

フォーマッタ階層を媒介とした 構造化文書とハイパーテキストの統合

林 浩一 関島 章文

富士ゼロックス株式会社 システム・コミュニケーション研究所

文書アイデア創出から文書の割り付け/印刷までの一連の作業は、(I)ネットワーク化によるアイデア創出、(II)階層化による論旨の明確化、(III)線形化によるレイアウト生成、という枠組で捉えられ、それぞれのフェイズを支援するシステムが開発されてきた。しかしながら、この枠組では一旦論旨を明確化した後でのアイデア創出を支援できない。我々は各フェイズを任意に繰り返すことのできる文書処理モデルを提案する。我々は(1)アイデアの創出と論旨の明確化の過程を同時に表現できる REN 構造を導入し、(2)REN 構造に対する割り付け処理メカニズムを規定した。本稿では、現在開発中のプロトタイプ SUIREN を用いて、この文書処理モデルを説明する。

Formatter Hierarchy Unifies Hypertext and Structured Documents

Koichi Hayashi Akifumi Sekijima

Systems & Communications Lab., Fuji Xerox Co., Ltd.

YBP East Tower, 134 Godocho, Hodogaya-Ku, Yokohama-Shi, Kanagawa 240, Japan

Authoring process involves various activities, from creating ideas to formatting or printing documents. Conventional systems divide the process into the sequential 3 phases: 1) Network structuring to support idea creation; 2) Hierarchical structuring to organize a document outline; 3) Linear structuring to generate a formatted form. However, this framework does not allow additional idea creation after formatting the document. We proposed a document processing model to enable us to repeat each phases in any order. We introduce the REN structure, which can be used for representing ideas and constructing outlines; and the formatting mechanism for the structure. This paper illustrates the document processing model and the first prototype SUIREN.

1. はじめに

文書の作成は人間の最も知的で創造的な作業の一つである。文書作成のプロセスには様々な作業が含まれており、それらを支援するためのシステムが、ハイパーテキスト、文書処理、データベースなど、種々の立場から提案され、開発されてきた^{[Con87][AFQ89]}。この中で、特にアイデアの創出から文書の割り付け印刷までを一貫して支援するハイパーテキストシステムの原型は、論文[SWF87]によって示された WritingEnvironment である。

論文[SWF87]は、文書の作成プロセスを人間の認知モデルの立場から、次の3フェイズのモデルとして捉えることを提唱した。

(I) ネットワーク化: アイデア創出のフェイズ

未整理の収集情報やアイデアのメモを蓄積し、相互に関係づけを行なって、ネットワーク構造を構築する。

(II) 階層化: 論旨の明確化のフェイズ

ネットワーク構造に表現された情報を基にして、文書のアウトラインを表現する階層構造を構築する。この構造のことを、本稿ではアウトライン構造と呼ぶ。

(III) 線形化: レイアウト生成のフェイズ

アウトラインの構造からレイアウト構造を生成し、表示、印刷する。このフェイズは割り付け処理と呼ばれ、自動処理が実現されている。

WE は、これらのフェイズを順次支援するプロトタイプシステムである。各フェイズで異なるシステムを用い、ファイルを媒介にしてデータ交換する場合も含めると、今日の計算機によるアイデアの創出から文書の割り付け印刷までの支援は、全てこのモデルに従っていると考えられる。

ところで、実際の文書作成においては、各フェイズが一回ずつ、この順序に進むとは限らない。我々は、論旨が十分に整理されていない段階であっても、 \surd 切が近づけばドラフトを作成する。そして、他の人の意見も取り入れながら内容を吟味し、編集を繰り返す。最終版の文書が完成するま

で、アウトラインが大きく変わることも、重要なアイデアが加わることもある。言い替えると、最終版の文書を得るまでには、各フェイズは様々な順序で繰り返される。さらに、こうして得られた最終版の文書にも、いつかまた大幅な改訂が必要になるかもしれない。各フェイズは、より大きな情報創造過程を形成するのである。

しかし WE のモデルでは、一旦論旨を明確化してしまった後で、もう一度アイデア創出のフェイズに戻ることはできない。論旨の明確化が、ネットワーク構造からアウトライン構造の生成という、不可逆な変換の過程であるからである。このため、WE のモデルがその有効性を十分に発揮できるのは、最初のバージョンの作成時に限られてしまう。

