

# バージョン管理機能を持つファイルシステムのアプリケーションレベルでの実現に関する研究

99T259 藤本 哲郎 (最所研究室)

## あらまし

バージョン管理機能を持つファイルシステムをアプリケーションレベルで実現する研究を行った。バージョン管理機能を実現するライブラリの設計や実装を行った。

## 1 序論

近年のパソコンの急速的な普及により、これらを利用した作業が一般的となり、利用されるプログラム及びデータの種類や個数は飛躍的に増加している。これらはユーザによって常に変更を繰り返されているが、全ての変更が改善とはならなかったり、また人為的ミス等でデータが破損した場合、過去の状態にまで戻さなければならない。そのためには、更新される可能性がある全てのファイルの変更の状態を把握する必要があり、ファイル管理等の人的な負担は増大する。このため、各ファイルの管理コストを軽減するバージョン管理が必要である。

そこで、本研究ではユーザのファイル管理の負担を軽減するファイルシステムの研究を行った。本研究の目的は提案しているシステムの有効性を示すとともに、その問題点を洗い出すことにある。昨年度の研究 [1] ではファイルシステムとして実装を行ったが、カーネル内のデバックなどが困難であったため、実装に至らず、提案システムの有効性を確かめることができなかった。このため、本論文ではバージョン管理システムの有効性を確かめることを第一の目標とし、実装が容易なアプリケーションレベルでの疑似ファイルシステムとして実装を行うことにした。

## 2 システムの設計

既存のファイルシステム上のファイルをブロックデバイスとして見なし、その上にバージョン管理機能を持つ疑似ファイルシステムをアプリケーションレベルで構築する。バージョン管理機能として以下のような特徴を持つ。

- バージョンファイルの保持方法として、ファイルシステム化が容易なように、UNIX の i-node をモデルとする。  
バージョンを木構造で管理し、バージョンを管理するためのデータ構造を持つ。
- 各バージョン間の変更前、変更後の親子関係を保持する。

- 過去のバージョンの一部を削除し、過去のバージョン間の親子関係を付け直すことができる。  
例えば、親 (P) と子 (C) を持つバージョンを削除する場合、P から C へのリンクを付け直す。
- 自動的にバージョンを作成する。
- バージョンの作成方法を選択できる。  
バージョンの作成方法として、ファイルのクローズ等内容の変更時、周期的、あるいは手動もしくは自動でのバージョン作成等を提供する。
- 同じバージョンに対して、複数ユーザが同時に使用できる。

このバージョン管理の機能をライブラリとして提供する。バージョン管理情報の構成及びバージョン関係を図 1 に示す。バージョン管理情報には、バージョン管理を行うファイル自身の情報、バージョン間の関係及び各バージョンのバージョンデータの管理情報の位置等の情報が含まれる。またバージョンデータの管理情報には、バージョンの実データを管理する情報を保持する。これらの情報は、ブロックデバイスと見なしたファイル上に図 2 に示すように配置される。

