SGML 구조 기반의 문서 저장 및 검색 시스템

김 학균, 조 성배
연세대학교 컴퓨터과학과

A Document Storage and Retrieval System based on SGML Structure

Hak-Gyoon Kim and Sung-Bae Cho
Dept. of Computer Science, Yonsei University

요약

플랫폼에 관계없이 한번 작성된 문서의 정보를 이기종간의 시스템에서 공유하고, 다양한 문서 형식을 지원하기 위해 ISO 8879로 표준화된 SGML이 사용되고 있다. SGML 문서는 문서의 내용 뿐만 아니라 문서의 구조적 정보를 가지고 있다. 점차 증가되고 있는 SGML문서의 구조적 정보를 이용하여 데이터베이스의 구축 및 검색을 위한 검색 시스템에 대한 필요성이 대두되고 있다. 그러나, 기존의 색인어를 이용한 전문 검색 엔진으로는 문서의 구조적 정보를 이용할 수 없다. 따라서, 본 논문에서는 문서의 구조적 정보를 이용하여 SGML 문서의 저장 및 검색을 지원하는 도구를 설계하였다. SGML문서를 DSSSL 및 HyTime의 문서 모델인 Grove를 통해 엘리먼트 단위로 분해하고 그 중 구조적 정보를 추출하여 데이터베이스에 저장하였다. 그리고, 구조 기반 검색을 위해 다양하게 문서의 구조적 정보를 제시해 주는 효과적인 사용자 인터페이스를 설계하였다. 본 연구는 웹에서의 검색이 가능하도록 사용자 인터페이스는 자바 애플릿으로 개발되었고, 대상 데이터베이스는 Informix Universal Server를 사용하였다.

1. 서 론

컴퓨터를 이용한 텍스트 처리 장치의 보급이 확대되어 감에 따라, 플랫폼에 관계없이 한번 작성된 문서 정보를 이기종간의 시스템에서 공유할 수 있는 데이타베이스의 구축 및 검색 그리고 상호교환의 중요성이 날로 증대되고 있다. 또한, 정보는 메모, 전자 메일, 작업 매뉴얼, 공문서 등의 다양한 문서 형식으로 되어 있다. 이러한 문서들로부터 원하는 정보를 효율적으로 관리, 공유하기 위해서는 문서를 일관성 있게 구조화하는 기술이 필요하다. 이러한 요구에 따라 SGML(Standard Generalized Markup Language)이라는 문서의 논리 구조를 표현하는 표준안이 등장하였다[1].

SGML은 DTD(Document Type Definition) 라는 구조적 정보와 이에 따른 실제 문서(Document Instance)로 구성되어 있다. SGML은 구조적인 문서를 저작하는데 널리 이용되어지고 있다. 이에 CALS, EC/EDI, 디지털 도서관 등에서 SGML이 차지하는 비중이 나날이 커질 뿐만 아니라, SGML 문서로 변환하는 작업이 활발히 진행되고 있다.

점차 증가하고 있는 SGML 문서를 체계적으로 공유, 저장, 검색하기 위한 SGML 문서 데이타베이스의 구축 및 검색 시스템에 관한 연구가 활발히 진행되고 있다. 본 논문에서는 SGML 문서의 저장 및 검색을 위한 데이타베이스 엔진 및 사용자 인터페이스를 개발하고자 한다.

2절에서는 현재까지 연구된 SGML 문서 모델들에 대해 살펴보며, 3절에서는 데이타베이스의 저장 및 검색 시스템의 설계 및 구현 방향에 대하여 살펴본다. 마지막으로 이에 대한 개선 방향을 제시하고 결론을 맺는다.

2. SGML 문서 데이타베이스 모델

SGML 문서 저작 도구의 개발이 저변화되어 있는 반면에, 대용량의 SGML 문서의 저장 및 검색을 위한 관련 분야의 진보는 뒤쳐지고 있다. SGML 구조적 정보의 대상 데이타베이스로의 전환, SGML 문서의 빠른 처리를 위한 적절한 인덱스 구조의 생성 등이 큰 문제로 부각되고[1], 또한 구조적 정보에 대한 질의 인터페이스의 부재라는 문제를 안고 있다[2]. 본 절에서는 현재 연구되고 있는 SGML 문서의 데이타베이스 모델에 대해 살펴본다.

