システムソフトウェアと
オペレーティング・システム 64−25
マルチメディア通信と分散処理 65−25
（1994. 5. 20）

共同利用計算機におけるファイル管理に関する一考察

金澤正憲、平野彰雄、赤坂浩一

（京都大学大型計算機センター）

概　要

多くの研究者が共同で利用する大型計算機システムでは、利用者の保存ファイルおよびセンターの提供するデータベースの容量は、増加の一途を辿っている。しかし、すべてのファイルが毎日参照されるわけではなく、3ヶ月から数ヶ月間まったく参照されないファイルもある。このような状況に対して、ファイルの階層化が採用されるのが通常である。ここでは、京都大学大型計算機センターで導入した利用者保存ファイルに対する階層ファイルの構成・概念と運用について述べる。さらに、利用者保存ファイルのバックアップとリカバリの方式についても述べる。

A Consideration on File Management for a Large Scale Computer System

Masanori KANAZAWA, Akio HIRANO, and Hirokazu AKASAKA

(Data Processing Center, Kyoto University)

Abstract

The users' data files and the databases have been increasing in volume in the large scale computer system shared by many researchers. It is unusual that all of the users' data files are accessed everyday. There are some files which are not accessed for several months. Hierarchical file system is, thereby, introduced in most of the computer systems In this paper, we discuss the concept of the hierarchical file management system to users' data files, and the design of backup and recovery method at the Data Processing Center, Kyoto University.

1. はじめに

京都大学大型計算機センターでは、大規模計算機システムを設置し、毎年5000人以上の大学等の研究者が自由に共同利用できるようになっている。個々のジョブやセッションで使用できるリソース(CPUや主記憶など)の量には制限があるが、ジョブの依頼できる総件数や利用者保存ファイル(以下利用者ファイルという)の総容量は実際には制限がないといえるであろう。利用者ファイルに関しては、大型計算機がファイルサーバとして機能し、汎用OSのMSP/EXでは、1994年2月末現在、個数が約13万8千個、容量が約130GBである。

利用者ファイルに関しては、アクセスの頻度と装置のコストの問題、及び、保全性の問題が、計算機センターでは重要である。ここでは、階層化されたファイル管理により、両方の問題を解決する方法を検討し、どのような運用がもっとも合理的であるかを論じ、その適応について説明する。

2. ファイルの階層化

2.1 利用者ファイルの使用状況と記憶階層システム

システムの処理効率および応答時間から見れば、利用者ファイルはすべて磁気ディスクに格納しておくのが好ましい。このためには、多数台の磁気ディスク装置が必要である。一方、利用者ファイルは、いつアクセスされるか判らないのが現実であり、コスト面から考えると、低速で大容量のファイル装置を加えた階層ファイルシステムを構成するのが適当である。

本センターでも一時期、大容量磁気テープ装置(MSS)を導入し、階層ファイルシステムを構成しようとしたことがあった。しかし、MSSは、磁気テープと同様の順アクセス装置であり、記憶媒体の後方に記録されたファイルへのアクセスは3分以上かかることが判った。このため、記憶階層を担う装置よりもむしろ、自動装填可能な磁気テープ装置として運用した。即ち、利用者が性能の異なるファイル装置に格納していることを意識して利用する方式にした。それでも、アクセスの時間の長さは、会話型処理では我慢できるものではなかった。

その後、磁気ディスクの記録の高密度化の技術革新があり、大容量化・コンパクト化され、数年の間、磁気ディスクのみで構成することができた。しかし、より大容量の使用の要求に加えて、バックアップの問題が大きくなってきた。

本センターでのこの10年間の利用者ファイルの使用量とファイル個数を表1に示す。但し、年度内の最大値(通常2月の値)である。また、1993年3月末の利用者ファイルの大きさの分布は、図1に示すように、1および2トラックで半分以上になる。

表1. 利用者ファイルの利用状況

年　度	'84	'86	'87	'89	'90	'91	'92	'93
容量(GB)	13	15	84	92	112	118	129	130
個数(千個)	60	68	122	143	145	147	148	138