本稿では、アイデア創出、論旨の明確化、割り付け処理の各フェイズを、任意の順に繰り返すことのできる文書処理モデルを提案する。(図1)

我々のアプローチは、ネットワーク化と線形化のフェイズの守備範囲を広げ、階層化のフェイズを不要とするもので、(1)新しいデータモデルの導入と、(2)割り付け処理モデルの規定を行なった。

新しいデータモデルとして、アイデア創出と論旨の明確化の両方に用いることのできる構造、REN(Restriction Enclosure Network)を導入した。さらに、REN 構造に対して割り付け処理を行なうために、インタラクティブなフォーマッタ階層の構築手法を採用した。フォーマッタ階層は、構造化文書の割り付け処理手法の一つ「多重レベル流し込み」で用いられる、割り付け処理のための制御構造である。現在、このモデルに基づいた文書処理環境のプロトタイプ SUIREN を開発中である。

本稿では、まず、我々の提案する文書処理モデルの背景となる関連技術を検討する。次に、データモデル、割り付け処理モデルについて述べる。その中で、開発中のプロトタイプ SUIREN についても触れる。なお、本稿では主にデータモデルについて述べ、割り付け処理モデルの詳細については別報に譲る。

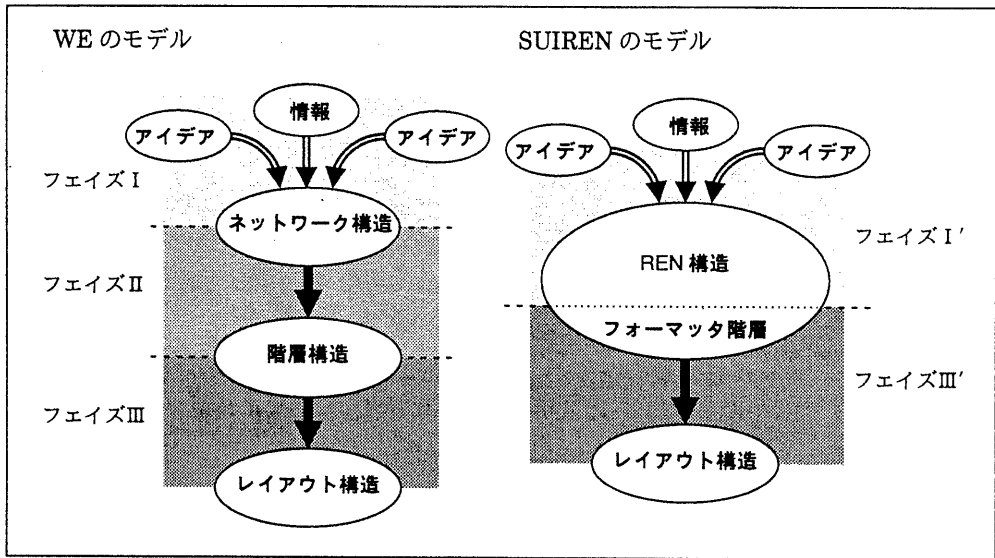


図1 文書作成プロセス

2. 関連研究

論旨の明確化のフェイズ(II)で用いるアウトライン構造は、前後のフェイズ(I)(III)の橋渡しの役割を果たしている。本章では、アウトライン構造を境にした前工程と後工程について、関連研究を概観する。

2.1. アイデア創出からアウトライン生成

ネットワーク構造からアウトライン構造への変換には、大きく分けて対話的変換と自動変換の二つがある。それぞれについて問題点を検討する。

2.1.1. 対話的変換

ハイパーテキストでは、構造リンクと参照リンクの二種類のリンクで、アウトライン構造とネットワーク構造を表現する手法が用いられる^[Con86]。このようなシステム(例えば NoteCards^[HMT87])では、ユーザは手作業でリンクを書き換えることによって、ネットワーク構造をアウトライン構造に変換できる。

