

本とハイパーテキストを融合したユーザ指向メディア： 3 H-10 OpenBook*(2)

— ユーザ・インターフェース —

工藤 正人 市村 哲 荒井 恭一 岡田 謙一 松下 温†
慶應義塾大学†

1 はじめに

ハイパーテキストでは、データの管理形態としてn次元的な資源管理を行っているが、ユーザがその情報にアクセスする際には、リンクをたどってのシーケンシャルなアクセスしか出来ず自由度に乏しい。一方、本では、情報は基盤となるものから目的とするものまで一次元に管理されているが、拾い読みのようにランダムなアクセスが可能である。これらのことから、我々は情報の管理形態としてハイパーテキストを用い、その情報へのアクセス形態として本を用いることにした。そこで、ハイパーテキスト中から必要な情報を本の形態で示すことの出来るシステムを考え、Sun ワークステーション上で実装した。

2 基本コンセプト

2.1 インターフェースとハイパーテキスト

ハイパーテキスト・システムでは、情報は個別の小さな構成単位(ノード)に分解され、そのノード間にリンクをはることによって非線形的に個々の情報を関連付けている。情報を連想的にリンクすることによって、人間の情報記憶/検索メカニズムを模倣し、すばやく直感的に情報へアクセスすることが出来る。いわばn次元的な資源管理を行っているといえるだろう。データの管理形態という面で見ると、これは大きな利点であり、我々にとって馴染みやすいものといえる。

がしかし別の面から見ると、あまりにもリンクの自由度が大きいため、リンクが視角的にイメージできないほど無次元化、複雑、多岐にわたってしまい、ユーザがハイパーテキスト空間をさまよっているときに、なにかしら物理的なもの(例えば道)に自分の位置をマッピングすることが出来ないことになってしまい、ユーザがデータを見失ったり、関係を把握できなくなってしまう(迷子の問題)。

また、ハイパーテキストでは、リンクの作成は大変に煩わしく手間のかかる作業であり、ハイパーテキストの構築者に多大な負担を課す可能性がある。加えて、新しい

ノードを付け加えるときには、ハイパーテキスト内の既存のリンクを全て把握していなければならない、付け加えるノードがどのノードから参照されるべきか、またどのノードを参照するべきかを知っていなければならない。

2.2 OpenBookのアプローチ

先に述べた問題点を解決するために、OpenBookでは次に挙げるようなアプローチを行っている。

2.2.1 本メタファの使用

本メタファをインターフェースの面からみると以下のような利点がある[1]。

- 高速ブラウジング(パラパラめくり)が可能。
- 馴染みの深いメディアである。
- ページ、本の厚みは読み手に対し自分の位置を明確にさせることができる。
- 空間的記憶を利用して目的の情報を検索することができる。
- ページめくりの間に目に入る情報から偶然の知識が得られる。

が、情報の管理形態の面から見ると、本メディアでは情報は順序付けされて記述され、ユーザはシーケンシャルに読み進めて情報を得ていくため、時として我々人間の非線形的な思考プロセスと合わないことがある。この点では既に述べたようにハイパーテキストは優れている。我々は、この双方の利点を活かすことのできるメディアとしてOpenBookを位置付けた。以下にOpenBookの利点をあげておく。

- 本メディアは、シーケンシャルに読み進めるものであるが、この記述の線形性を、ユーザが読み進めるための基本的なガイドとして用いることができる。
- パラパラとページをめくることで、情報の移動方向や進み具合を視覚的に把握できる。
- 各ノードの関係、情報の量などをうまく本の厚みに反映させることで、全体の中での自分の位置を明確にでき、ユーザに対し大きな安心感を与えることができる。

*OpenBook: A User-Oriented Hypertext Integrated with Book Interface — User Interface —

†Masato Kudo, Satoshi Ichimura, Kyoichi Arai, Ken-ichi Okada and Yutaka Matsushita

‡Keio University

2.2.2 知識ベースの利用

ハイパーテキストのデータ管理に知識ベースを用いることでリンクの構築の簡略化を目指した [2]。この知識ベースでは、ハイパーテキストの既存のノードやリンクに関する情報が管理されている。ユーザがリンクを構築する際には、この知識ベースがユーザを支援する。例えば、ハイパーテキストのノードの作成の際には、データはキーワード（ユーザがそのデータに対して付加する属性、付加情報）が付加された形でノードとして格納されるが、このキーワードの情報は知識ベースにも蓄えられ、ノードの検索やリンクの生成に用いられる。この知識ベースを用いて検索されたハイパーテキストのデータを、理解しやすい形にソートし本の形態に変換するが、ここでアルゴリズムとして、ISM ノード構造化法 [2],[3] を用いることで、ハイパーテキストの構造を反映した順序付けが行われる。

