

ディレクトリ階層を利用した分散形 ハイパーテキストシステムの設計と実現

二村祥一 菊池正城 脊古裕司

futamura@csis.oita-u.ac.jp

大分大学 工学部 知能情報システム工学科

〒870-11 大分市旦野原700

ファイルシステムのディレクトリ階層を利用した分散形のハイパーテキストシステムを設計・実現した。本システムでは、ユーザのディレクトリ階層を直接に利用することにより、容易にかつ効果的にハイパーテキストを構築・利用できる。ハイパーテキストのノードとしてはファイルとディレクトリを割り当て、リンクとしてはディレクトリ階層に従った階層リンクと任意のノード間を接続する動的リンクを用意した。システムはサーバ・クライアントモデルに従ってオブジェクト指向設計し、システム機能としてノード航行、情報検索、リンク操作などを実現した。

Design and Implementation of Distributed Hypertext System
Based on File/Directory Hierarchical Structures

Shouichi FUTAMURA Masaki KIKUCHI Hiroshi SEKO

Department of Computer Science and Intelligent Systems
Oita University
700 Dannoharu, Oita 870-11, Japan

We implemented a distributed hypertext system based on file/directory hierarchical structures which made us to construct hypertext easily and to use them effectively. As the nodes of our hypertext we used files and directories, and as the links we prepared hierarchical links and dynamic links; the hierarchical links correspond to the directory structures, and the dynamic link is for the connection of two arbitrary nodes. We designed our system as a server-client model under object-oriented paradigm, and implemented the hypertext facilities such as node navigation, information retrieval, link operation, and the file operations.

1. はじめに

文書を文書要素（ノード）の集合と要素間の関連（リンク）として捉え、文書の理解や作成など人の知的活動のために役立てようとするハイパーテキスト技術が応用段階に達し、WWWに見られるようにその応用の空間的な広がりも見せている。しかし、現状のハイパーテキストシステムでは、情報の提供に時間がかかる；最新情報の継続的な提供が困難である；また情報の見通しができにくい；などの短所をもつ場合が多い。

ユーザは、各自の利用する計算機システムにおいて、ディレクトリ階層を使って情報を整理している。ここでは、ディレクトリ階層を利用することにより、容易にかつ高速にハイパーテキストを作成・利用できるようにと考えた。特に、分散計算機環境での情報資源の統合利用について配慮した。基本的には、ユーザはディレクトリやファイルを準備するだけで、ハイパーテキストを構築できるようにした。

2. ディレクトリ階層とハイパーテキスト

UNIXを例にして、ディレクトリ階層とハイパーテキストとの関連を述べる。UNIX以外のMacintosh-OSやMS-DOSなどについても同様の取り扱いを考える。

UNIXでは、一つのディレクトリのもとに複数のサブディレクトリやファイルを配置できる。ここでは、ディレクトリやファイルをハイパー

テキストのノードと見なし、それらの所属関係によりノード間のリンクを想定する。

特定のディレクトリ以下のディレクトリ階層構造（以下、木と呼ぶ）をハイパーテキストと考え、木の集合（以下、森と呼ぶ）を統合してハイパーテキスト群としてデータ操作できるように設計した。ここで、森の要素である木はそれぞれが分散計算機環境の別の計算機上に散在してもよい。

ディレクトリ階層とハイパーテキストの関連を図1に示す。この図はディレクトリAを根とするディレクトリ階層である。リンクにはディレクトリ、ファイルの所属関係による階層リンクと、任意のノード間を結ぶことのできる動的リンクの2種類を用意した。

3. ハイパーテキスト構成法

この章では、ハイパーテキストの構成法とハイパーテキストシステムのユーザインタフェースについて述べる。

3.1 ノードの管理

ディレクトリとファイルをノードとする。ノード名としてディレクトリ名、ファイル名を使う。ファイルの場合はファイル内容をノード内容とする。ディレクトリの場合は、該当ディレクトリに.adoc という名前のファイルを作り、そこにディレクトリのノード内容を設定する（図2を参照のこと）。ノード内容として、現在はテキスト、画像、音声などのモノメディア情報を置いている。

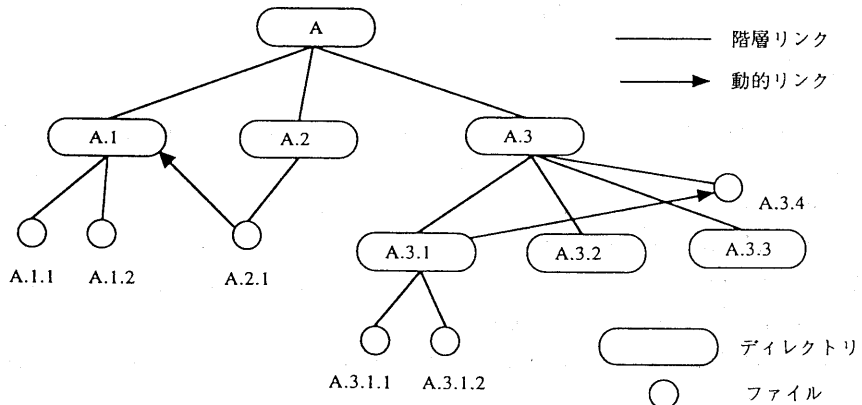


図1. ディレクトリ階層とハイパーテキスト

ファイル・ノード (file1に対して)

