

縦空間・レイヤー融合構造直感的楽曲ファイルブラウザ —視覚化されたタグ関連による新しいファイル管理とブラウジング体験—

1. 背景

PC においてファイル管理を行う際には、現状では様々な問題点が存在する。これまで個人利用のPC内でのファイル管理は、フォルダを階層状に作成し、その中にファイルを格納することで分類を行うという方法が主流であった。しかしこれは、管理する場所が一意なものになるという性質がある。そのため一度格納場所を忘れてしまうと、探索に時間を要する。検索機能を利用する場合においても、ファイル名などを記憶していなければならないなど不都合が生じる。ある特定のファイルやそれを纏めた最小限のフォルダを探し出す際に、人為的な探索経路は複数あって然るべきである。なぜなら、ユーザがある記憶を辿る行為は常に一定とは限らないからである。また、ファイルを管理するためにフォルダ階層を作る行為自体、ファイルを含まない、分類のためだけの無駄な階層構造を量産することになる。その階層構造自体、またその内部をを順に辿るという操作方法の双方ともが非効率的である。

このような、従来のUNIXディレクトリ構造におけるファイル管理の不都合に対して、その枠組みの中でとられている対策は、その代表として以下のようなものがある。

- i. アプリケーション毎に扱うファイルを決めてしまい、ファイルの管理はそれぞれのアプリケーション内でのみ行うことを推進する方法。
- ii. それぞれのファイル自身に、タグと呼ばれる分類記号を附属させ、このタグをもとに分類、管理を行う方法。

前者 i の方法は、音楽なら iTunes、写真なら iPhoto というように、Apple (iTunes、iPhoto など) や Adobe (Bridge など) のアプリケーションにおいて近年顕著である。しかし、管理したい単位が音楽単体や画像単体では無い場合についての分類、管理方法については疑問が残る。また、大量の音楽や画像をブラウジングし目的のファイルを探し出す際には、ファイルに附属する属性値ごとに並び替えを行うという方法がとられる。この並び替えの弱点である、例えば複合的な条件をもとに探索を行いたい場合、単一属性による並び替えではそれに対応できないという点は改善の余地がある。またあくまで並び替えなので、目的のファイルに瞬時にアクセスできるとは言い難い。逆にメリットとしては、管理の体系がアプリケーションごとというある程度分散された環境であることから、管理の為だけの無駄なフォルダ構造を作成しなくとも良いということがある。

後者 ii は近年ウェブアプリケーション、特にブログサービスなどにおいて顕著に見られた管理方法である。タグによる管理のメリットとしては以下がある。ファイルやコンテンツ自体に附属されたタグを元に分類を行うという性質のため、ファイルの実体が実際のディレクトリ位置を問題としないこと。タグは一つのファイルに任意の個数附属させることが出来るため、そのファイルにアクセスする方法が一通りに限られないこと。これらの利点は、固定的なディレクトリ管理から脱却する一つの方法である。しかし現状では問題点もある。まず、タグは任意に何通りでも登録することが可能であるが、それ故に、あるファイルをどのようなタグで管理していたかということ忘れてしまった場合には不都合が生じる。さらに、タグというのはそれぞれが独立した情報であるため、体系的な管理には労力を要する。体系的に管理する場合もそうでない場合も、ファイルに複数のタグを登録する際には手間が掛かる。タグ登録インタフェースの現状は、ファイル一つ一つに、分割されたトークンとして文字を打ち込みながらタグを登録するというものである。このように、分類登録後には、探索時に複数の方法を提供できるというメリットのあるタグによる

管理も、登録の際に着目すると、フォルダに対し一纏めに格納していたこれまでと比較し必要以上に手順が煩雑であると言える。ウェブサービスを個人利用する程度のコンテンツ数では問題ない場合もあるが、大量の画像や音声、プロジェクトファイルを管理する PC での分類作業において、この手間は膨大である。PC 内でのファイル管理方法としてタグが浸透していない理由には、この点が重要な位置を占めていると想像できる。またタグによる情報の探索では、階層型管理の利点でもある、経路途中における当初の目的ファイル以外の条件該当ファイルの発見という要素は消滅している。

また例外として、TRON における実身、仮身システムに代表される特異なファイルシステム等の調査も行ったが、この方法が個人レベルの運用において浸透するには相当の期間を要すると判断、従来の UNIX 方式でのファイルシステム、ファイル管理システムを根幹とした管理体系を改善する方針をとった。

以上により本プロジェクトでは、ディレクトリによる管理、タグによる管理の双方のメリットを十分に引き出し、またこれにより双方の問題を最小化するに足り得る、ファイル管理の方法とその UI の考案、およびその実装を行う。

