

整理と検索のための労力を考慮した 操作履歴に基づく電子ファイル整理

A File Organization Method with Consideration of Efforts for Rearrangement and Search Based on Operation Logs

寺井 政文[†]
Masafumi Terai

原田 史子[†]
Fumiko Harada

島川 博光[†]
Hiromitsu Shimakawa

1. はじめに

社会の急速な情報化に伴い、電子ファイルに対する重要性が高まってきている。企業では、省エネなどの視点から紙資料の電子ファイル化が進められている。また、各種製品のマニュアルなどの電子ファイル化も進んでおり、個人が計算機上で扱う電子ファイルの数は増加傾向にある。

計算機上で効率よく電子ファイルを検索するためには整理が重要となり、一般的にはフォルダを用いた階層構造による整理がなされる。しかし、適切な整理を行うには労力がかかるため、不要なファイルが削除されずに残されていたり、一時的な場所に保管されたファイルがいつまでもそのままになっているなど、効率的なファイル整理ができていない場合が多い。一方、一時的な場所にファイルを保管する整理では、整理に時間はかからないが、どのファイルがどの目的のためのファイルであるかわかりにくく、検索に多大な時間を要する [1]。そのため、特定の目的のためにユーザが必要とするファイルを効率よく検索できず、作業に支障をきたすことがある。

そこで、ファイルの整理と検索のための時間の両方を考慮し、ファイルアクセスのための労力を減らすことで作業の効率化を実現する整理手法を考える。特に本論文では、整理に要するユーザの労力を減らすために、操作履歴を取得することで整理を行い、検索に要する労力を減らすために、目的ごとに必要なファイルをすぐに操作できるような整理手法を提案する。本手法では、使用頻度が高いファイルに対して、検索に要する時間を減らすための階層構造の構築を支援できるという特徴を持つ。また、目的に応じてユーザの作業に必要なファイルがまとめて表示されるファイル整理がなされる。

提案した整理手法の仕組みを実装し、階層構造による整理との比較実験を行った。100個のファイルを整理するために、本手法で提案した整理では2時間0分38秒、階層構造による整理では2時間44分56秒かかった。一方、1個のファイルを検索するために、本手法で提案した整理では41.6秒、階層構造による整理では20.7秒かかり、検索回数130回で両検索法の整理に要する時間と、各回検索に要する時間を含めたファイルアクセスのための作業時間が入れ替わった。結果より、一時的に使用するファイルであれば、本手法で提案した整理を用いる方が短時間となった。

2. 電子ファイル整理

人は計算機上で海外出張についての仕事や卒業研究などの目的を持って作業する。過去のファイルを探すさい

には、ファイル名でなく、あの時のあのファイルというようにあいまいな情報で探す場合が主であるため、電子ファイルの整理は重要である。電子ファイル整理においては、フォルダを用いた階層構造を構築する整理法が一般的である。一度階層構造を構築すれば、フォルダ内で名前順や時刻順による並べ替えができ、長期的にファイルを使う場合には、検索が容易となり作業効率上がる。しかし、この整理には、フォルダの作成やファイルの移動、コピーなどの操作が必要となるため労力がかかる。整理が適切になされていないと必要なファイルにすぐにアクセスできないため、時間をかけて階層整理を行うことになる。一方、一時的な作業にしか使わないファイルでは、整理時間が無駄となり作業に支障をきたす原因となる。

これに対して一時的なフォルダにファイルが保管され、階層構造を用いた整理が行われない場合がある。ある目的のための作業を単独で行う場合、一時的な保管場所にファイルを配置する方がすぐに操作でき便利である [2]。しかし、実際には複数の目的のための作業を切り替えながら並行して行うので、一時的な保管場所はファイル群ですぐに埋め尽くされ、作業に必要なファイルを見つけ出すのが困難となる [1]。両整理法は、整理と検索にかかる労力において一長一短である。ファイルの整理と検索のための時間の双方を考慮した整理法を実現できれば、ユーザの作業を効率化できる。

3. ユーザの労力を軽減する操作履歴による電子ファイル整理

3.1 整理に要する時間と検索に要する時間

本研究では、ユーザが目的を持って作業するさいの、ファイルにアクセスするための労力を減らし、ファイルを効果的に整理できるようにすることを目的とする。ファイルにアクセスするための労力として、ファイル整理時間とファイル検索時間が挙げられる。本研究ではこれらを合わせてファイルアクセス作業時間と呼ぶ。頻繁に検索されるファイルほど、検索にかかるユーザの労力の総和が大きくなるといえる。フォルダを用いた階層構造による整理と一時的な保管場所での整理において、ファイルアクセス作業時間と、ファイル検索回数との相関関係は図1のようになると予想される。