注. 1984~86年はMSSがあり、40~50GBが利用者ファイルとして使用されていた。85,88年は紙面の都合で省略。

最近、磁気ディスクを補う装置として大容量の光磁気ディスクが現れた。ランダムアクセスが可能で、さらに、記録の高密度化、読出しだけでなく書込みの高速化、実装のコンパクト化などにおいて、優れたファイル装置である。本センターに設置されている光磁気ディスクの諸元を磁気ディスクと対比させて表2に示す。

表2. 光磁気ディスクの諸元

項　目	大容量光磁気ディスク	磁気ディスク
装置当たりの容量	644 MB	2.8 GB/15面 (187MB/面)
転送速度	2.1 MB/秒	4.5 MB/秒
平均シーク時間	44 ms	12 ms
平均回転遅れ時間	5.6 ms	6.9 ms
平均ロード時間	5.5 秒	—

2.2 階層ファイルシステムにおける基本方式

階層ファイルシステムを運用するには、利用者ファイルのアクセスを分析する必要がある。そこで、VTOC(Volume Table Of Contents)の情報から、個々のファイルが最後にアクセスされてからどれ位の日数が経過しているか、本センターの計算機システムで調査した。

アクセスがあるとは、そのデータセットを実際にOPENしたことを指す。例えば、利用者ファイルの名前の一覧を打ち出すTSSコマンド

(LISTCAT)を用いても、これはアクセスに該当しない。区分ファイルのメンバー一覧を打ちだすことは、そのファイルの先頭部分を打ちだすことであるから、ここでいうアクセスに該当する。

利用者ファイルへの最終アクセスからの経過日数は、1994年3月末に調査した結果は、図2のとおりである。横軸の単位は週で、縦軸は該当するファイルの個数である。1~10週では、指数分布的に個数が減少しているが、20週を越えると、緩やかな減少というよりほぼ横ばいになっている。さらに驚くべきことは、図1には示していないが、156週(3年)以上のファイルが数多く(約28,000個)残っていることである。

一度アクセスされてから3ヶ月位までは、アクセスされる確率がかなり高いが、それ以上になると期間にあまり関係しないことが判った。即ち、利用者は時々思い出したかのように長期間アクセスしなかった利用者ファイルをアクセスする。従って、常時は倉庫にしまわれて、必要な時にそこからファイルが呼び戻されてできるだけ速くアクセスできることが必要である。

大容量の光磁気ディスクライブラリ(以後、光ディスク装置という)は、記憶媒体が常時回転しているわけでない。必要な光ディスクを、ドライブ(読み書きを行う部分)にアクセッサで持ってきて、回転させて読み書きを行う。大容量で、省電力で、コンパクトで、かつ、アクセス時間が短いため、光ディスク装置が階層ファイルシステムを構成する装置として適していると判断した。

2.3 マイグレーションとリコールの方式

磁気ディスクの使用容量が多くなると、新しくファイルを割り付ける、または、既存のファイルを拡張するときに充分な領域が確保できなくなり、プログラムが異常終了することになる。このために、ある判断基準で持って、ファイルを光ディスクに追い出す(マイグレーションする)必要がある。マイグレーションのアルゴリズムとして、次のようなものが考えられる。

[マイグレーションのアルゴリズム]

ⓐ 経過日数方式:最終アクセスから予め設定された日数を越えているファイルをすべて追い出す。

ⓑ 限界値方式:磁気ディスクの領域割当て率が、予め設定された値よりも低くなるまで、最終アクセスの古い順位にファイルを追い出す。

経過日数方式は、利用者から見ると、ファイルが追い出されているかどうかの基準が明確なため、違和感がすくない。しかし、日の変わり目に、該当するファイルを一斉に追い出しをかけることになる。従って、同時にマイグレーションされるボリュームの数を制限するか、リコールの優先制御など、応答時間が極端に長くならないようにする対策が必要である。

限界値方式は、一度に一ボリュームが対象となるので、両ディスクのビジー率への影響は少ないと考えられる。しかし、利用者から見ると、同じ最終アクセス日の利用者ファイルが、一方は追い出され、他方は追い出されていないという不明確な状況になる。

つぎに、磁気ディスクへのファイルの割当てを考える。利用者ファイルは不特定ボリューム指定で割当てられるので、どのようなアルゴリズムのもとで実際に割当てるべきボリュームを選択するかが問題である。本センターでは、磁気ディスク装置への新規ファイルの割当ては、領域が確保できなくてジョブが異常終了しないように、以下のような方式を採用している。光ディスクからリコールされたファイルもこのアルゴリズムが適用される。

