

객체지향 데이터베이스 기반의 XML 응용을 위한 통합 설계 방법론

김경수*, 최문영**, 주경수***

A Unified Design Methodology for XML Applications based on OODB

Kyung-Soo Kim*, Mun-Young Choi**, Kyung-Soo Joo***

요 약

본 연구는 UML을 이용해 유스 케이스에 의한 순차 다이어그램을 도출하여 클래스 다이어그램을 만든 후 그 클래스 다이어그램에 의해서 XML 모델링과 데이터 모델링을 구현하고자 한다. XML 모델링을 위해 UML 클래스를 XML 문서로 변환시키는 가이드라인을 제시하고, 제시한 방법에 따라 UML 클래스를 XML DTD로 도출하는 예를 보일 것이며, 한편 데이터 모델링은 UML 클래스를 객체지향 데이터베이스로 변환시키기 위한 변환 방법을 제시하며, 제시한 변환 방법에 따라 구현 예를 보일 것이다.

Abstract

In this paper, we implement the XML modeling and data modeling by the class diagram that is made by the use case after drawing out the sequence diagram. For the XML modeling, the guiding line to map the UML classes on the XML document will be presented. Then, an example to generate the XML DTD from the UML classes will be shown under the presented guiding line. In the data modeling, the transformation method from the UML classes into the object-oriented database also will be proposed. Finally, we will give an example developed by the proposed transformation method.

본 연구는 정보통신부의 ITRC 사업에 의해 수행된 것임

* 천안외국어대학 컴퓨터정보과 조교수

** 순천향대학교 전산학과 박사과정

*** 순천향대학교 정보기술공학부 교수

논문접수 : 2002. 1. 28

심사완료 : 2002. 3. 14

떻게 저장할 것인가 하는 제안들을 하고 있으며, 주어진 XML 데이터나 DTD로부터 정보의 의미에 대하여 데이터베이스 제약사항들을 다루는 방법을 제안하고 있다(5,6,7,8).

I. 서론

프로그램 개발은 요구분석, 설계, 구현, 시험 등의 과정을 거쳐 완성된다. 프로그램 개발 과정을 일관된 관점에서 진행하고 이들 업무 담당자들간의 의사 소통을 용이하게 하여 프로젝트가 효율적으로 진행될 수 있는 공통된 객체 지향적 설계 언어가 필요하다. 아울러 XML 프로그램이나 XML을 이용한 B2B 시스템 구축과 같은 XML 프로젝트에서는 객체 지향적 설계 언어인 UML을 이용하여 개발의 효율을 필요로 하고 있다[16]. 그러므로 UML을 이용한 XML 어플리케이션 개발에 있어 UML 클래스 다이어그램을 XML DTD로 변화하는 기법을 그리고 UML 클래스 다이어그램을 객체지향 데이터베이스 스키마를 도출하는 변환 기법과 제안된 변환 규칙을 이용하여 객체지향 스키마 스크립트를 구현했다. 이를 위하여 먼저 2장에서는 관련 연구, 3장에서는 UML을 이용한 모델링, 4장에서는 UML 클래스를 XML로 모델링하는 방법, 5장에서는 UML 클래스를 객체지향 데이터베이스 스키마로의 변환을, 마지막으로 결론을 제시한다.

II. 관련 연구

XML이 단순한 콘텐츠에서 데이터베이스로까지 그 적용 분야가 확장되면서 XML로 표현된 정보들을 어떻게 효율적으로 저장하고 관리할 것인지에 대한 기술도 XML과 함께 발전되어 가고 있다. 그러나 기존의 DBMS가 XML 데이터를 효율적으로 저장·관리할 수 있는가 하는 것이 큰 관심사이다.

많은 연구들이 복잡한 XML 데이터를 관계형 데이터베이스에 저장할 수 있도록 데이터베이스와 XML 문서의 연결에 대하여 발표되었다[12,14]. 발표된 연구들의 내용 분류하면 주어진 XML 파일을 관계형 데이터베이스에 어

또한 XML 문서가 객체-관계 시스템으로 구조화된 데이터 타입을 저장하는 방법이 제안되고 있다(4,7). 다른 한편으로는 객체-관계 데이터베이스로부터 XML로 변환하는 내용을 다루고 있으며, 그 내용은 어떻게 XML 파일을 객체-관계 데이터베이스에 저장하여 빠르게 질의할 것인가를 다루고 있다(3,9). 또한 객체-관계 데이터베이스 시스템은 XML 문서들을 서로 연결하기 위한 것이며, XML의 중첩된 문서들을 객체-관계 모델로 변환하는 내용을 다루고 있다.