2. 目的

本プロジェクトにおいて目標とするのは、現状の OS 標準ファイルブラウザにおけるファイル探索、管理方法と、そのインターフェースの新規な置き換えである。よって実装形態は、ファイルシステムにアクセスしこれを視覚化、管理する GUI アプリケーションとなる。

従来のディレクトリ階層型管理の特徴であるツリー型と、ファイル自体にメタ情報を格納し、それをもとに管理、参照を行うタグ型の特徴をハイブリッド化を行う。ツリー型の多重条件参照における難点に対しては、タグ型の特徴を取り入れることによりカバーしつつ、タグ型の難点である体系的な管理に対しては、タグをツリー状に管理するという方法によりカバーする。

3. 開発の内容

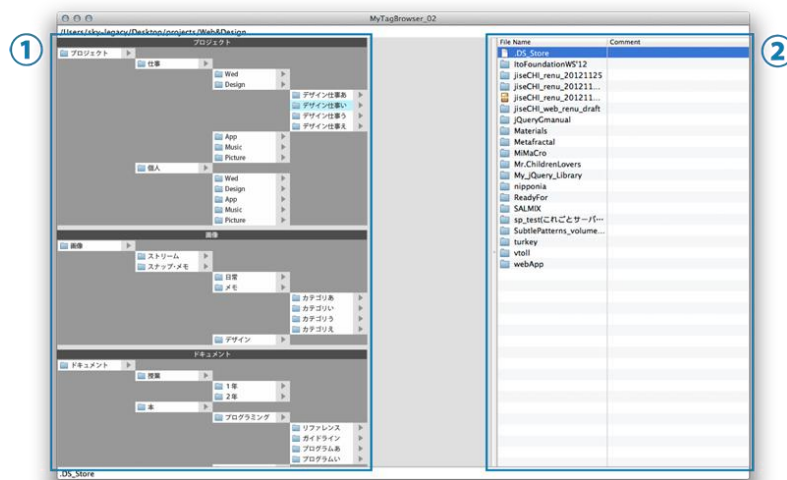


図 1 開発したシステムのインターフェース操作手順

- 基本は 2 分割画面。左側①がツリー表示、右側②が一覧表示。
- ツリー表示は、左上端からノードが開始される。ノード右端の展開ボタンをクリックまたはタップすることで、その右下に子ノード 1 階層分が展開される。それぞれのツリ

- 一で展開を繰り返すと図の①のようになる。
- ツリー表示は展開を繰り返すことで自動的に描画領域を拡張する。
- このツリー表示を用いて通常のファイルシステムを辿る方法に加え、任意に「タグツリー」を作成可能。
- 通常のファイルシステムのツリー表示と、タグツリー表示は、同じ画面座標領域内においてどちらか一方に切り替えて使用する。切り替えは切り替えボタン(図では未実装)、Tab キー、スワイプ操作などで行う。

- 通常のファイルシステムのツリーを用いて探索を行う場合
 - 探索行為はツリー表示領域①で行う。選択したノードの内容が一覧表示領域②に描画される。
 - 任意の位置をルートとして登録、また展開状態の記憶が可能。
- タグツリーを用いて分類や探索を行う場合
 - タグツリーの特性
 - ◇ タグツリーでは、一つのノードが一つのタグを表す。
 - ◇ タグ同士には親子関係などが保持させる。各々のタグノードは、その親子関係に基づき、UNIX ファイルシステムのように親から連なる一意なパスを持つ。これをタグパスと呼ぶ。
 - ◇ タグは、ファイルやフォルダ(以降アイテムとする)に付加される情報である。
 - ◇ 任意のアイテムは、任意のタグパス情報を複数個同時に保持することができる。
 - タグツリーによる分類の方法
 - ◇ タグツリーに、任意の分類用ノードを作成、または新しいタグツリーを作りそれを繰り返す。新しいノードやツリーは、それぞれ任意のノード、任意のタイミングで作成が可能。
 - ◇ ファイルシステムツリーを使用するなどし、予め②の領域に、分類したいアイテムを含むディレクトリを表示しておく。
 - ◇ 分類したいアイテムを、②の領域やその他の場所から、作成済みタグツリーの任意ノードにドラッグアンドドロップする。
 - ◇ アイテムがノードにドロップされると、そのノードが持つタグパス情報が、アイテムの拡張属性に書き込まれる。
 - ◇ この操作以降、タグツリーのノードをクリックし選択した際には、その情報を持つ該当アイテムが②に表示される。
 - ◇ つまり、ドロップされたアイテムは、あたかもタグツリーのそのノードに格納されたように振る舞う。
 - ◇ アイテムは、任意の複数ノードへドラッグアンドドロップが可能。その際アイテムは複数のタグパス情報を持つ。
 - ◇ それぞれのタグツリーのルートには、プロジェクト、文書、画像、音楽・音声、などのように、プリミティブなものを指定。
 - ◇ 例えば、ある複数のプロジェクトのフォルダ群を、タグツリーの「プロジェクト>仕事>デザイン」の位置にある「デザイン」ノードに D&D すると、それらのフォルダは「プロジェクト>仕事>デザイン」という分類に格納されたことになる。
 - ◇ 前述の分類操作のあと、このフォルダ群は、単に「プロジェクト」のノードをクリックしただけでも表示される。さらに、「プロジェクト>仕事」のノード、「プロジェクト>仕事>デザイン」のノードを選択した場合でも表示される。