ファイル名	ファイル内容
file1	file1のノード内容
.link.file1	file1の動的リンク情報
.keyword.file1	file1のキーワード情報

ディレクトリ・ノード (dir1に対して)

ファイル名	ファイル内容
.adoc	dir1のノード内容
.link.adoc	dir1の動的リンク情報
.keyword.adoc	dir1のキーワード情報

図2.ファイル情報によるハイパーテキスト管理

3.2 リンクの管理

ディレクトリ階層に対応して暗黙に階層リンクを張る。また、任意のノード間に動的リンクを設定可能にした。動的リンクは、ノードの所属ディレクトリに .link. ノード名のファイルを置き、そこにリンク先のノード情報を設定する(図2参照)。

3.3 キーワードの管理

ノードの内容識別のために、ノードにキーワード情報を付加することができるようにした。この情報はハイパーテキスト内でのノードのキーワード検索時に使用される。これは画像や、音声などの非テキスト情報に対する情報検索で有効である。キーワード情報は、ノードの所属ディレクトリに .keyword. ノード名 のファイルを置き、そこに複数のキーワードを設定する(図2参照)。

3.4 ユーザインタフェース

システムのユーザインタフェースを図3に示す。システムウィンドウとして、コンソール、ブラウザ、文書ウィンドウを用意した。

コンソールには検索可能なハイパーテキストの一覧を表示する。一覧からメニュー選択することにより、ハイパーテキストのデータ航行や情報検索を始めることができる。複数のハイパーテキストの同時選択も可能にした。

ブラウザには、ディレクトリ階層の森を構造表示する。森を構成するそれぞれの木は指定ディレ

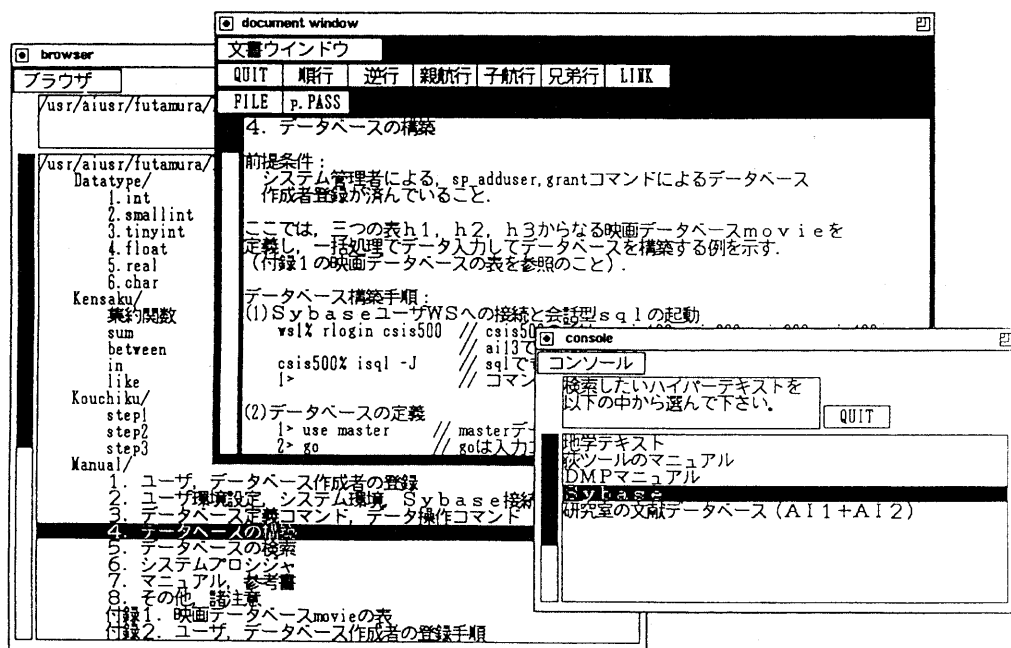


図3.ハイパーテキストシステムのユーザインタフェース

クトリを根とし、ディレクトリの深さに応じて段付けして表示した。各行にはノード名を表示し、ディレクトリに対しては最終文字として” / ”を付加している。ブラウザの行のうち、注目ノードについてはノード名を反転強調表示した。

