オペレーティング・システム 36－1
（１９８７．９．１１）

追記型光ディスクを用いた
世代管理ファイルシステムの開発

横関　隆　、　中川正樹　、　高橋延匡
東京農工大学　数理情報工学科

ワークステーションなど、比較的小規模なシステムにおいて追記型光ディスクを利用する場合、光ディスクのファイルシステムをどのようにするかが大きな問題となる。

本報告では、追記型光ディスクを見かけ上書換え可能なディスクとして扱う方法を提案する。この方法を用いることにより、既存のOSで光ディスクをそのまま利用できることになる。

また光ディスクの利用によって、安いコストで実現可能となる世代管理機能についても触れる。

Development of a multi-generation file system on write-once optical disks

Takashi Yokozeki , Masaki Nakagawa , Nobumasa Takahashi
Tokyo University of Agriculture and Technology
2-24-16, Naka-machi, Koganei, Tokyo, 184, Japan

One of the hot issues in the computer science is how to utilize write-once optical disks, huge but unrewritable and rather slow devices, in the computer architecture. We propose a virtual optical disk method which makes an optical disk look like a rewritable device. By using this method, we can use optical disks on current operating systems without any reconstruction of their file systems. Then we refer to multi-generation file management on the virtual optical disk method.

1. はじめに

磁気ディスクなどの2次記憶装置の容量は年を追って増加してきた。5.25インチフロッピーディスクは1Mバイトのものが主流になりつつあり、1枚で10Mバイトもの容量を持つものも発表されている。また3.5インチフロッピーディスクにおいても1Mバイトのものが使われ始めており、記録する情報の高密度化が進んでいる。

しかしこれらは記録媒体として磁性体を用いており、読み書き可能ということは、取扱い上信頼性の面に一種の不安を感じる。またプログラムミスや操作ミスなど人間の不注意による、情報の破壊の可能性を考えると、はなはだこころもとない。

オーディオ製品においてはレコードがCD(コンパクトディスク)に代わりつつあるが、計算機の世界において光ディスクは、磁気ディスクに次ぐ第3の記憶装置として普及しようとしている。光ディスクは傷に強く、記録密度も磁性体のそれよりも非常に高くすることができる。また磁気による情報の破壊の心配もないため、データ保存用の記憶装置として大きな期待が持てる。

現在、情報の価値は以前に比べ非常に高くなっており、また今後ますます高くなる傾向にある。例えば設計仕様書やプログラムなどはそれ自体、知的財産として高い価値を持つものであるが、近年になってそれらが作成される過程も、重要な情報として認識されるようになっている。

ここにおいて情報の保存が重要なものとなり、保存の技術に高い信頼性が要求されるようになった。

追記型光ディスクは、大容量、高信頼性、低価格と、この要求を満たすための最適な条件を持ち合わせている。現在、光磁気ディスクなどの書換え可能型光ディスクの研究が行われているが、それらが市場に出現したとしても、情報保存の技術における追記型光ディスクの立場は変わらないであろう。

本研究では情報コストの増大、特に知的所有物としての設計、仕様、プログラムの重要性、教育、保守等のための文書の必要性、また社会背景として、情報生成の過程を法的に立証できるようなシステムの意義を考え、これらのニーズの認識のもとに、消去されない情報のファイルシステムの開発を行う。

2. 研究目的

本研究の目的は次の3点である。

(1) 既存のOSに光ディスクファイルシステムを実現する

既存のOS上に光ディスクファイルシステムを実現できれば、実現即実用可能という状態になる。またそのことにより光ディスク利用効率(光ディスク全体におけるユーザーデータの占める割合)、世代管理の有用性などの評価も行いやすい環境ができる。

この一石二鳥を狙い、まず既存のOS上での実現を目指す。

(2) 光ディスク及び世代管理対応のファイルシステムに必要な機能を模索する

たとえ上記のように既存のOS上に光ディスクファイルシステムが実現できたと

しても、それが光ディスクにとって最良のものであるとは限らない。
今回は新しいOS上に光ディスクファイルシステムを実現する際に、どのような機能が必要であり、どのような形を取るのが好ましいかを考える。

（３）世代管理の利用法を考える
世代管理機能は、プログラムやその仕様書を管理する際に有効であることは、ほぼまちがいない。しかしその他の分野でも世代管理が有効となるものがあると予想される。
世代管理ファイルシステムを実現し、実際に使用してみることにより、有用な使い方を考える。

（４）追記型大容量記憶の利点を考える
書換えのできない大容量記憶は、重要な情報の記録に最適である。ここでは光ディスクを実際に利用することにより、この情報化社会において追記型光ディスクのあるべき立場を考える。