- ◇ まとめると、アイテムがあるノードに分類された場合、それらはルートノードからそのノードまでの道筋の間においてのみ表示されるということ。
- ◇ 「デザインあ」「デザインい」というアイテムを、「プロジェクト>仕事>デザイン」ノードに格納、また、「アプリあ」「アプリい」アイテムを「プロジェクト>仕事>アプリ」ノードに格納、また、「ウェブあ」「ウェブい」アイテムを「プロジェクト>個人>ウェブ」ノードに格納した場合「プロジェクト」ノードを選択すると、②に、「デザインあ」「デザインい」「アプリあ」「アプリい」「ウェブあ」「ウェブい」が表示される。
- ◇ 「プロジェクト>仕事」ノード選択すると、②に、「デザインあ」「デザインい」「アプリあ」「アプリい」が表示される。
- ◇ 「プロジェクト>仕事>デザイン」ノードを選択すると、②に、「デザインあ」「デザインい」が表示される。

4. 従来の技術(または機能)との相違

本プロジェクトではあえて木構造を可視化したタグツリーを操作することにより分類を行うという手法を採用している。これは、スムーズな操作性を優先するためである。近年のドキュメント整理インターフェースでは、この表現方法とその直接操作による実装は、避けられる傾向にあった。これは、描画領域の移動操作が煩雑になり、画面領域を占有するという理由からであると想像できる。今回これを敢えて採用し推進した理由として、近年一般個人レベルへの流通が浸透し始めた、マルチタッチインターフェースによる操作が挙げられる。タッチインターフェースの特徴は、画面内の要素の選択と、画面表示範囲の移動が単一の物理的インターフェースにより統合されていることにある。これにより、これまでマウスのように選択はボタン、画面移動はスクロールホイールまたはスクロールバーを掴んでというように、ハードとソフト双方において分散されていた操作が、タッチインターフェースでは、ハードとソフト双方で操作の統一が行われている点が重要である。これは、細やかな点ではあるが降り積もれば無駄となるUIの操作を、これまで以上に省略し直感的たらしめている。このようなインターフェースが一般的な環境として浸透しつつあり、ユビキタスの環境においても今後更に推進されるであろう現在においては、これまでとは逆に率直な視覚化による操作効率の向上を見込むことが可能である。更に、空中でのジェスチャを検出可能である技術の実現により、現在以降においてこの傾向は更に強まると判断したことも理由の一つである。

本UIの最大の特徴は、タグのツリーを視覚的に表現した探索領域と、そのノード選択による該当ファイルの一覧表示という点にある。この特徴により、階層的な管理体系を持ちながら階層探索時の視界を開いたまま確保、さらに複数タグの付与、複数条件によるコンテンツ探索を、一つのUIによって提供することに成功している。これは従来のファイルマネージャには存在しない操作方法、操作環境である。タグツリーにおける管理の中で、ユーザはコンテンツの最小単位を、そのコンテンツごとに適切な単位に自然と限定することが自然に促される。また、コンテンツの実体は、画像、本、プロジェクトなどの最大限の大きさの枠組みごとにそれぞれ一つのディレクトリに保存しておくことが可能となる。これにより、コンテンツを実際に利用する際そのHFSパスが要求されるような場合においても、簡潔なパスを提示することが可能であり、またPC更新時などの可搬性においても優れる。

今回、タグツリーによるファイル管理と探索の根幹となる機能を実装したアプリケーションを作成したが、実際に筆者が使用した感想としては、是非既存のOS附属ファイルブラウザの代替として使用したい使用感であった。

5. 期待される効果

今後例えば個人や組織が本アプリケーションを導入することにより、コンテンツ、ドキュメント管理とその取り出しにおいて、大幅な負担軽減を望むことが可能である。細やかな機能は今後拡張する予定だが、本プロジェクトの形式、つまりメタデータをメインとしたファイル管理とその関連視覚化、それによる非常に簡易で高速なメタデータ付与とファイル探索は、今後のスタンダードとなるべきであると感じた。

6. 普及(または活用)の見通し

先述のように、本プロジェクトによるコンテンツ管理の形式は、特に個人利用のユビキタス環境において、タッチ、空間ジェスチャデバイスとの相互な普及が望まれる。

7. クリエータ名(所属)

鶴田真也(筑波大学情報学群情報メディア創成学類、株式会社ジセカイ)