WE^[SWF87]では、手作業の変換手続きを簡便にするための手段が提供されている。ネットワーク構造を操作するブラウザから部分構造を選択し、ア

ウトライン構造を操作するブラウザにコピーすることによって、アウトライン構造を構築できる。選択した構造にループが含まれている場合には、リンクが自動的に切断される。

これらのシステムにおいて、二つの構造は完全にユーザの制御下にある。したがって、アイデアの創出とアウトラインの生成のステップが同時に進行する場合、二つの構造の一貫性を維持するのはユーザの責任になる。様々な変更が繰り返される場面では、ユーザの負担が大きくなり、結局どちらかの構造が破棄されることになる。

2.1.2. 自動変換

KJエディタ^[MKO89]では、KJ法の図解中のアウトラインを自動的に評価し、線形化まで行なう機能を持っている。しかしながら、この変換は一方にしか進まない。INSPIRATION[™]^[Cer91]では、二つの構造の双方向の変換機能を提供している。それぞれの構造に対して、ブラウザが用意されているが、一時には一方のブラウザしか提示されない。ブラウザの切替えの際に、データ構造が変換される。変換時に不足する情報はシステムによって補

* INSPIRATION[™]は Ceres Software 社の登録商標である。

足される。

このような自動変換では、変換がシステムの解釈に基づいて行なわれるため、ユーザの意図した結果となる保証はない。たとえば、INSPIRATION™でループを持つネットワーク構造をアウトライン構造に変換し、編集を施した後で再びネットワーク構造に戻した場合、変換結果を予測することは事実上不可能である。

2.2. アウトラインから割り付け処理

アウトライン構造の重要な役割の一つは、割り付け処理の対象となることである。どのような構造が割り付け処理の対象となり得るかを知らるために、構造化文書の割り付け処理手法を検討する。

構造化文書では、アウトライン構造を論理構造と呼び、割り付け指示と処理の体系化がなされている^[AFQ89]。アウトライン構造は、文書内容を表現する構造であるだけでなく、割り付け処理の制御構造でもある。割り付け処理は、原理的にはアウトライン構造をルートからトラバースし、順番に割り付けオブジェクトを生成、配置すればよい。しかし、多様なレイアウト指示が可能であるODA^[ISO87]のような体系では、複雑な処理が必要に

なる。

多重レベル流し込みは、割り付け処理制御のための構造を、論理構造から分離することによって、この処理を高機能化する手法である^{[MH92][SKHM92][MHHK92]}。論文[MH92]では、割り付け処理の対象となる構造として「ネストした論理ストリーム」が、また割り付け処理制御のための構造として「フォーマッタ階層」が、アウトライン構造から抽出される。(図2)

論理ストリームとは、他と独立に割り付け処理のできるノード列のことである。章や節を表す部分木のルート、あるいは文字などのプリミティブが、論理ストリームを構成する。下位構造に論理ストリームを持つノードが、一つ上の階層の論理ストリームを構成してよい。ネストした論理ストリームは、章や節に指定されている下位構造に対する割り付け規則、レイアウトテンプレートを手がかりにして、自動的に抽出される。

フォーマッタは、一つの論理ストリームに対して部分割り付け処理を行うプロセスである。ネストしたストリームのそれぞれに対応したフォーマッタは階層構造を形成する。各フォーマッタは、上位と下位のフォーマッタと情報交換しながら処