階層構造による整理では、階層構築のための労力がかかるが、後でファイルを検索するための労力は軽減される。一方、一時的な保管場所にファイルを保管する整理では、階層構築のための作業は不要であり、整理に要する時間は階層による整理と比べると短縮されるので、一時的にファイルを使うさいには、ファイルアクセス作業時間は短時間となる。長期的にファイルを使う場合には、

[†]立命館大学大学院理工学研究科

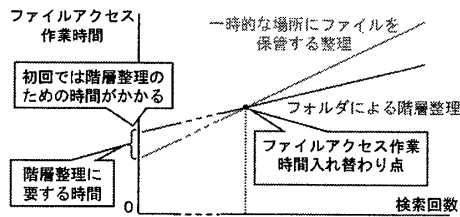


図1: 階層整理の有無によるファイルアクセス作業時間の関係

階層による整理のほうが検索時間が短くなり、ファイルアクセス作業時間は短時間となる。このようにファイルの使用頻度によって、ファイルアクセス作業時間が入れ替わる点が存在する。この点を求めることで、ファイルの使用頻度に応じたファイル整理を支援できる。

3.2 操作履歴による目的に応じたビュー作成

一時的な保管場所による整理でも、できるだけ検索の労力を減らせることが好ましい。目的に応じて各ファイルに対する扱いは異なり、目的ごとにファイルがまとめられていると作業効率がよい。一方、同一のファイルを異なる目的のための作業で扱うことはしばしば起こり、扱いが困難である。これらを踏まえ、図2のように、ファイルそのものの移動やコピーをするのではなく、ファイルへの参照の集めで作られるビューを利用すれば、目的ごとに独立したファイル整理ができ、ユーザは作業に集中できる。そこで、階層構造構築を必要としない、ユーザの負担を減らした整理法として、ファイルへの参照を利用した、目的ごとに必要なファイルをまとめるビューの作成手法を提案する。

本手法では、目的ごとにビューが作成される。ビューの名前はユーザが入力する目的と対応しており、目的とするビューがない場合はユーザが作業名を入力することで作成される。作成した時点でのビューは空の状態であるが、閲覧、更新、コピー、ペーストなどの操作がなされたファイルを随時ビューに加える。ビュー内のファイルは階層構造構築を必要としない一覧表示を採用する。ビューの内容は常に保存しておき、作業を一時中断する場合も、再開時に対応するビューを選択することにより、前回のビューの状態を閲覧できるようにする。

3.3 枢軸ファイルによるフェーズ分割

どの目的に対する作業においても、進捗に応じて各ファイルに対する重要性も変化する。ある1つの目的に向けた作業はいくつかのフェーズに分割することができ

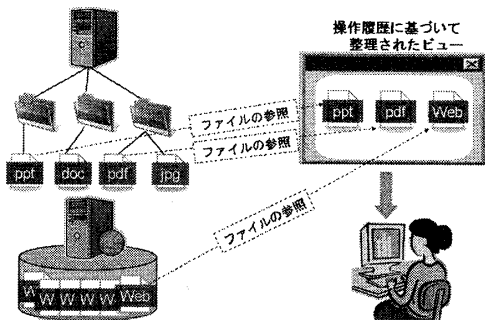


図2: ファイルへの参照を利用したビューによる整理

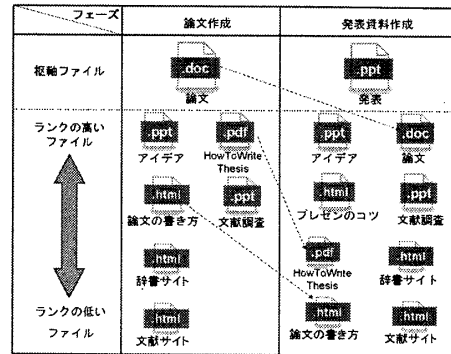


図3: フェーズに応じたビュー内の整理

る。例えば、「卒業研究」という目的であれば実験、論文作成、発表資料作成の3つにフェーズを分割することができる。同じ目的で扱うファイルでもフェーズによって重要性は変化する。各フェーズにおいて、使用頻度の高いファイルが最も重要といえる。例えば、実験フェーズでは実験結果を記したファイル、論文作成フェーズでは論文の原稿となるファイルが挙げられる。このとき、フェーズごとの各ファイルの重要性に基づいて、フェーズごとにビュー内のファイルが整理されていると作業ははかどる。前のフェーズで重要性が高いファイルが次のフェーズで必ずしも重要性が高くなるとは限らないので、重要性に基づいた整理を行うことで、ユーザの作業効率はよくなる。

