

家計簿の管理や組織構造の作成には、一般に表計算ソフトやデータベース・ソフトが利用されています。その場合、データをアプリケーション内でたんなる数字として扱うのではなく、個々のデータ項目をファイルシステムのファイルというかたちで表現すれば、大規模な階層的データをうまく扱える可能性があります。

そこで、今回はデータ項目をファイルで表し、階層型データベースとして活用する方法を考えてみました。

ファイルシステムと階層的データ

階層的な構造をもつデータについて計算をおこなったり、全体の構造を直感的に把握したい場合がよくあります。身近なところでは、ディスクの使用状況を調べたり、ファイルを検索したりといった例が思い浮かびます。家計簿、名簿、組織構造、書誌データ、予算など、多くのデータは階層的に表現できます。また、国家予算、図書館の蔵書目録、地理情報データ、ディレクトリ・サービスのデータのような規模の大きいものも、かならずといってよいほどなんらかの階層構造で表現されています。したがって、階層構造を直感的に把握できる視覚化システムの応用範囲はかなり広いと思われます。

大規模な階層構造に対応した情報視覚化システムは、以前からいろいろと提案されています。しかし、Yahoo!ディレクトリや国家予算のようなものをうまく視覚化した例はあまりみられません¹。一方、UNIX や Windows などで利用されている階層的なファイルシステムについては、

ディスク領域の使用状況を視覚化したり、ファイルを管理するソフトウェアがたくさんあります。ということは、国家予算や図書館の蔵書目録などの大規模な階層的データをファイルシステムおよびファイルとして表現すれば、ディスク領域を視覚化したり、ファイルを管理するツールをそのままデータベースの管理に活かそうです。

多くの表計算ソフトや家計簿ソフトには、内訳を円グラフなどで表す機能がありますが、深い階層をもつデータベースをうまく視覚化できるものはあまりありません。これらのソフトウェアの代わりに高度なディスク領域視覚化ツールを使えば、予算の内訳などがより細かく把握できるはずです。

データ項目をファイルとして表す方式には、階層構造の視覚化以外に次のような利点があります。

- 家計簿ソフトなどで扱う情報はたんなる文字や数字なので、数字の内容にあまり実感が湧かず、“1,000 円”も“100,000 円”も大差ないように感じられてしまいます。しかし、これらが 1MB のファイルと 100MB のファイルとして表現されていれば、中身を眺めたりコピーしたりするときに相応の時間がかかるため、相違が直感的に認識しやすくなります。ファイルにはタイムスタンプが付いているので項目の新旧も分かりますし、拡張子や内容による区別も可能なので、たんなる数字や文字による表現とくらべて、はるかに多くの情報を伝えることができます。
- 表計算ソフトなどを使う場合は、カット&ペーストなどによってデータを生成したり移動したりします。一方、データ項目がファイルとして表現されていれば、ls や mv、find といった馴染みのある UNIX コマンドが使

¹ 国家予算などは、誰もが一目で分かるように視覚化されると、困る人が出るのかも知れませんが。

えるので、UNIX ユーザーなら操作方法にあまり悩まないでしょうし、操作ミスも減らせます。

cd や find など、階層的なディレクトリ構造を扱うコマンドはたくさんありますから、これらを活用してデータの管理ができます。さらに、GUI を使えばディレクトリ間でのドラッグ&ドロップによる項目の移動も可能です。あるいは、tar などを利用することで、バックアップ作業も直感的におこなえます。さらに、作成日時やファイルの内容などの属性を活用すれば、もっと複雑な処理も簡単にこなせるようになるでしょう。

- 家計簿をつける場合、通常は専用のアプリケーションや表計算ソフトを使います。もちろん、ごく基本的な使い方しかしないのならそれで十分ですが、ちょっと凝ったことをしようとする専用アプリケーションなどでは対応できなくなる可能性があります。

一般的な家計簿ソフトでは、あらかじめ用意されている処理しかおこなえません。また、データベースや表計算ソフトの操作には、そのシステム専用の言語を使うのが普通です。これに対し、データがファイルで表現されていれば、検索や各種の処理をおこなうプログラムを C や Perl、Ruby、Java などの任意の言語で書くことができます。

