関係データベースによる構造化文書データベースの実現

2C－3

岩崎 雅二郎　　小川泰嗣
（株）リコー 情報通信研究所

1 はじめに

近年文書の構造記述言語である SGML(Standard Generalized Markup Language) のためのアプリケーションが徐々に製品化され始め、SGML 文書の処理環境が整いつつある。これに伴い SGML 文書が増加し SGML 文書データベースの要求が高まっている。SGML 文書を単にプレーンテキストとしてデータベースに蓄積することは容易であるが、SGML で記述された文書の論理構造に基づく検索はできない。そこで、SGML で記述された構造情報を関係データベース (RDB) に実装し、構造情報に依存する検索が可能な実用的な SGML 文書データベースを実現した。本稿ではこの RDB による構造化文書データベースについて述べる。

2 構造化文書データベースの概要

文書は物理的な構造や論理的な（内容からしか判断できない）構造をもつが、通常の文書ではこれらの構造が明示的に記述されていない。しかし、文書の構造が記述されれば、その情報を基に飛躍的に文書の加工が容易になるだけでなく、構造に基づく検索が可能であり文書検索が効率的にできる。

文書の構造を表現するマークアップ言語である SGML が標準化された。図 1に SGML で記述された報告書の構造を示す。この例では報告書は二つの章からなりそれぞれ二、三段落から構成され、さらに二つのコメント文書が付けられている。コメント文書と報告書との文書間リンクは HyTime[1] で記述されている。

SGML を利用して文書を処理するシステムも数多く現れてきた。しかし、このようなマークアップ情報を基に検索できる実用的な文書データベースは少ない。本システムは SGML で記述された構造化文書データベースであり、文書から木構造の構造情報と文書間リンク情報を抽出し、それぞれ RDB で管理する。

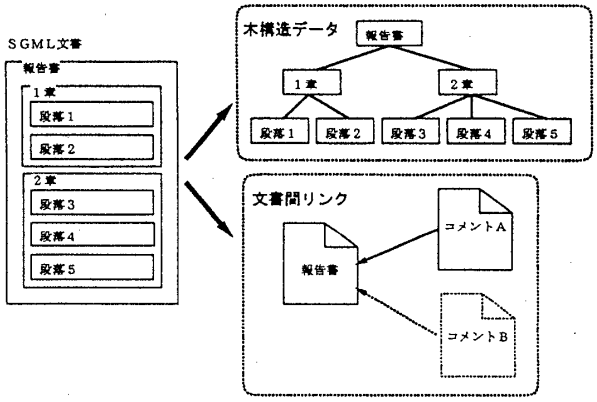

図 1: SGML による報告書

しかし、単純な文書でも SGML で記述した場合、かなり複雑な構造となり前述のように構造情報をそのままデータベースで表現した場合には、RDB 上のデータ構造が複雑になる。その結果、処理時間が増加し実用的な文書データベースを構築できない。

そこで、実際に検索対象となる構造情報はすべての構造情報の一部であることに着目し、本システムでは検索に必要な構造情報のみを木構造で管理し、また、検索を補助する情報を追加することにより高速な検索を可能とする構造化文書データベースを実現した。

3 データ構成

本システムでの構造化文書のデータ構成を以下に示す。

- 木構造データ
- 構造化文書ソースデータ
- 文書間リンク
- 書誌情報

SGML で記述した場合の会議開催通知を例に文書の木構造を図2に示す。SGML では木構造の各ノードをエレメントと呼ぶ。各エレメントはエレメント ID、エレメント名及びエレメントの内容から成る。このような文書を構成するすべてのエレメントを木構造として RDB に

Implementation of a structured document database by a relational database
Masajirou IWASAKI, Yasushi OGAWA
Information and Communication R & D Center
RICOH Co.,Ltd.

