文書の構造に注目した全文データベース検索システム

原　正一郎（国文学研究資料館）
根岸正光（学術情報センター）
芝野耕司（東京国際大学）
安永尚志（国文学研究資料館）

要　旨　全文データベースは利用者自身による多様な検索要求に応えられるものとして期待されている。国内の学術情報機関においてもデータベース・サービスの一環として，さまざまな全文データベースの研究・開発およびサービスが行われている。一方，これらの過程でユーザ・インタフェイスの貧弱さ，あるいは検索ノイズの混入といった，全文データベースの抱える諸問題も明らかになってきた。本稿は，これまでの全文データベース開発の成果を踏まえた，新しい全文データベースシステム開発への取り組みを述べたものである。

Full text database is expected to be useful for the various types of information retrieval. National Institute of Japanese Literature (NIJL) has been creating full text data of Japanese classical literature, but the problems of creating effective user interfaces for classical text manipulation is still untouched. On the other hand, National Center for Science Information Systems (NACSIS) has been servicing full text databases as its service repertories, which appeared many problems on the full text databases, i.e. poor user interfaces, difficulty of arranging text data and a lot of noises. Base on above experiences, we have been pursuing the research to resolve these problems.

This paper presents our trial of devloping "new full text data base system". Peculiarities of the system are, 1) Data model is based on the logical structure of the text, 2) SGML is introducted to present the logical structure of the text, 3) New query language (DQL: Document Query Language) is created to manipulate nested structure of the text, 4) Graphical user interface is examined for intuitive information retrieval, 5) Client-server system is introducted for effective data processing.

Though, this system is made for current Japanese texts, we believe it is available to the classical ones.

1．はじめに

全文データベース（Full Text Data Base：以下では全文DB）は，抄録型データベースあるいはファクト型データベースに対する用語で，雑誌・書籍等の文書全体を対象としたデータベースであり，検索のみならず全文の入手がオンラインで可能であるという特徴を持つ。

オンライン・データベース・サービスによる全文DBの成長は著しい。Directory of Online Databases [1]によれば，1980年版に収録されているデータベース500件のうち25件（5％）が全文DBであったが，1984年版では2453件中422件（18％），1987年版では3369件中842件（25％），1989年版では4062件中1381件（34％）というように，絶対数のみならず比率も増加している。これに伴い，ユーザ・インタフェイスの貧弱さあるいは検索ノイズの混入といった，全文DBのかかえる諸問題も明らかになりつつある。

本稿では，このような問題を解決する手段として，「文書の論理構造に注目した全文データベースシステム開発」への取り組みを，学術情報センター（NAtional Center for Science Information Systems：以下NACSIS）における学術雑誌を対象とした全文DBMS（以下では新システム）の開発を例に述べる。以下，第2章では情報検索サービスの具体例としてNACSISの化学全文データベースを取り上げ，現行の全文DBに関する問題点を整理する。第3章では新しい全文DBの枠組みについて考察し，第4章では文書の論理構造を考慮した全文DBの概要について紹介する。

2．全文データベースの現状と問題点

NACSISでは1987年から国内各学会の協力のもとに，学会誌の全文DB化についての検討を開始し，1989年から化学系学会誌掲載論文の全文DB（以下では化学全文DB：正式名称は「学術論文データベース第二系」）の利用者向けサービスを開始した。本章では化学全文DBを取り上げ，現行の全文DBにまつわる問題点および問題解決の方針について整理する[2]。

2．1　化学全文DBの検索[3]

化学全文BDの利用者は検索コマンドを用いて希望する文献を検索する。検索コマンドはNACSISが行っている他の情報検索サービスとほぼ同じである。つまり，一次検索は検索語とフィールドの指定が基本であるが，検索語の指定に際しては前方一致・後方一致・範囲指定などの部分指定や，パラグラフ単位の検索が許される。また二次検索用として文書単位の検索用にAND・OR・NOT・DIFFのブール演算，パラグラフ単位の検索用にPAND（パラグラフ単位での論理積）・POR（パラグラフ単位での論理和）・PDIFF（パラグラフ単位での論理差）のブール演算，さらにいくつかの補助検索コマンドが用意されている。化学全文DBの結果表示も，基本的には他の情報検索データベースと同様であるが，パラグラフ単位あるいは全文出力といった全文DBならではの機能がある。さらに，化学全文DBの特徴として図表のFAXサービスを挙げることができる。つまり，本文データベース中に挿入されている説明文を参照して，希望する画像IDをファクシミリ出力コマンドのパラメータとして指定すると，その図表が利用者のもとへFAXされてくる。このような全文DBに対する利用者側の長所としては以下の事項が指摘できる。

1）全文がオンラインで入手できる。これにより，雑誌類の保管スペースの節約や周辺領域の雑誌購読経費の節約が可能となる。
2）本文をブラウジングできるため，検索の有効性の判定が容易である。
3）網羅的な検索ができる。
4）固有名詞や実験手法など，特定あるいは周辺の情報を利用した検索ができる。

一方，欠点としては以下の点が指摘されている。

1）図表の出力にG3ファクシミリを利用しているため，表示の品質が印刷媒体に比べて劣る。
2）検索ノイズが多い。
3）検索手続きが複雑で，初心者には使いにくい。

出力品位の低さはハードウェアの問題であり，G4ファクシミリなど高性能の製品が開発・普及するにしたがって解決可能な問題であると考えられる。これに対して検索ノイズは本質な問題である。全文DBでは文書全体が収録されているので検索が容易であるように思われがちであるが，全文であるがゆえに日常的に使用している言葉は殆ど含まれている。したがって，普通に思いつく言葉（キーワード）で検索を行うと，かなりの量の不適切な文書（ノイズ）がヒットしてしまう。これがノイズの問題である。ノイズが増える原因として，全文DBが従来の情報検索データベースと同様に，文書を単なるフィールドの集合とみなしている点をあげることができる。一方，我々が雑誌を通覧して必要な文献を探す場合，「Aという言葉が章タイトル中に現れていれば探している論文の可能性があり，さらに，その章中にBという言葉があれば，その可能性はさらに高くなる」といったように文書の構造を考慮することで，言葉の適