データを自動入力する場合、データベースや家計簿ソフトの API を利用する必要がありますが、ファイルとして表現されていれば、新しいファイルを作成するだけです。

- 関係データベースや表計算ソフトは“表”の扱いやすさに重点を置いているため、階層的なデータの表現に不向きなこともあります。階層型データベースはそれほどポピュラーではないので、ファイルシステムのほうが階層的なデータを扱いやすい場合も多いでしょう。

以下では、既存のファイル視覚化システムを用いて大規模な階層データの構造を把握する方法を考えてみましょう。

階層的情報の視覚化

階層構造の視覚化については、これまでにさまざまな手法が考案されています。

TreeView

TreeView は、Windows のエクスプローラをはじめと

図 1 TreeView



する多くのシステムで利用されている階層情報視覚化ツールです。現在の GUI における、もっともポピュラーな階層構造視覚化ツールといえるかもしれません。TreeView では、階層構造の可視/非可視を“+”と“-”の 2 つのボタンで切り替え、全体を俯瞰したり、詳細を調べたりすることができます。図 1 は、地図アプリケーションで使われている TreeView の例です。

TreeView は、さまざまなツールキットで手軽に扱える点はいいとしても、たくさんの + や - を操作するのはやはり面倒です。また、構造は捉えやすくて、全体のサイズなどは把握しにくいという問題があります。

Hyperbolic Tree

Xerox PARC²で開発された Hyperbolic Tree は、非ユークリッド空間である“Hyperbolic Space”に配置した情報を 2 次元空間にマップすることにより、大きな階層データを 1 つの円のなかに表示するシステムです。中心から遠いノードほど小さく表示されるため、原理的には無限のデータを表現することができます(図 2)。Hyperbolic Tree を含む各種の視覚化システムが、InXight³という会社で製品化されています。

Hyperbolic Tree では階層構造をすべて円のなかに入れて眺められるので、全体的な構造の俯瞰やリンクをたどりながらの検索などには便利ですが、ディスクの残り容量を直感的に把握したいといった用途には不向きです。

TreeMap

メリーランド大学で開発された TreeMap⁴は、画面を

脚注 4: URL を追加しました

2 現在は Xerox から独立し、Palo Alto Research Center (<http://www.parc.xerox.com/>) という会社になっています。

図 2 Hyperbolic Tree

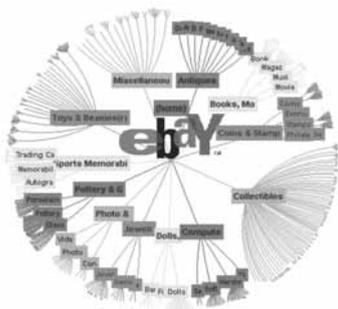
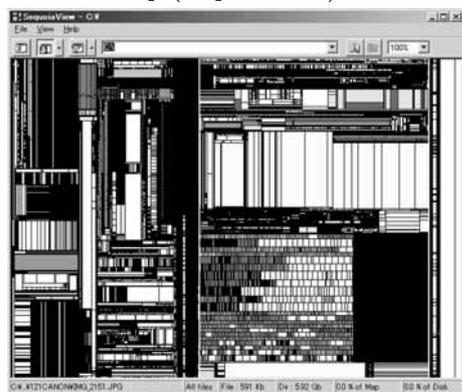


図 3 TreeMap (SequoiaView)



縦横に分割して情報の階層構造を表現する視覚化手法です。マッピングやレイアウトのアルゴリズム自体は単純で、データのサイズや重要度を矩形の大小によって表現するだけです。この方針にもとづき、階層のトップから順番に矩形を縦横に分割していくことにより、階層データ全体が1つの大きな矩形のなかに表示されます。

図 3 は、オランダのイントホーヘン工科大学で開発された SequoiaView⁵の画面です。これは TreeMap を Windows 上で実装したもので、ディスク上にある大きなファイルが一目で分かります。

TreeMap は、データの大きさは分かりやすいのですが、階層構造の把握は TreeView や Hyperbolic Tree の場合よりも難しいかもしれません。

