

XML-UGL : XML 문서 질의를 위한 UML 기반 그래픽 언어

하 안[†] · 김 기 한^{**}

요 약

웹 상에서 XML 문서가 표준으로 자리잡아감에 따라 웹 질의 언어에 대한 연구가 활발히 진행 중이다. 그리고, UML은 객체지향 분석과 설계의 결과를 표현하기 위한 그래픽 언어이다. 따라서, 본 연구에서는 UML을 기반으로 한 XML 문서를 위한 그래픽 질의 언어(Graphical Query Language)를 제안한다. 본 연구는 XML 문서의 내용이 복잡하더라도 시각적인 형태의 사용으로 XML 문서의 질의 문법과 의미를 직관적으로 표현 가능하도록 한다. 그리고, XML 문서의 내용과 질의의 문법과 의미를 동일한 시각화를 적용함으로써 XML 문서의 객체지향 DBMS에 저장, 검색의 모든 과정을 일체시킬 수 있다.

XML-UGL : UML-based Graphical Language for Querying XML Documents

Yan Ha[†] · Gi Han Kim^{**}

ABSTRACT

The widespreading of XML as a standard for semi-structured documents on the Web opens up challenging opportunities for Web query language. And UML is a graphical language to represent the result of object-oriented analysis and design. In this paper, we introduce an UML-based graphical query language for XML documents. The use of a visual formalism for representing the syntax and semantics of queries enables an intuitive expression of queries, even when they are rather complex. And, it is matched a series of processes to store and retrieve XML documents to OODBMS with the use of an uniform visualization for representing both the content of XML documents (and of their DTD) and the syntax and semantics of queries

키워드 : XML 문서(XML Document), 질의(Query), UML, 그래픽 언어(Graphical Language), OODBMS

1. 서 론

웹 문서의 형태가 HTML에서 XML로 빠르게 변함에 따라 XML 문서로부터 정보를 추출하는 새로운 언어의 필요성이 대두되고 있다[1]. 그리고, 최근에 멀티미디어 문서들의 저장, 관리를 용이하게 하는 측면에서 XML 문서를 모델링하고 저장하는 많은 연구나 시스템들이 객체지향적인 접근을 하고 있는 추세이다[2, 3].

한편, UML은 객체지향 분석과 설계의 결과를 표현하기 위한 그래픽 언어이며, 다양한 다이어그램이 존재한다. 현재 상용되고 있는 객체 모델링 도구인 Rational Rose나 Together 등에서는 클래스 다이어그램을 비롯한 다양한 다이어그램들을 지원하며, 이들로부터 각종 데이터베이스 스키마와 객체지향 코드를 생성해 준다.

본 연구는 UML 기반 XML 문서 그래픽 질의 언어인 XML-UGL을 제안한다. XML-UGL은 XML 문서를 UML 클래스 다이어그램에 사상시킨 데이터 모델을 기반으로 한다. XML-UGL은 XML 기반 웹 문서를 질의하기 위한 시각적이고 직관적인 언어로서, UML의 표기법을 사용한다. XML-UGL의 가장 큰 특징은 다른 질의 언어에서 표현하지 않았던 XML 링크 등을 지원하는 등 풍부한 표현력을 갖는다. 그리고, 별도의 표기법을 제시한 것이 아니라 UML 메카니즘을 그대로 이용함으로써, UML 관련 도구들을 그대로 활용하고, 객체지향적인 문서 처리를 용이하게 한다는 점이다.

본 연구의 의의는 XML 문서의 내용과 질의의 문법(syntax)과 의미(semantic)를 UML 클래스 다이어그램이라는 동일한 시각화를 적용함으로써 XML 문서의 OODBMS에 저장, 검색의 모든 과정을 일체시키는 것이다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구로 XML 질의 언어들을 살펴보고, 3장에서는 UML과 UML을 이용한 XML 문서 모델링에 대해 설명한다. 4장에서는 본 연

[†] 준 회원 : 경인여자대학 컴퓨터정보기술학부 교수

^{**} 종신회원 : ETRI 부설 국가보안기술연구소
논문접수 : 2002년 9월 27일, 심사완료 : 2003년 1월 14일

구에서 제안하는 객체기반 질의 언어에 대해 기술한다. 구체적으로 구성 요소와 표기법을 제안하고, 대표적인 XML 질의 언어들과 비교 분석해 본다. 끝으로 5장에서는 결론 및 향후 연구과제에 대해 논한다.

2. 관련 연구

XML 질의 언어의 대표적인 예로는 LOREL, XML-QL, XML-GL 등이 있으며, 이들을 분류해보면 (그림 1)과 같다 [4, 7].