切な使い分けと検索集合の効率的な絞り込みを行っていると考えられる。このような文書の構造を考慮した検索法という視点は，効率的な検索システムの実現において重要であると考えられる[4]。

ところで，現状のシステムではノイズを除去するために，検索を幾重にもかけたり，ヒットした周辺をブラウジングする事によって検索の妥当性を推測している。しかし，検索そのものがコマンドによるものであり，しかもブラウジングの範囲も限られているため，初心者には使いにくいものになっている。ページをめくる要領で全体的に眺めたり，コマンドを意識せずマウスなどによる直感的操作で検索できるGUI（Graphical User Interface）の開発が望まれるところである。このような直感的操作を可能にするためには，文書構造を意識した分かりやすいレイアウトを作成する必要がある。

2．2　化学全文DBの作成

化学全文DBは，化学系各学会（高分子学会，日本農芸化学会，日本薬学会）から雑誌印刷用CTS（Computer Type Set：電算写植）用磁気ファイルの提供を受けてNACSISにおいて作成されている。現在，化学系学会誌に限らず多くの学会誌がCTSにより作成されており，このような学会には機械可読な文書ファイルが存在している。しかし，これらのファイルは写植用であるため，データベース化に際してNACSISでは以下のようなデータの変換作業を行っている[5]。

1）学会より提供されたＣＴＳファイルから制御コードなどを削除して，平文のテキストファイルに変換する。
2）平文テキストファイルをエディタ上に呼び出し，著者名・所属機関名などの論理項目識別子を挿入して本文データファイルを作成する。
3）全文中からキーワードの抽出を行う。化学全文DBでは日本語のわかち書

きと「単語」の抽出にHAPPINESSを利用しており，原則としてストップワード（不要語）を除く全単語がキーワードとして抽出される。キーワードとそれに対応したデータベース内のレコード情報は検索用のインバーテッドファイルに蓄積される。

4）図表や数式を，NACSISが割当てた画像IDをキーにして光ディスクに焼き込む。ここでは文字データとして処理可能な簡単な表や数式と，画像データとして蓄積すべき図表や数式を仕訳して，後者は画像データベース上の画像IDを割付け，同時に本文中の該当箇所にこのIDを挿入するなどの作業を伴う。

5）本文データファイルとインバーテッドファイルをオンライン情報検索システムにロードして利用者に公開する。

データベース作成過程において以下のような問題点が指摘されている。

1）写植機ごとに独自の制御コード体系を持っているので，学会誌ごとに変換プログラムを作成しなければならない。

2）時間上の制約から，最終校正は必ずしも写植ファイルの更新によらず，版下に校正部分を直接貼り込むという便法が行われていることが多い。校正によって生じたCTSファイルと雑誌間の相違部分は，校正原稿を読みながら手作業で修正せねばならない。

3）CTSファイルはフォントの都合上，論文単位ではなく表題・著者など論理項目別に編集されている場合が多い。そのため，論理項目単位のファイルを論文単位のファイルに再編集する必要がある。

4）論理項目識別子挿入の際には，著者姓名の逆転などの正規化，著者と所属機関との対応，本文と脚注との対応など，編集者の能力が要求される。

5）特殊記号や化学記号の読み下しなど，専門知識が要求される。

6）キャプションのない図，数式，化学構造式などにキャプションと画像IDを補う。

1～3は印刷とデータベース作成における工程差によるものであるが，この差を埋めるには相当の作業量を要する。4～6は学術論文についての編集知識が要求される事項である。実際，データベースの作成においてはかなりの水準の要員が必要であるが，これらの要員を継続的に確保することは困難である。つまり，現状の全文DBの作成法は作業的にも費用的にも効率が低いと言わざる得ない。文書ファイルを受け取る側で，データ変換作業を効率的に行えるようにするには，印刷物としての書式つまり割付構造（layout structure）情報だけでは不十分で，表題・章・段落など文書の論理構造（logical structure）情報が不可欠である[6]。

以下の章では，これらの点を含め，新しい全文データベースシステム開発のための基本的考察を行う。

3．新しい全文データベースシステムの枠組みについて

本章では新しい全文DBシステム開発のための基本的な枠組みを，1）データの構造，2）データの記述法，3）データモデル，4）データの操作，という視点から考察する。

3．1　全文データベースにおける文書の論理構造

文書をイメージとして見ると，文書は字下げ・改行・改ページなどの割付構造によって区別された要素から構成されているように見える。本稿でいう「文書の論理構造」とは，章・見出し・段落など要素間の相互的構造のことである。

つまり，意味的にまとまった単位であり，かつ表示上も一定の規則によって相互に区別できる単位を文書の論理要素とし，これらの論理要素の相互関係から構成される構造を文書の論理構造とする[7]。直感的には表題・著者名・要旨・章・章見出し・段落などが論理要素に相当し，これらが組み合わされて文書を構成する。文書の論理構造をこのように考えると，これは「文書」をルートとする木構造で表現できる（図1）。

2章で述べたように，従来の全文ＤＢにおける検索ノイズの多くは，データベース中に文書の論理構造情報が欠落していたためであった。一方，今日では多くの文書がワードプロセッサなどで作成されている。したがって，これらの機械可読化された文書がその論理構造とともにデータベースに取り込めれば，データベースの作成・検索にかかわる労力を大幅に削減することが可能となる。