객체지향 데이터베이스 시스템의 풍부한 데이터 모델링 때문에 전자문서의 저장이 잘 이루어진다(2,8). 그러나 객체지향 데이터베이스 시스템은 효과적인 방법으로 대량의 데이터를 다루기에는 적합하지 않다(13).

III. UML을 이용한 모델링

UML은 소프트웨어 시스템의 구조물을 명세화, 구조화 시각화, 문서화뿐만 아니라 비-소프트웨어 시스템 그리고 비즈니스 모델링을 위한 하나의 언어이다(15). 또한 XML 프로그램이나 XML를 이용한 B2B 시스템 구축과 같은 XML 프로젝트에서는 객체 지향적 설계 언어인 UML을 이용하여 개발의 효율을 높일 수 있다.

순차 다이어그램은 응용 프로그램에서 사건의 흐름을 분명하게 하는 역할을 하며 글로 풀어서 쓴 유스 케이스(Use case)를 메시지로 표현한 그림으로 변환할 수 있다 또한 그 그림은 항상 더 간결하게 이해되고, 특히 우리가 클래스 다이어그램을 만들 때 작업을 쉽게 해준다. 그러므로 UML을 이용해 유스 케이스에 의한 순차 다이어그램을 도출하여 클래스 다이어그램을 만든 후 그 클래스 다이어그램에 의해서 그림 1과 같이 XML 모델링과 데이터 모델링으로 변환할 수 있다.

그림 2의 순차 다이어그램은 객체들간의 메시지과정을 설명한다. 이 다이어그램은 메시지 객체들과 진행과정에

관계된 actors를 포함하고 있다[11].

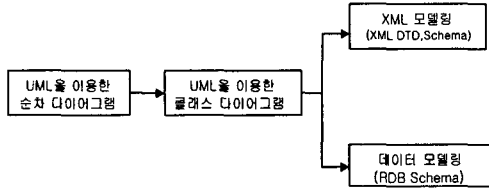


그림 1. UML을 이용한 XML 모델링과 데이터 모델링

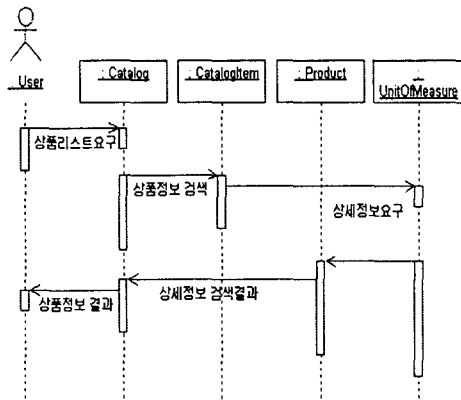


그림 2. 메시지 객체와 순차 다이어그램

그림 2에서 객체들과 actors 이외에 수직의 축은 시간을 표현한다. 화살표는 행위 또는 객체와 actors간에 발생하는 사건을 표현한다. 이 순차 다이어그램은 3개의 클래스와 2개의 새로운 actors(User, UnitOfMeasure)를 한다. 이것은 시스템에서 actors를 설명하는 객체를 의미한다. 그림 3의 클래스 다이어그램은 3개의 클래스, 그리고 그들의 연관된 속성과 서브-객체를 예를 들어 설명한다.

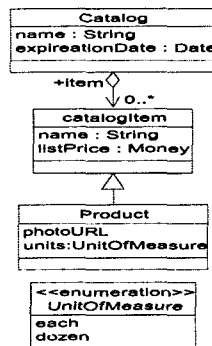


그림 3. 클래스 다이어그램

그림 3은 Catalog 와 CatalogItem간에 관계성에서 연관 관계를 갖는다. 왜냐하면 CatalogItem은 Catalog 객체에 의존하고 있기 때문이다. 또한 CatalogItem은 Product의 부모 클래스가 된다. 이것은 Product 클래스에서 CatalogItem의 일부 속성을 상속받기 위해서이다. 속성의 상속은 하위 클래스에서 유용하게 사용될 수 있기 때문이다. 다음의 <표 1>은 각 클래스의 객체의 특성을 서술한 것이다. 또한 각각의 객체에 대한 속성, 사상수, 관계성은 <표 2>와 같다.

