インターフェイスの街角 (91) - 並べる!技術 増井俊之

並べ替えの手間

文章やメモなどを新しく作成するとき、情報を分類しな がら追加したり、編集することがよくあります。アイデア やメモをまとめるときには、新しい情報は分類しながら追 加していくとうまく扱えますし、役に立つ Web ページ のリンク集を作るような場合も、内容別に分類しながらリ ンクを追加していくことが多いでしょう。買い物リストや TODO リストなどのちょっとしたメモはもちろんですが、 プレゼンテーションの資料や原稿などの比較的長いものも、 たいていは内容をすこしずつ考えながら書き足していくの ではないでしょうか。

そのような場合、編集する情報の量が少ないうちは並べ 替えもたいして苦になならないので、普通のテキストエデ ィタを利用すればよいでしょう。しかし、データの量が多 くなってくると、並べ替えや編集に手間がかかるようにな り、単純なテキストエディタでは効率的な編集ができなく なります。

編集作業では、関連する情報を近くに置いたり、なんら かの順番でソートしたり、あるいは分類の階層構造を修正 したりといった操作が必要になります。しかし、大きな文 書を扱う場合、単純なテキストエディタでは効率が極端に 落ちてしまいます。たとえば、2つの大きなブロックの順 番を入れ替えるとしましょう。そのようなときは、一方の ブロックの先頭と末尾の位置を指定してコピーし、その後 にもう一方のブロックの場所に移動して貼り付けたりしま す。この方式では、コピーする領域や貼り付ける場所を間 違えたり、貼付け操作そのものを失敗する可能性があるの で、手軽にというわけにはいきません。

テキストの全体像を把握しながら移動などの操作をおこ なうには、全体を縮小表示したり、なんらかのフィルタで 内容を絞り込んだり、データを階層的に管理するといった 工夫が必要です。そんなとき、階層的な情報を並べ替えな がら、入力や編集を簡単におこなえるツールがあれば便利 そうです。

大規模な情報を管理する場合、階層的な構造を利用する のが普通ですし、計算機上でこの種の情報を整理するツー ルもいろいろとあります。UNIX や Windows のファイ ルシステムは階層構造ですから、ファイル管理ツールで階 層型データが扱えます。2005 年 3 月号で紹介したような 方法を使えば、これらのツールをデータ管理に応用するこ とも可能でしょう。メニューや設定ファイルなども、 階層 的データ構造を前提としたつくりになっています。Windows では TreeView という GUI 部品で階層情報を管 理することが多いようですが、その他の環境でも同様の方 法で情報管理ができます。

このように、ファイルやメニューの管理については、階層 的データ構造を前提としたツールがひろく使われています。 ところが、通常のメモやテキストの編集では、データの追 加や階層的並べ替えを補助する方法はあまり利用されてい ないようです。たいかに、テキストを階層的に扱う "アウ トライン・エディタ"があり、Emacs や MS Word に も "アウトライン・モード"があります。しかし、これら を十分に活用している人はあまりみかけません。たいてい は、コピーや貼付けなどの一般的な編集コマンドで間に合 わせているのが実情のようです。

Emacs のアウトライン・モードの機能は、テキストを

UNIX MAGAZINE 2005.10

1

階層化し、それらのレベルの表示/非表示を切り替えるのが 基本です。しかし、階層のレベルの変更や、ブロックの並 べ替えがとくに簡単にできるわけではありません。このあ たりが、活用している人が少ない理由かもしれません。

ちょっとしたメモやテキストの作成や編集でも、分類や 並べ替え操作は頻繁におこなわれます。ですから、簡単な 操作でテキストの追加や階層化、並べ替えができるシステ ムがあれば便利でしょう。メモ、名簿、予定表、TODOリ スト、ブックマークなどでは、追加や編集、並べ替えが頻 繁におこなわれるので、このようなシステムは有効だと思 います。もちろん、大規模なデータの場合も、階層的な管 理や並べ替え操作が簡単にできれば扱いが楽になります。