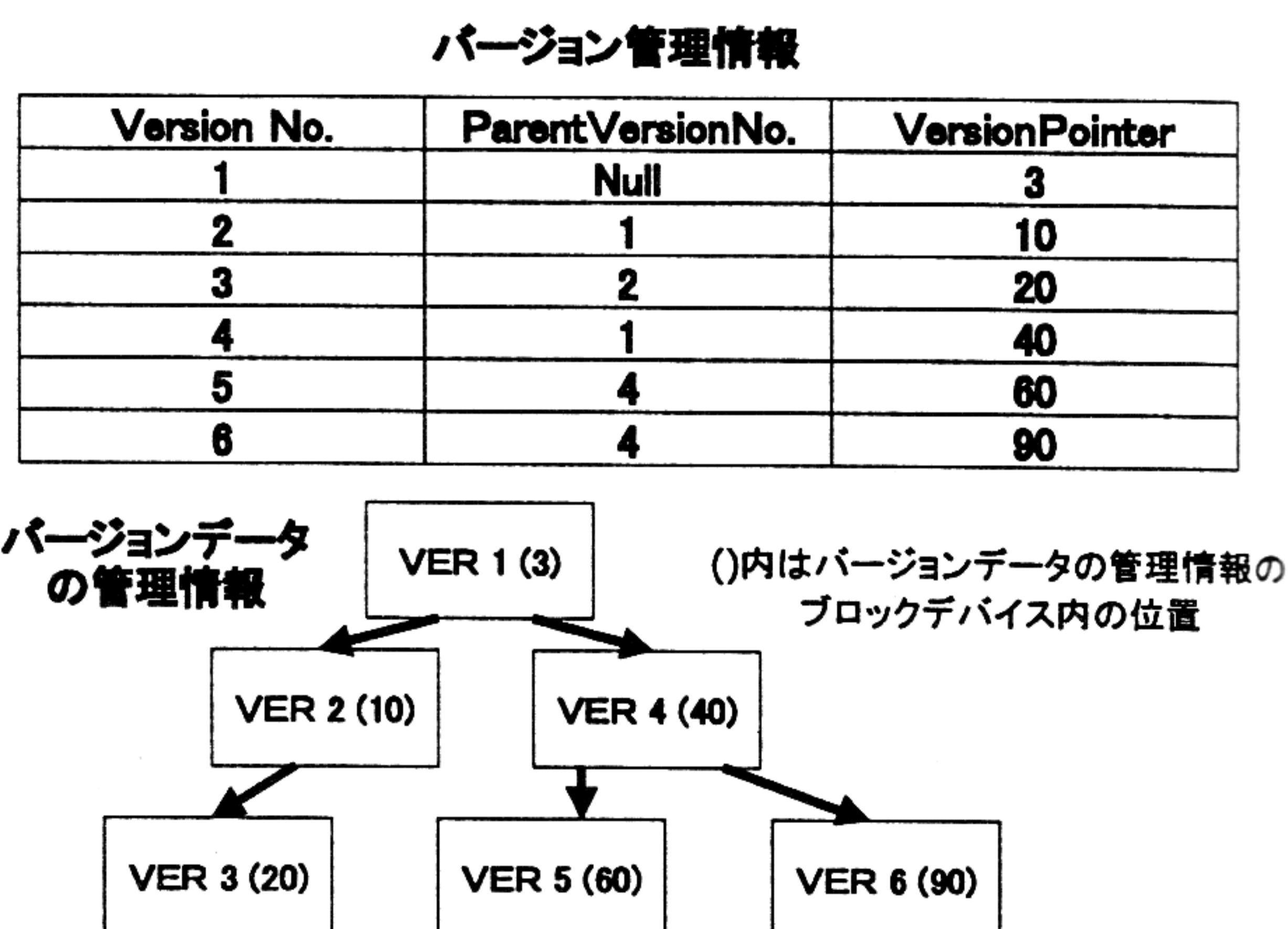


図 1: バージョン管理情報とバージョン関係

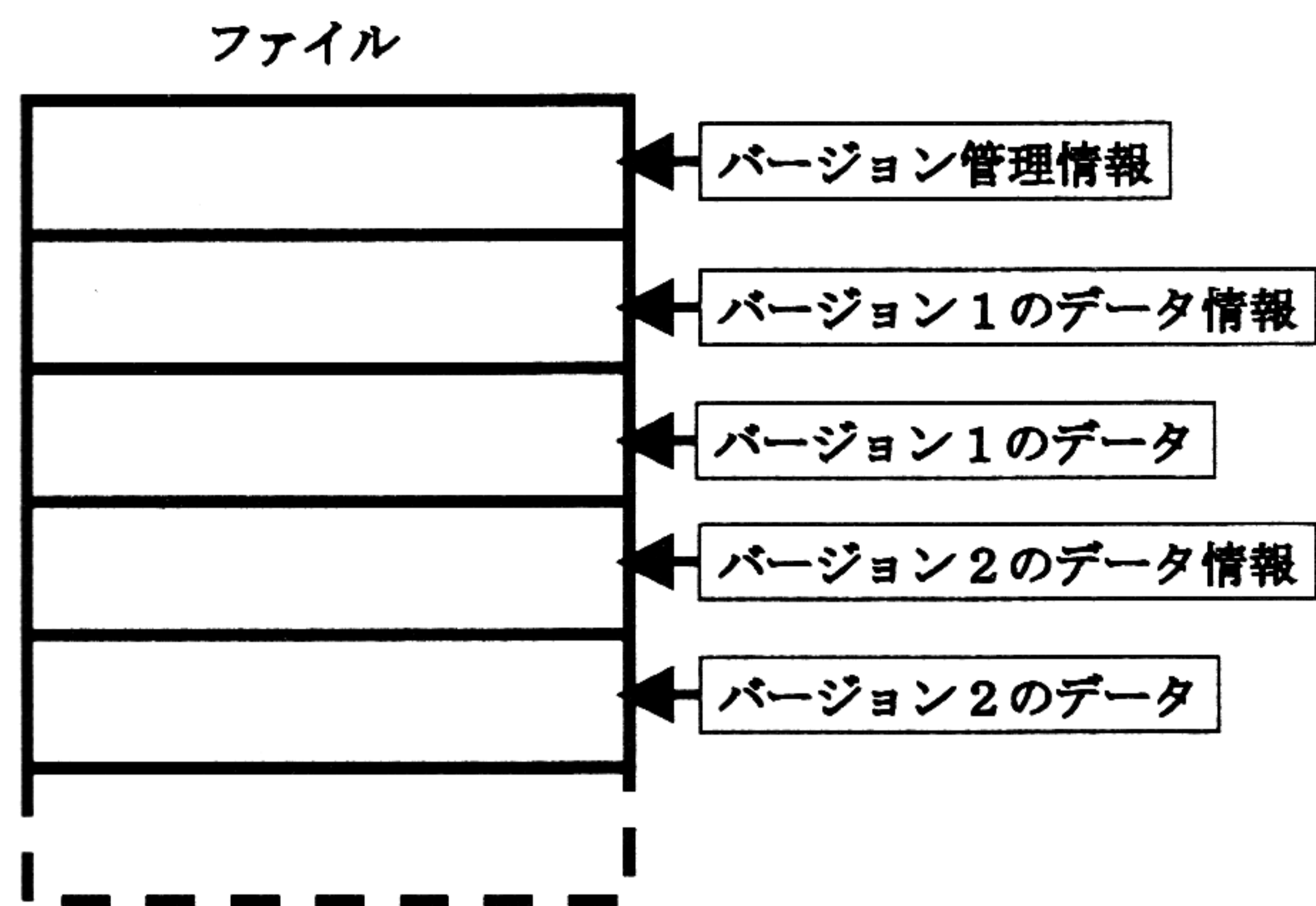


図 2: バージョンファイルの構造

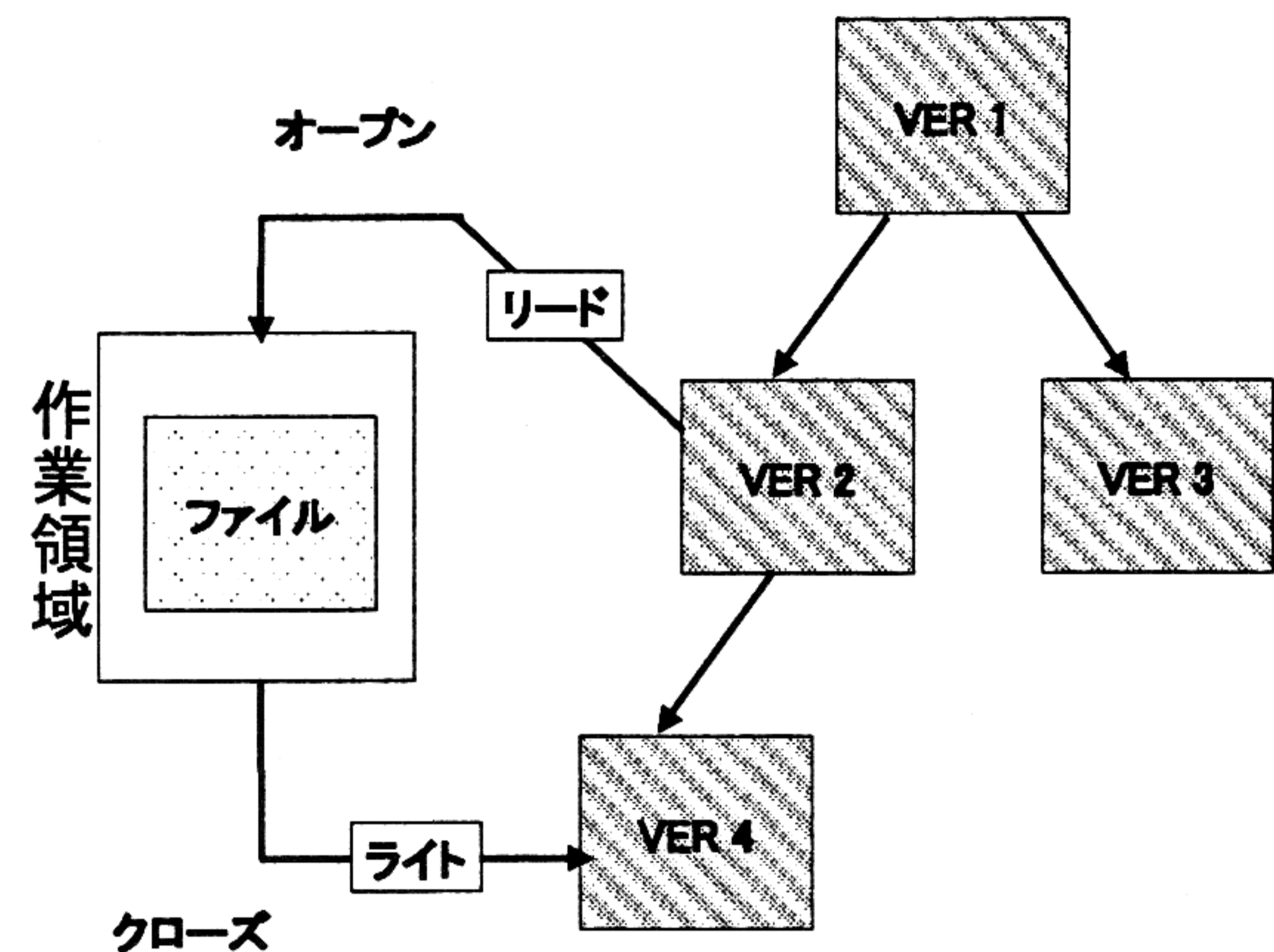


図 3: ライブラリの動作概要

### 3 ライブラリの実装

バージョンの管理を行うために必要な以下の機能を含むライブラリを用意する。これらの機能を用いることにより、バージョン管理機能を持つ擬似的なファイルシステムを実現する。

- ファイルオープン
- ファイルクローズ
- ファイルリード
- ファイルライト
- バージョンファイルの作成
- バージョンの削除
- バージョンデータのコピー
- ポリシーの設定

バージョンファイルのデータを利用する場合、図 3 に示すようにファイルオープンを用いて、バージョンファイル内の目的のバージョンデータを、ファイルとして作業領域にコピーしオープンする。実際のコピーはファイルリードを用いて行う。

通常ファイル操作は作業領域内で行われ、作業終了後にファイルクローズを用いて新たなバージョンデータとしてバージョンファイルに書き込む。実際の書き込みはファイルライトを用いて行う。

また、既存のファイルをバージョン管理する場合や、バージョン管理が行われているバージョンファイルの一部のバージョンデータを新たにバージョンファイルにコピーする場合、図 4 に示すようにバージョンファイルの作成及び、バージョンデータのコピーを利用する。バージョン管理機能を持つファイル操作の実装は C 言語で行っている。

