

文書構造と編集処理とに関する検討

坂下善彦 土田泰治

三菱電機㈱ 情報電子研究所

1. はじめに

種々の情報を表現しかつ保存する文書は今日重要な位置を占めている。手紙、報告書、案内書、伝票、メモ等々と我々の日常生活において、情報を保持し、かつ、伝達の媒体となる有力なものとして文書が存在する。

計算機システムの発展、特にオフィスオートメーション(OA)が叫ばれて久しい。OAにおける処理の主な対象はこの文書である。近年までは、種々の情報はきわめて特化された形で、計算機システムで扱われて来ており、その一例がEDPでもある。ワードプロセッサ(WP)の飛躍的発展と相まって、テキスト処理への期待が大きなものとなり、日常、机上で作成する種々の形態の文書を計算機システム上で作成、編集するようになってきた。

文書には、三種類の形態の情報、すなわち、コード情報の文字列、ベクトル情報の図形、そして、ピクセル情報のイメージ・絵が基本的に存在すると、我々は定義し、そして、この三種類の形態の情報により表現される文書の作成・編集システムについて報告した。〔1〕作成されたテキストは、削除、挿入などの編集の対象となる。文字のテキスト、図形のテキスト、そしてイメージ・絵のテキストそれぞれを独立に編集することは、比較的容易に処理できる。しかしながら、文書においては、必ずしも、これらのテキストは独立には存在せずかえって混在した状態で表現されることが多い。例えば、地図における、地形を現す図形又はイメージのテキストと、その地形を説明する文字テキストのようにである。文書の各テキストは、先に述べたように、3種の情報形態から作成されている。文書が一過性のものであれば、一度出来上がってしまった文書は、元の情報形態によらず、その装置にとって都合良い情報形態、例えばピクセル情報にて保存すれば良い。しかしながら、OAシステム、及びローカルエリアネットワーク(LAN)等の発展により、文書は、流通し、参照され、再利用される状況が、一層増大すると予測される。従って文書はいついかなる場合でも、再編集可能な状態にしておくことが必須となる。この事は文書情報をどのような形で管理するかは、作成・編集処理との関係からも重要である。我々

は文書の作成・編集処理を計算機システム上で実施するために、文書構造の概念を改めて定義した。〔1〕〔2〕同じような目的で、文書構造の概念をISO, CCITTでも提案している。〔3〕〔4〕

本報告は、ビットマップメモリを備えたワークステーション〔5〕〔6〕を前提として、先に述べた文書構造の概念に従った文書構造について論じ、次に、この構造に依る文書処理方法、データ構造について論ずる。

2. 文書構造

2.1 文書における情報の表現

(1) テキストの種類

文書に現れるテキストの種類としては、

- ・文字列を主体としたテキスト
 - ・図形・グラフ等を主体としたテキスト
 - ・イメージ・絵等を主体としたテキスト
- が基本的に存在する。更に
- ・図形・グラフ等と文字列
 - ・イメージ・絵等と文字列
- のテキストが存在し、そして
- ・全てが混在するテキスト

がある。特に全てが混在するテキストとしては表テキストがあり、日常多く用いられる文書の一つでもある。

以上のようなテキストの存在を考慮して文書は図2-1に示すような表現形態をとると定義した。文書はページの集合からなり、ページは余白部分を除いたテキスト領域の内側に、ヘッダー領域、フッター領域、図テキストが表現されている図領域、イメージテキストが表現されているイメージ領域、そして本文テキストが表現されている本文領域等から構成される。これ等の領域は互いに重なり合うことはなく、レイアウト編集の処理単位と定義している。

(2) 編集と編成と表現

一般に文書処理の処理フェーズから分けると、

- 1) テキストを作成・編集する編集フェーズ
- 2) テキストを書式情報に従って装置類に依存しない形式で書式を整える編成フェーズ

3) 装置類に依存した表示、出力形態に表現する表現フェーズとなる。〔7〕

編集フェーズにおいては、文字テキストに対しては、削除、挿入、移動、等の編集処理、図テキストに対しても、移動、拡大、縮小、等の編集処理、そして、イメージ・絵テキストに対しても、削除、追加、移動、回転、拡大、縮小、等の編集処理が実行される。編成フェーズにおいては、文字テキストに対しては、文字の送りピッチ、行の送りのピッチ、左・右の寄せ、中央揃え、ジャスティフィケーション、等の処理、図テキストに対しては基本単位への数値の変換、そしてイメージ・絵テキストに対しても基本単位への変換、等の処理が実行される。