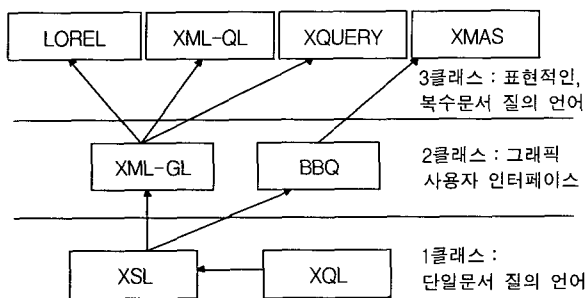
(그림 1)을 살펴보면 세 가지 클래스로 구별되어 있다.

1클래스는 단일 분석 질의 언어로 XSL과 XQL이 해당된다. XSL(Extensible Stylesheet Language)은 패턴과 템플릿으로 구성되며, 다른 질의 언어를 위한 기반이 된다. 여기서, 패턴은 소스 트리에서 노드에 해당하는 것이며, 템플릿은 결과 트리 부분을 형성하기 위해 실체화된다[5].

XQL은 엘리먼트와 XML 문서의 텍스트를 선택하고 필터링 하기 위한 표기로서, XSL 패턴 문법에 대한 확장이다. 이것은 제한된 표현능력을 갖고 문법적으로 간단하고 간결하게 설계하는 것을 목표로 한다[4].

3클래스에 해당하는 것은 표현적인 복수 문서 질의 언어로서 LOREL, XML-QL, XQUERY 등이 있다. LOREL은 원래 반구조화된 데이터를 위해 설계되어 XML 데이터로 확장되어온 것으로 SQL/OQL 스타일로 사용자에게 편리하도록 설계된 언어이다. 문서의 구조를 미리 알지 못할때, 매우 유용하고 강력한 경로 표현 능력을 갖는 장점이 있다[5].

XML-QL은 SQL을 확장한 언어로 다른 자원으로부터 XML 데이터를 통합하기 위한 변형을 포함하는 특징이 있다[6].



(그림 1) 대표적인 XML 질의 언어의 분류

그리고, 1클래스와 3클래스 사이에 2클래스에 해당하는 그래픽 사용자 인터페이스 클래스가 있다. 웹 문서의 특징 상 단일 문서에 대한 질의 보다 복수 문서에 대한 질의의 필요성이 더 부각되고 있는 실정에서 복잡한 문서 구조를 보다 쉽게 접근할 수 있는 시각화 방법으로 XML-GL과 같은 그래픽 사용자 인터페이스가 제안되었다[1]. 3클래스에 해당하는 XMAS(XML Matching And Structuring)의 경우도 질의를 생성하기 위해 별도의 그래픽 사용자 인터페이스인

BBQ(Blended Browsing and Querying)를 정의한 바 있다. BBQ는 트리 구조에 기반하여 XML 데이터를 시각화하는 윈도우를 사용하였다[7].

그래픽 사용자 인터페이스의 대표적인 예인 XML-GL은 XML 그래프에 의해 XML 문서와 DTD를 그래픽적으로 표현하는 방법이나, [6]에서 제안한 별도의 XML 데이터 모델(XML-DML)에 의존해서 사용되며, XML 링크 부분 등을 지원하지 못하는 단점이 있다.

본 연구는 XML 문서를 객체지향 데이터베이스에 저장하는 연구가 활발히 진행되고, UML 관련 도구들이 꾸준히 상용화되고 있는 실정에서 UML 기반 그래픽 사용자 인터페이스인 XML-UGL을 제안한다. UML의 다양한 메커니즘을 활용한 XML-UGL은 XML 링크 부분을 지원할 수 있으며, 객체지향 개념을 그대로 적용할 수 있다.

3. UML과 XML 문서 모델링

본 장에서는 UML의 주요 메커니즘과 클래스 다이어그램을 간략하게 기술하고, UML을 이용한 XML 문서 모델링에 대해 설명한다.

3.1 메커니즘(Mechanism)

UML은 여러 가지 객체 지향을 결합한 형태로 다양한 확장 메커니즘을 제공한다. 제한조건, 엘리먼트 특성, 스테레오타입이 이에 해당된다.

3.1.1 제한조건(Constraints)

조건이나 가정을 규정하는 모델 엘리먼트들 간의 의미적인 관계로 항상 참이 되어야 한다. 제한조건의 표기는 '{ }' 안에 텍스트 스트링을 기술한다.

3.1.2 엘리먼트 특성(Element properties)

많은 엘리먼트들은 시각적으로 표현되지 않은 세부적인 특성을 갖는다. 따라서, 사용자는 태그된 값을 이용하여 새로운 엘리먼트의 특성을 정의할 수 있다.특성에 대한 표기는 '{ }' 안에 한 개 이상의 '키워드=값'을 ','로 구별하여 기술한다.

3.1.3 스테레오타입(Stereotypes)

스테레오타입은 엘리먼트의 새로운 클래스이다. 엘리먼트와 같은 형태를 가지나 다른 의도로 사용되는 하위클래스이다. 스테레오타입의 표기는 « »안에 스테레오타입의 이름을 키워드 스트링으로 기술한다.