2.1 기존 전문 검색 시스템

기존의 전문 검색 시스템들은 문서에 색인어를 부여함으로써 문서검색을 도왔다. 문서로부터 수동 혹은 자동으로 색인어를 추출하여, 사용자의 질의가 주어졌을 때, 질의에 사용된 단어와 문서 색인어 사이의 유사성을 계산하여 결과를 제공하였다. 이러한 시스템은 문서를 단순히 단어의 집합으로 보아, 문서 내의 구조적 정보는 무시되었다.

2.2 엘리먼트 기반 모델

SGML 문서는 구조적 정보를 담고 있는 엘리먼트로 구성되어 있기 때문에, 엘리먼트를 기반으로 한 검색이 필요하다. 엘리먼트 기반 모델의 경우, 엘리먼트에 대해 질의하기 위해서 전체 문서를 분해하여 엘리먼트를 추출한다. 그리고, 엘리먼트에 대해 직접 질의를 수행한 후 결과 엘리먼트가 속한 텍스트를 재구성하여 결과를 제시하는 방법을 취한다. 엘리먼트 기반 모델은 분해된 엘리먼트로부터 문서를 재구성해야하기 때문에 검색결과를 생성하는데 오랜 시간이 걸린다는 단점이 있다[4].

2.3 객체지향 모델

관계형 접근방식이 갖고 있는 문제점에 대한 해결책으로 구조의 네스팅(nesting)과 참조(reference)를 지원하는 확장형 관계 모델[5] 혹은 복잡한 오브젝트 모델[6]로 표현하는 방법이 있다.

INRIA의 VERSO 프로젝트에서는 O_2라는 OODB를 이용하여 SGML 문서의 DTD를 엘리먼트 단위로 분해하여, O_2 스키마로 변환하고 문서를 관련 객체와 값으로 매핑시켰다. 여기서는 DTD에 정의된 각각의 엘리먼트는 타입과 제약조건 및 디폴트 연산자를 갖고 있는 클래스로 해석된다[6].

GMD의 HyperStorm은 OODB를 이용하였으며, DTD에 대한 상위 DTD(Super-DTD)를 새로 정의하여, 특정 DTD에 관계없이 일관되게 SGML 문서를 엘리먼트 단위의 오브젝트로 저장하였다. 또한, 문서의 엘리먼트를 구조적 엘리먼트와 비구조적 엘리먼트로 나누어 비구조적인 엘리먼트는 상위 구조적 엘리먼트의 오브젝트에 포함되어 있는 구조를 가지고 있다[7].

3. 문서의 저장 및 구조기반 검색

본 시스템의 개요는 그림 1과 같다. SGML 문서를 파싱하여 데이타베이스에 저장하는 데이타베이스 생성시스템과 저장된 데이터에 따라 구조 정보에 따라 검색할 수 있는 구조 기반 검색 엔진으로 구성되어 있다.

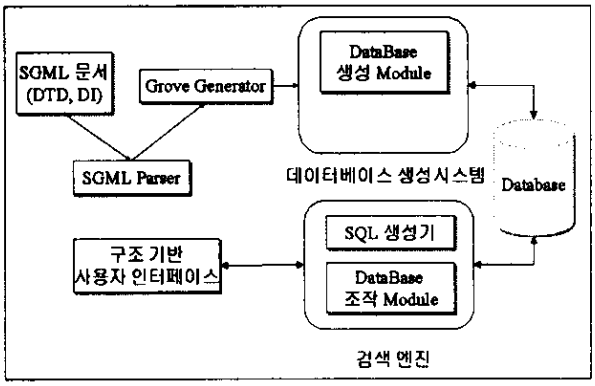

그림 1 전체 시스템의 개요

이 논문의 목적은 구조를 기반한 검색 도구의 개발이다. 이러한 검색 도구를 위해서 크게 두 가지의 측면을 고려하여 연구 개발이 진행되고 있다.