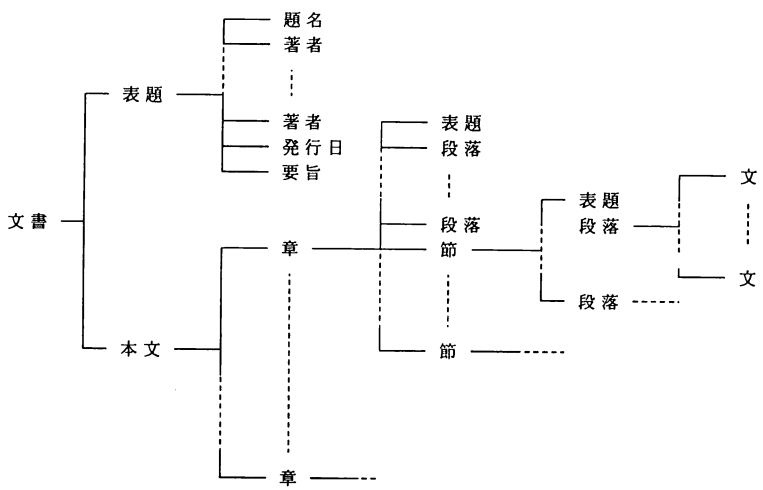

図1　文書の論理構造例

3．2　全文データベースにおけるデータの記述法

文章の論理構造を記述できる国際規格としては，ODA（Open Document Architecture）とSGML（Standard Generalized Markup Language）が普及しつつある。

（1）ODA

近年では，ワードプロセッサやファクシミリなどのOA機器を利用して文書の作成や通信を行うオフィスが増えている。しかし多くの文書はオフィスごとの独自な書式に基づいて作成されているため，別のオフィスとの間では文書を交換しにくいという問題が顕在化している。

このためCCITT（Consultative Committee for International Telegraph and Telephone：国際電信電話諮問委員会）では，どのようなオフィス文書でも相互に交換し共通に扱えることを目的とした文書通信のための標準を作成した。これは，文書をオフィス間で相互に解釈・編集できるように文書の構造を統一的に規定したODA（Open Document Architecture：開放型文書体系）と，これに準拠した文書（ODA文書）を通信するための文書通信プロトコルDTAM（Document Transfer And Manipulation：文書の転送および操作）からなる。これらは，ISO（International Standard Organization：国際標準化機構）のODA（Office Document Architecture）と同一の概念である。ODAはビジネス上の文書を扱うものであるから，文字のみならず図形などの多様なオフィス文書（マルチメディア文書）の相互交換を主眼としている点に特徴がある。

ODAでは，文書の構造を論理的な側面（論理構造）とレイアウトの側面（レイアウト構造）から捉えている。文書の構造を表現する要素はオブジェクトとよばれ，論理構造では章・節などが，レイアウト構造ではページ・フレームなどがこれに相当し，文書の構造はオブジェクトの階層的な木構造として表

現される。これらのオブジェクトは性質に応じて，一般の文書に共通する構造（ジェネリック構造）と，ある文書に特定される構造（特定構造）に大別される。

オフィスにおける手紙を例にすると，社章・日付・住所・本文・挨拶・署名などは共通の様式をとる場合が多い。オブジェク指向的な視点からは，共通の性質をもつオブジェクトはオブジェクトクラスを形成するとみなせるが，このクラスの持つ共通の構造をジェネリック構造という。ジェネリック構造を導入することにより，文書の持つ情報が共通化でき，伝送効率の向上や，生成された文書の一貫性を保つなどの効果が期待できる。これに対して，特定構造は具体的な文書（オブジェクトと指向でいうところのインスタンス）に固有な構造であり，各オブジェクトには，特定の文書内容が対応づけられている。

ODA 文書を作成するにはジェネリック論理構造に従って原稿を入力すればよいが，ジェネリック論理構造の定義記述言語は規定されていないので，任意に作成される。ODA では，文書編集の過程で作成された特定論理構造（構造記述子＋テキスト）の記述子を認識する簡単なパーサ・プログラムを用いることにより，自動的にデータベース用のデータを作成することが可能である。

（2） SGML

SGML（Standard Generalized Markup Language）は1986年に ISO8879-1986として制定された標準一般化マーク付け言語で，文書の表題・著者・要旨などの書誌データに加えて本文中の章タイトル・節タイトル・段落・文などの文書の論理構造を表現するための言語規格である。つまり，ODA が通信分野から生まれた規格であるのに対し，SGML は出版の分野から発生した規格であると言える。

SGML には，

1）文書構造のメタ記述，つまり文書構造を記述するための文書型定義機能

（DTD : Document Type Definition）

2）DTD で定義されたタグを用いてマーク付けされた文書を解析する機能

3）省略された文書内容についてのマーク付けの解釈機能

4）文書構造定義の省略機能

5）図表など端末から入力ができない要素を文書中で扱えるようにする要素参照機能

6）文書清書系サポート

7）形式的記述

といった多くの機能を持つ。SGML は文書構造記述能力の高さから，文書情報交換のための規格として注目を集めつつある。SGML にはいくつかの文書構造記述子が用意されており，これらを用いることによって文書要素型の命名と構造を自由に記述できる。例えば，図1の木構造は次のようにマーク付けできる。

```
<!ELEMENT  文書（ 表紙(題名，著者+，発行日，要旨)，
                    本文(章(表題，段落x，節(表題，段落*)*)+) >
<!ELEMENT 題名        (#TEXT)                                  >
<!ELEMENT 著者        (#TEXT)                                  >
<!ELEMENT 発行日      (#TEXT)                                  >
<!ELEMENT 要旨        (#TEXT)                                  >
<!ELEMENT 表題        (#TEXT)                                  >
<!ELEMENT 段落        (#TEXT)                                  >
<!ELEMENT 節          (#TEXT)                                  >
```

さらに，この DTD にしたがって本稿のマーク付けを行うと以下のようになる。

〈文書〉

〈表紙〉

〈題名〉文章の構造に注目した全文データベース検索システム〈/題名〉

〈著者〉原　正一郎〈/著者〉

〈著者〉根岸　正光〈/著者〉

〈著者〉芝野　耕司〈/著者〉

〈著者〉安永　尚志〈/著者〉

〈発行日〉1993年3月31日〈/発行日〉

〈要旨〉全文データベースシステムは.....〈/要旨〉

〈/表紙〉

〈本文〉〈章〉〈表題〉はじめに〈/表題〉

〈段落〉全文データベース.....〈/段落〉

〈段落〉オンラインデータ.....〈/段落〉

〈段落〉本稿では.....〈/段落〉

〈/章〉

〈章〉〈表題〉全文データベースの現状と問題点〈/表題〉

〈段落〉NACSIS .....〈/段落〉

〈節〉〈表題〉化学全文 DB の検索〈/表題〉

〈段落〉化学全文 DB .....〈/段落〉

.....