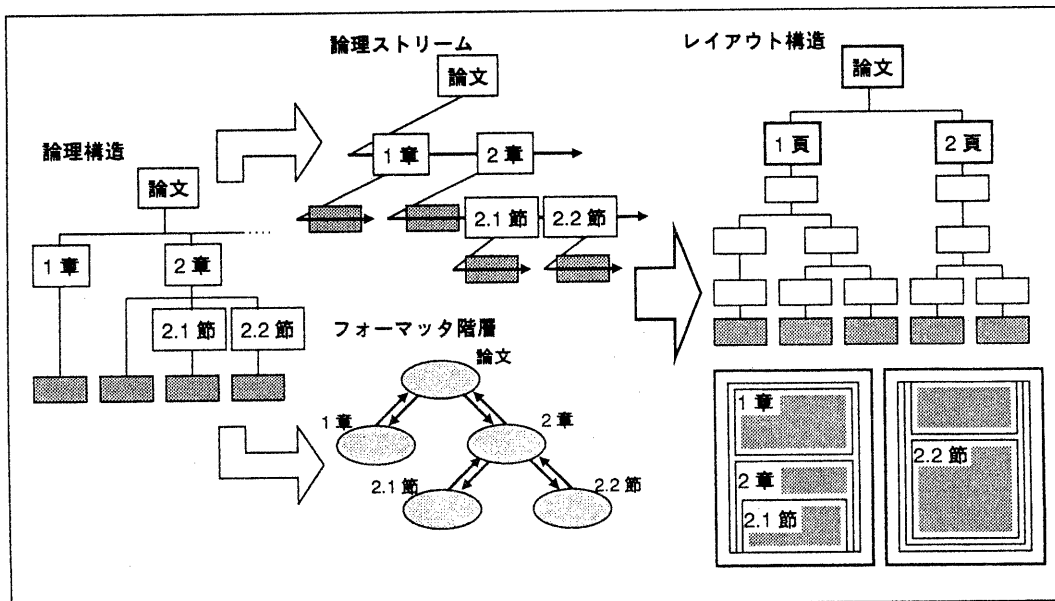


図2 論理ストリームとフォーマッタ階層

理を行ない、全体の論理構造を割り付ける。

割り付け処理の対象は論理ストリームだけではない。論文[MH92]は、構造化文書モデルの一つ tnt^[Fur90]に対する割り付け処理も与えている。tnt 構造は、木構造で表される文書構造と、非木構造で表される内容を相互にネストさせることができる。tnt 構造は、内容を割り付けるフォーマッタと、ストリームを割り付けるフォーマッタから構成される階層によって割り付けられる。

3. 文書処理モデル

関連研究についての検討の結果、我々は以下の方針で文書処理モデルを構築することにした。

● データモデル

ネットワーク構造とアウトライン構造の両方の性格をもった構造を導入する。我々はネットワークとアウトラインの二つの構造を扱うことは、アイデア創出と論旨の明確化の繰り返しには向かないと、結論した。二つの構造の間で一貫性を維持しながら、両方を更新することは著しく困難だからである。

● 割り付けモデル

新たに導入する構造に対してフォーマッタ階層を構築するメカニズムを提供する。フォーマッタの階層構造が用意できれば、割り付け処理は適用できるからである。

3.1. REN 構造

我々は、ネットワーク構造とアウトライン構造の両方の性格を持った構造、REN(Restriction Enclosure Network)を導入した。REN 構造は、フィールド、アンカー、プリミティブの三つの要素から構成されるデータ構造である。(図 3)

プリミティブ

操作、表示の対象となる最小の単位。例えば、文字はプリミティブである。

フィールド

プリミティブの集合に対して関係を規定する。規定する関係の種類ごとに異なるフィールドがある。例えば、ストリームは文字の集合に順序

関係を規定するフィールドである。

アンカー

プリミティブの一つで、他のフィールドを参照する指標である。アンカーは任意のフィールドを参照できるので、全体の構造はネットワークはネットワーク構造を形成する。

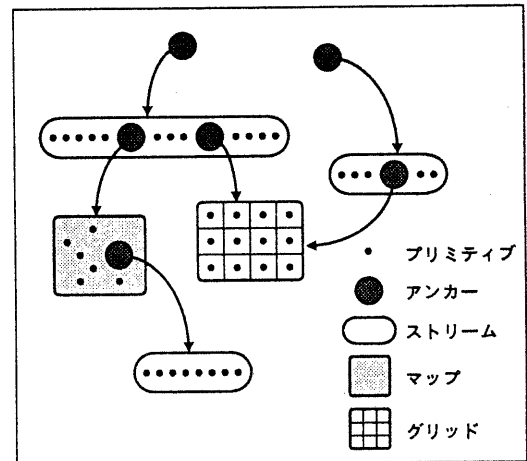


図 3 REN 構造

前章で述べた割り付け処理手法では、割り付け対象となる構造として、ネストした論理ストリームを木構造から抽出した。アウトライン構造は正確には順序付の木構造である。ストリームのネストと木構造という二種類の表現のどちらを選択するかは、深さと幅のどちらを優先すべきと考えるかによる。多重レベル流し込みで、ストリームによる表現を用いているのは、ストリームが一階層分の順序関係を保持し、フォーマッタによる局所的な割り付け処理の動作を保証するからである。