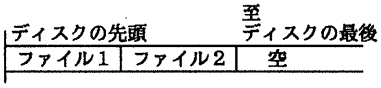
3．世代管理
プログラム等を作る際、一般にはその仕様書をまず作成する。また、仕様書の作成がプロジェクトによって行われているのであれば、原案の提出とそれに対する議論を繰り返し、仕様を固めることになる。
その中で原案の資料、会議の議事録、仕様書などの大量の文書が生まれる。情報の価値が高まっている現在、これらの文書は破棄できない性質のものとなっている。プログラムやその設計仕様書自体も重要な情報であるが、それに加えてどのような過程を経てその設計になったのか、なぜそのような設計になったのかということも重要な情報なのである。
そこで大量の文書（ファイル）をいかに管理するかを考える必要が出てくる。この管理を計算機のファイルシステムで行う場合、現存するファイルシステムではいくつかの問題が出てくる。
ファイル間の因果関係がつかみにくい、ファイルの数が多くなるにつれて管理しにくくなる等、致命的ではないにせよ決して最適とは言えない。
そこでこれらの管理をある程度自動的に、計算機に行わせようというのが世代管理である。具体的には、ファイルがいつ作られ、いつどこがどのように変更されたか等、各ファイルの変化の過程を管理するものである。
これにより仕様の変化等を管理する手間が軽減するのみならず、変化そのものが情報として残ることになり、ドキュメントとしての価値が非常に高くなる。
この世代管理は、ますます高度化、大規模化する情報化社会において重要な役割を果たすと考えられる。
当研究室では各自の提案や議事録をワープロで作成しており、大量のファイルを管理しなくてはならない状態にある。また、それらを編集して仕様書を作成するのであるが、その仕様書の世代管理が実現されれば“何故そのような設計にしたのか”という内容の含まれた、高い質のドキュメントが残せると考えている。

4. 光ディスクファイルシステムの基本形

光ディスクを用いたファイルシステムの最も基本的な形は、その名の通り追記という形である。書換えを書き加えで代用するものである。

簡単な例を挙げると、1つのディスクに1つのファイルしか格納しないと仮定して、ディスクの端にまずファイルを書き込む。ファイルの書換えが起こった場合には、前に書いたファイルの後ろに改めて書き込む。これを繰り返していき、もし最新のファイル内容が欲しい場合には、まだ書き込まれていないセクタを探してその直前のファイルを読むことになる（図1参照）。

これを少し拡張して、書き換えたファイルをリンクでつなぐ方式を取れば、階層ディレクトリをサポートした、外見の上では既存のOSのものと比較しても見劣りしないファイルシステムが出来上がる。



ファイル1が書換えられて
ファイル2になった場合

図1　追記の例

例えば図2のようにファイルの先頭にそのファイルの大きさや、ファイルかディレクトリかのフラグ、その他日付など必要な情報を書き込む。そして後からリンク用のポインタとして書き込める場所を、各々のファイルに1つずつ持たせる。

ファイルの書換えが起こった場合は、新たなファイルを追記し、古いファイルのポインタにその先頭セクタ番号を書き込む。最新のファイル内容を探すには、書き込まれていないポインタが見つかるまでリンクをたどれば良い。

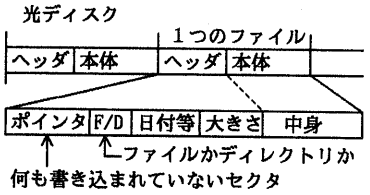


図2　簡単な光ディスクファイルシステム

またディレクトリもファイルと見なし、ディスクの先頭に書かれたファイルをルートディレクトリとすれば、階層ディレクトリが構成できる。

上記のファイルシステムは一見単純で良さそうに思えるが、様々な欠点・欠陥を持ち合わせている。

5. 追記型光ディスク

追記型光ディスクは磁気ディスク同様、計算機の外部記憶装置としてデータの記録・読出しを行えるデバイスである。ただ磁気ディスクと大きく異なる点は、一度書き込んだデータの書換えができないことである。従って追記型光ディスクを扱う場合には磁気ディスクとは別のファイルシステムを用意しなければならない。

構造上の制約からくるこの特徴は一見欠点に見えるが、逆手に取れば一度書き込んだデータは消去されずに残る、という利点となる。

次に今回使用する光ディスクの主な仕様を示す。

光ディスクドライブ	パイオニア　DD-8001
インターフェース	SCSI
回転数	450rpm CAV
データ転送速度	660K bytes/sec max.
平均シーク時間	250msec
読みだしビットエラー率	10^{-12} 以下