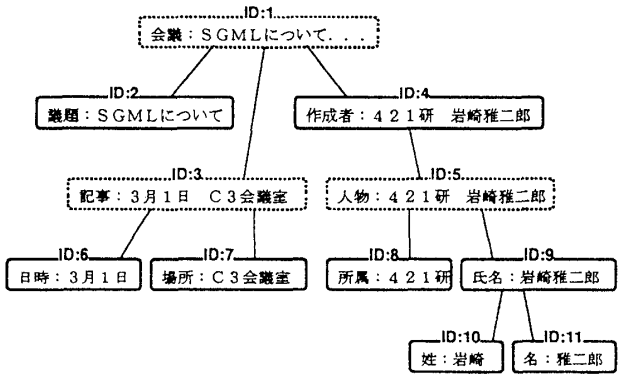

図 2: 構造化文書の構成

表 1: 木構造データ

ID	名前	エレメントパス	親ID	内容
2	題題	／会議／題題	1	SGMLについて
4	作成者	／会議／作成者	1	４２１研．．．
6	日時	／会議／記事／日時	3	3月1日
7	場所	／会議／記事／場所	3	C3会議室
8	所属	／会議／作成者／人物／所属	5	４２１研
9	氏名	／会議／作成者／人物／所属／氏名	8	岩崎雅二郎
10	姓	／会議／作成者／人物／所属／氏名／姓	9	岩崎
11	名	／会議／作成者／人物／所属／氏名／名	9	雅二郎

保管するとデータ構成が複雑になり処理時間が無視できなくなる。そこで、実用上で検索の対象となるエレメント（予めシステムで規定したエレメント）のみをRDBに登録する。図2において点線のエレメントはRDBに登録されていないエレメントを示す。このように本システムでは一部のエレメントしか管理しないので、木構造データとは別にSGML文書の元の文書のデータ（構造化文書ソースデータ）も保管している。

RDB上では図2の木構造データは表1に示すテーブルで表現される。エレメント名、エレメントID、エレメントの内容の他に、親エレメントのID及びルートエレメントから木構造を辿ったエレメントパスをもつ。エレメントパスはUNIXのディレクトリ指定と同様に"/"で各エレメントを区切る。

4 機能

本システムの主な機能を以下に述べる。

4.1 構造情報に依存する検索

各エレメントは木構造とは関係なく独立に検索対象となるだけでなく、木構造に依存する検索も可能である。木構造に依存する検索には２つの方法がある。

- 木構造を辿る検索：エレメントIDを用いて木構造を辿る。図2を例とするとID:10のエレメントからID:11のエレメントを得ることができる。つまりID:10のエレメントの親エレメントIDからID:9のエレメントを得て次に親エレメントIDが9であるエレメントを検索することでID:11のエレメントが得られる。しかし、辿るエレメントごとに検索しなければならないので処理が遅い。そこで前述のエレメントパスによりこの問題点を解決した。
- エレメントパスによる検索：例えば会議開催通知の作成者の姓を知りたい場合、"/会議/作成者/人物/所属/氏名/姓"というエレメントパスを持つエレメントを検索すれば良い。この検索方法ではエレメントを一回の処理で検索できるので処理速度が速い。

4.2 文書間リンクの管理

HyTimeを用いて記述した文書間リンクを一つのテーブルで管理する。このテーブルはリンクを定義している文書、リンクの始点である文書及びリンクの終点である文書の３つを記述するフィールドを有する。ユーザが文書間リンクを定義した文書を作成したり削除したりすることによって自由に文書間のリンクを結合したり切ったりすることが可能である。

4.3 アクセス権の管理

本システムでは文書単位だけでなく、文書のエレメント単位にアクセス権を設定できる。したがって、文書内のある部分は参照できるがある部分は参照できないというように、文書単位だけでなく文書内の細かなセキュリティ管理が可能である。

5 おわりに

検索対象となる構造情報のみをRDB上で管理し、各エレメントの木構造における位置をエレメントパスで表現することで木構造に依存する検索速度が速く実用的な構造化文書データベースを実現することができた。

今後は木構造に依存する複雑な検索要求を高速に処理できるように改良を行なっていく予定である。

参考文献

[1] Information technology – Hypermedia/Time-based Structuring Language (HyTime), ISO/IEC 10744, 1992

[2] F.J.Burkowski, AN ALGEBRA FOR HIERARCHICALLY ORGANIZED TEXT-DOMINATED DATABASES, Information Processing & Management, 28(3), 333-348, 1992

[3] 田中洋一, 文書記述言語SGMLとその動向, 32(10), 1118-1125, 1992