

リンク構造ファイルシステムにおける ガベージコレクションの設計と実装について

06T217 大橋 洸一（最所研究室）

ファイルを可変長ブロックの集合として扱う「リンク構造ファイルシステム」に、不要なデータ領域の回収を行うガベージコレクション機能を追加するため、必要な機能の設計と開発を行う。

1 はじめに

当研究室では、リンク構造ファイルシステムの開発を行っている [1]。このシステムはファイルを従来のデータ構造のないバイト集合として扱うファイルシステムとは異なり、ファイルを可変長ブロックの集合として扱うファイルシステムである。ファイル操作をブロック同士のリンクの変化で表現することにより、従来のファイルシステムでは実現できなかった指定位置へのデータ挿入や削除を行うことができる。

現在このファイルシステムでは、ファイルの履歴管理を行うために、削除操作では実際にブロックを削除せず、またファイルを操作するたびに履歴を管理するための LinkBlock と呼ぶ構造体を作成する [2]。この履歴管理により、誤った操作により削除された、または変更されたファイルの復元を可能にしているが、一度作成されたブロックがディスク上から削除されなくなることや、不必要な履歴でさえも確保してしまうことから、ディスク領域の肥大化が問題となる。また昨年度の実装では、リンクブロックの作成操作が、ファイルの先頭、および末尾のブロックの挿入、削除操作には対応しておらず、これらの操作が行われたファイルに対しては正しく履歴を管理できないという問題があった。

そこで本研究では、これらの構造体が利用している領域のうち、古くなり復元する必要の無くなった構造体が利用している領域を解放し、再び空き領域として利用可能にするガベージコレクション機能の設計と実装を行った。また、履歴管理機能を全てのファイル操作に対して対応できるように、必要な機能の設計と実装を行った

2 ガベージコレクションの概要

一般的なガベージコレクションの手法にはマーク・アンド・スイープや、コピー GC 等の技術が存在する [3]。しかし、これらの技術は本来メモリ上でのガベージコレクションを前提とした技術であり、リンク構造ファイルシステムに導入するには問題が多い。そのため本研究では、リンク構造ファイルシステムで効率的に動作する“ マネージメントブロック・ガベージコレクション”を提案する。

マネージメントブロック・ガベージコレクションの概要を図 1 に示す。不要になった LinkBlock に、空き領域を管理するためのブロックからリンクを繋ぎ、それ以外のブロック、および他の LinkBlock から繋がる全てのリンクを削除することにより、領域の回収を行う。また、削除操作により作成された LinkBlock の場合、その LinkBlock が管理していたブロックも別の空き領域を管理するためのブロックから新たにリンクを繋ぐことにより、復元する必要の無くなった削除データも同時に回収する。この空き領域を管理するためのブロックを ManagementBlock と呼称する。

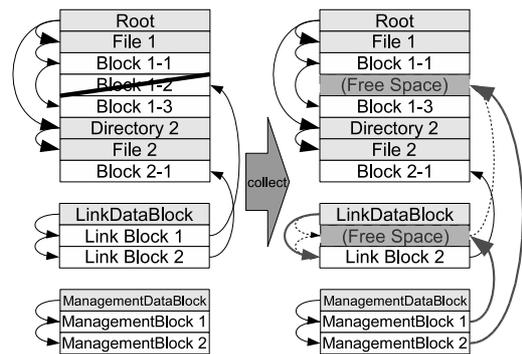


図 1: 提案するガベージコレクションの概要

3 追加構造体

本研究では、リンク構造ファイルシステムにおけるガベージコレクション機能の実装のため、現在のリンク構造ファイルシステムの機能に加え、以下の 2 つの構造体を追加し、実装を行った。

• ManagementBlock 構造体

ManagementBlock 構造体は、ファイルシステムにおいて空き領域の管理を行うための構造体である。ManagementBlock に繋がれた構造体は完全に削除された構造体として扱われ、以降はファイルシステムが自由に扱うことのできる空き領域として利用される。

• ManagementDataBlock 構造体

ManagementDataBlock 構造体は、

ManagementBlock の管理を行うための情報を持つ構造体である。

また、ファイルの先頭、および末尾のブロックに対しても履歴を作成できるようにするため、あらかじめファイルの先頭と末尾に1つずつ作成する空のブロックを作成する。以降のファイル操作はこれらのブロックにより、ファイルの先頭、および末尾のブロックに対するファイル操作を他のブロックと全く同じ方法で操作することが可能になる。この空のブロックを DummyBlock と呼称する。