本手法では随時各ファイルの重要性を求め、それに基づいてフェーズの切り替わりを判定する。

ファイルの重要性は各フェーズでの更新回数と比例すると考えられる。例えば論文作成フェーズでは重要性の最も高いファイルは論文の原稿となるファイルであり、更新回数も多くなる。以降、更新回数の最も多いファイルを枢軸ファイルとよぶ。更新回数と同じであれば閲覧時間が長いファイルを枢軸ファイルとする。1つの枢軸ファイルが決まると、その枢軸ファイルと更新が行われている他のファイルの更新頻度を監視する。現在の枢軸ファイルの更新頻度が減少し、他のファイルの更新頻度が増加した場合、後者を新しい枢軸ファイルとする。枢軸ファイルの切り替わりと同時にフェーズが切り替わったと見なす。新しく枢軸ファイルとなったファイルが初めて更新された時刻から次のフェーズとし、ユーザは新しいフェーズの名前を決定する。フェーズが切り替わった直後のビューは、直前のフェーズのビューの状態を表示する。

3.4 フェーズに応じたビュー内の整理

各フェーズにおいて図3のように、各ファイルの重要性は変化する。枢軸ファイルと関連性の高いファイルも変化する。枢軸ファイルと関連性の高いファイルがまとめて表示されるとユーザの作業効率は上がるため、フェーズごとに、ビュー内のファイルを枢軸ファイルとの関連性や重要性によって整理する。

各ファイルの重要性は、更新回数と、計算機上でアクティブになっていた時間によって決定する。関連性は各ファイル間でのコピー、ペーストといった操作の回数によって決定する。ビュー内のファイルは、まず枢軸ファイルを中心に整理され、次に枢軸ファイルとの関連性と

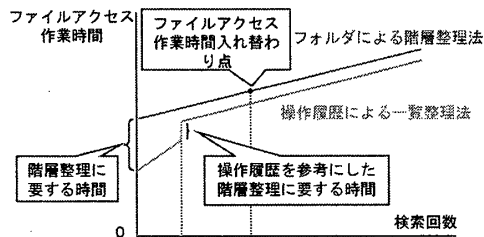


図 4: 階層構造支援による整理と検索の関係

重要性を基に算出される値によって整理される。これをランクとよぶことにする。ここで、重要性を I とする。枢軸ファイルとの関連性を Re 、重み係数を α とすると、ランク R は、 $R = I \cdot \alpha \cdot Re$ という式で導くこととする。ビュー内のファイルは一覧表示され、ランクが高いものを上から順に表示する。このように、1つの目的に対応するビューでも複数のフェーズに分割され、それぞれのフェーズごとに整理がなされるため、ビューは複数の整理された状態を保持する。

3.5 ビューによる階層構造構築支援

長期的に使用するファイルであれば、階層構造を構築した方が短時間で検索することができるが、階層構造構築に時間がかかる。ビューにはフェーズごとに必要なファイル群が集合としてまとめられているため、対応するフェーズのビューを参考にすることで、図4のようにユーザは適切な階層構造を短時間で構築可能である。

これにより、階層構造構築に時間を費やさずにユーザは目的に応じたファイルを効率良く検索でき、本手法はユーザの労力を軽減し効果的な整理を実現できる。

4. 実験

4.1 実験目的

3章で提案した操作履歴による整理がなされたビューと、フォルダを用いた階層構造による整理とで、整理に要する時間とファイル検索に要する時間を求める実験を行う。結果は3.1節で紹介したグラフのようになると予想され、実際にそれを検証する。また、ファイルの使用頻度に応じてファイルアクセス作業時間が入れ替わる点が存在することを確認し、両整理法による整理と検索にかかる時間を明らかにすることが目的である。

4.2 実験内容

本実験の被験者は20歳～24歳の男子13名、女子1名である。被験者14名を提案手法を用いた整理を行うグループ7人(被験者A～G)と、階層構造の構築によって整理を行うグループ7人(被験者H～N)に分けて実験を行った。実験は複数の作業を切り替えながら並行して行うことを想定し、3段階に分けて実施した。