表現フェーズにおいては、文字テキストに対しては、装置類の物理特性に依存した寸法に変換して処理を実行して実際に目に見える状態にして表現する。図テキストに対しては、同様に物理特性に依存した寸法に変換してベクトル情報からピクセル列の情報に変換して目に見える状態として表現する。イメージ・絵テキストに対しても同様に物理特性に依存した線密度に従って表現する。一般に、第2、第3のフェーズは同時実行処理される場合が多い。

又、イメージ・絵テキストを論理的に、どのように表現すべきかは今後の重要な課題と考える。

2.2 文書構造の概念

第1章で記したように、計算機システム上で処理するために、即ち、作成、編集、表示、そして変換される文書の構造、表現の概念定義、そして、文書交換の際のプロトコル定義が必要である。

(1) 論理構造

一般に、文書は、章、節、項、…といった階層構造により構成され、各々の階層構造における最下位の位置に実際のテキストが存在する。このテキストは先に述べた、文字、図形、イメージのテキスト、及びその複合化されたテキストである。この構造は、文書の骨格を表し、更に、表すべき情報としてのテキストを保持している。この意味において文書の論理構造と称す。文書の論理構造の概要を図2-2に示す。図中の節が論理対象である。

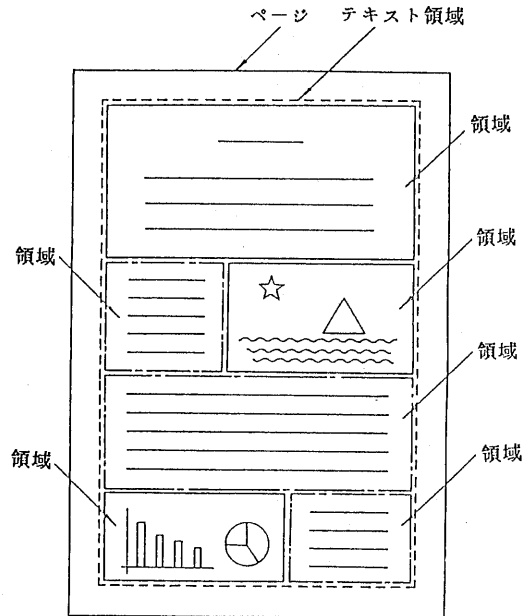


図2-1 文書構造 ページと領域

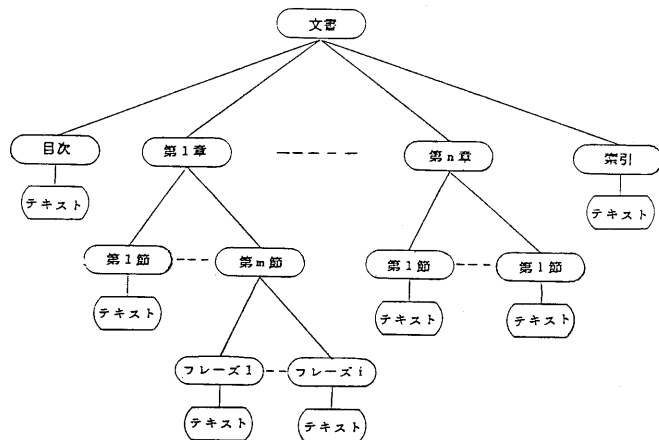


図2-2 文書の論理構造

論理構造のテキストに対する処理が、2.1節で述べた編集処理に相当する。又この構造それ自体をテキスト実体がなくとも編集することにより、その骨格を編集することが可能となる。テキスト自体を処理対象とすることにより、キーワードの抽出、インデックスの作成、表記誤り検出、表記のゆれの検出、等の処理が行われる。

(2) レイアウト構造

論理構造が論理対象（論理オブジェクト）及び論理対象間の関係を定義しているのに対し、レイアウト構造は文書内容を紙面等に物理的に配置するための構造である。

2.1節で述べたように、テキスト領域は、幾つかの副領域より構成されるレイアウト構造として定義した。〔1〕この副領域はテキスト情報それ自身を保持していない、情報は領域に従属しているブロックに保持される。一つのブロックには一つの形態の情報のみが保持でき、更に、複数のブロックを重畳し、その結果を先の領域に射影することにより、各種テキストが表現される。このように定義したレイアウト構造を図2-3に示す。又、ブロックと領域への射影の関係を図2-4に示す。