3.2 클래스 다이어그램

UML 클래스 다이어그램은 클래스와 그들간의 관계로 구성된다[9]. 이에 대한 설명은 다음과 같다.

3.2.1 클래스(Class)

클래스는 클래스 이름, 애트리뷰트 리스트(Attribute list)

와 오퍼레이션 리스트(Operation list) 부분으로 구성된다. 애트리뷰트 리스트는 클래스의 멤버들을 나타내고, 오퍼레이션 리스트는 멤버들을 다루는 함수를 나타낸다.

3.2.2 집단화(Aggregation)와 합성화(Composition)관계
 집단화 관계는 상위클래스와 하위클래스가 'part-of' 관계이며 '◇'로 표기한다. 강한 집단화 관계로 합성화 관계가 있는데, 이것은 하위클래스가 상위클래스와 같은 생명주기를 갖는 강한 형태의 구성 관계를 나타낸다. 표기는 '◆'로 한다.

2개 이상의 집단화 관계에서 선택적으로 하위클래스가 발생하는 경우 'xor' 관계를 갖는다. 이 때, 여러 개의 집단화 관계를 점선으로 연결하고 'xor'이라는 제한조건을 준다.

3.2.3 의존(Dependency) 관계

한 쪽 클래스의 변화가 다른 클래스에 영향을 줄 수 있는 관계로 점선으로 나타내며 방향성을 나타낼 수 있다.

3.3 XML-OGI의 데이터 모델

UML 기반 그래픽 질의 언어인 XML-OGI을 생성하기 위해서는 먼저, XML 문서에 대한 객체 모델링이 전제되어야 한다. 본 연구에서는 이를 XML-OGI의 데이터 모델이라 한다. XML-OGI의 데이터 모델을 만들기 위한 XML과 UML 간의 사상 규칙을 테이블로 정리하면 <표 1>과 같다[11].

<표 1>은 XML 문서의 구성요소를 객체지향 개념을 적용하여 UML 클래스 다이어그램의 구성요소에 사상시킨 것이다.

<표 1> XML-OGI 데이터 모델 생성 규칙

XML 구성요소		UML 클래스 다이어그램			
엘리먼트 선언	선언 부분	엘리먼트	클래스		
	내용 부분	연결자	' '	집단화 관계 'xor'제한조건	
			','	집단화 관계	
		발생 지시자	'?'	다중성	'0...1'
			'*'		'0...*'
			'+'		'1...*'
		'()'	임시 클래스		
	PCDATA	단일	'text' 클래스로부터 상속		
		혼합	임시클래스 생성, 상속		
	링크	단순	의존 관계	'<<HREF>>' 스테레오타입	
확장		'<<LOCATOR>>' 스테레오타입			
애트리뷰트 리스트선언	애트리뷰트	Private 애트리뷰트			

4. XML-OGI

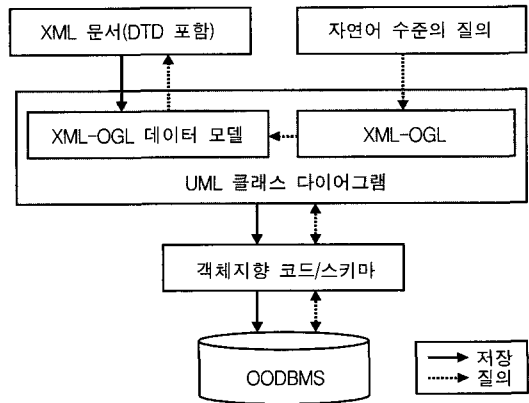
본 장에서는 3.3절의 데이터 모델을 기반으로 XML-OGI

를 제안한다.

4.1 XML-OGI 및 데이터 모델

XML-OGI 데이터 모델은 XML 문서를 객체지향 데이터 베이스에 저장하기 위해 모델링하는 언어이라면, XML-OGI은 이를 기반으로 자연어 수준의 XML 질의를 대신하는 모델링 언어로서, 검색된 결과를 XML-OGI 형태를 통해 XML 문서로 출력하도록 해준다.

(그림 2)는 XML 문서를 XML-OGI 데이터 모델을 통해 OODBMS에 저장하고, 자연어 수준의 질의가 들어왔을 때 XML-OGI을 통해 검색하며, 이에 대한 결과를 XML-OGI 데이터 모델로 출력해주는 전체 시스템 구성도이다.

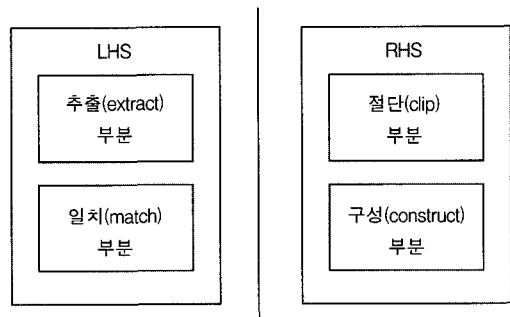


(그림 2) XML-OGI 및 데이터모델

4.2 구성 요소

XML-OGI은 크게 LHS(Left-Hand Side)와 RHS(Right-Hand Side) 두 부분으로 구성된다. LHS는 다시 추출 부분과 일치 부분으로 구성되며, RHS는 절단 부분과 구성 부분으로 구성된다[4].