これらの構造を実装したファイルシステム全体の構造を図2に示す。

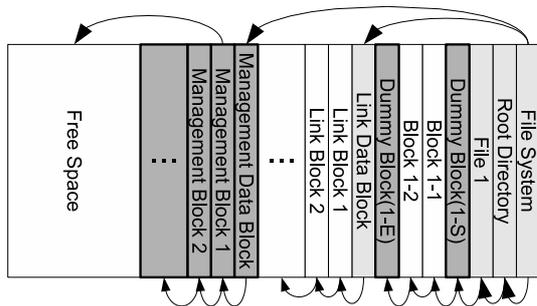


図2: ガベージコレクション機能を実装したリンク構造ファイルシステムの構造図

4 追加機能

本研究では、リンク構造ファイルシステムにおけるガベージコレクション機能の実装、および DummyBlock の実装のため、現在のリンク構造ファイルシステムの機能に加え、以下の4つの操作を追加した。

- **空き領域の回収を行う操作**
古くなり復元する必要の無くなった構造体を特定し回収する。LinkBlock が利用していた領域を回収する場合は、その時点で存在する一番古い LinkBlock を回収し、その他の構造体の利用していた領域を回収する場合は、その構造体を削除した操作の履歴にあたる LinkBlock を回収した後、その LinkBlock からリンクを辿ることで回収対象の構造体を特定し、回収を行う。操作の手順は図1で示している手順と同様である。
- **回収した領域を再使用する操作**
操作の概要を図3に示す。空き領域の回収操作により回収された領域の中で一番大きな領域を、ディスク書き込みに使用する領域に割り当てる処理を行う。ファイルシステムが書き込み操作に使用している領域が使い切られると、回収した空き領域から最も大きな領域を走査し、その領域を

ファイルシステムが書き込み操作で使用する領域として設定する。

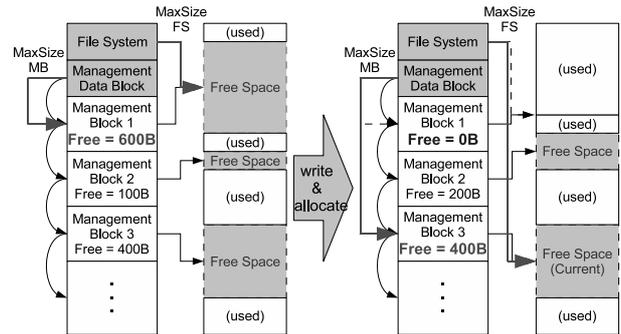


図3: 回収した領域を再使用する操作の概要

- **ブロック構造体の削除を禁止する保護フラグを変更する操作**
特定のブロックを削除操作により削除されないようにするための操作である。対象のブロックをシステムによる削除禁止状態、ユーザーの操作による削除禁止状態、削除可能な状態のいずれかの状態に変更する。
- **LinkBlock の回収を禁止する保護フラグを変更する操作**
特定の LinkBlock を回収されないようにするための操作である。対象の LinkBlock をシステムによる回収禁止状態、ユーザーの操作による回収禁止状態、回収可能な状態のいずれかの状態に変更する。

5 まとめ

以上、リンク構造ファイルシステムのガベージコレクション機能の実装のために追加した機能、構造について述べた。現在のところ、設計および実装を行い、動作確認を完了している。しかし、性能評価には至っていない。今後はこれらの性能評価を行うとともに、File 構造体、および Directory 構造体に対するガベージコレクション機能の実装や、空きブロックの断片化を防ぐためのデフラグ機能の実装を行う必要がある。

参考文献

- [1] 津紀孝, 最所圭三, “可変長単位ブロックで操作を行うファイルシステムの開発” 香川大学研究科 修士論文, 2007
- [2] 森俊文, 最所圭三, “リンク構造ファイルシステムへの履歴機能の追加” 香川大学工学部卒業論文, 2008
- [3] Fenichel, R.R, ”A LISP garbage-collector for virtual-memory computer systems”, (1969), Communications of the ACM