(1) 被験者は「研究調査」という目的の作業を行う。論文、ゼミ資料、実験風景の写真など、複数の研究に関する与えられた種々のファイル100個を閲覧し、感想やまとめをプレゼンテーションファイルとして作成する。被験者A～Gは、「研究調査」という名前のビューを作成し、その結果、閲覧したファイルの参照が作成したビューに格納され、整理がなされる。被験者H～Nは、与えられたファイル100個に対し、階層構造を用いて自由にファイル整理をする。

(2)(1)の実験から約1週間後に実施し、被験者は「イベント調査」という目的の作業を行う。(1)の作業と同様に各イベント時に撮影した写真や行程表など、複数のイベントに関する与えられた種々のファイル100個を閲覧し、感想やまとめをプレゼンテーションファイルとして作成する。被験者A～Gは、「イベント調査」という名前のビューを作成し、閲覧したファイルの参照がそのビューに格納され、整理がなされる。なお、被験者H～Nは与えられたファイル100個を階層構造を用いて自由にファイル整理をする。

(3) ファイル検索を目的とし、(2)からさらに約1週間後に実施され、ここでファイル検索時間を計測する。(1)と同じ「研究調査」という目的の作業の次のフェーズとして、(1)で閲覧した研究に関するファイル100個から9個のファイルを検索する。被験者には「FITの学会に提出した特定のある学生Wの論文」という情報を与え、その情報から1つのファイルを検索する。検索し終わると、次に検索するファイルの情報が与えられ、これを繰り返し行い、それらのファイルへのリンクが一覧表示されるWebページを作成する。

4.3 作業時間の比較結果と分析

各被験者ごとに、(1)でファイル整理に要した時間、(3)にてファイル検索に要した平均検索時間をそれぞれ計測した結果を、表1と表2に示す。

平均整理時間は、操作履歴による整理を用いるグループ(被験者A～G)では2時間0分38秒、階層構造による整理を用いるグループ(被験者H～N)では2時間44分56秒要した。100個のファイルを整理するさいには、前者のほうが整理に要する時間は大幅に短くなり、階層構造を構築するために44分18秒要した。また、9個のファイル検索に要した平均検索時間は、操作履歴による整理を用いるグループでは41.6秒、階層構造による整理を用いるグループでは20.7秒となった。この結果から、操作履歴による整理を用いるグループと階層構造による整理を用いるグループでファイル検索時間に二倍近く差が出た。これは、ファイル検索のさいにビューを用いる被験者A～Gは、マウスフォーカスは検索するファイルの行に合っているが、被験者が求めているファイルだと気付かずに、次の行に移っている光景が多々見られたことが関連していると考えられる。ビュー内のファイルが一覧表示となっており、似た名前をもつファイルや同じ種類のファイルが並んだことと、ファイル数に比例して閲覧が困難となるからである。このため、重要性や関連性の低いファイルを排除する機構とユーザインタフェースを検討する必要がある。

4.4 ファイルアクセス作業時間入れ替わり点に関する結果と分析

両整理グループでの、整理時間を含めた累計ファイル検索時間であるファイルアクセス作業時間を図5に示す。

同一のファイルを検索する時点で、両整理法のファイルアクセス作業時間が入れ替わる。ここがファイルアクセス作業時間入れ替わり点とみなせる。予想したように、3.1節で示した図のようになった。

以上より、1個のファイルを探すさいには検索回数130回がファイルアクセス作業時間入れ替わり点となる。こ

表 1: 操作履歴による整理の実験結果

| 被験者 | A | B | C | D | E | F | G | 平均 |
|---------------|---------|---------|---------|---------|---------|---------|---------|---------|
| 整理時間 (時間:分:秒) | 1:57:09 | 1:31:28 | 2:26:01 | 2:47:17 | 1:44:56 | 1:53:36 | 1:44:02 | 2:00:38 |
| 検索時間 (時間:分:秒) | 0:00:43 | 0:00:27 | 0:00:57 | 0:00:41 | 0:00:43 | 0:00:33 | 0:00:47 | 0:00:42 |

表 2: 階層構造による整理の実験結果

| 被験者 | H | I | J | K | L | M | N | 平均 |
|---------------|---------|---------|---------|---------|---------|---------|---------|---------|
| 整理時間 (時間:分:秒) | 3:39:03 | 2:58:15 | 2:02:43 | 1:26:46 | 3:22:24 | 3:44:26 | 2:00:54 | 2:44:56 |
| 検索時間 (時間:分:秒) | 0:00:20 | 0:00:25 | 0:00:16 | 0:00:19 | 0:00:24 | 0:00:28 | 0:00:14 | 0:00:21 |