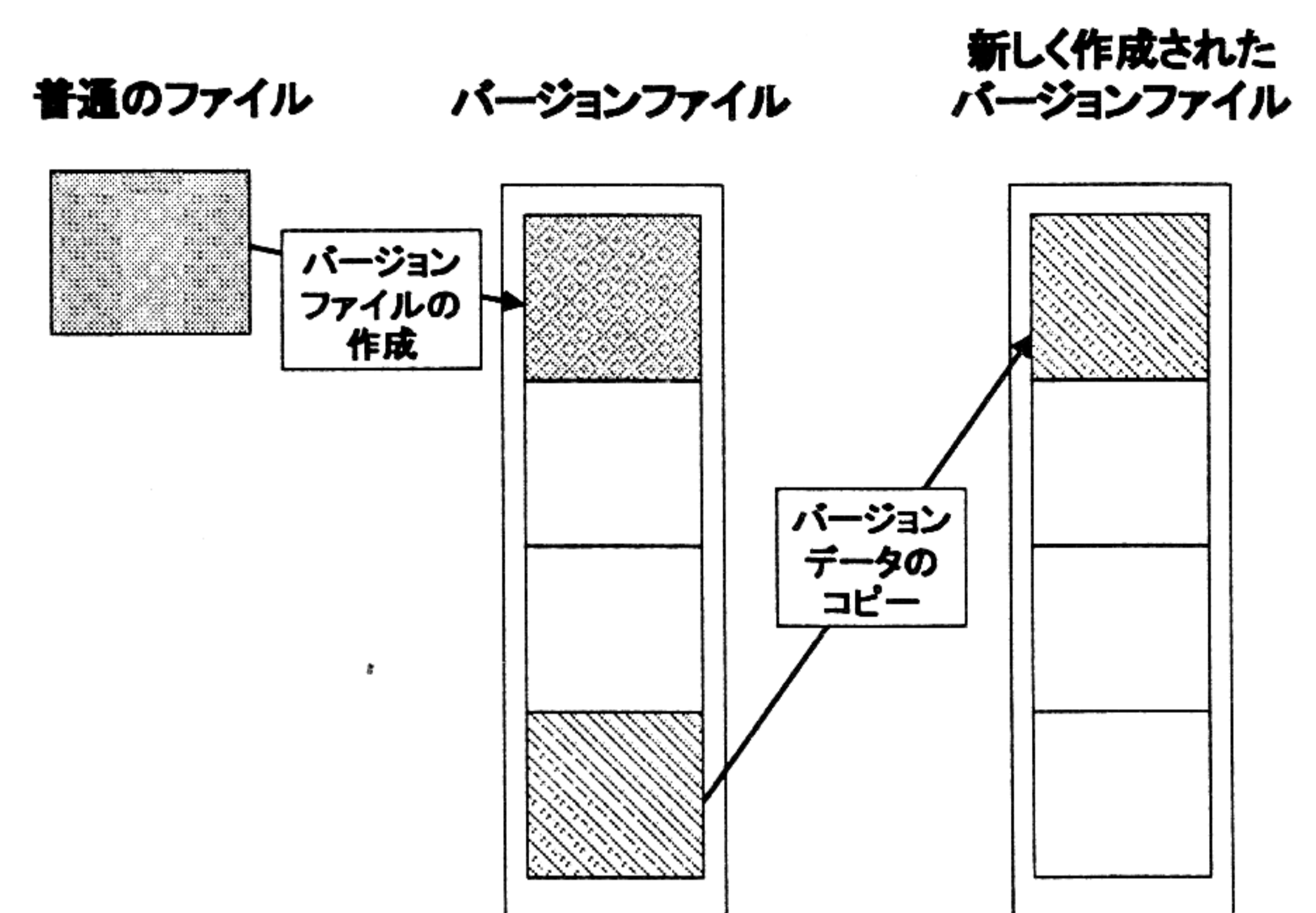


図 4: ライブラリの動作概要

### 4 まとめ

本論文では、バージョン管理機能を持つファイルシステムをアプリケーションレベルで実現する疑似ファイルシステムの研究について述べた。現在はライブラリの設計を行った後、バージョンデータの管理情報上にデータブロックを配置した上での、ブロックデバイス上の位置の保持の処理を実装している。今後は、残りの機能の実装を行い、システムの有効性の確認を行う必要がある。また、データブロックの有効利用のためのバージョンデータ間の差分処理とブロックデバイスの再利用の処理について、再設計を行う必要がある。

### 参考文献

- [1] 平井貴浩「バージョン管理機能を持つファイルシステムの設計」香川大学工学卒業論文 2002.