이에 대한 개념도는 (그림 3)과 같으며, 각 구성 요소에 대한 설명은 다음과 같다.



(그림 3) XML-OGI의 구성 요소

4.2.1 추출(Extract) 부분

문서에서 목적이 되는 문서와 엘리먼트를 가리키므로써 질의의 영역을 명시한다. SQL과 비교해보면, 질의에 의해

목적이 되는 관계를 설정하는 **from 절**에 해당하는 부분이다.

4.2.2 일치(Match) 부분

목적 엘리먼트가 질의 결과를 만족하기 위한 논리적인 조건을 규정한다. SQL과 비교해보면 조건은 결과의 부분이 되는 목적 튜플을 선택하는 **where 절**에 해당한다.

4.2.3 절단(Clip) 부분

조건을 만족하는 추출된 엘리먼트의 하위엘리먼트를 규정한다. SQL과 비교해보면, 질의의 최종 결과에서 보유하고 있는 결과 튜플의 열을 정의하는 **select 절**에 해당한다.

4.2.4 구성(Construct) 부분

결과 문서와 추출된 엘리먼트들간의 관계에서 포함된 새로운 엘리먼트를 규정한다. SQL과 비교해보면, 구성 부분은 질의의 결과로부터 새로운 관계를 설계하도록 하는 **create view** 문장의 확장이라고 볼 수 있다.

4.3 생성 규칙

XML-OGI은 UML 표기법을 이용하여 질의 언어를 생성한다. 각 질의 언어에 대한 생성 규칙은 <표 2>와 같다.

<표 2> XML-OGI 생성 규칙

질의 요소		UML 구성요소	
부 정		클래스 제한조건	
포 함		집단화 관계	'{all}' 제한조건
링 크	단 순	의존관계	'<<HREF>>' 스테레오타입
	확 장		'<<LOCATOR>>' 스테레오타입
바인딩		의존관계	'<<bind>>' 스테레오타입
조 인		의존관계	'<<Use>>' 스테레오타입
리스트	일 반	다중성	'1...*'
	그 룩		'1...n'

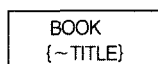
각 질의 요소에 해당하는 생성 규칙을 예를 통해 기술하면 다음과 같다. 질의 요소는 부정, 포함, 링크, 바인딩, 조인, 리스트로 구별할 수 있다.

4.3.1 부정과 포함

질의 언어에서 부정 혹은 임의(arbitrary) 포함을 표기하는 규칙들은 다음과 같다.

[규칙 1] 부정(negation)은 클래스의 제한조건이 된다. 즉, 해당 클래스에 태그된 값(tagged value)이 되며, '~'을 앞에 붙인다.

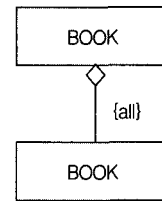
예) “책(BOOK) 제목(TITLE)이 알려지지 않은 책을 찾아라”.



(그림 4) [규칙 1]의 질의 언어(LHS 부분)

[규칙 2] 하위 클래스들을 모두 포함하기 위해 집단화 관계에 '{all}' 제한조건을 사용한다.

예) “웹의 특정한 문서에서 책(BOOK) 엘리먼트를 모두 찾아라”.



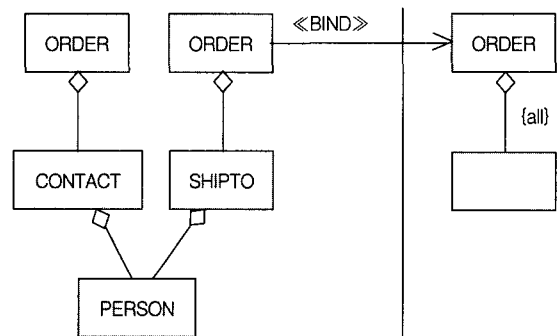
(그림 5) [규칙 2]의 질의 언어(RHS 부분)

4.3.2 링크

XML의 가장 큰 특징인 단순 링크와 확장 링크를 표현하는 질의 언어의 생성 규칙은 다음과 같다.

[규칙 3] 단순 링크(simple link)는 의존 관계에 '<<HREF>>' 스테레오타입을 갖는다.

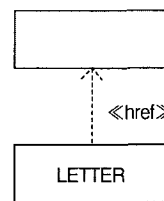
예) “해당 편지(LETTER)와 단순 링크된 웹 문서를 찾아라”.



(그림 6) [규칙 5]의 질의 언어

[규칙 4] 확장 링크(extended link)는 의존 관계에 '<<LOCATOR>>' 스테레오타입을 갖는다.