〈/節〉

.....

〈/章〉

.....

〈/本文〉

〈/文書〉

したがって，CTS用ファイルがSGML準拠であり，DTDと文書を同時に受け取ることができれば，データベース作成の問題点である，テキストの切り出しやタグの挿入などについてはほぼ解決できることになる。つまり，DTDで規定した論理構造に従って原稿を入力してSGML文書を作成し，これ（タグ＋テキスト）を簡単なパーサで解析することにより，自動的にデータベース用のデータを作成することが可能となる。もっとも，SGML自体はシンタクス記述のためのメタ言語であるから，実質的な規格はこれを用いてさらに規定する必要がある。米国などではAAP（American Association of Publisher）の国内規格などが成立している。

このように，文書が論理構造情報とともに機械可読化されていれば，ODAあるいはSGMLのいづれの規格であっても，文書情報をその論理構造情報と共にデータベースに取り込む事が可能であるが，

1）ODAは学術出版物を対象とした場合の論理構造記述能力が低く，執筆からデータベース作成・印刷までを一貫したシステムとして実行するためのデータ源としては不十分である，

2）ＳＧＭＬは論理構造記述能力は高いが，実際の印刷に必要な標準ができていないため，実際に採用できるようになるまでには暫く時間がかかる，

といった問題点がそれぞれの規格に存在する。しかし，論理構造の記述能力の高さに注目し，本稿で述べる新システム用の文書構造記述にはSGMLを採用した。

3．3　全文データベースにおけるデータモデル

全文を対象としたデータベースを考えた場合，対象となる文書を性質に応じて幾つかの種類に分けて考える必要がある[8]。

（1）明確な構造をもたない情報

抄録あるいは新聞記事などは短い文書であり，これらは明確な文書構造を持たない。見出・記事本体・段落などといった構造を捉えることもできるが，個々の文書は量的にも内容的にも一塊のものとして捉えられ，検索対象も文書全体である場合が一般的であり，構造が無いものとしても問題は少ない。

（2）表として捉えられる情報

従来の二次情報のように，一群の文書を行に，内容を属性に応じて列にマッピングできる情報は，表として捉える事ができる。辞書を例にとれば，一つの項目を行とし，著者や発音記号などの情報を列として捉えることにより，これらに関する情報を関係データベースの表として格納することができる。

（3）同一の構造をもつ情報

一種類の学術雑誌を取り上げると，そこに掲載されている論文の文書構造はほぼ同一とみなせる。これは，執筆要綱が学会ごとに規定されているためである。この場合，SGML による DTD を設定し，これに従って文書を作成することが可能となる。

（4）異なった構造をもつ情報

複数種の雑誌等を対象とする場合，個々の雑誌に対しては特定の文書構造を設定する必要がある。しかし，各々の文書構造からの一般化が可能である場合も多い。例えば，学会誌を対象とすれば，表題，抄録，著者，所属，章題および段落などといった一般的な文書構造を抽出することができよう。

これらのうち（1）については新聞記事データベースなどとしてサービスが行われている。（2）は，明らかに従来の関係データベースで直接扱うことが可能である。また（1）についても，何らかの属性情報を設定する事により，

（2）と同様に関係データベースととして処理することも可能である。新システムでは学会誌用全文データベースを目指しているので，基本的には（3）を対象とし，将来的には（4）までを扱えることを目標としている。

3．4　全文データベースにおけるデータ操作

データベース管理システムが情報技術基盤の一つとみなされていた反面，検索システムはデータベース管理システムのアプリケーションとして位置づけられていた。このため検索システムはデータベース管理システムの一部として実現されてきた。SQL（Structured Query Language）も，関係データベースシステムの管理機能とインタフェース機能を規定するために開発された言語であったが，次第に事実上唯一の標準としての地位を固めつつある。

SQLは関係データモデルに基づく言語であるため，対象データを行と列からなる表として扱う。具体的には，表に対して関係演算を基本とした操作対象の指定（SELECT），挿入（INSERT），更新（UPDATE），削除（DELETE）の各操作を行うことができる。

初期のSQL規格には，表間の関連に関する外部キー制約が規格化されていないなど多くの問題点を抱えていたが，1990年の改正では，参照制約を保証する機能が導入され，同時に日本語をサポートする機能も規格化された。JIS/ISOでは引き続きSQL言語の拡張を行っており，例えばSQL2では，一時表とすべての関係演算子を直接サポートすることができる。このSQL2の新しい機能により，文字列の部分照合検索を含めた対話的な情報検索システムは，ほとんど直接的に実現できるようになった。

しかし，SQLで扱えるデータ構造はきわめて限られている。すなわち，SGMLの記述を用いるとSQLで対象となるデータ構造は，次のようなもの（平坦な表）に限られる。

〈!ELEMENT 表 ((列1，列2，列3)*)〉

3．6　新しいデータベースシステムの要件

さて同一の構造を持った（学術雑誌）全文DBを対象とする場合，3．2で述べたように文書の論理構造はSGMLで規定されるDTDの記述能力の範囲で十分に対応可能である。実際，新システムではSGMLのDTDのうち，グループ化，出現標識，連結子の三つのクラスの構造化演算子がサポートされている。

ところで，SGMLの持つ諸機能のうち全文DBに関連する機能として重要なものは，

1）文書構造のタグによる識別

2）タグによる文書構造の識別

である[9]。1）の機能を利用することにより文書データを自動的にデータベースに投入する事が可能となる。一方，2）の機能を利用すると，「〈章〉の〈表題〉に"国文学研究資料館"を含む章で，"データ転送"を含んだ〈節〉の〈表題〉を取り出せ」という問い合わせが可能となる。これは，文書中にSGMLのDTDによって〈章〉，〈表題〉，〈節〉のタグと，〈章〉が〈表題〉と〈節〉を包含するなどの定義がなされていることを前提としている。