3.1 데이타베이스의 설계

문서의 구조에 따른 검색을 위해서는 문서의 구조 또한 데이타베이스에 저장되야 한다. 즉, SGML의 문서와 데이타베이스 포맷과의 상호 변환이 가능하여야 한다. SGML 문서의 구조는 계층적으로 이루어져 있기 때문에, 평면적인 테이블에 의존하는 기존의 RDBMS로는 해결이 어렵다. 이에 반해 객체 기반 DBMS는 이러한 계층적 정보의 표현은 가능하나, 저장된 데이타베이스를 다루기 위한 질의 언어의 표준 부재 및 처리 속도의 저하라는 문제가 있다.

데이타베이스의 모델에 따라 장단점이 존재하지만, 기본적으로 데이타베이스로의 변환 시에 다음과 같은 조건을 만족시켜야 효과적인 구조 기반 검색이 가능하다. 이러한 조건을 기반한 데이터 모델링을 통해 구조 기반 검색 시스템을 개발한다[7].

1. 엘리멘트 레벨까지의 문서 구조 브라우징이 가능한 모델이어야 한다.
2. SGML 문서의 모델과 데이타베이스에 사용되는 데이타 모델과의 차이를 최소한으로 줄여야 한다.
3. 특정 DTD에 관계없는 범용적인 데이터 모델을 설계하여야 한다.
4. 대용량의 데이터의 관리가 가능하여야 한다.

현재 연구되고 있는 SGML 데이타베이스 시스템 및 검색 엔진의 주요 이슈는 SGML과 대상 데이타베이스의 연관성에 관한 것이다. 즉, SGML의 구조정보를 효과적으로

담아내는 데이타베이스 설계이다.

본 논문의 데이타베이스의 설계과정에서 큰 축은 두 가지로 볼 수 있다. DSSSL 및 HyTime의 문서 모델인 Grove와 검색 필드 정보를 가지고 있는 특정 DTD에 대한 템플릿이다. 이 두 가지 정보를 가지고 실제 대상 데이타베이스에 문서의 구조정보와 내용 정보가 저장된다.

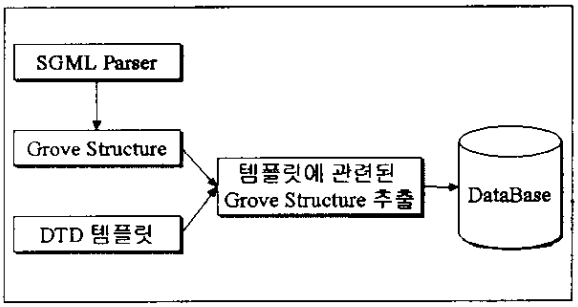

그림 2 구조적 정보의 저장

Grove는 DTD 정보를 가지고 있는 일종의 상위 DTD(super-DTD)로서, 여러 종류의 DTD를 일관되게 관리할 수 있는 트리 형태로 만들어진 데이터 구조이다. DSSSL에서는 문서의 포맷팅 정보로 사용되고, HyTime에서는 문서의 의미 정보를 이용하는데 사용된다. Grove는 여러 클래스와 클래스에 소속되어 있는 속성들(Property set)로 구성되어 있다.