예) “해당 웹 문서와 확장 링크된 웹 문서를 모두 찾아라”.



(그림 7) [규칙 3]의 질의 언어(RHS 부분)

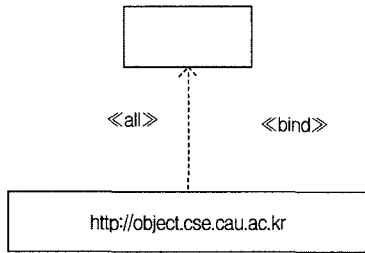
4.3.3 바인딩과 조인

질의 언어에서 LHS와 RHS 부분의 클래스를 연결하는 바인딩에 관한 질의 언어는 [규칙 5]와 같고, LHS 부분에

서 2개 이상의 클래스의 애트리뷰트 값을 비교하는 질의 언어는 [규칙 6]과 같다.

[규칙 5] 바인딩(bindings)은 의존 관계에 ‘<<bind>>’ 스테레오 타입을 갖는다.

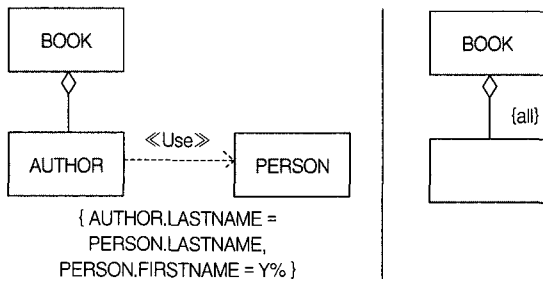
예) “다른 주문(ORDER)에서 계약(CONTACT)된 사람(PERSON)에게 발송(SHIPTO)된 주문을 발견하라.”



(그림 8) [규칙 4]의 질의 언어(RHS 부분)

[규칙 6] 조인(Join)은 ‘<<Use>>’ 스테레오 타입을 갖는 의존 관계에 제한 조건('{ }')을 갖는다.

예) 성(LASTNAME)이 같은 저자(AUTHOR)에 의해 쓰여진 모든 책을 찾아라. 이 때, 저자의 이름은 'Y'로 시작한다.



(그림 9) [규칙 6]의 질의 언어

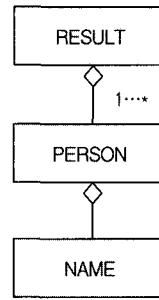
4.3.4 리스트와 그룹 리스트

질의 결과로 새로운 엘리먼트를 구성하는 경우 RHS 부분에 표현되는 리스트 관련 질의 언어들은 다음과 같다.

[규칙 7] 리스트(List)는 '1 ... *'을 다중성으로 갖는 집단화 관계이다.

```

예) <RESULT >
  <PERSON id = "1" >
    <NAME > Yan Ha </NAME >
  </PERSON >
  <PERSON id = "2" >
    <NAME > John </NAME >
  </PERSON >
  <PERSON id = "3" >
    <NAME > Amy </NAME >
  </PERSON >
</RESULT >
    
```

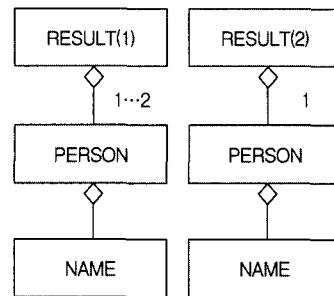


(그림 10) [규칙 7]의 질의 언어(RHS 부분)

[규칙 8] 그룹 리스트(Grouped List)는 '1 ... n'을 다중성으로 갖는 집단화 관계이다.

```

예) <RESULT >
  <PERSON id = "1" >
    <NAME > Yan Ha </NAME >
  </PERSON >
  <PERSON id = "2" >
    <NAME > John </NAME >
  </PERSON >
</RESULT >
<RESULT >
  <PERSON id = "3" >
    <NAME > Amy </NAME >
  </PERSON >
</RESULT >
    
```



(그림 11) [규칙 8]의 질의 언어(RHS 부분)

4.3 예 제

4.2절의 규칙들을 종합한 예제를 살펴해보도록 한다. 구성 부분에 엘리먼트를 확장하고, 포함하는 예제들을 기술한다. 첫 번째, 구성 부분에서 엘리먼트를 확장한 예제이다.

예) “저자(AUTHOR)의 성(LASTNAME)과 이름(FIRST-NAME)으로 사람(PERSON)이라는 엘리먼트에서 주소 (ADDRESS)를 찾아 저자에 주소가 포함된 새로운 엘리먼트(NEWORDER)를 만들어라.”