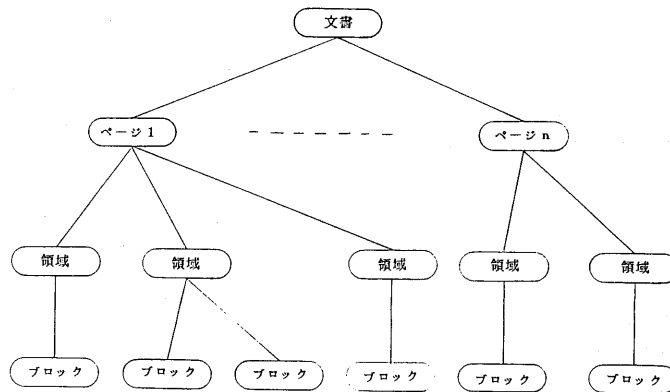


図2-3 文書のレイアウト構造

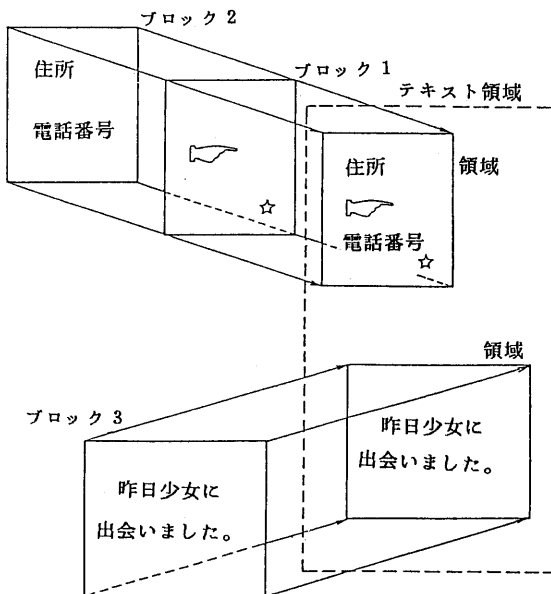


図2-4 テキスト領域と領域とブロックの関係

3. 文書の編集と処理

本章では、第2章で述べた各種のテキストを含んだいわゆるマルチメディアを対象とする文書処理システムを構築するための、文書処理システムのファイル構造や文書処理システムのデータ構造、およびその処理方式を述べる。

本システムを構築するにあたり以下の点を考慮した。

- (1)各種のメディアを統一して扱う文書処理システムである。
- (2)文書処理システムとして汎用性を持ち高速な処理ができるシステムである。
- (3)移植性が高く、H/WやOSに依存しないシステムである。
- (4)容易に機能の拡大や縮小が行えるシステムである。

3.1 文書のファイル構成

本システムは文書の実体とその構造に関する情報を以下の様なファイル構成により管理する。図3-1に、文書ファイルの構成を示す。

3.1.1 ファイル構成

(1)ドキュメント・ファイル

文書の情報本体のファイルである。文字列の情報、書式制御情報、フィールドの参照情報等を保持する。

(2)パラメータ・ファイル

文書名、著者名、作成年月日、変更年月日等の文書全体に関する情報を保持し、また文書全体に関する書式制御情報（文字ピッチ、行ピッチ、文書サイズ等）も保持する。

(3)領域・ファイル

参照領域の識別子、ブロックの識別子、各ブロックの重ね合わせに関する情報等の情報を保持する。

(4)ブロック・ファイル

各々のメディアの情報の本体のファイル。その情報形式は、各メディア毎の表現形式に従う。またそのブロックファイルが参照されている回数等の情報も保持する。

(5)論理構造ファイル

文書の論理構造（目次、章、節、項等）に関する情報の管理を行う。論理構造ファイルはドキュメントの内容と論理構造とを対応させたファイルである。論理構造を用いてドキュメントの検索や編集を行う際に用いる。

(6)レイアウト・ファイル

文書の物理構造（ページに対するレイアウトの情報等）に関する情報を持つ。レイアウト・ファイルはドキュメント・ファイルの内容とページへのレイアウトを対応させるファイルである。表示や印刷等を行う場合や、文書交換用通信プロトコルへの変換を行う場合に用いる。