SGML DTD는 구조적 정보를 가지고 있는 엘리먼트들과 비구조적인 정보를 가지고 있는 엘리먼트로 구성되어 있다. 따라서 구조 기반 검색을 위해서는 DTD의 구조적 정보에서 이러한 구조적 정보를 가지고 있는 엘리먼트를 추출하여야 한다. 현재 구조적 정보를 담고 있는 엘리먼트는 수동으로 추출되어 일종의 템플릿으로 저장된다. 이것들을 자동적으로 추출하는 기법은 아직 연구중이다.

```
<MEMO>
<FROM><PERSON>Person1</PERSON>
<TO><PERSON>Person2</PERSON>
    <PERSON>Person3></PERSON>
<BODY><STATEMENT><PARA>Paragraph
  </PARA><PARA> Paragraph2
  <KEYWORD>memo</KEYWORD>remainder
  </PARA></STATEMENT>
<RECOMMAND><PARA>Recommandation
  </PARA></RECOMMAND>
<P.S>comment</P.S>
</BODY></MEMO>
```

그림 3 메모 DTD에 대한 샘플 문서

예를 들어 그림 3과 같은 SGML 문서에서 구조적 정보를 가지지 못하는 <PARA>, <KEYWORD>는 상위 레벨에 있는 구조 정보에 포함되어 데이타베이스에 저장된다. 그림 3과 같은 문서는 실제로 데이타베이스에 그림 4와 같은 블록으로 저장된다.

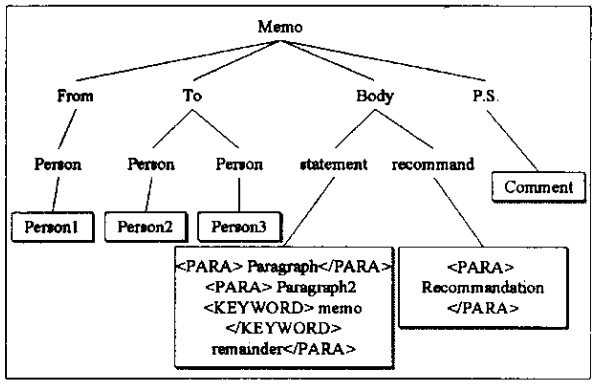

그림 4 데이타베이스의 구조

이와 같은 Grove 데이터 구조와 문서의 템플릿 정보를 이용하여 생성된 각각의 오브젝트들은 데이타베이스에 저장된다. 이 방법의 장점으로는 Grove를 이용함으로써, 특정 DTD에 관계없이 데이타베이스를 설계할 수 있으며, 검색 필드를 지정함으로써 모든 엘리먼트를 데이타베이스에 맵핑하는 것보다 실행상의 이점을 제공한다.

3.2 인터페이스의 설계

문서의 구조 정보를 이용한 검색을 시각적으로 사용자에게 제시할 인터페이스의 설계가 필요하다. 기본적으로 문서의 구조를 브라우징하는 기능과 문서의 엘리먼트 단위에서의 검색 기능이 있어야 한다[2]. 이러한 구조 정보를 표현하기 위하여 크게 두 가지의 검색 화면을 제공하고 있다. SGML 문서의 구조적 정보를 트리 구조로 보여주는 것과, 문서의 템플릿 정보를 이용하여 문서의 형식을 보여주는 방법이 있다.

첫째, 그림 5와 같이 문서의 구조적 정보를 트리로 표현함으로써, 사용자가 문서의 구조를 브라우징하여 문서의 구조를 명확히 이해하면서 검색을 할 수 있다. 검색할 필드를 더블 클릭함으로써, 검색할 구조 정보를 선택하고, 보내기 버튼을 클릭하여 검색할 구조 정보에 대하여 키워드를 입력하여 질의를 서버에 보낸다.

둘째, 그림 6과 같이 문서의 템플릿 정보를 보여주는 검색 화면으로써, 사용자가 문서의 구조적 정보를 이해하지 못하더라도 문서의 템플릿에서 원하는 검색 필드를 클릭하

면서 질의를 할 수 있다. 이와 같은 방법은 트리 구조를 통해 구조적으로 검색하는 것보다 편리하게 질의를 보낼 수 있는 장점이 있다.