光ディスク	パイオニア　DC-801　（片面ディスク）
ディスク外形	8 インチ
記憶容量	750Mバイト
ディスク分類	色素ディスク
トラック形状	スパイラル
フォーマット	185,000 セクタ／面 8 セクタ／トラック 4096 バイト／セクタ
ディスク寿命	15年以上

6．当初の方針

ここで解くべき問題は“信頼性が高くデータの一貫性が取れた世代管理を行いたい、そのためのファイルシステムを光ディスク上にどのように実現するか”である。その基本的な考え方を以下に示す。

ファイルの作成はすべて追加という作業のみで行えると考えられる。例えば設計仕様書などを書く場合、普通は一度に書き上げずに少しずつ書き加える形で作っていく。修正や削除なども有り得るがそれらについては、どこからどこまでをどのように修正あるいは削除したのかという情報を追加すると考える。すると設計仕様書の作成はすべて追加（ファイルのアペンド）のみで行えることになる。

ただその追加、削除、修正の情報をどこで作り出すかが問題となる。大きく分けて２つの案がある。１つはファイルシステムあるいはアプリケーションが古いファイルと新しいファイルを比較し、その差分を取り出す方法である。もう１つは、ファイルの差分情報をエディタやワープロのコマンドで代用するものである。

前者の場合はDPマッチングなどを用いて差分を取り出す、一般的な方法である。後者は例を挙げると、ある文書のある行を削除する場合、ユーザーはその行までカーソル（行エディタの場合は行のポインタ）をもっていき、そこで一行削除のコマンドを実行する。エディタはその一部始終を記録しておく。そしてエディタを終了するときに、それまでにできあがったファイルの代わりに、今まで行われてきたコマンドの列をディスクに記録するというものである。

どちらもファイルに情報を加える形を取れるため、追記型光ディスクにとっては好都合である。

またファイルの変化の過程がすべて残るため、開発過程の記録として後で役立つ。これはソフトウェア工学に高く寄与すると考えている。例えば、この開発過程の記録を後で評価し文書化することにより、どのような問題が起き、それに対してどのような対処をとったか等のノウハウが、プロジェクトに参加しなかった

人間にも得られるようになる。

7. 設計方針

追記型光ディスクへ書き込んだデータは二度と書き換えることはできない。ディスク上のデータは全て残ることになる。よって何らかの形で過去に書いたデータを選び出せるようにすれば、世代管理機能は実現できることになる。

このように考え、基本方針は“光ディスクファイルシステムを考え、その副産物として世代管理機能を取り出す”という形を取った。

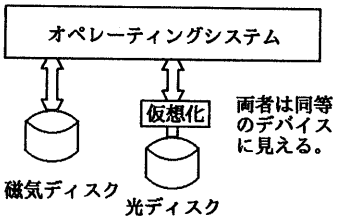


図3 光ディスクの仮想化

そこで光ディスクのファイルシステムをどうするかである。本来光ディスクは既存のOSのファイルシステムで扱えるものではない。独自のファイルシステムを考え出さなければならない。しかしそれでは研究目的である“既存のOSに光ディスクファイルシステムを実現する”ことが困難であるため、ここでは追記型光ディスクを仮想的に何度でも書き換えられるものとし（仮想ディスクの構成）、ファイルシステムは既存のOSのものをそのまま利用する。つまり光ディスクとファイルシステムを切り離し、追記型である光ディスクをあたかも磁気ディスクのように見せかける方法を考えた（図3参照）。

追記型光ディスクの容量は磁気ディスクに比べ非常に大きく、ビット単価も安い（現状でもフロッピーディスクの 1/10 程度であり一層の低下傾向にある）ため、多少のムダ使いをしたところで、それほど大きな問題となるとは思えない。また光ディスクの用途がもともとデータの保存が主であることと、それが既存のOS上でそのまま利用できることを考えれば、多少アクセス速度が遅くとも我慢ができる。これはディスクキャッシュメモリの利用によってある程度解決できる問題でもある。

以上のことから第一段階として上記の方針を実現し、新しい光ディスクファイルシステムへの足掛りにすることにした。

8. 光ディスクの仮想化

光ディスクの仮想化は、仮想－物理セクタ対応表を持つことで実現する。図4にその基本的な構造を示す。読み書きの要求は常に仮想セクタに対して行われる。要求があると、仮想－物理セクタ対応表を引いて対応する物理セクタをアクセスしてその結果を返す。