フィールドは論理ストリームを一般化したものであり、構成要素の関係を保持し、フォーマッタによる局所的な割り付け処理の動作を保証する役割を果たす。

ストリームは、テキスト内容中に保持される順序関係を抽出したものである。すなわち、テキスト内容が文字の順序であることと、章が節の順序であることが、ストリームとして同一視される。フィールドは、文字列内容に限らず、内容の種類

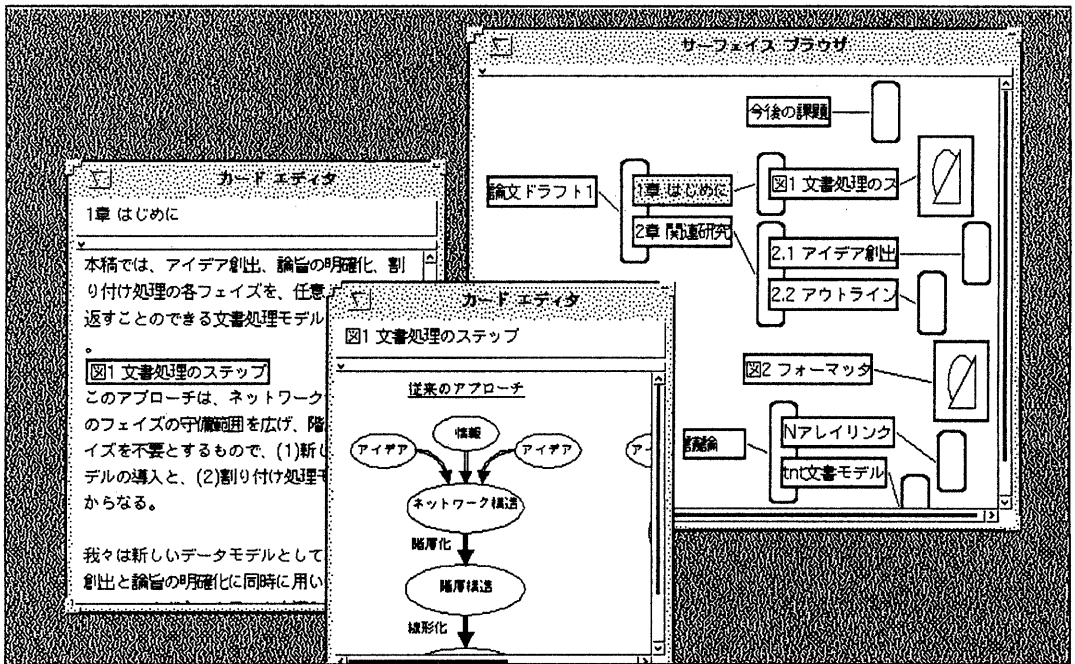


図 4 SUIREN の概観

ごとに定義できる。例えば図形内容に対応し、二次元平面上の位置を規定するマップ、表内容に対応し、二次元格子内での配置を規定するグリッド、などが考えられる。

REN 構造は、tnt 構造をネットワークへ拡張したデータモデルでもある。tnt 構造と異なり、我々は木構造を基本的なデータ構造として扱わず、各階層ごとのストリームで構成される構造と考える。REN 構造では、フィールドの中に閉じ込められた制約のみを扱い、木構造のような構造全体に及ぶ制約は扱わない。REN 構造はフォーマッタ階層を与えることによって、tnt 構造と全く同様にして割り付け処理が適用できる。

3.2. 文書編集環境 SUIREN

我々は、REN 構造を扱う文書編集環境のプロトタイプ SUIREN を、Objectworks\Smalltalk** 上に開発中である。(図 4)

3.2.1. フィールドエディタ

SUIREN は、REN 構造の探索や編集を行なうためのツールとして、フィールドエディタを提供している。フィールドとアンカーは、ハイパーテキストのノードとリンクにほぼ対応するので(詳しくは次章で論じる)、フィールドエディタは、通常のハイパーテキストと同様のインタフェイスを提供する。すなわち、一つのフィールドを一つのウィンドウ中表示する。アンカーはウィンドウ中でボタン状に表示される。SUIREN では、フィールドを表示するウィンドウをリーフ(leaf)、アンカーを表示するボタン状の表示をバド(bud)と呼んでいる。