こういったことを考えると、規模の大小にかかわらず、 手軽に使えて有効な編集/並べ替えの技術が求められてい るように思います。

並べるシステム

そこで、ブラウザ上で手軽にデータを編集するために、 Wiki で使える入力/編集システムを試作してみました。

私は、TODO リストや予定表、ブックマークなどをす べて 2001 年 11 月号で紹介した Wiki システムで管理し ています。そこでまず、その上でデータの追加や編集を簡 単におこなう方法を試してみました。これは、以下のよう な機能の実現を目指したものです。

単純な操作でデータの追加や並べ替えができる
矢印キーを使い、選択したテキストの位置や階層を移動したり、表示する階層レベルを変化させたりします。

●編集モードや書込みボタンを不要にする

Wiki システムの大きな欠点は、編集モードと閲覧モー ドが分かれていることにあります。そのため、編集に手 間がかかり、編集結果も書込みボタンを押さなければ反 映されません。

NOTA¹などの先進的な Wiki では WYSIWYG 編 集や自動書込みも可能ですが、テキストベースの従来型 Wiki の大半は、編集モードと書込みボタンをベースに しています。今回のシステムでは、編集をできるだけ簡 略化しつつ自動書込みをおこなうことにします。

1 http://rakusai.org/nota/

図1 Wiki ページ画面



ズーミングとフィルタリングを活用する
以前に、大規模なデータを効率よく検索するためのシステムとして LensBar(2000 年 2 月号)や Q-Pocket
(2000 年 5 月号)を紹介しましたが、Wikiページ上でもこれらの手法が使えるようにします。

動作例

Wiki 掲示板システムを JavaScript で拡張し、上記の ような特徴をもつシステムを作ってみました。このシステ ムの概要を紹介します。

簡単な編集と保存

図1は、 ^{*}並べる!技術^{*} についての情報を Wiki ページ 上で閲覧しているところです。

通常の Wiki ページでは、[編集] ボタンを押して図 2 のような編集モードに移行し、テキストの編集を終えたら、 [Write] ボタンを押して書込みをおこないます。しかし、ち ょっとした編集のために、これらのボタンをいちいち操作 するのは面倒です。

今回のシステムでは、ボタン操作で編集モードに移行す る代わりに、テキストをクリックするとその行が編集可能 になるようにしてみました。たとえば、図2で^{*}既存シス テムの問題点[~]の行をクリックすると、画面が図3のよう に変化してこの行が編集可能になります。

ここで、"大きな"という文字列を追加すると、画面は図 4 のように変わります。モノクロなので分かりにくいかも しれませんが、テキストの編集中は文字入力枠の背景色が



図2 Wiki ページ編集画面

◎基本記状論 = Motille Firefox ファイルの 編集(0) 表示(2) (移動(2) ファクマーク(3) フール(1) ヘルブ(4)	-10/2
並べる!技術	·2014年 王 第四十一覧
	1

図 3 Wiki ページ編集画面

⑦並べ記技術 - Mozil	la Firetox				_101 ×
7ヶイル田 編集田 日	表示公 移動回	ブックマーク(日)	ウール田	人为700	00
並べる!技	術			22.44	▲ <u>→</u> 双型 編集 一致
Search					
既存システムの間鐘点					
アウトライン 補助ななどです 声別のコネスカより 書通のテネストエ・ のけなけの点 イングリンンの加 しensBac, O-Poda 適応とニュー アウトラインゴー 自動セージTenace 自動セージTenace 通用塩所 ブレゼンデーション TODO ジックマーク 本種ore	エディタはあるがト ドキスト人力で優好 ドネスト人力で優好 に が高いたい ディクロノトの称 ドチクロノトの称 ドチクロノトの称 ドチクロノトの称 ドチクロノトの称 ドチクロノトの称 ドチクロノトの称 ド オート パロ イ 注 料	たど 差したらくことはよく 大学事なのたわらう 動などを行ちなうの ない	あ <i>るのだが3</i> ブラウザ上で 11結構差の1	E金/ニフリーフォーイ 岳 べかえなどの編 単い	わったて入力するのノス変変 制作でをサポートすると除しい

図 4 テキスト編集中



変化します。編集結果は、2005年5月号で紹介したAjax の手法を用いて随時サーバーに送られ、保存されます。保 存されると、文字入力枠の背景色が通常の白に戻ります。