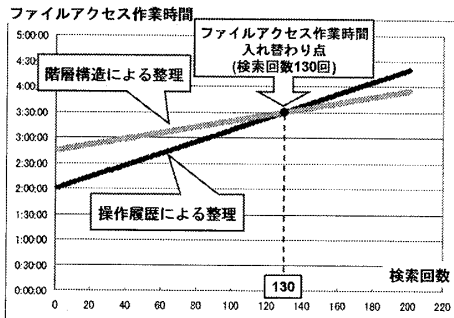


図 5: 検索するファイルが1つである場合のファイルアクセス作業時間入れ替わり点

の点を境に作業に要する時間は入れ替わる。また、1つの作業においてよく扱うファイルは1個とは限らず、数個のファイルが扱われる。検索するファイル数と、ファイルアクセス作業時間入れ替わり点との相関関係を図6に示す。

今回の実験で実際に検索したファイルは9個であり検索回数15回が交差点となる。1つの作業において扱うファイル数が少ない場合は、操作履歴による整理の方が作業効率はよい。扱うファイル数が多い場合には、階層構造による整理とファイルアクセス作業時間が入れ替わる前に、階層構造による整理に移行すればよいというアドバイスを提案できる。そのさいにユーザは本手法によって作成されたビューを参考にして階層構造を構築でき、最初から構築するよりも短時間で整理できる。したがって本手法は、ユーザの労力を減らし、ファイルの使用頻度に応じて効果的な整理が実現できる。

5. 関連研究

本研究と関連のある研究として俺デスク [3], time-machine computing (TMC) [4] が挙げられる。本研究とこれらの研究との比較を示す。

俺デスクは、ユーザが計算機上で過去に操作したデー

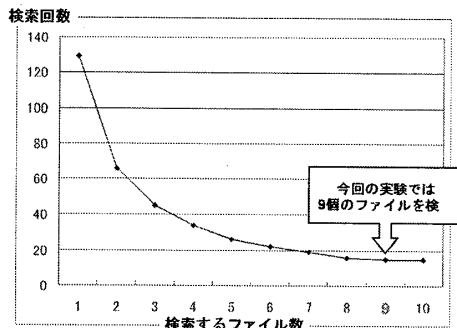


図 6: 検索するファイル数とファイルアクセス作業時間入れ替わり点との関係

タの想起を効率化するために、ユーザのデータに対する着目度と個々のデータ間の関連度を基にデータ群を時系列で整理することで情報想起を促すツールである。一方、Time-Machine Computing (TMC) [2] は、デスクトップの状態を時系列で記録しておき、任意の時刻のデスクトップの状態を復元可能にする。また、メーラーやスケジュール帳などのアプリケーションを使用した時刻も記録して連携させることで情報の想起を促す。これらの時系列による整理手法では、作業目的ごとには整理されておらず、複数の目的のための作業を並行して行っている場合は、不要なファイルなども併せて提示されていまい、必要なファイルをすぐに操作できない。加えて、どのタイミングで階層整理に移ればよいかを教えてくれるような機構もない。これに対して本手法では、ビューを用いて目的ごとにファイル群を整理し、操作履歴からビュー内のファイルを整理するので、ユーザは目的に応じて必要なファイル群をすぐに操作できる。時系列による整理がなされていないこと、ビュー内のファイル数に比例して検索時間がかかることが問題点としてあげられる。

6. おわりに

本論文では、ユーザのファイルアクセスのための労力を減らし、作業の効率化を実現するために、ファイルの整理と検索のための時間を考慮した整理手法を提案した。

本手法に基づく操作履歴による整理法を実装し、階層構造による整理との整理に要する時間と検索に要する時間の比較実験を行った。検索回数130回で両検索法の整理でのファイルアクセスのための作業時間が入れ替わり、一時的に使用するファイルでは、操作履歴による整理を用いることで、ファイルアクセスのための労力を削減することができた。

今後は、階層構造支援についての研究を進め、本研究の有効性を高めたいと考えている。

参考文献

- [1] Fertig, S., Freeman, E. and Gelernter, D, ACM SIGCHI Bulletin, Vol.28, No.1, Jan. 1996.
- [2] Nardi, B. and Barreau, D. K., ACM SIGCHI Bulletin, Vol.29, No.1, Jan. 1997.
- [3] Ohsawa, R., Takashio, K. and Tokuda, H., IEEE Eighth International Symposium on Multimedia (ISM'06), Vol. 12, pp. 762-765, 2006.
- [4] Rekimoto J., In Proceedings of the ACM Symposium on User Interface Software and Technology (UIST'99), pp. 45-54. ACM Press, Nov. 1999.