書込みの場合は、未使用の物理セクタに書込みを行い、仮想セクタにそのセクタを割り当てる。仮想－物理セクタ対応表はそれに応じて書き換えられる。

ここで問題となるのは仮想－物理セクタ対応表をどのように管理するかである。別の磁気デ

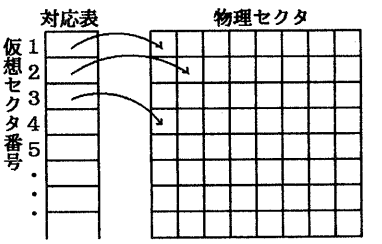


図4 仮想ディスクの構成法

ィスクにこの対応表を置く等方法は様々あるが、本方針ではこの対応表をも光ディスク上に置いて管理する。

具体的な方法を簡単に述べると、まず仮想－物理セクタ対応表を数値の１次元配列として構成する（図４左側の表）。この対応表は比較的大きい（今回の設計では 64Kバイト）のでさらに対応表の対応表を構成する（図５参照）。表を２段構成にすることにより、最終的に書き換えなければならない情報が小さくなる。

最後にこの情報を先に述べた追記という形を用いて管理すれば、光ディスクの仮想化が実現できる。

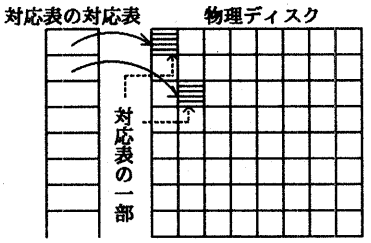


図５　仮想－物理セクタ対応表の対応表

９．世代管理の実現

世代管理で行えるべき機能としては次のようなものが重要と考える。

（１）ファイルの書込み、読出し、書換え
（２）過去のファイル（書き換えたり消去したりする前の内容）の取出し
（３）ファイルに関する過去の情報の一覧表示（更新日時、大きさ等）
（４）現在のファイルと過去のファイルとの差分の取出し

このうち（１）は光ディスクの仮想化によって実現する。（３）と（４）は（２）の実現と関連して設計時点で十分に検討しておくことにより実現可能である。ここではまず（２）の“過去のファイルの取出し”が一番基本的かつ重要な機能となるので、まずこの機能の実現を図った。

今回は過去のファイルの取り出しを、アプリケーションがデバイスドライバを直接操作し、仮想ディスク全体の世代管理を利用して行うものにする。

ディスク全体の世代とはディスクバックアップのイメージに近いものである。ディスクの世代を終了すること（仮想－物理セクタ対応表を光ディスクに書き込むこと）はバックアップを１枚作ることに対応する。

ただ普通バックアップは１日とか１週間に１度というように、さほど頻繁には行われないが、ここではファイルが１つ書き込まれたり削除されるたびに光ディスク側が自動的にバックアップを作る。

ディスク全体の世代を管理することは、大量に生成されたバックアップを管理することに対応する（図６参照）。

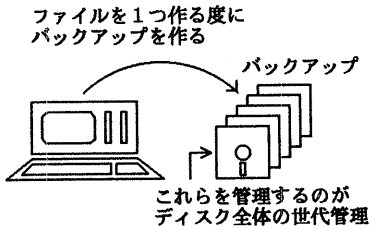


図６　仮想ディスク全体の世代管理

アプリケーションはこれを利用し、ファイルごとの世代管理を実現する。

１０．実現環境

本研究の実現環境は以下のとおりである。

ホスト計算機	NEC PC-9801VM2
光ディスク装置	パイオニア　DD-8001
光ディスクインターフェース	パイオニア　DDI-8098
メモリ	主記憶640Kバイトのほかに4Mバイト増設
OS	MS-DOS Ver 3.1

１１．おわりに

現在MS-DOS上の光ディスクデバイスドライバ実現のための設計仕様書の作成が完了し、インプリメント中である。今後は光ディスクファイルシステム第一版を実現した後、それを実際に使用することにより、光ディスク利用効率、世代管理の有用性等の評価を行う予定である。

光ディスクには追記型の他に、光磁気ディスクなどの書換え可能なものも存在する。しかしそれらが商品化され、一般の計算機に使用されるようになるには時間がかかるであろうし、またアクセスが遅いなどまだ多くの問題を抱えているため、磁気ディスクに取って代わるまでには、さらに時間がかかると思われる。

これに対し追記型は書換えができないが故に、第３の記憶装置として受け取られ、ほかのディスクデバイスとは別の道を歩んでいる。

メモリ素子の容量が大きくなるにつれ、プログラムを組む際にはメモリの節約よりも、プログラムの生産性を意識することが一般的になっている。ディスクにおいてもビット単価の非常に安い追記型光ディスクの出現により、ムダ使いが可能になったのである。節約よりも使いやすさ、使い道を重視したものを考えるべき時期を迎えているのではないだろうか。

１２．参考文献

（１）DD-8001 ８インチ光ディスク装置仕様書　　パイオニア株式会社
（２）標準MS-DOSハンドブック　　株式会社アスキー