しかし，SGML自身には検索機能は用意されていないので，データ操作を行うためには外部の機能を導入する必要があるが，このような問い合わせをサポートしうる標準的データベース用インタフェイスはSQLのみである。しかし，SQLは対象を単純な表形式のデータに限定しており，「入れ子」構造などに対しては不十分である。

このような問題を解決する手段として，新システムではSQLを拡張してSGMLにおける文書構造記述子をサポートできるような文書ベース言語DQL（Document Query Lanuage）を設計した。

4．文書の論理構造に注目した全文データベースシステムの構築

新システムはメインフレームを主体としたサーバ系とワークステーションを主体としたユーザ系および両者を結ぶ通信系から構成される。

サーバ系では大量のデータを蓄積する必要があること，大量のデータに対して高速検索で実現できなければならないことから，メインフレームを利用した。

一方，検索要求は多様であり，検索モデルを予め作成しておく困難である。さらに，DQLは繰り返しなどを含む複雑な検索命令を記述できるが，これを命令文として書き下すには熟練を要する。そこで，新システムではグラフィックを利用して検索操作を直感的に行えるようなインタフェイスを考えており，これにはワークステーションが適している。

本章では，新しい全文DBシステムの概要を，データ作成，サーバ系，検索言語DQLおよびユーザ系に視点からのべる。

4．1　データ作成

新システムのデータベース作成にはSGMLに基づいてタグ付けされた文書を利用しているが，その際のネックはタグ付きの文書を作成にまつわる煩雑さである。理想的にはSGML用のワードプロセッサーが利用できればよいが，このようなソフトウェアはようやく入手可能になった段階で，普及にはなお時間を要するであろう。

次善の方法としては簡易タグによる執筆要領の作成とSGMLパーサの利用が考えられている。つまり執筆者向けの簡易タグと執筆要領を作成し，執筆者はこの要領にしたがって各文書要素の始めに簡易タグを挿入しながら文書を作成する。このように作成された文書をSGMLパーサに掛けて正式なSGML文書に変換するのである。簡易タグによる文書の例を以下に示す。

〈論文〉

〈論文情報〉

〈和文誌名〉国文学研究資料館紀要

〈巻〉19

〈号〉1

〈出版日〉〈年〉1993〈月〉Mar.〈日〉31

〈ページ〉〈開始ページ〉1〈終了ページ〉15

〈柱〉〈和文題名〉文章の．．．．．

〈著者名〉原　正一郎

〈要旨〉〈和文要旨〉

〈段落〉全文．．．．．．

このようにして作成された文書をSGMLパーサにかけることによって，以下のようなSGML形式の文書を作成することができる。

〈論文〉

〈論文誌情報　言語="J"〉

〈論文誌名〉

国文学研究資料館紀要〈/和文誌名〉〈/論文誌名〉〈巻〉

19〈/巻〉〈号〉

1〈/号〉〈出版日〉

〈年〉

1993〈/年〉〈月〉

Mar.〈/月〉〈日〉

30〈/日〉〈/出版日〉〈ページ〉

〈開始ページ〉

1〈/開始ページ〉〈終了ページ〉

15〈/終了ページ〉〈/ページ〉〈/論文誌情報〉〈柱〉

〈和文柱題名〉文章の．．．．．〈/和文柱題名〉〈柱著者名〉

原　正一郎〈/柱著者名〉〈/柱〉〈題名　論文タイプ="一般論文"〉

〈和文題名〉

文章の．．．．．〈/和文題名〉〈/題名〉〈要旨〉

〈和文要旨〉

〈段落〉全文．．．．．．

〈/段落〉〈/和文要旨〉〈/要旨〉

SGMLによるタグ付けは非常に煩雑であり，正規のSGML文書の作成を執筆者に依頼することはかなり困難である。この例のように，執筆者には簡易的なタグだけを挿入しながら文書を作成し出版側でSGML文書に変換する方法は，SGML普及の上では大きな示唆を与えるものである。本研究で使用する文書データも同様の方法で作成された。具体的には，学術論文データベースに収録されている論文を中心に収集したサンプルに対して「執筆要領」に基づいたタグを挿入した後に，DTDとともにSGMLパーサを利用してSGML文書を作成した[10]。

4．2　サーバ系

新システムでは，開発コストの軽減をはかるためデータベースの実装レベルでは既存の関係データベースシステムを利用している。ところで，多くのシステムで採用されている関係データモデルには，「リレーションが第一正規形でなければならないとい」う制約がある。一方，文書の論理構造は図1のように木構造として記述できるが，これを素直に表形式に変換すると図2のように，

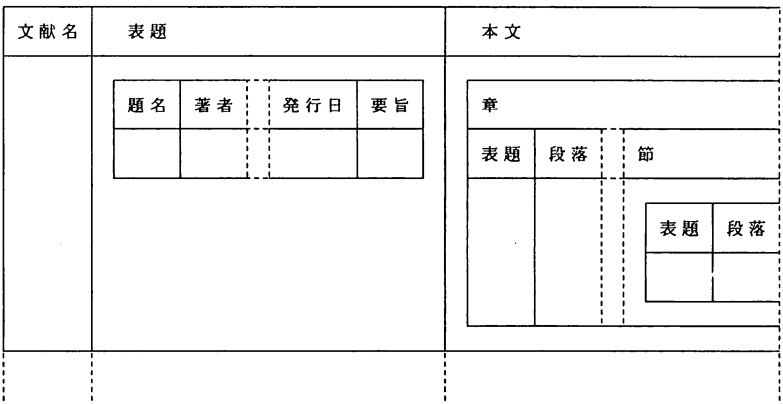

図2　表形式による文書論理構造の記述

表の中に表が含まれる「入れ子」構造となり第一正規形とはならない。そこで本来のDTDが指示する入れ子形式のスキーマに対して第一正規形への正規化操作[11]を行い，平坦な表形式のスキーマに変換した。データはこの変換されたスキーマにしたがってデータベース上に記録されている（図3参照)。