文書ウィンドウには注目文書を表示する。ブラウザと文書ウィンドウは対になって動作する。(以後、これをブラウザ・文書ウィンドウ対と呼ぶ)。ブラウザ・文書ウィンドウ対は複数表示できる。

4. ハイパーテキストシステムの機能

この章では、システムの基本機能であるノード航行と情報検索、リンク操作について述べる。

4.1 ノード航行

ノード航行は文書ウィンドウのツールボタン、またはブラウザウィンドウを使って行う。

文書ウィンドウのツールボタンを利用したノード航行は、注目文書から出ているリンクを辿ってノード移動する。このうち階層リンクを利用したノード航行としては、順行、逆行、親航行、子航行、兄弟行の5個の航行法を用意した。それぞれ以下の航行を行う。

順行：ディレクトリ階層を先行順 (preorder) にノード航行する。

逆行：順行の逆方向に航行する。

親航行：所属するディレクトリ・ノードに移動する。

子航行：該当ディレクトリ内の先頭ノードに移動する。

兄弟行：所属ディレクトリ内の次ノードに移動する。

動的リンクによるノード航行は、文書ウィンドウのLINKボタンにより起動され、それによる動的リンク先一覧表示の中から移動先を選択できる。

ブラウザを利用した直接航行も可能である。この場合は、ブラウザ上で航行位置を選択する。

4.2 情報検索

情報検索機能としてフルテキスト検索とキーワード検索の2種類を用意した。いずれも指定キーワード文字列が検索対象に含まれるかどうかをチェックする。検索対象としては、フルテキスト検索ではノード内容 (ファイルの場合はファイル内容、ディレクトリの場合は .adoc ファイルの内容) を割り当て、またキーワード検索では .keyword. ノード名 のファイル内容を割り当てた。

情報検索ではパラメタとして、検索種別、検索範囲、質問式の三つを指定する。検索種別にはフルテキスト検索、キーワード検索の区別を、検索範囲はブラウザにより検索対象ディレクトリ (複数指定可) を、質問式はキーワードを論理演算子を伴って指定できるようにした。

情報検索においては、検索対象と検索結果を共にディレクトリ階層で表示するようにし、処理を一貫して捉えられるようにした。

4.3 リンク操作

リンク種別として階層リンクと動的リンクの2種類があるが、このうち階層リンクについては、ファイルシステムのディレクトリやファイルの追加・削除により自動的にリンクが設定・削除される。

動的リンクについては、リンク元のノード位置に .link. ノード名 のファイルを置き、そこにリンク先のノード情報を追加・削除する。動的リンク設定のために、リンク元とリンク先の両方の文書ウィンドウ (正確にはブラウザ・文書ウィンドウ対) を同時に表示して、確認しながらリンク設定ができるようにした。

動的リンクの管理方式として、現在はすべて .link. ノード名 のファイルを用いて管理する方式をとっているが、不特定多数のユーザ利用に備えて、動的リンクをリンク管理サーバで集中管理する方式を併用することを考えている。

5. システム実現法

本システムはUNIXの分散計算機環境のもとで開発した。プログラミング言語はC++を、ユーザインタフェースはX-Windowを、また通信プロトコルはTCP/IPを用いた。

5.1 システム構成

システム構成を図4に示す。システムは分散計算機環境で動作できるように、クライアント・サーバモデルに従って設計した。

システム要素として ForestServer, TreeServer, Clientの3個を考えた。ForestServerは森レベルのハイパーテキストを登録提供するサーバで分散計算機環境に原則として一つだけ配置させる。TreeServerは木レベルのハイパーテキストを処理するサーバで、分散計算機環境のそれぞれの計算機上に配置し、自己のファイルシステムに対する処理を行う。

ハイパーテキストの森は、例えば次のように定義される。

ハイパーテキスト名=地学

WS1:/usr/aiusr/futamura/地学テキスト/

WS2:/home/宇宙/

WS3:/home/惑星/写真/

このとき、ForestServerはこの情報そのものを提供し、WS2のTreeServerはディレクトリ/home/宇宙/以下のディレクトリ階層の処理を担当する。

Clientは、ForestServer, TreeServerへの情報要求やその結果受信、結果マージ処理などを行うとともに、ユーザインタフェースとして機能する。

5.2 システム要素の実現法

システム要素のForestServer (対応するForest Clientを含む) とTreeServer (対応するTreeClientを含む)、およびClientの構成部品である DocWindowについては、オブジェクト指向を応用して、それぞれをクラス設計した。DocWindowはユーザインタフェースのためのクラスで、これはブラウザ・文書ウィンドウ対に対応するものである。このクラスによりブラウザと文書ウィンドウを連動させて処理することが可能になった。

それぞれをクラス設計することにより、以下の効果を得ることができた。

- ① 異種の計算機システムのクラスインタフェースを統一できた。
- ② TreeServerのインスタンス化だけで、容易に分散計算機環境に対応できた。
- ③ DocWindowのインスタンス化により、容易に複数文書ウィンドウを実現できた。複数文書ウィンドウはリンク設定などで有効である。