표 1. 객체 정의

클래스	기술
Catalog	상품목록의 이름과 그 정보에 대한 만기일자를 가지고 있다. 사용자가 특정 카탈로그의 목록을 원하면, CatalogItem 객체를 통해서 정보를 가져온다.
CatalogItem	특정한 목록의 이름과 가격정보를 가지고 있다. 자식 클래스인 Product를 이용하여 각 상품에 대한 정보를 표현할 수 있다.
Product	CatalogItem에서 상속받은 속성과 자신이 가지고 있는 속성을 이용하여 하나의 제품에 대한 여러 가지 정보를 표현한다.

표 2. 객체들의 속성(서브-객체), 사상수, 관계성

클래스	속성/서브-객체	사상수	관계	기술
Catalog	name	1..1	복합	상품의 목록을 기술해야 함
	expirationDate	1..1	복합	만기일자를 기술해야 함
CatalogItem	name	1..1	복합	세부 상품이름을 기술해야 함
	ListPrice	1..1	복합	상품의 가격을 기술해야 함
Product	PhotoURL	1..1	복합	상품의 사진이 있는 URL을 기술해야 함
	Units	1..1	복합	단위 수량을 기술해야 함

IV. XML 모델링

XML 어플리케이션에서 기본이 되는 XML DTD 설계 과정에서 최적의 DTD가 되도록 하기 위해서 물리적 모형이나 도해를 만드는 것을 말한다.

4.1 UML를 XML로의 변환 방법

① UML 클래스를 XML 엘리먼트로 변환

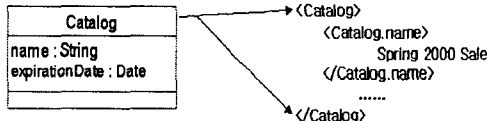


그림 4. UML 클래스의 변환

UML 클래스의 경우 XML로의 변환은 그림 4과 같이 하나의 XML 요소로 완전히 변환된다. XML 태그 이름은 UML 클래스 이름과 같으며, 태그 이름은 공백을 포함하지 않고, 알파벳, 숫자만으로 제한된다. 특수문자인 '.', ':', '_'를 포함할 수 있으며, 태그 이름은 각각의 글자나 '_'로 시작한다. UML 클래스나 패키지 이름은 XML 문자로 사용되지 않는다. 그림 4은 UML 클래스를 XML 엘리먼트로 변환한 것이다[1,10].

② 상속(Inheritance)

상속을 사용하는 것은 객체 지향 모델링에서 기본적으로 적용되는 특성이다. 그림 5는 상속을 UML 다이어그램으로 표현한 것이며, 상속받은 클래스는 부모 클래스의 어트리뷰트를 사용할 수 있다. 이것은 XML DTD로의 변환에서도 중요하게 사용된다.

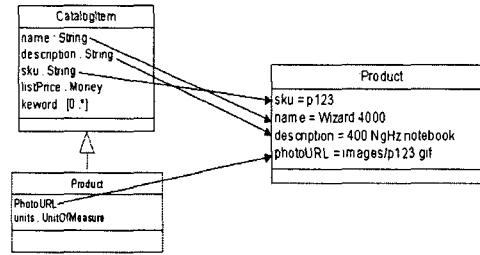


그림 5. 상속을 UML 다이어그램으로 표현

기존의 XML 표준은 상속을 표현하여 생성을 하는 메커니즘을 가지지 못하였으며, DTD는 엘리먼트를 정의하는데 상속을 표현하지 못하였다. 단지 다른 엘리먼트 안의 엘리먼트가 포함된 집합 구조를 표현할 수밖에 없었다. 위와 같이 XML에서의 기술은 어트리뷰트, 관계 참조, 문장에서의 상속을 그대로 복사해서 사용한다.

③ UML 어트리뷰트를 XML 엘리먼트로 변환

UML 클래스의 인스턴스가 주어졌을 때, XML에 어트리뷰트들이 각각 매칭하게 된다. 이것은 XML 변환기법들을 디자인할 때, 자식 엘리먼트를 분리하는 것과 같이 각 데이터 값을 정의하는 일반적인 관례이다. 그림 6은 UML 객체 인스턴스 다이어그램을 표현하였으며, 이것을 XML 문서로 변환한 결과를 보여준다.