サーバ系とユーザ系はネットワーク（TCP/IP）で結合されている。サーバ系ではユーザ系から転送されてくるDQL命令をDQLサーバと称する論理構造解析ソフトに掛けてSQLを主体とした検索命令に変換する。データベースの検索部ではこのSQLに基づいて上記のデータを検索する。検索結果はSGML形式のデータに再構成されてユーザ系に転送される。

これらの仕掛は既存の機能を利用するための方便であり，入れ子構造のデータとDQLを素直に利用できる検索系の開発は今後の重要な課題である。

4．3　検索言語 DQL [12]

（1）文書構造定義

DQL（Document Query Lanuage）は，SQL に SGML でサポートしている文書構造記述子（グルーピング・出現標識・連結子）を追加したものである。これにより，

1）文書構造定義において入れ子関係を扱える

2）それぞれの要素の出現頻度数を定義できる

3）要素間の順序に関する定義が可能になった

という従来の関係データベースにはなかった機能が付与されたため，SGML と同等の文書構造が定義可能となった。DQL では Won Kim らの complex Object における定義法を基礎にして，SGML のデータ構造記述子をサポートできる記法を採用した[13]。具体的には ISO SQL2 で規定されている拡張 BNF に準じた記法を用いている。DQL による文書構文定義の概要は以下のようになっている。

〈文書定義〉::='CREATE DOCUMENT'〈文書型名〉〈文書構造定義〉

〈文書構造定義〉::='('〈文書要素構造〉')'

〈文書要素構造〉::=〈文書要素名〉〈出現標識〉

{'TEXT' | 〈文書構造定義〉 | 〈連結子〉}

〈出現標識〉::='?' | '*', | '+' | ' '

ただし' 'は1回出現する，'?'は0回または1回出現する，'*'は0回以上出現する，'+'は1回以上出現することを意味している。

〈連結子〉::=',' | '|' | '&'

ただし，','は順序関係がある，'|'はいづれかが出現する，'&'は順序関係がないことを意味している。

この文書構造定義を用いると図1は以下のようになる。

```
CREATE DOCUMENT 学術文献
        (表紙
                (題名 TEXT,
                 著者+ TEXT,
                 発行日  TEXT,
                 要旨 TEXT
                ),
                 本文
                        (章+
                                (表題 TEXT,
                                 段落* TEXT,
                                 節*
                                        (表題 TEXT,
                                         段落* TEXT
                                        )
                                )
                        )
        )
```

なお，この文書構造定義をサーバ系のテーブルに展開すると図3のようになる。

学術論文

論文誌ID	表紙ID	本文ID
A001	B001	C001

表紙

表紙ID	題名ID	著者ID	発行日ID	要旨ID
B001	D001	E001	F001	G001
B001	D001	E002	F001	G001

本文

本文ID	章ID
C001	H001
C001	H002

章

章ID	章No	表題ID	段落ID	節ID
H001	1	I001	J001	K001
H001	1	I001	J002	K002
H002	2	I002	NULL	NULL

節

節ID	節No	表題ID	段落ID
K001	1	L001	M001
K001	1	L001	M002
K002	2	L002	M003
K002	2	L003	M004

題名

題名ID	順序	テキスト
D001	1	文章の. . .

図3　SGML文書のデータベースへの展開（部分）

（2）文書問い合わせ

DQLの問い合わせ式はSQLと類似のキーワードを用いており，その概要は以下の通りである。

〈文書問い合わせ式〉::=〈文書問い合わせ指定〉〈集合演算子〉
〈文書問い合わせ指定〉

〈集合演算子〉::='UNION' | 'INTERSECT' | 'EXCEPT'

〈文書問い合わせ指定〉::=SELSCT〈部分文書構造指定〉
FROM〈原文書型指定〉AS〈部分文書構造指定〉
WHERE〈探索条件〉

〈部分文書指定〉::=--〈文書型定義に準ずる〉

〈原文書型指定〉::=〈文書型名〉

〈探索条件〉::=--ＳＱＬにおける〈探索条件〉に準ずる

具体的には，問い合わせ式は次のような形になる

SELECT〈部分文書構造指定〉 ...
FROM 〈原文書型指定〉 ...
AS 〈部分文書構造指定〉
WHERE〈探索条件〉
〈集合演算〉
SELECT〈部分文書構造指定〉 ...
FROM 〈原文書型指定〉 ...
AS 〈部分文書構造指定〉
WHERE〈探索条件〉

この問い合わせの導出過程は以下の順で行われる。

1）FROM で指定される原文書定義から AS の指定された部分文書構造 SD1 を作成する

2） 1の結果に対して〈探索条件〉を真とする部分文書を選択する

3） 2によって選択された部分文書から，SELECT で指定される部分文書構造指定に基づいて，新たに部分文書構造 SD2 を生成する

4） SD2 を元とする集合に対して〈集合演算〉をとる

ここで，SELECT〈部分文書構造指定〉を SELECT〈式値〉，FROM〈原文書型指定〉を FROM〈表式〉とみなせば，DQL が SQL の拡張になっていることがわかる。

4．4　ユーザ系

DQL は適用範囲の広い検索言語であるが，実際に利用しようとするとかなり面倒である。例えば，「ある節に“認識”と“画像”とが含まれ，これと同一の章内の別の節に“医学”と“眼底”が含まれる本文」のような問い合わせは，DQL により次のように記述される。

```
SELECT 学会誌.本文
FROM 学会誌
WHERE 学会誌.本文.章 IN
    (
      SELECT 学会誌.章
      FROM 学会誌
      WHERE 学会誌.本文.章.節 IN
          (
            SELECT 学会誌.章
            FROM 学会誌
            WHERE 学会誌.本文.章.節 LIKE '%認識%' AND
```

```
                学会誌.本文.章.節 LIKE '%画像%'
        )
    )
AND 学会誌.本文.章 IN
    (
      SELECT 学会誌.本文.章
      FROM 学会誌
      WHERE 学会誌.本文.章.節 IN
          (
            SELECT 学会誌.本文.章.節
            FROM  学会誌
            WHERE 学会.誌本.文.章.節 LIKE '%医学%' AND
                  学会.誌本.文.章.節 LIKE '%眼底%'
          )
      AND NOT 学会誌.本文.章.節 IN
          (
            SELECT 学会誌.本文.章.節
            FROM 学会誌
            WHERE 学会誌.本文.章.節 LIKE '%認識%' AND
                  学会誌.本文.章.節 LIKE '%画像%
          )
    )
```