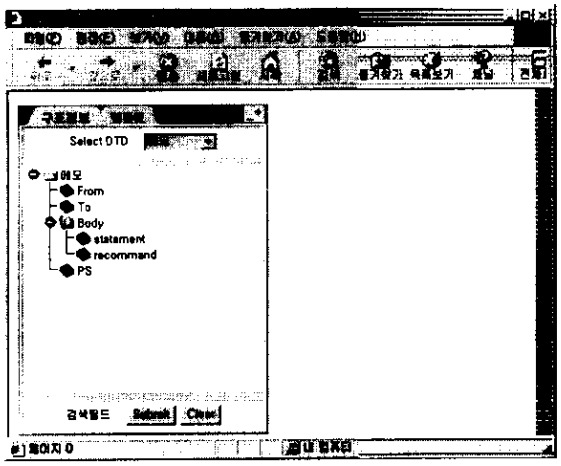

그림 5 문서의 구조적 정보

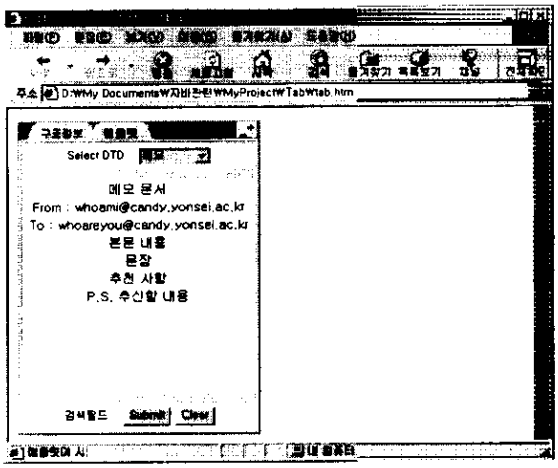

그림 6 문서 템플릿 정보

이러한 구조 기반 검색은 이해하기 편하고 사용하기 편하지만 사용자가 문서의 구조를 정확히 모르는 경우 원하는 정보를 찾기가 어려우며, 구조가 복잡한 경우 바로 찾아가는데 시간이 걸리는 단점이 있다. 따라서 구조 검색뿐만 아니라 일반적인 전문 검색 기능을 포함하여야 한다.

4. 결 론

본 논문은 SGML 문서의 구조적 저장 및 검색을 위한 데이타베이스의 변환 과정 및 구조적 정보를 이용한 질의를 가능케 하는 사용자 인터페이스를 설계하였다. DSSSL 및 HyTime의 문서 모델인 Grove를 이용하여 특정 DTD에 관계없이 일관된 데이타베이스 스키마를 제공할 뿐만 아니라, 구조적 정보만을 나타내는 엘리먼트 단위로 데이타베이스에 담고 있기 때문에, 기존의 시스템에 비해 좋은 성능을 보일 수 있다.

아직까지는 문서의 구조적 엘리먼트의 지정, 즉 템플릿 정보의 작성을 수동으로 하고 있으나, 이를 SGML 문법 요소를 고려하여 자동화하는 문제가 남아있다.

참고문헌

[1] E.V. Herwijnen, *Practical SGML*, Kluwer Academic Publishers, 1993.

[2] A. Seungupta and A. Dillon, "Extending SGML to accommodate database functions: a methodological overview," *Journal of the American Society of Information Systems*, pp. 629-637, 1997.

[3] A. Sengupta, "Demand more from your SGML databse. Bringing SQL under the SGML limelight," *In <TAG> The SGML Newsletter*, 9(4), pp.1-7, 1996.

[4] G.E. Blake, M.P. Consens, P. Kilpelainen, P.A. Larson, T. Snider and F.W. Tompa, "Text / relational database management systems: harmonizing SQL and SGML," *Proc. Applications of Databases*, pp. 267-280, 1994.

[5] R. Sacks-Davis, T. Arnold-Moore and J. Zobel, "Database systems for structured documents," *IEICE Trans. on Information and Systems*, 78-D(11), pp.1335-1342, 1995.

[6] V. Christophides, S. Abiteboul, S. Cluet and M. Scholl, "From structured documents to novel query facilities," *SIGMOD*, 1994.

[7] K. Aberer, K. Böhm and C. Hüser, "The prospects of publishing using advanced database concepts," *Conf. on Electronic Publishing*, 1994.