同心円視覚化

最近、同心円状にディスク容量を表示する OverDisk⁶や

3 <http://www.inxight.com/>

4 <http://www.cs.umd.edu/hcil/treemap/>

5 <http://www.win.tue.nl/sequoiaview/>

図 4 OverDisk

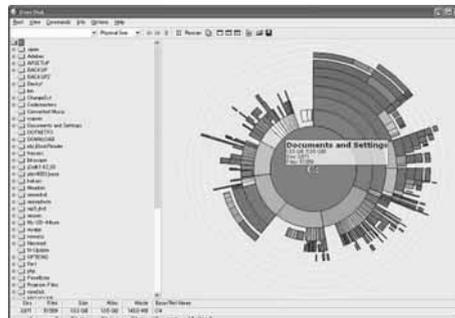
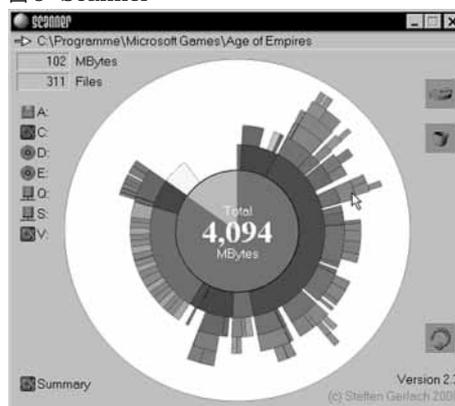


図 5 Scanner



Scanner⁷というシステムが注目を集めています。

階層構造のトップを中心とする同心円で全体的な構造を表現し、部分階層のサイズに応じて扇形に分割して内容を表現します。Hyperbolic Tree や TreeMap などと比較すると、より直感的に構造が把握できるように思います。

現在のところ、これらの視覚化システムの多くはそれほど普及していないようです。ただし、SequoiaView や OverDisk、Scanner などのディスク領域管理ツールについては、フリーの Windows 版が公開されているので手軽に利用できます。

ファイルシステムとの対応づけ

階層的なデータをファイルシステムに対応づければ、ここまでで紹介したツールがほぼそのまま使えます。この方

6 <http://users.forthnet.gr/pat/efotinis/programs/overdisk.html>

7 <http://www.steffengerlach.de/freeware/>

表 1 2004(平成 16)年度一般会計歳出の内訳(一部)

社会保障関係費	
生活保護費	1,748,858
社会福祉費	1,633,871
社会保険費	15,380,234
保健衛生対策費	503,351
失業対策費	530,691
文教および科学振興費	
義務教育費国庫負担金	2,512,846
国立学校特別会計へ繰入	0
科学技術振興費	1,284,115
文教施設費	144,261
教育振興助成費	2,057,249
育英事業費	134,571
.....	

(単位: 100 万円)

式は、とくにデータの規模が大きい場合に効果的です。

国家予算の視覚化

日本の国家予算は約 80 兆円ですが、このような莫大な金額も、階層的に表現すれば全体像が把握しやすくなります。

財務省の Web ページで公開されているデータ⁸によると、2004(平成 16)年度一般会計歳出の内訳は表 1 のようになっています。

これらの数字を実感するのは難しいと思いますが、金額と同じバイト数のファイルを作成して Scanner などでも閲覧すれば、どの項目にどのくらいのお金が支出されているかが一目で分かります。たとえば、「社会保障関係費」というディレクトリの下に、サイズが 1,748,858 バイトの「生活保護費」、1,633,871 バイトの「社会福祉費」、……といったファイルを作り、全体を Scanner で閲覧すると図 6 のように表示されます。80 兆円の歳出のうち、国債費がかなりの割合を占めているのが一目瞭然ですし、防衛費が 5 兆円であり、社会保障関係費が 20 兆円にもなることがすぐに分かります。社会保険にかかわる歳出の内訳を詳しく知りたければ、「社会保険費」ファイルディレクトリに変更し、その下で細かな内訳をファイルとして表現すればよいでしょう。