フィールドエディタには、カードとアウトラインの二種類のインタフェイスがあり、必要に応じて使い分けられる。カードエディタでは、バドを選択して open コマンドを送ると、参照しているフィールドを表示する別のリーフを開く。アウトラインエディタでは、リーフの入れ子構造によってネストしたフィールドを表示する。

** Objectworks\Smalltalk は ParcPlace Systems 社の登録商標である。

3.2.2. サーフフェイスブラウザ

SUIREN は、REN 構造を図化して表示するツールとして、サーフェイスブラウザを提供している。サーフェイスブラウザは、通常のハイパーテキストのブラウザとは異なり、ネットワークの部分構造を選択的に表示する。編集の際に、ユーザが常に REN 構造全体に関心があるとは限らないからである。

SUIREN では、REN 構造上におけるユーザの関心領域と、そうでない領域を分離するために、「水面」(Surface)という概念を導入した。ユーザの関心領域は、水面上(on the surface)に現れた状態であり、そうでない部分は水面下(below the surface)にある状態と考える。サーフェイスブラウザは、水面上にある領域について次の情報を表示する。

(1)フィールドによる多項間関係

アンカーは、フィールドエディタと同様に、バドによって表示される。フィールドは、バドを内包する図形要素によって表示される。この図形はフィールドの種類ごとに異なり、図形中のバドの位置は、フィールド中のアンカーの位置を反映する。

(2)アンカーによる参照関係

アンカーが参照するフィールドは、対応するバドとフィールドを表す図形の間を線で結ぶことによって示される。

ユーザはサーフェイスブラウザ上で、データ構造の変更、関心領域の変更、フォーマッタ階層の定義、を行なうことができる。

4. 議論と今後の課題

4.1. フィールド/アンカー vs. ノード/リンク

前章で述べたように、フィールドとアンカーは、ハイパーテキストのノードとリンクに相当するとみてよいこともあるが、いつも一致するわけではない。これらは、異なるデータモデルである。

ハイパーテキストにおいて、リンクは、ノード間の関係を記述するとともに、ユーザがリンクに沿ってノードを探索するためのエントリーの役割を果たす。一方、ノードの役割は内容を保持することである。このモデルでは、ストリームなど、割り付け処理の動作を保証するための、複数項目の間の関係は内容の中に埋没してしまう。

フィールドとアンカーのモデルでは、アンカーはフィールドを参照するためのエントリーの役割のみを果たす。内容はプリミティブの単位に分解され、フィールドが複数のプリミティブの間の関係を記述する。フィールドは内容の持つ関係記述の能力を抽出したものである。

アンカーが実現するのは最も単純なリンク、すなわち、名前のない、点からノード全体への、有向リンクである。しかし、複数項目間での関係記述能力を持つフィールドによって、より高度なリンクを提供するモデルに匹敵する表現能力が提供される。例えば、Aquanet^[MHRJ]などが採用している N アレイリンクでは、複数項間の関係を記述できる。これと同等な記述能力を得るには、属性名とプリミティブの組を持つフィールドを導入すればよい。

4.2. フォーマッタとレイアウトテンプレート

SUIREN の割り付け処理モデルは、多重レベル流し込みの手法を仮定している。元々、多重レベル流し込みの割り付け処理は、論理ストリーム、フォーマッタ、レイアウトテンプレートの三者によって制御される。REN 構造では、ストリームはフィールドの種類の一つに過ぎない。フィールドの種類ごとに、異なるフォーマッタの部分割り付け処理アルゴリズムとレイアウトテンプレートが必要になる。

また、フォーマッタの考え方は、レイアウト生成だけでなくウィンドウの生成にも採用できる。この場合、インタフェイスの違いはレイアウトテンプレートの違いとして捉えることが可能である。ディスプレイ上では、スクロールやボタンとしての表示など、ハードコピーにない機能が利用