[新規ファイルの割当てアルゴリズム]

(i) 磁気ディスクの領域割当て率が、予め設定された値(本センターでは80%)以上の場合、そのボリュームへの新たな割当ては抑止する。即ち、選択されるボリュームの対象外とする。[1]

(ii) 領域割当て率が、設定値より低いボリュームの中から、最もビジー率の低いボリュームを選択する。なお、ビジー率は、アクセスによって当該ボリュームが占有されていた時間的割合で、適当な時間間隔ごとにOSで測定された値を用いる。

新規ファイルの割当てアルゴリズムと合わせて考えると、マイグレーションのアルゴリズムは、つぎの理由で、限界値方式の方が優れていると判断できる。

(i) アクセス日の分析結果から、20週を越えると殆ど日数に依存しないので、マイグレーションのための日数を予め設定する根拠が乏しい。

脚注1 すべてのボリュームが越えている場合は、この限りではない。

(ii)用意された磁気ディスクには、できるだけ多くの利用者ファイルを残しておいて、できるだけアクセス時間を短くしたい。

(iii) 反対に、利用者ファイルは増加の一方なので、それを監視して、日数を漸次短くしていく必要がある。

この場合でも、新規割当ての限界値(A)とマイグレーション開始値(M)との関係という問題がある。

もしA=M(ほぼ等しい)ならば、領域割当て率がAに近づくと、新規割当てでAを越えるとマイグレーションが発生する。

もしA<M(有意を持って小さい)ならば、新規割当てによるマイグレーションは殆ど無くなり、領域拡張の場合にマイグレーションが起こることになるであろう。さらに、この場合は、新規の割当てアルゴリズムが有効に機能しない可能性がある。

もしA>M(有意を持って大きい)ならば、新規割当て毎にマイグレーションが発生する事態が容易に考えられる。

以上から、A=Mとせざるを得ない。

そこで、解決策として、限界値方式に加えて、マイグレーションは予め決められた時刻、例えば、深夜に行うことにした。なお、この方式ではAがMより少し大きな値に設定するのも適切かもしれないが、ここではこれ以上取り上げない。

もう一つの問題として、マイグレーションを利用して、磁気ディスクのフラグメンテーションをいかに少なくするがある。これについては、さまざまな方式が考えられるが、別の機会の検討としたい。

リコールに関しては、マイグレートされた利用者ファイルへのアクセス要求が発生した時点でリコールするデマンド・リコール方式を採るのが適当である。なぜならば、リコールされる利用者ファイルの予測は不可能であるからである。

現在は、運用して間もないので、リコールが殆どない。そのため、待ちがなくて平均ロード時間程度(10秒程度)で、殆ど問題がない。ところが、マイグレーション中や、後述するバックアップ中に、リコールが発生すると、ロード時間が極端に長くなり、問題になっている。

3. ファイルのバックアップ

3.1 バックアップの考え方

磁気ディスクのヘッドクラッシュなど予期せぬハードウェアのトラブルからファイルを守るため、バックアップを取るということは昔から考えられていた。本センターも利用者ファイルをサービスした時点から、その影響を最小限に抑えるため、磁気テープへのバックアップを続けてきた。現在バックアップは、カートリッジテープで行っているが、日々カートリッジテープ群を交換するために人手を要している。幸いなことに、ここ十年以上大きな事故はなかった。

一方、利用者の観点からは、利用者のファイルに関するトラブルがなかったかといえば、それは正しくない。ハードウェアのトラブルに起因するものではなくて、利用者自身による操作ミスに起因するものが多い。例えば、ファイル名の指定を誤って別のファイルを消去してしまったとか、コピーによる上書きとかである。バックアップということから考えれば、これらに対しても、修復の方法を提供することが重要である。