DQLによる記述が複雑になる理由の１つとして，DQLがその基礎としたSQLの言語的特性をそのまま受け継いでいる点を指摘できる。実際，SQLでも複雑な検索条件を容易に構成するための試みがなされており，その１つとし

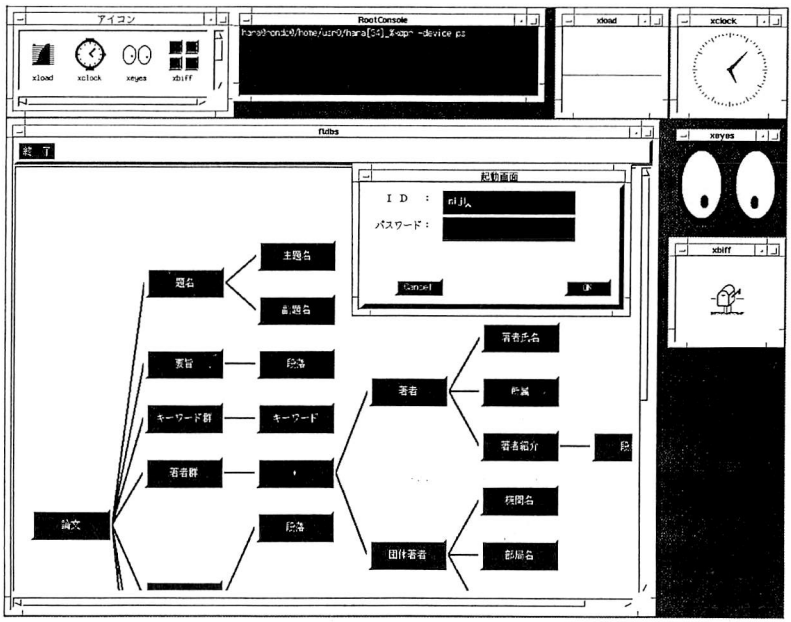

図4　ユーザ系のインタフェイス画面（例）

てGUIの利用を挙げることができる[14]。本システムでも検索条件の構築にはGUIを採用している。本システムのユーザ系では，システムの起動時に4．3．（1）に示されたようなDTDを解析し，文書の論理構造を樹系図として端末上に図形表示する（図4）。この樹系図の各ノードはオブジェクトであり，検索命令はこれらのオブジェクトに対するメッセージ伝達であるとみなされる。

ユーザ系の検索過程を，「題名中に“全文データベース”があり本文中に“DQL”を含む論文を表示する」を例にして述べる。まずノード「題名」の位置でマウスの左ボタンをクリックすることにより検索条件入力ウィンドウをオープンされる。このウィンドウは，要素的検索条件（単体の検索条件）を入

力するテキストウィンドウ，検索述語を選択するメニュー，検索条件の肯定あるいは否定を指定するボタン等から成り立っている。ここで検索条件文“全文データベース”をキーボードから入力し，さらに検索述語“like”を選択すると，要素的検索命令

```
SELECT 論文.題名
FROM   論文
WHERE  論文.題名 LIKE '全文データベース'
```

が生成される。生成された検索命令はノード「題名」の下にアイコン化されて表示される（図5）。同様に，本文ノードに対して“DQL”を入力すると要素的検索命令

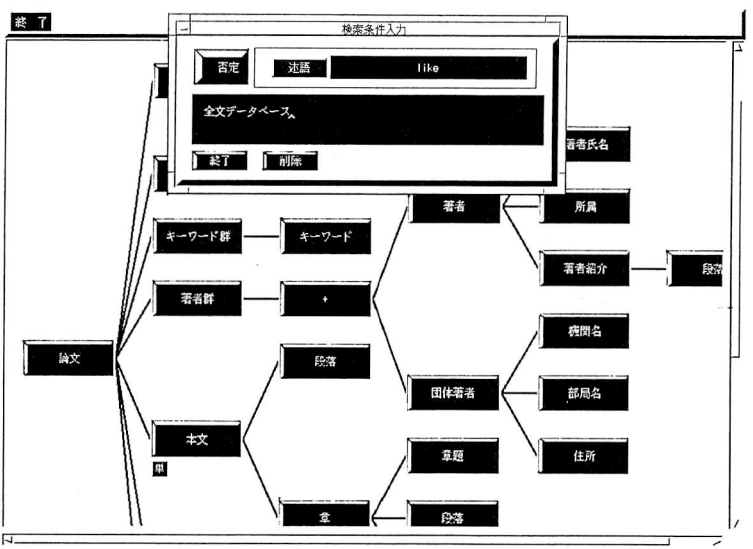

図5　検索例

```
SELECT 論文.本文
FROM   論文
WHERE  論文.本文 LIKE 'DQL'
```

が生成されアイコン化される。要素的検索命令の合成はこれらの要素的検索命令を示すアイコンをマウスで操作することにより行う。ここではノード「題名」とノード「本文」の下にある2つのアイコンの位置でマウスの左ボタンをクリックすることにより，これらの要素的検索命令が合成の対象となる条件であることを指示する（アイコンの色が反転する）。次にノード「論文」の位置でマウスの右ボタンをクリックして，条件合成の対象（合成された検索命令の一番外側のSELECT文の対象）であることを示す。このとき合成する際の条件がプルダウンメニューとして提示されるので，この場合は「AND」を指定する。その結果，検索命令

```
SELECT 論文
FROM   論文
WHERE  論文 IN
  SELECT 論文.題名  FROM 論文  WHERE  論文.題名 LIKE '全文データベース'
AND 論文 IN
  SELECT 論文.本文  FROM 論文  WHERE  論文.本文 LIKE 'DQL'
```