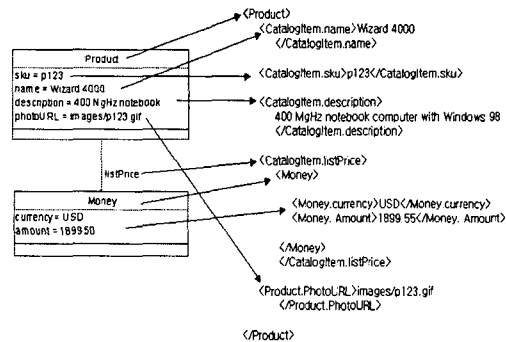


그림 6. 어트리뷰트 값을 가진 하나의 객체와 변환된 XML 문서

④ UML 어트리뷰트를 XML 어트리뷰트로 변환

UML 어트리뷰트는 XML 어트리뷰트로 표현이 가능하다. 그림 6의 클래스 다이어그램을 XML로의 변환하는 것을 기초로 한다면, product 엘리먼트의 sku와 photoU UML 어트리뷰트는 XML 문서의 어트리뷰트로 변환된다

currency와 amount는 money 엘리먼트의 XML 어트리뷰트로 변환된다. 만약 UML 어트리뷰트의 값이 string 값을 포함한다면, XMI(XML Metadata Interchange)는 각각의 엘리먼트나 어트리뷰트를 허용한다. 반면에 XML 어트리뷰트의 값이 항상 공백이 된다면, XML 파서는 모든 공백 문자(탭, 개행 문자, 리턴, 여러 가지 공백)를 삭제한다(11).

⑤ UML 복합 연관관의 변환

UML 복합연관은 Catalog와 CatalogItem과 같이 그림 3에 보여진다. 복합연관은 항상 UML 다이어그램에서 보여지며, 관계의 끝에 채워진 다이아몬드 모양으로 나타난다. 이와 유사한 것으로는, XML 엘리먼트가 자신의 자식 엘리먼트를 가지는 것이다. 그림 7에서는 세 가지의 Catalog Item으로 구성된다. 그림 7에서 하나의 Catalog 객체는 item 규칙 이름을 사용하였으며, 이는 세 개의 Product 객체와 같다.

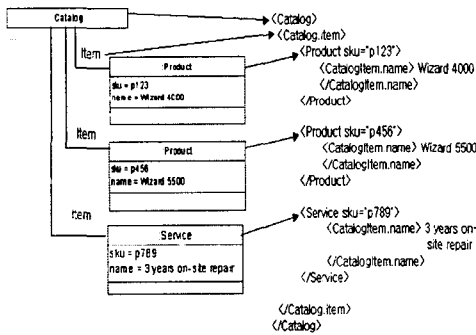


그림 7. CatalogItem의 구성과 변환된 XML 문서

5.2 UML을 XML DTD로 변환

XMI는 UML 문서로부터 XML DTD 생성시 유연하게 처리하거나 강제적으로 느슨하게 변환한다. 이 방법은 DTD의 표현에 제한을 두었기 때문에 부분적인 모델의 변환을 가능하게 만들었다. 반면에, UML에서 엄격한 DTD의 생성을 가능하게 한다. UML 스테레오 타입은 디자이너의 컨트롤을 지원하는데, 이러한 엄격한 DTD는 표준화된 느슨한 DTD의 서브셋이기 때문이다(11). 그림은 UML 클래스 다이어그램을 XML DTD로 변환한 결과이다.

V. 데이터 모델링

데이터베이스 스키마 변환 과정에서 정보 구조로부터 논리적 개념을 이용하여 어떤 논리적(데이터) 구조로 표현하는 것이 필요한데, 데이터 모델링이란 이 변환과정을 말한다(17).

5.1 변환 방법

다음은 UML 클래스로부터 객체지향 데이터베이스 스키마로 변환하는 방법이다.

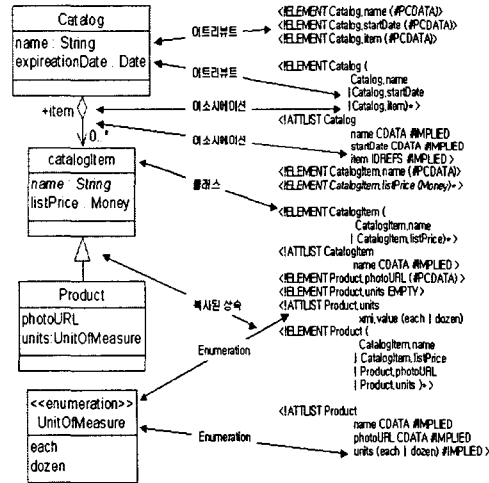


그림 8. UML을 XML의 느슨한 DTD로 변환

- ① “지속적인” 클래스는 지속적인 OODB 클래스가 된다.
- ② 인터페이스가 “지속적인” 클래스 OODB 인터페이스로 실행되기 위한 인터페이스를 지원하는 언어 (JAVA, ODL)나 또는 추상적인 베이스 클래스를 위해 오직 순수한 가상멤버 그리고 no data 멤버의 그 언어는 인터페이스를 가지고 있지 않는다.
- ③ type 분류자는 enum 또는 typedef로 표시된다.
- ④ 클래스의 attribute는 적당한 type 변환과 제한은 OODB 클래스의 attribute가 된다.