3.2 バックアップのタイミング

利用者ファイルをどの時点でバックアップするか、分類し、その特徴を検討する。

ⓐ 即時方式:ファイルを更新して、クローズした時点で直ちにバックアップする方式。

(i) ファイルを新規作成した場合は、磁気ディスクへの書き込みと同じ量を光ディスクにコピーする必要がある。

(ii) ファイルの一部を更新した場合は、ファイル全体を光ディスクにコピーする必要がある。区分ファイルの1メンバを修正した場合がこれに当たる。

(iii)直接ファイルを更新した場合は、最終状態をコピーすればよい。

ⓑ ジョブ終了時方式:ジョブまたはセッションが終了した時点で、更新されたファイルのみをバックアップする方式。

(i) 即時方式に対して、1ジョブまたはセッション内で、何度も更新し、クローズしたファイルは、バックアップは1回で済む。

(ii) さらに、1ジョブまたはセッション内で、作成し、終了時に消去したファイルは、バックアップの必要がない。

ⓒ 1日1回方式:1日1回、更新されたファイルのみをバックアップする方式。

(i) 方式ⓒに対して、1日に何度も更新したファイルでも、バックアップは1回で済む。
(ii) さらに、あるジョブで作成し、計算結果の確認の後、不要になり消去したファイルは、バックアップの必要がない。
(iii)更新後、1日1回のその時刻までにトラブルが発生すると、バックアップがないことになる。

一方、操作ミスによる破壊からファイルを救済するという観点から検討すると次のようになる。

即時方式:
(i) 誤ってファイルに上書きすると、直ちに、そのコピーがとられるため、救済できない。救済するためには、1レベル古いファイルも保存するという世代管理が必要である。
(ii) ファイルを誤って消去した場合も同じである。

ジョブ終了時方式:
(i) 誤ってファイルに上書きしたり、削除した場合でも、そのセッション中であれば、救済可能である。しかし、バッチジョブの場合は、通常救済できない。従って、救済するためには、1レベル古いファイルも保存するという世代管理が必要である。

1日1回方式: 例えば、深夜の2時にバックアップするとする。
(i) 誤ってファイルに上書きしたり、削除した場合でも、午前2時までであれば、救済できる。但し、深夜の2時の少し前に終了したバッチジョブで破壊した場合は、実際的には救済できない。

本センターでは、利用者ファイルが多量であるため、2重にバックアップを取ることは容量的に無理がある。さらに、昼間は、TSSの同時アクティブ数が大きく、できるだけ余分な負荷をかけたくない。利用者の操作ミスによる利用者ファイルの消去・破壊は、操作後直ちに気付くのが殆どである。また、更新をかけたファイルは一日程度のの作業で行われたものであれば、バックアップされている古い版を取り出し、再実行させて修復可能と推測できる。

このようなことから、本センターでは、1日1回方式を採用することにした。さらに、消去したファイルのバックアップ版の消去は、一日か二日遅らせる方式を採り、より多く救済できるように機能アップしたいと考えている。

この1日1回方式では、マイグレーションも同じ深夜の2時にすると、一時に光ディスクへのアクセスが集中し、応答時間が急激に遅くなるという事態を生じさせるであろう。少し、時刻をずらす必要があると考えている。

3. まとめ

ここでは、利用者ファイルに対する階層ファイルシステムの構成と方式について提案し、検討した。その結果、マイグレーションは限界値方式で、バックアップは機能アップされた1日1回方式で行うのが最適であるという結論が得られた。即ち、磁気ディスクをできるだけ有効に利用するとともに、ハード的障害および人間的ミスから利用者ファイルを守れることが判った。深夜に自動的にマイグレーションを行い、その終了後バックアップを引き続き行うという手順で実施するよう、現在準備している。

マイグレーションを運用開始し、やがて定常的な状態になると、磁気ディスクの領域割当て率がどれもマイグレーションの設定値近くになる。従って、一日で増加する(リコールされたファイルも含めて)ファイルの容量だけマイグレーションすればいいことになる。利用者ファイルに使用量の増加率を見れば、年間20GBであろう。運用は300日であるので、一日平均67MBとなる。リコールを含めて、その6倍(同量のリコールがあり、異常値が平均の3倍とする)を考えても、600MBで、光ディスクの諸元から見れば、転送は10分程度で済む。

しかし、問題は光ディスクの取り出しと返却に時間を要することである。1トラックの利用者ファイルをマイグレーション/リコールする時間は、約30秒と測定された。この結果から、利用者ファイル1つごとに取り出し・返却をしていたら、1アクセッサ当たり1時間に120個の利用者ファイルしか扱えないことになる。光ディスクの使用量にアンバランスができても、できるだけ同じ光ディスクにマイグレーション/バックアップする必要がある。