できるので、部分割り付け処理アルゴリズムと、レイアウトテンプレートはレイアウト構造生成のためのものとは異なると考えられる。

現在 SUIREN で提供しているフィールドの種類は、ストリームとマップにすぎない。また、フィールドエディタとして提供しているインタフェイスも、カードとアウトラインの二種類のみである。今後、SUIREN で提供するフィールドとインタフェイスの種類を増やすとともに、部分割り付け処理アルゴリズム、レイアウトテンプレートを体系化して行く予定である。

5. まとめ

本稿ではアイデア創出、論旨の明確化、割り付け処理の各ステップを、任意の順に繰り返すことのできる文書処理モデルについて述べた。我々は、(1)アイデア創出と論旨の明確化の両方に用いることのできる REN 構造の導入、(2)REN 構造に対してフォーマツタ階層を与えるメカニズムの提供を行なった。本稿ではまた、このモデルに基づいて開発中の文書処理環境のプロトタイプ SUIREN を紹介した。

我々は、ハイパーテキストの研究において、文書処理との接点が過小に評価されてきたのではないかと考えている。実際、ハイパーテキストの研究では伝統的な文書処理からの制約を積極的に排除することで、新しいビジョンを提示してきた。しかし、このことは、マルチメディアの制御を目的としたもの以外、ハイパーテキストシステムが大きな市場を獲得できていない一因でもある。大多数のユーザにとっては、ハイパーテキストは現在と断絶した未来の技術としか映らない。本稿は、部分割り付け処理の技術を媒介にして、ハイパーテキストと構造化文書処理の両者を統合することで、新しい展望が開けることを示したものである。

謝辞

本研究にあたり多くの方々から助言と励ましを頂きました。加藤守通氏、上林憲行氏、村田真氏、

申吉浩氏、旧文書編集交換グループの皆様感谢您的いたします。

参考文献

- [AFQ89] Andre, J., Furuta, R. & Quint, V., *Structured Documents*, Cambridge University Press, 1989.
- [Cer91] Ceres Software™, Inc., *INSPIRATION™* オナーズマニュアル, (株)スリーズカンパニー, 1991.
- [Con87] Conklin, J., "Hypertext: An Introduction and Survey" *IEEE Computer*, Vol.20, No.9, pp.17-41, 1987.
- [Fur89] Furuta, R., "Concepts and Models for Structured Documents", *Structured Documents*, Andre, J., Furuta, R. & Quint, V., Cambridge University Press, pp.7-38, 1989.
- [HMT87] Halasz, F. G., Moran, T. P. & Trigg, R. H., "NoteCards in a nutshell" *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pp.45-52, 1987.
- [ISO87] ISO, *Information Processing - Text and Office Systems - Office Document Architecture (ODA) and Interchange Format*, ISO/IS 8613, 1987.
- [MH92] Murata, M. & Hayashi, K., "Formatter Hierarchy for Structured Documents", *EP92*, Vanoirbeek & Coray editors, Cambridge University Press, pp.77-94, 1992.
- [MHHK92] 村田真, 林直樹, 林浩一, 小林健一, 構造化文書モデル ODA(Open Document Architecture)の拡張, 富士ゼロックステクニカルレポート, No.7, 1992.
- [MHI92] 村田真, 林浩一, 磯部俊哉, "構造化文書体系(ODA)の拡張: Nested Content", 情報処理学会第 44 回全国大会講演論文集(3), pp.299-300, 1992.
- [MHRJ] Marshall, C. C., Halasz, F. G., Rogers, R. A. & Janssen, W. C., "Aquanet: A Hypertext Tool to Hold Your Knowledge in Place", *Proceedings of Hypertext '91*, pp.261-275, 1991.
- [MKO89] 森田哲司, 河合和久, 大岩元, "図を用いた思考展開における文章化支援 図式エディタ PAN/KJ の場合", 情報処理学会研究報告 89-DPHI-23-2, 1989.
- [SKHM92] 申吉浩, 小林健一, 陌間端, 村田真, "構造化文書体系(ODA)の拡張: 多重レベル流し込み", 情報処理学会第 44 回全国大会講演論文集(3), pp.299-300, 1992.
- [SWF87] Smith, J. B., Weiss, S. F. & Ferguson, G. J., "A Hypertext Writing Environment and its Cognitive Basis", *Proceedings of Hypertext '87*, pp.195-214, 1987.