Yahoo!のディレクトリ構造の視覚化

図 7 は、数年前の Yahoo!のディレクトリ構造をファイルで表現し、それを Scanner で視覚化したものです。こ

⁸ <http://www.mof.go.jp/seifuan16/yosan004.pdf>

図 6 歳出割合の視覚化

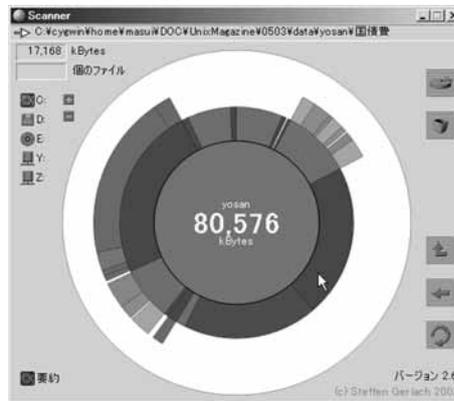


図 7 Yahoo!のトップカテゴリ

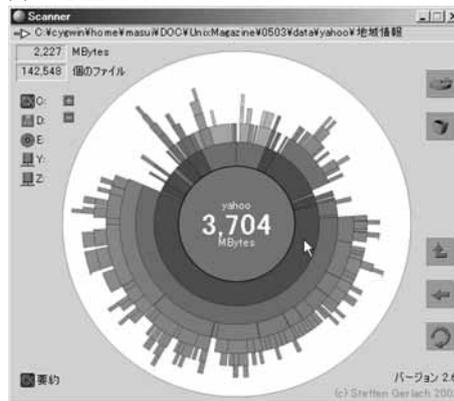
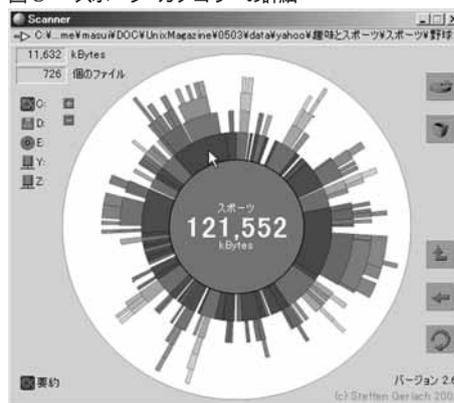


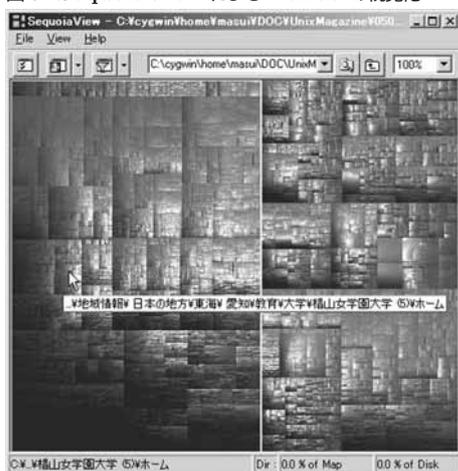
図 8 「スポーツ」カテゴリの詳細



の時点で、Yahoo!のエントリの半分以上が「地域情報」カテゴリに属していることが分かります。

図 8 は、図 7 の「趣味とスポーツ」カテゴリを選びさらに「スポーツ」カテゴリを選択してその内訳を表示

図9 SequoiaView による Yahoo!の視覚化



したものです。カーソル位置にある“野球”も大きな割合を占めていますが、右側のサッカーのほうがより大きな領域となっています。

一方、SequoiaView を用いて Yahoo!の情報を視覚化すると図9のようになります。数十万件の項目が1画面に表示されているので見にくいのですが、この場合も半分以上が地域情報であることなどが分かります。

このような視覚化システムと ls、mv、findなどを組み合わせれば、大規模な階層データでも構造を直感的に把握しながら操作できるはずです。

おわりに

すこし前までは、金額の多寡を示すためだけに大きなファイルを作るなどという考えはバカげていると思われたでしょう。しかし、最近は大容量のディスクが安価になったので、今回のようにファイルのサイズでデータを表現する方法も十分に検討する余地があると思います。皆さんも、家計簿の管理などに応用してみてもいいかもしれません。

(ますい・としゆき 産業技術総合研究所)