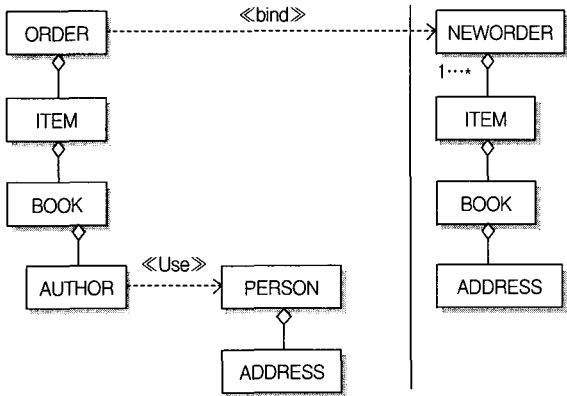
```

<NEWORDER >
  <ITEM >
    <BOOK >
      <AUTHOR FIRSTNAME = "Yan"
        LASTNAME = "Ha" >
    </BOOK >
  </ITEM >
</NEWORDER >
    
```

```

    <POSTAL> 407-740 </POSTAL>
      <CITY> Incheon </CITY>
    </ADDRESS>
  </AUTHOR>
</BOOK>
</ITEM>
</NEWORDER>
  
```

이에 해당하는 질의 언어는 (그림 12)와 같다.



(그림 12) 엘리먼트 확장 예제

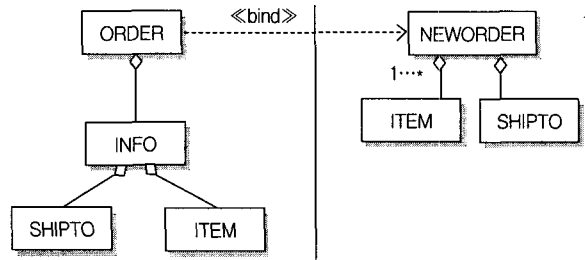
두 번째, 구성 부분에서 엘리먼트를 포함한 예제이다.

예) 엘리먼트 집합으로부터 정보를 추출하고 새로운 주문 (NEWORDER)들을 만들어라. 각 주문들은 모든 아이템 (ITEM)을 포함하고 이에 해당하는 배달정보(SHIPTO)도 갖는다.

```

<NEWORDER id = 1>
  <SHIPTO>
    <ADDRESS>
      <POSTAL> </POSTAL>
    </ADDRESS>
  </SHIPTO>
  <ITEM>
    <NAME> easy XML </NAME>
  </ITEM>
  <ITEM>
    <NAME> easy JAVA </NAME>
  </ITEM>
</NEWORDER>
<NEWORDER id = 2>
  <SHIPTO> </SHIPTO>
  <SHIPTO>
    <ADDRESS>
      <POSTAL> 407-740 </POSTAL>
      <CITY> Incheon </CITY>
    </ADDRESS>
  </SHIPTO>
  <ITEM>
    <NAME> easyXML </NAME>
  </ITEM>
  <ITEM>
    <NAME> easyJAVA </NAME>
  </ITEM>
</NEWORDER>
  
```

이에 해당하는 질의 언어는 (그림 13)이다.



(그림 13) 엘리먼트 포함 예제

4.4 비교 분석

본 질은 기존에 제안되어 온 질의 언어들과 본 연구에서 제안한 XML-OGL질의 언어간의 표현력과 특징을 비교한다. 질의 언어들간의 비교항목을 나열하고 설명해 보도록 한다.

4.4.1 데이터 모델

별도의 데이터 모델을 명확하게 갖고 있는지 여부를 나타낸다. XQL의 경우, XML 그래프에 의해 나타내기 때문에 자신만의 데이터 모델은 갖고 있지 않다. XML-GL과 XML-OGL은 데이터 모델을 갖고 있다. 그러나, 이 둘의 데이터 모델은 큰차이를 나타내고 있다. XML-GL의 데이터모델인 XML-GDM은 객체, 관계, 속성으로 구성되어 있으나, 객체지향 개념과는 다르다. 이에 비해 XML-OGL에서 사용하는 UML 기반 객체 모델링 방법은 UML의 메카니즘을 그대로 적용한 모델링 방법으로 별도의 표기법을 익힐 필요가 없으며, UML 관련 상용 도구들을 그대로 이용할 수가 있다.

4.4.2 기본 질의 축약

㉠ 문서 선택

문서와 질의가 주어졌을때, 질의 조건을 만족하는 문서를 선택하는 것으로 모든 질의 언어에서 이를 지원한다. XML-GL과 XRML-OGL의 경우 수직선으로 나뉘어진 2개의 집합으로 구성된다. 좌측은 선택된 문서와 만족해야할 조건을 나타내고, 우측은 결과로서 생성되는 문서를 표현한다.

㉡ 조인

같은 문서나 혹은 다른 문서에 포함된 데이터나 2개 이상의 애트리뷰트를 비교하는 것이다. XML-OGL에서는 '«Use»' 스테레오 타입을 갖는 의존 관계에 제한조건('}')을 붙여 비교 구분을 삽입하도록 한다.

㉢ 질의 결과의 의미

질의의 결과는 데이터베이스의 현재 내용이나 새로운 문서로서 정의될 수 있다. LOREL의 경우 새로운 엘리먼트를 나타내는 객체 식별자의 집합이며, 그 외 언어들은 데이터베이스 관리에 독립적인 새로운 XML 문서를 질의 결과로 반환한다.