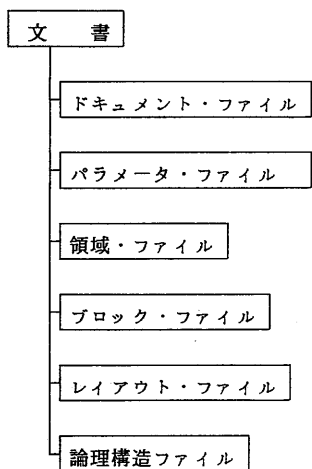


図3-1 文書のファイル構成

3.1.2 文書の構成

本システムでは、文書に関連する各種属性の情報（文書名、著者名等）、及び文書全体に係わる書式制御情報をパラメータ・ファイルに保持する。ページ単位、またはページ内の1部分のみの書式制御情報はドキュメント・ファイル中に書式制御情報として保持する。このような構成にする事により文書全体に影響する書式の変更は、パラメータ・ファイルを変更することにより変更でき、文書の部分的な書式の変更は、ドキュメント・ファイルを変更することによりでき、容易に文書の書式の変更を行うことが出来る。

3.1.3 ドキュメント・ファイルの構造

ドキュメント・ファイルは以下の情報により構成される。

- ①文章情報
- ②書式制御情報
- ③領域参照情報

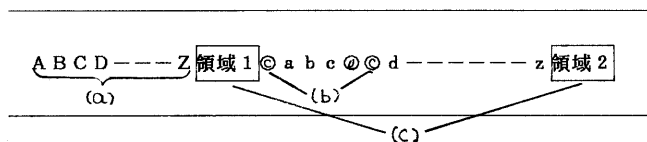


図3-2 ドキュメント・ファイルの構成

図3-2にドキュメントファイルの内容を示す。文章情報は、図3-2の(a)に示す用に文字列(英数字、漢字、記号等)の情報列である。

書式制御情報は、図3-2の(b)に示す部分的(ページやページの1部分)に書式制御(インデント、行組、フォントの変更など)を行うための情報列である。領域参照情報は図3-2の(c)に示すように領域・ファイルを参照している事を示す情報列である。

ドキュメント・ファイルはこれらの情報を1次元的に配置しそれぞれの情報列を一つの情報単位として同等に扱ったファイルである。この1次元情報を用いて順次2次元的に配置することにより文書を作成する。図3-3にその情報を用いて2次元的に配置した図を示す。このように全ての情報を1次元的に、また同等に扱うことにより文字列の移動、複写、削除は1次元情報にたいして操作を行うことにより可能である。書式制御情報も移動、複写、削除の文字列の操作と同様な処理で

行えるので容易に書式の変更ができる。また、領域の移動や複写、削除も同様に1次元情報に対して領域参照情報を移動、複写、削除を行う事により可能であるので容易にできる。本文と領域はあるルールのもとに接続されており、自動的に2次元に配置を行う事が可能である。

3.1.4 領域・ファイルの構造

領域・ファイルは、図3-4に示すようにブロック・ファイルや他の領域・ファイルへの参照を示す情報、領域内のブロックの位置情報、ブロック/領域の重なり順序、文字列の排斥の有無、ドキュメント・ファイルや他の領域・ファイルから重複参照可能なように参照回数の情報を保持する。

3.1.5 ブロック・ファイルの構造

ブロック・ファイルは、各メディア情報の本体と、領域・ファイルから重複参照可能なように参照回数の情報を保持する。ブロック・ファイルと領域・ファイルの構造を図3-4に示す。領域・ファイルとブロック・ファイルは、階層的な構造をしているためブロックや領域の参照、移動、削除、ブロック同志の重ね合わせ等も容易に出来る。

3.2 本システムの処理方式

本システムの処理方式を考える上で3.1で述べた点を考慮すると以下の点が問題となる。

- ①H/WやOSに文字列処理やグラフィック処理やイメージ処理、入力処理等のサポートレベルに違いがある。
- ②各々システムで必要な機能のレベルに差がある。

以上の問題点を解決するために、図3-5に示すように、本システムでは、

- (1)H/Wに依存する部分(IHP: Imaginary H/W Primitive)と、H/Wに依存しない部分(ISP: Imaginary S/W Primitive)に分割し、各々影響しないようにする。
- (2)OS機能に依存しやすいイベント入力やキーボード入力等は、IHPにイベント(キーボード)バッファを備えるなどによりOSに対して独立性の高いシステムにする。
- (3)IHPは高速処理を行うような処理形態にする。