テキスト以外の場所をクリックすると文字入力枠が消 え、編集後の Wiki ページが表示されます(図5)

図 5 編集後の Wiki ページ

이불<장扶衛 - Mozilla Firefox	
ファイル(1) 編集(1) 表示(1) 移動(1) ブックマーク(1) ツール(1) ヘルブ(1)	0
並べる!技術	
122-14	1 川田 山田 一覧
earch	
<u>ま存システムの大きな問題点</u>	
リストを簡単に語うことを目的にしたシステムは少ない	
Wikiなどでテキスト人力や爆発したいことはよくあるのだが完全にフリーフォーマ	ットで入力するのは原例
実際コは入力よりも編集操作の方が大事なのだからブラウザ上で基べかえなどの編集	操作をサポートすると嬉しい
普通のテキストエディタでリストの移動などを行なうのは結構面倒臭い	
WY版学が必要なので。 ・Cアグリメンタル検索を融合されていない	
既存技術	
LensBar, Q-Pocket	
通応メニュー	
アウトラインプロセッサ	
自動セーブEmacs	
ab (13 44) 7ac	
20111-10171	
間田福田 ゴレゼンテーション資料	
間田 11回加 ゴレゼンテーション資料 TODO	
週田 編加 ゴレゼンテーション資料 TODO ジックマーク	
1997日1997) フレゼンテーション次回 TODO ジックマーク 本種のを	

図6 項目の移動



このように、該当箇所の文字列をクリックするだけでテ キストが編集でき、しかも自動的に保存されるため、Wiki ページ上での編集が手軽におこなえるようになりました。

項目の移動

選択した行とその下の階層の行とのあいだは、Shift キ ーを押しながら上下の矢印キーを押すことで移動できます。 たとえば、図 5 の状態で "既存システムの大きな問題点" をクリックし、続けて Shift キーを押しながら下向き矢印 キーを押すと、この階層以下に含まれる行がまとめて下に 移動します(図 6)

項目の追加

Shift キーを押しながら行をクリックするか、編集中に Enter キーを押すと、図 7 のように新しい行(項目)が追 加されます。

UNIX MAGAZINE 2005.10

3

9日本3日 7元(14回	橋 - Moz 副版(E)	illa Fire 表示(2)	lox 1946(Q)	ブックマーク国)	7-4D	人物力的	,	- <u>n</u>
並べ	くる店	大術					「204 48 」 成型 編集 ー)	- -
Search		-						-
概存技速 LensB 通にメ アウト 自動t	lar, Q-Pock Cユー ラインプロt オープEmac	ut 277						
- 既存シス5 リスト 実際に 普通の 通用場所 ブルゼ TO00 ブック 本単の	テムの大 を開単に執 た に か の か か か か か か か か か か か か か	きな問題 たことを目 シエディタ デキスト、 バーターの の また の の の の の の の の の の の の の	増点 約コスたシ、 はあるけれ、 人力や爆集 作の方が少 フストの移動 で。 されていな	ステムはまかない だ したいことはよく て事なのたから: かよどを行なうの い	あ <i>るのだが</i> 5 デラウザ上で は結構面例	<i>日金にフリー</i> 基ペかえな 臭い	ンナーマットでんかするのは政府 どの編集時代をサポートすると知人い	1
28 I	頭の)畳み	込み					
08488	lif – Moz	illa Fire	1546(0)	ゴックスーク(名)	2-500	A \$ 700		-미지

図 9 インクリメンタルなフィルタリング

100 差不む技術 - Mazilla Firelox ファイル(在) 編集(2) 表示(公) 移動(3) ブックマーク(2) ウール(2) ヘルブ(4) -101×1 0 並べる!技術 |24.1488 三 単型 | 編集 | 一覧 Search Iso 実際コネスカよりも単築時代の方が大阪なのだからブラウザ上で並べかえなどの編集時代をサポートすると際し、 新人・データを追加する方法 ー 空デーが追加時代 具体的な操作法 図10 表示される階層レベルの制御 10日本な社員 - Mozilla Firetox ファイルの 編集の 表示の 移動の ブックマークの ツールの ヘルブの -101× 並べる!技術 |231488 三 川田 | 福美 | 一覧 Search Dou

