カーネルの設計”, 共立出版, 1990