リコールはデマンド方式のため、要求があれば、できるだけ短時間でリコールしなければならない。マイグレーションおよびバックアップ中にリコールが発生すれば、リコールを優先処理しなければならない。少なくとも、マイグレーショ

ン/バックアップ・キューより、リコール・キューを高い優先度にしなければならない。

その他にも、例えば、TSSのALLOCATEコマンドの場合、リコールの必要が判明すれば、別タスクにまかせて、端末入力状態に戻すということが考えられる。しかし、このような単純な方式では済まないであろうから、今後、検討しなければならない。

バックアップの容量は、現在のシステムの測定結果からでは予測ができない。従って、実際のシステムで運用開始してから、磁気ディスクや光ディスクの動作状況を測定し、解析を進めたいと考えている。

最後に、ここで提案した方式の実システムへのインプリメントは、富士通㈱によって行われている。ここに、感謝の意を表します。

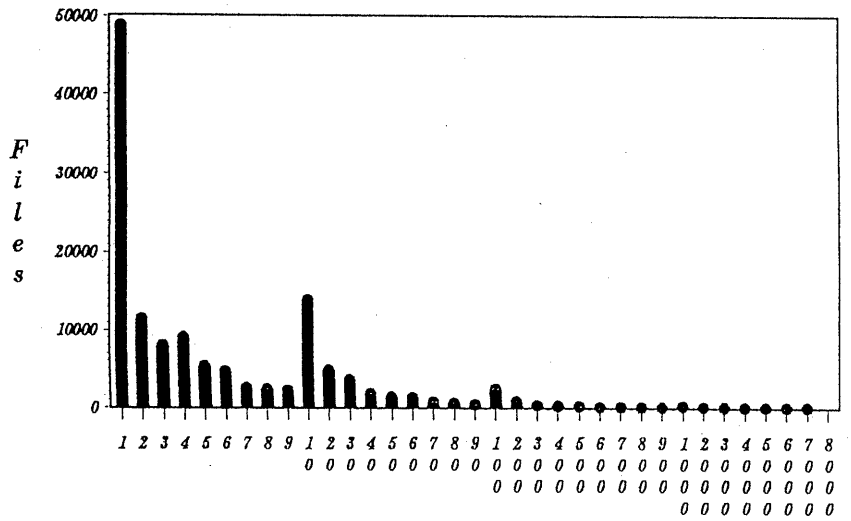

図1. ファイルの大きさの分布

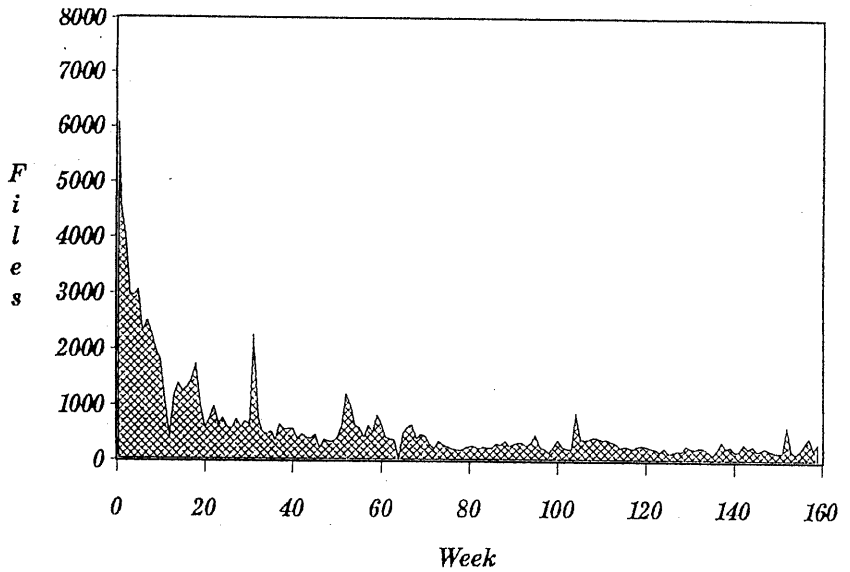

図2. 利用者ファイル最終アクセスからの日数の分布