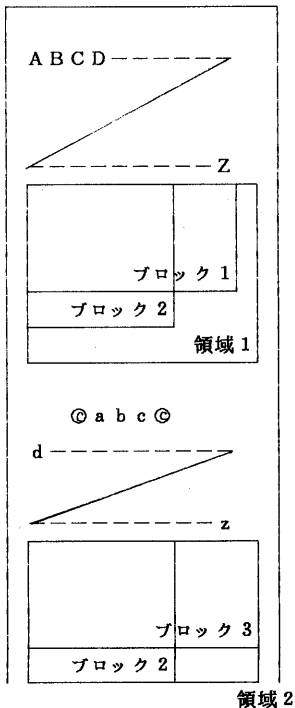


図3-3 2次元配置の構成

(4)ISPは、3.1で述べたファイル構造を持ち、その処理は機能を分割しモジュール化することにより、機能を簡単に抽出する事ができるようにする。

以上のように作成することにより、H/W独立性およびOS独立性の高いシステムを構成することができる。

3.2.1 IHPの処理機能

IHPの基本機能として、以下の機能が必要である。

①文字表示処理

文字や文字列の表示処理をおこなう。

②ベクター表示処理

直線や円や矩形、多角形、円弧等の表示処理をおこなう。

③イメージ表示処理

イメージ・情報の入出力、イメージの拡大/縮小、回転等の処理をおこなう。

④ファイル処理

ファイルに対す入出力やファイルの作成、移動、複写、削除等の処理をおこなう。

⑤入力処理

キーボードやポインティングデバイス（マウスやタブレットなど）に対する入力処理をおこなう。

⑥拡張機能処理

個々のH/Wに特有の処理をおこなう。たとえばプロッタの制御や、音声入力等の処理をおこなう。

3.2.2 ISPの機能

ISPの機能として、以下の機能が必要である。

①文字列の操作

ドキュメント・情報に対して、入力や削除、移動、複写等の処理を行う。（書式制御情報や領域参照情報も同様に扱う。）

②図、表の表示処理

ビジネスグラフや帳表等の表示を行う。

③ベクトルの管理及び表示

ベクトル（直線、円等）のセグメント管理やクリッピングの処理、ベクトルの表示をおこなう。

④入力処理

文字列の入力に対してプロンプトの表示や文字列のエコー出力、ポインティング・デバイスからのイベント入力、日本語処理（かな漢字変換等）の処理をおこなう。

⑤文字列の表示

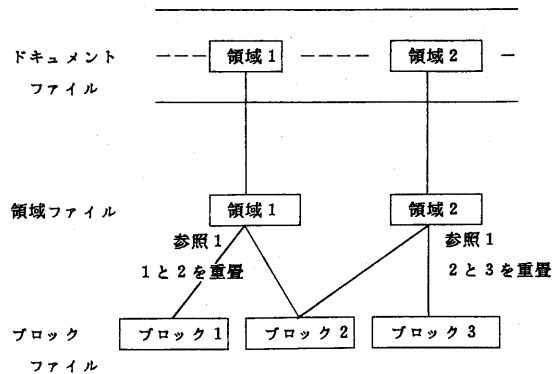


図3-4 領域とブロックの関係

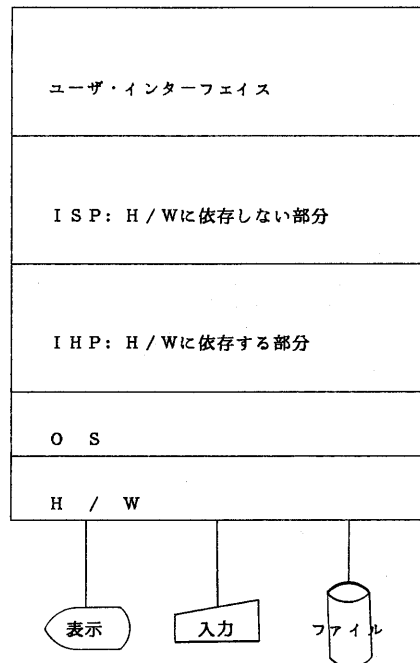


図3-5 本システムの機能レベル

ドキュメント・情報とパラメータ・ファイル及びレイアウト・ファイルによって文字列の表示をおこなう。

⑥各メディアの表示

各メディアの情報を、ブロック・ファイルの情報とレイアウト・ファイルにより表示する。

⑦レイアウト・ファイルの再構成及び論理構造ファイルの再構成

レイアウト・ファイルの新規作成やドキュメント・ファイルやパラメータ・ファイルの変更にともなうレイアウト・ファイルの再構成をおこなう。

論理構造ファイルも同様に行う。

⑧パラメータ・ファイルの再構成

パラメータ・ファイルの新規作成や、ファイルの内容の変更をおこなう。

⑨拡張機能に対する処理

H/Wに固有な処理によって必要な処理を行う。

⑩共有処理

そのほか各プログラムにおいて共通に用いられている処理を行う。

3.3 本システムの使用例

以上のようにIHPとISPの機能を明確に分割することにより、H/WやOSの変更に伴う部分はIHPの変更を行うことによりでき、文書処理システムの機能の追加は、ISPの変更によりでき、機能の縮小は、必要なモジュールのみ抽出することにより行う事ができる。