5.3 木情報の作成処理

木情報はTreeServerによって作成・提供される。

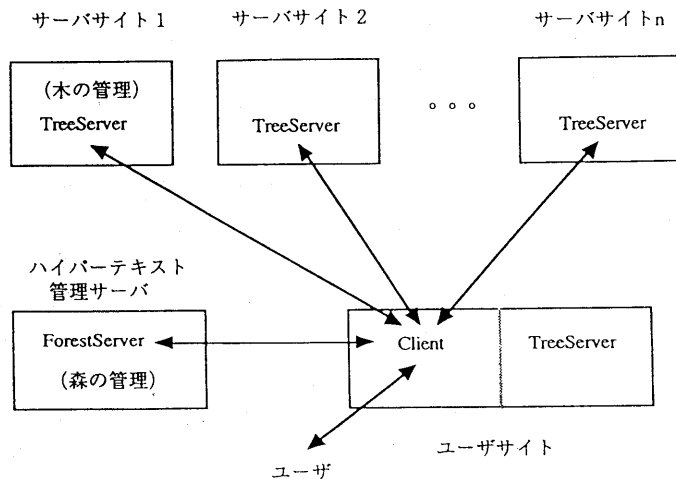


図4. ハイパーテキストシステムの構成

木情報は指定ディレクトリ以下のディレクトリ階層として作成されるが、作成内容としてFiletree, Dirtree, Dottreeの3種類のものを選択できるようにした。Filetreeは(ディレクトリ+ファイル)の階層を、Dirtreeはディレクトリだけの階層を、またDottreeはFiletree情報に更にドットファイル(.login, .cshrcなど)を含めた階層を作成する。

木情報は以下の手順で作成する。

①指定ディレクトリにFiletree (またはDirtree, Dottree) があれば、それを木情報とする。

②指定ディレクトリにそれらがなければ、ディレクトリ階層を使って木情報を新規に作成する。

手順①での木情報の提供により、整列された木情報の提供; 木情報の提供処理の高速化; が達成できた。

6. 応用

この章では、研究室での利用例にあげて開発システムの応用について述べる。

高校地学テキスト

編、章、節、小節などの階層構造をもった文書のハイパーテキスト化は容易である。高校地学テキストについては章タイトル、節タイトルなどをディレクトリに、文章パラグラフや図、表などをファイルに割り当てるだけでよかった。これにより階層リンクによる航行が可能である。

現在は、テキストや図、表などのモノメディアを単位にノード分割しているが、それらを組み合わせたマルチメディアの定義ファイルを用意することにより、より効果的な文書管理を行なう計画をもっている。

Sybase利用マニュアル

学科授業科目のデータベース演習のための冊子体マニュアルをハイパーテキスト化した。これに、データベース構築用スクリプトや検索コマンド例集、質問回答集などのハイパーテキストの木をマージして、Sybase利用マニュアルのハイパーテキストの森を作った。これは、分散配置された情報の統合を示す例である。それぞれの木が別の管理者によって管理されているが、最新情報を

適切にユーザに提供できる。

研究室文献データベース

研究室の文献データを、特定ディレクトリのもとに文献単位にファイル化して、ハイパーテキストとした。それぞれの文献にはキーワードを付加し、キーワード検索を可能にした。

文献を分類して、それにサブディレクトリを割り当てることにより、ディレクトリ階層を生かした情報検索を行なうことも可能である。1研究室の文献データを木として、複数研究室の文献データの森の中を情報検索して、文献の位置づけを知ることにも有効であろう。

7. おわりに

開発したハイパーテキストシステムは、ディレクトリ階層で管理された情報から、高速かつ効果的にハイパーテキストを構築できる。本システムは分散計算機環境に対応しており、分散環境に存在するデータを容易に統合できる。現在、このシステムは主に研究室での情報管理に使っているが、広域ネットワーク上での情報統合のために利用されることが十分に期待できる。

本研究は一部、一般研究(C) No.06808036の科学研究費の援助で行われた。

参考文献

- [1] 上林弥彦編著：ハイパーメディアとオブジェクトベース、共立出版、1995。
- [2] 石田晴久監訳：UNIXプログラミング環境、ASCII、1985。
- [3] 木下凌一ほか：X-Window Ver.11 プログラミング(第2版)、日刊工業新聞社、1993。
- [4] 二村祥一ほか：文献関連情報をもつ研究室用文献データベースの構築、情報処理学会研究報告、94-CB-21, pp.17-24, 1994。
- [5] 二村祥一ほか：分散型データベースシステムのオブジェクト指向設計、情報処理学会マルチメディア通信と分散処理ワークショップ論文集、pp.137-144, 1993。