4.4.3 경로 표현

정확한 구조가 알려지지 않았을 때, 반구조화된 데이터를 질의하게 되면, 경로 표현에 기반한 항해적 질의 언어를 사용하는 것은 무척 편리하다. 모든 질의 언어에서 부분적이거나 경로 표현을 지원하고 있다. XML-OGI의 경우는 경로 표현을 지원하기 위해 집단화 관계에 '{all}'이라는 제한 조건을 갖는다.

4.4.4 부 정

인스턴스가 해당하는 서술을 만족하지 않을 때 사용한다. XML-OGI에서는 해당 클래스에 대한 제한조건을 사용한다. '{ }'안에 '~'를 사용한다.

4.4.5 재구성되는 추상화

㉠ 새로운 엘리먼트 설계

질의 구성 메카니즘에 의해 새로운 XML 엘리먼트가 생성되는지를 나타낸다. XQL에서는 구성 메카니즘이 없기 때문에 엘리먼트가 생성되지 않으나 다른 질의 언어들은 이를 지원한다.

㉡ 그룹화

결과 엘리먼트들을 집단화하거나 재구성할 수 있는지를 나타낸다. LOREL에서는 group by 절이 있고, XML-GL이나 XML-OGI에서 추출된 객체는 PCADATA나 애트리뷰트의 구별된 값에 따라 그룹이 될 수 있다. 각 클래스는 대표적인 값을 수행하는 엘리먼트 인스턴스와 관련있고 나머지는 하위 엘리먼트가 된다.

4.4.6 순서 관리

㉠ 결과 순서

결과의 일부 데이터의 값을 정렬하므로써 엘리먼트 인스

턴스의 순서를 구성할 수 있다. XML-OGI은 제한조건을 이용하여 '{ordered}'라고 라벨을 붙일 수 있다.

㉡ 순서 유지 결과

원래 문서와 같은 방식으로 결과에서 엘리먼트의 순서를 구성한다. XML-OGI에서는 데이터 모델에서와 같이 하위 엘리먼트의 왼쪽에서 오른쪽으로 우선 순위를 높게 갖는다.

㉢ 인스턴스 순서 질의

주어진 순서 관계에서 XML 엘리먼트나 애트리뷰트의 순서를 질의할 수가 있다. XML-OGI에서는 클래스의 오퍼레이션 리스트 부분에 순서를 나타내는 함수를 추가할 수 있다.

4.4.7 추상화된 데이터 타입 지원

멀티미디어 내용의 다른 종류를 선택할 수 있도록 데이터 타입을 지원할 필요가 있다. XML-OGI에서 사용하는 클래스들은 'text'와 같은 기본 데이터 클래스로부터 상속을 받도록 한다.

4.4.8 XML 링크 지원

XML-OGI은 XML의 가장 큰 특징인 다양한 링크 형태를 지원한다. 단순 링크와 확장 링크에 대해서 구별하여 지원한다.

4.4.9 객체지향 개념 사용

XML-OGI은 UML 클래스 다이어그램을 사용하므로써 OODBMS에 저장, 검색할 수 있는 기반을 제공한다.

종합해 보면, XML-OGI은 XML-GL과 같은 그래픽 질의 언어로서 직관적이고 시각적이며, 추출, 일치, 절단, 구성 부분을 갖는다. 그러나, XML-OGI은 XML-GL에 비해 풍부한 표현력을 갖는다. 즉, 인스턴스 순서를 질의, 추상화된 데이터 질의, XML 링크 부분 등을 지원할 수 있다.

<표 3> 대표적 XML 질의 언어 비교 분석

특 징	XML 질의 언어	LOREL	XML-QL	XML-GL	XSL	XQL	XQuery	XML-OGI
별도 데이터 모델		○	○	○	○	×	○	○
기본 질의 축약	문서 선택	○	○	○	○	○	○	○
	조인(Joins)	○	○	○	×	×	○	○
	결과의 의미	OID들의 집합	XML 문서	XML 문서	XML 문서	XML 문서	XML 문서	XML 문서
경로 표현	○	○	부분적	○	○	부분적	부분적	
부 정	○	×	○	○	○	○	○	
재구성되는 추상화	새 엘리먼트 구성	○	○	○	○	×	○	○
	그룹 구성	○	×	○	×	×	○	○
순서 관리	결과 순서	○	○	○	○	×	○	○
	순서 유지 결과	○	○	○	○	○	○	○
	인스턴스 순서를 질의	○	×	×	○	○	○	○
추상화된 데이터 타입	○	×	×	×	×	○	○	
XLink의 지원	×	×	×	×	×	×	○	
객체지향	×	×	×	×	×	×	○	

그리고, 별도의 표기법 대신 UML 표기법을 사용하기 때문에 UML 관련 도구들을 그대로 사용하며, 데이터베이스 스키마나 객체지향 코드로 쉽게 변환되어 객체지향적인 문서 처리를 용이하게 한다.