例えばこのシステムの一部の機能を用いて別のサービスを作成することができる。例えばOHP作成システムのようにベクトル処理と文字列処理のみが必要な場合はそれらの機能のみを抽出してシステムを作成することにより、小規模なシステムが作成される。また、帳表システムのような場合はISPの表の機能のモジュールを改造し高機能にすることにより、文書処理システムより表機能が充実した帳表システムが作成される。このようなOHP作成システムや帳表システムで作成された情報は文書処理システムと同じ情報形式であるため情報の交換が可能である。

IHP、ISPの機能を図3-6に示す。

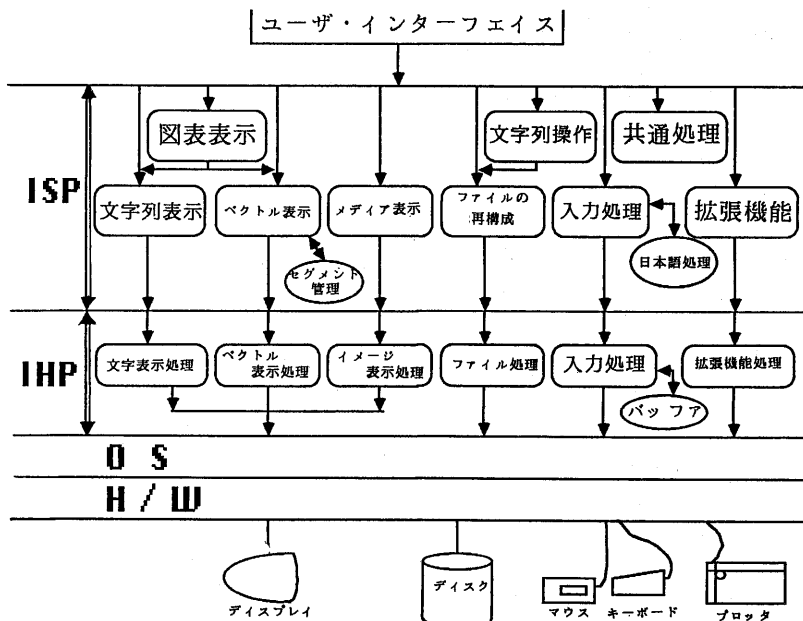


図3-6 IHP、ISPの処理の流れ

4. おわりに

第3章で挙げた諸要件を満たす文書処理を目指し、文書構造を定義し、これに基づいた、文書情報の管理体制を検討した。更にこの管理体制下で、作成・編集処理を実施する方法について述べた。三種類のテキストを基にした本報告による文書処理方法により、今日、我々が一般的に用いる文書の類の、作成・編集は可能であることを、我々の試作システム上で確認した。

文書は必ずしも始めの方から順次作成されるとは限らない。又は、ある章は未完のまま他の章の作成・編集の作業に移ることもある。そして、表、図のテキストは、あらかじめ作成・編集されたり、後で作成・編集されることもある。このような、作成の順序が自由に変えられても、容易に対応できるように、処理形態、及びユーザインターフェースが柔軟に作成されることが重要となってくる。このような観点からも、文書処理モデルの検討が今後必要と考える。

参考文献

- [1] Y. Sakashita et al "Document Processing System (MIDOP)" ICTP'83 J1
- [2] 坂下善彦, 渡辺治『文書処理システム』昭和59年度電子通信学会総合全国大会 15-6
- [3] ISO DP 8613/2 "Information Processing - Text preparation and Interchange - Text Structure Part 2: Office Document Architecture
- [4] CCITT 勧告草案T'73 "Document Interchange Protocol for Telematic Services
- [5] 伊藤均その他『マルチウィンドウを用いた文書処理システム(II)』情報処理学会第26回全国大会 44-10
- [6] 清水剛, 福岡久雄, 坂下善彦『マルチウィンドウ表示方式』情報処理学会マイクロコンピュータ研究会資料 35-1
- [7] R. Furuta, J. Scofield, A. Show "Document Formatting Systems: Survey, Concepts, and Issues" ACM Comput. Survey Vol. 14 No. 3