3 OpenBook システム

本システムは、Sun ワークステーション上で作成されている。OS は UNIX 4.2BSD、言語は UNIX-C を使用しウィンドウ環境として OpenWindows 2.0、NeWS 2.1 を用いた。また Toolkit として XView、TNT-2.0 Toolkit を使用した。このシステムでは、データとしてテキストファイル、および EPSF 形式の PostScript ファイルを扱うことができる。また、独自のマークアップ言語がデータ処理に利用されている。

ユーザインターフェース：

システムを立ちあげると Fig.1 のようなウィンドウが現れる。

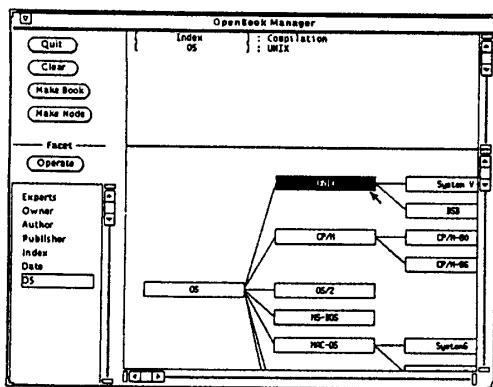


Fig.1 キーワード指定ウィンドウ

このウィンドウでは、ノードの作成や検索のためのキーワードの指定ができる。ノード作成の際には、Fig.1 のウィンドウでキーワードを設定する。するとこのデータはキーワードが付加された状態でハイパーテキスト中に格納され、知識ベースにキーワードが格納される。検索

の際には、Fig.1 のウィンドウでキーワードを指定した後、ここで指定されたキーワードによりハイパーテキスト中のデータがフィルタリングされ、選択されたノードを理解しやすいような形にソートした上で、Fig.2 にあるような本メタファを用いたウィンドウとなってあらわれる。

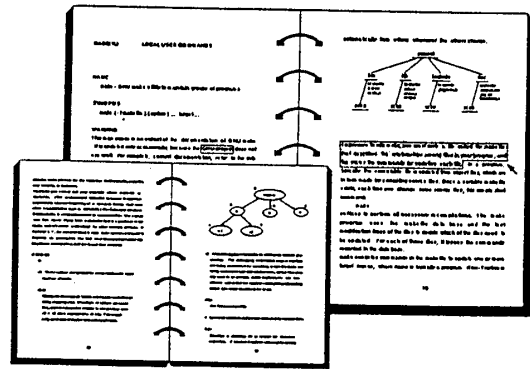


Fig.2 OpenBook ウィンドウ

このウィンドウは、テキストデータもビクチャーデータも同等に扱え、本メタファの持つページめくり機能や本の厚みによるページジャンプ機能の他に本来のハイパーテキストデータが持つリンクを用いたジャンプ機能を持っている。また、ユーザの任意のサイズにウィンドウをリサイズすることや複数の本ウィンドウを開くことも可能である。またこの複数の本ウィンドウの間で新たにリンクをはることも可能である。

4 おわりに

本稿では、ハイパーテキストと本のそれぞれの持つ利点を活かし、よりユーザ指向であることを目指したシステムである OpenBook システムを紹介した。これにより、ハイパーテキストの抱える問題点に対する 1 つの改善策を示した。

参考文献

- [1] 荒井, 佐藤, 木下, 横山, 松下, “ページめくり機能を持ったウィンドウインターフェース: BookWindow”, 情報処理学会研究会 HI-36-3, 1991 年 5 月。
- [2] 荒井, 市村, 塚田, 松下, “グループワークのための電子ブックの提案”, 情報処理学会第 43 回全国大会 5-77, 1991 年 10 月。
- [3] 佐藤, “教材構造化支援”, 電子情報通信学会誌 vol.71 No.4, 1988 年 4 月。
- [4] 前田, 市村, 金山, 松下, “本とハイパーテキストを融合したユーザ指向メディア: OpenBook(1) — グループ内での情報管理形態 —”, 情報処理学会第 44 回全国大会, 1992 年 3 月。