5. 결론 및 향후 연구 과제

본 연구에서는 XML 문서를 객체지향적으로 질의하기 위해 UML 기반 그래픽 질의 언어인 XML-OGI를 제안하였다. XML-OGI의 주요 특징으로는 UML 클래스 다이어그램으로 표현되는 XML 데이터 모델이 존재하며, 다른 질의 언어에서 지원하지 않는 XML의 단순 링크와 확장 링크를 지원하며, 객체지향 모델링 방법을 사용했다는 것이다. 본 연구의 가장 큰 의의는 XML 문서의 내용과 질의의 문법과 의미를 시각적으로 형식화함으로써 다소 복잡한 경우라도 쉽게 이해되도록 하며, XML 문서의 OODBMS에 저장, 검색에 관한 일련의 과정을 일치시킬 수 있다. 특히 UML은 객체지향 표준화 모델링 언어로써 Rose나 Together 등 상용 모델링 도구들을 이용하여 각종 객체지향 코드나 스키마로 변환 가능하다. 그러므로, 본 연구의 UML 클래스 다이어그램을 이용한 XML-OGI은 XML 기반 웹 문서를 질의하기 위한 시각적인 인터페이스와 언어로 자리잡고, XML 문서를 객체지향 데이터베이스에 저장, 관리를 위한 기반이 될 것이다.

향후 연구 과제는 본 연구의 질의 언어를 적용할 수 있는 OODBMS 구축이다. XML 문서를 OODBMS에 저장하고, XML-OGI에 의해 XML 문서의 일부분을 추출하여 보여주는 XML 문서 관리를 위한 시스템 전체의 설계 및 구현이다.

참고 문헌

[1] S. Ceri, S. Comai, E. Damiani, P. Fraternali, S. Paraboschi and L. Tanca, "XML-GL : A Graphical Language for Querying and Restructuring XML Documents," In Proc. of the 8th Int. World Wide Web Conf., Toronto, Canada, May, 1999.

[2] V. Christophides, S. Abiteboul, S. Cluet, M. Scholl, "From Structured Documents to Novel Query Facilities," In Proc. ACM SIGMOD Intl. Conf. Management of Data, pp.313-324, 1994.

[3] 하 안, 황용주, 김용성, "SGML DTD로부터 UML 클래스 다이어그램으로의 사상 알고리즘", 정보과학회논문지(B), 제26권 제4호, pp.508-520, 1999.

[4] Angela Bonifati, Stefano Ceri, "Comparative Analysis of Five XML Query Languages," ACM SIGMOD Record, 29(3), pp.76-87, 2000.

[5] S. Abiteboul, D. Quass, J. McHugh, J. Wiener. "The Lorel query language for semistructured data," Int. J. on Digital Libraries, 1(1), April, 1997.

[6] A. Deutsch et al., "XML-QL : A Query Language for XML," <http://www.w3.org/TR/NOTE-xml-ql>, 1998.

[7] B. Ludascher, Y. Papakonstantinou, P. Velikhov, "A Brief Introduction to XMAS," <http://www.npaci.edu/DICE/mix-system.html>.

[8] D. Chamberlin et al., "XQuery 1.0 : An XML Query language," <http://www.w3.org/TR/xquery>, 2001.

[9] OMG Unified Modelling Language Specification, <http://www.rational.com/uml>, 1999.

[10] James rumbaugh, Ivar Jacobson, Grady Booch, "The unified modeling language reference manual," Addison Wesley Longman Inc., 1999.

[11] 채원석, 하 안, 김용성, "UML 클래스 다이어그램을 이용한 XML 문서 구조 다이어그램", 정보처리논문지, 제6권 제10호, pp.2670-2679, 1999.



하 안

e-mail : white@kic.ac.kr

1992년 덕성여자대학교 전산학과(학사)

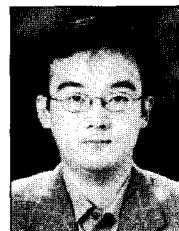
1994년 이화여자대학교 전자계산교육전공(석사)

2000년 전북대학교 대학원 전산통계학과(이학박사)

2000년~2001년 중앙대학교 정보통신연구소 연구전담교수

2001년~현재 경인여자대학 컴퓨터정보기술학부 전임강사

관심분야 : XML 응용, 객체지향 모델링, 컴포넌트 모델링, 애니메이션, 멀티미디어



김기한

e-mail : ghkim1@etri.re.kr

1997년 중앙대학교 컴퓨터공학과(학사)

1999년 중앙대학교 컴퓨터공학과(석사)

1999년~현재 중앙대학교 컴퓨터공학과 박사과정

2001년~현재 ETRI 부설 국가보안기술 연구소 연구원

관심분야 : 정보보증, 시스템 프로그래밍, S/W 개발 방법론