が生成される。

このようにして生成されたDQL検索命令はネットワークを介してホスト系へ送られる。ホスト系から検索結果が転送されると表示用ウィンドウがオープンし，検索結果が表示される（図6）。検索結果はSGML文書であるのでパ

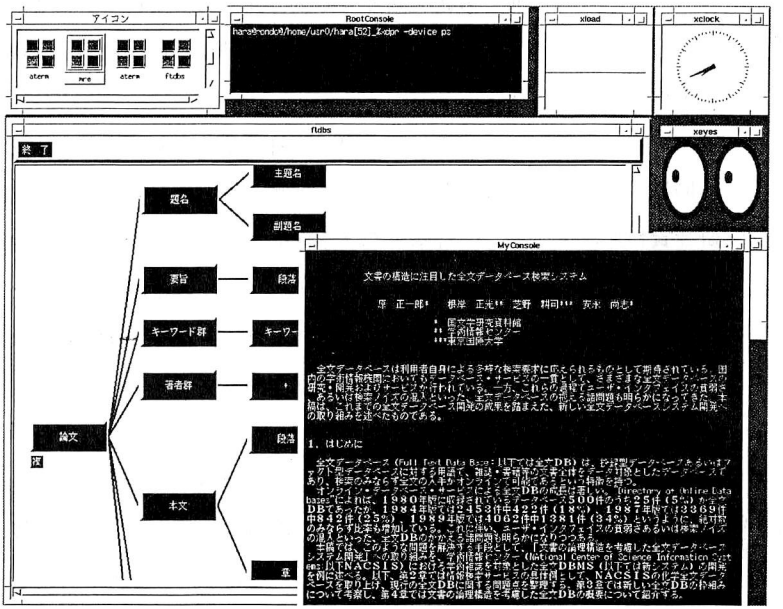

図 6　検索結果の表示例

ーサを利用することにより Tex 形式などに変換可能であるが，これは今後の課題である。

5．おわりに

本稿で述べた情報検索システムは，現在の全文 DB の持つ問題点に対する一つの提案と試みである。しかし，我々が論文誌の目次をながめ，また論文を通覧してゆくという日常的な方法が小量の文献検索に対して有効であることは日

頃経験しているとおりである。そこで大量の論文を収容する全文DBにおいても，このような自然で簡便な方法を通じて必要とする論文を得られるようにすることが研究の目的であり，こうしたシステムの原型が本研究において開発されことを期待している。

ところで，本稿で述べたシステムは「現代」日本語を対象としたシステムであるが，同様のシステムを「古典」文書に適用することは意義がある。実際，国文学研究資料館においても「日本古典文学作品本文データベース」の構築が進められている[15]。古典文献における全文データベース化の問題点は，古典文書の論理構造が現代文書と大きく異なっているため，従来のDTDでは対応できないことである。そのため「日本古典文学作品本文データベース」では作品ごとに特殊なDTDを設定している。一方，欧米を中心として人文科学分野向きの電子文書交換用の規約であるTEI（Text Encodeing Initiative）の作成が活発化しており，これへの日本語の対応が求められている[16]。今後はこのような動向を睨みながら，古典文学全般にわたる標準的DTDの構築が重要になるものと思われる。

本研究は，「文献の論理構造に基づく全文データベース検索システムの研究開発」（科学研究費補助金試験研究（B））の補助を受けている。またDQLの開発には芝野耕司教授（東京国際大学教授）のご尽力をいただいている。特に，本稿のDQLの部分は同教授が作成された研究会資料に負うところが大きかった。記して感謝いたします。

参考文献

[1] Directory of Online Databases, Vol.11, No.3, pp.826, Cuadra/Elsevier, 1990.

[2] 原　正一郎，宮澤　彰，根岸　正光：学術情報センターにおける全文データベース検索サービス，情報処理学会研究報告，vol. 91，No. 41，91－IS－34，1991，pp. 34－2．

[3] NACSIS－IR 総合マニュアル，電気・電子情報学術振興財団，1991.
[4] 長尾　真，谷口敏夫：図書・文献の新しい検索方式，学術研究支援のための高度情報システムに関する研究，(財)関西文化学術研究都市推進機構，pp. 73－125，1991.
[5] 根岸　正光：フルテキスト・データベースの実用化における諸問題――学術情報センターでの事例を踏まえて――，情報処理学会研究報告，Vol. 89，No. 66（89－FI－14－1），1989，pp. 1－9.
[6] 根岸　正光：学術分野における機械可読文書の作成と通信，学術情報センター紀要，No. 2，1989，pp. 43－52.
[7] 影浦　峡，大山　敬三，宮澤　彰，根岸　正光，鳥居　俊一，絹川　博之：文献の論理構造を考慮した全文検索システム，学術情報センター紀要，No. 3，1990，pp. 49－58.
[8] 芝野耕司：全文データベースのためのデータモデルについての予備的な考察，全文データベース研究会資料（NACSIS），1990.
[9] 芝野耕司：SGML と全文データベース，情報処理学会研究報告，Vol. 89，No. 66（89－FI－14－2），1989，pp. 1－8.
[10] 根岸正光：SGML に依拠する全文データベース・システムの研究開発，学術情報センター・ニュース，No. 15，1991，pp. 4－7.
[11] 増永良文：リレーショナルデータベースの基礎，オーム社，1990.
[12] 芝野耕司：DQL，全文データベース研究会資料（NACSIS），1991.
[13] Won Kim et.al.: "Operation and Implementation of Complex Objects", IEEE Trans. Software Eng., Vol.14, No.7, 1988, pp.985-996.
[14] J.Harvey Trimble,Jr., David Chappel: "A Visual Introduction to SQL", John Wiley & Sons, 1989.
[15] 安永尚志：日本古典文学作品本文データベースの開発とデータ記述文法について，国文学研究資料館紀要，No. 18，pp. 1－18，1991.
[16] TEI Steering Committee: "Guidelines for the Encoding and Interchange of Machine-Readable Tesxt", ACH, ALC, ALLC, 1990.