NALEXIM 属在システムの大きな問題点 実際のコネストルも毎年時代の方が大事なのだからブラウザ上で並べかえなどの編集時作をサポートすると知しい 適用単正 間見が必要なな法 取し、ドールを加ませる。また。

10.110-50法 10.117-50歳加すら方法 - 空デージ液加制作 基本がなる操作法 都方式のメリット 超自己 超基

項目の畳込み表示

並べる!技術

Search

既存技術 既存システムの大きな問題さ

県存システムの大き 適用場所 開発が必要な技法 具体的な操作法 新方式のメリット 面白さ 拡張

左右の矢印キーを押すと、階層表示のレベルを切り替え ることができます。たとえば、図6の状態で左向き矢印キ ーを2回押すと、下の階層が隠れて、一番上の階層だけが 表示されます(図8)

項目のフィルタリング

Search の欄に検索文字列を入力すると、Migemo²を用 いたテキストのフィルタリングがおこなわれます。例に使 っているテキストに対して "sou"と入力した場合、sou(そ う)という読みをもつ単語(この例では、 "操作")を含む行 だけが表示されます(図9)

ここで左右の矢印キーを押すと、LensBar と同じ要領 で、検索文字列にマッチした行と、それよりも上のレベル の階層の行も表示できるようになります。

2 http://namazu.org/~satoru/migemo/

今度は、図 9 の状態で右矢印キーを押してみましょう。 すると、一番上の階層の行も一緒に表示されます(図10) このような手法は "Focus+Context" と呼ばれ 注目し ている行と文書全体の構造を同時に眺められる点に特徴が あります。

評価

|24.1488 三 単型 | 編集 | 一覧

この手法を導入した Wiki ページをしばらく使っていま すが、たいへん快適です。従来は、Wiki にデータを入力 したあとで並べ替えるのが面倒だったため、よほどのこと がなければ入力したままにしていました。しかし、このシ ステムの場合は、入力したデータの編集や並べ替えが簡単 なので、Wiki 上での編集作業を負担に感じることもあり ません。これまで、Wiki データを編集するときは編集モ ードに切り替えていましたが、並べ替えや編集が直接的に

UNIX MAGAZINE 2005.10

実行できるのはたいへん便利です(最近は、なぜデータを わざわざ手作業で保存していたのだろうと不思議に感じる ほどです)。

現在の実装では、編集をおこなうたびに新しい Wiki ペ ージが生成されてしまいます。これが気になる人もいる と思いますが、そもそもテキストデータが膨大なサイズに なることはあまりありませんし、なによりも編集時のログ がすべて残るという安心感があります。実装方法に関して は、まだまだ考慮の余地が残っているものの、大まかな方 針としては今回の手法はかなり正しい方向にあるように思 います。

おわりに

今回の方法は、単純なテキストデータの編集を前提としています。しかし、テキストは方法によってはいろいろに 解釈できるので、実質的に制限はほとんどないといっても よいでしょう。

私の Wiki ページでは、予定表をテキストデータとして 編集し、それをフィルタにかけてカレンダー形式で表示す るようにしています。このように、適当なフィルタに通し たあとで閲覧する方式であれば、直接には単純なテキスト しか編集できなくても、多様な見せ方が可能になります。

今回の方法を使えば、大規模な YAML³ などのデータも 扱えるはずです。 YAML のデータも簡単な形式のテキス トですが、 XML と同様に複雑なデータ構造が表現できま す。 XML データを手作業で編集するの大変ですが。 今回 のようなシステムと YAML を併用すれば、手作業による データの作成や処理も難しくないでしょう。

計算機上でおこなうさまざまな仕事のなかでも、データ の入力や編集はもっとも基本的で重要な機能です。このシ ステムを足掛かりとして、さらに便利な入力/編集手法を開 発しようと考えています。

(ますい・としゆき 産業技術総合研究所)

3 http://www.yaml.org/

UNIX MAGAZINE 2005.10