- ⑤ 만약 nullable 태그가 나타나면 nullable 상수형 (nullable_short) 사용하고 바인딩을 지원하는 널값(ODL) 사용한다. 만약 그렇지 않으면, 그것을(C++, JAVA) 무시한다.
- ⑥ 만약 attribute가 initializer를 가지고 있으면, 생성에 초기화 코드를 추가한다. 생성자 메소드의 어느 한쪽의 부분으로서 또는 C++ 멤버에 초기화는 목록에 나열한다.
- ⑦ 서브 클래스는 클래스 선언에 슈퍼클래스 사양을 포함한다.
- ⑧ 연관 클래스는 연관 클래스의 attribute와 클래스를 생성한다.
- ⑨ 명백한 객체 일치 (oid) 또는 후보 키 (alterna oid)가 만약 바인딩을 지원하면 키의 선언을 명시한다. 만약 지원하지 않으면, 객체의 집합에 역제를 점검하는 적당한 메소드를 공급한다.
- ⑩ 각 명백한 역제에 대한 적당한 클래스에 메소드를 추가하고 안전하게 한다. 시스템은 메소드를 시스템이 그 역제가 만족할 만한 것을 요구할 때마다 호출한다.
- ⑪ 적당한 객체 또는 연관된 Multiplicity로부터 유래한 컬렉션 타입에 대해서는 연관 클래스를 가지고 있지 않은 각 이진 연관을 위해 관계성을 생성한다. 명백한 화살표들이 연관되어 있지 않다면 역 관계성들을 사용한다.
- ⑫ 다른 클래스의 각 역할 링크를 위한 연관 클래스에 관계성을 생성한다.
- ⑬ 코드를 생성하거나 또는 특별한 OODB의 요구처럼 어떠한 복합 집단화 연관을 위해 삭제를 전파하기 위하여 OODB 특징들을 사용한다.
- ⑭ 3진으로 이루어진 연결을 위해 연관 클래스를 생성한다. 그리고 연관 클래스에서 Multiplicity를 주던 적당한 자료 type의 연합한 클래스까지 관계성을 생성한다.

5.2 변환 예

그림 3의 UML 클래스 다이어그램을 5.1에서 제시한 변환 방법에 따라 ODL을 이용하여 객체지향 데이터베이스 스키마로 변환하면 그림 9와 같다.

VI 결론

XML은 웹에서 구조화된 정보나 반-구조화된 정보를 교환하기 위한 표준 마크업 언어로 채택되어 가고 있다. 한편 웹에서 정보교환을 위한 XML 메시지의 소스 데이터는 레거시 데이터베이스에 저장되어 있기 때문에, 이에 따라 XML 응용과 데이터베이스 시스템과의 원활한 연계가 요구되어진다.

XML 응용과 데이터베이스 시스템 사이의 원활한 연계를 위해서 XML DTD를 관계형 데이터베이스 스키마로 변환하는 방법에 대해서는 많은 연구가 진행되었으나, XML DTD를 객체지향 데이터베이스 스키마로 변환기 위한 연구는 미미한 실정이다. 멀티미디어 응용 등을 위하여 객체지향 데이터베이스의 보급이 확대되고 있는 현실에서 XML DTD를 객체지향 데이터베이스 스키마로 변환기 위한 연구는 조속히 요구되고 있다.

본 논문에서는 계층구조를 갖는 XML 데이터를 계층구조를 지원하는 객체지향 데이터베이스로 저장 가능하도록 하는 모델링 방법론을 제안하였다. 이를 위하여 우선적으로 객체지향 설계언어인 UML 이용해서 클래스 다이어그램을 도출한 후, 클래스 다이어그램에 의해서 XML DTD 설계를 위한 XML 모델링을 소개하고 이들의 모델링으로 교환되는 XML 데이터를 효율적으로 저장하기 위하여 객체지향 데이터베이스 스키마 설계를 위한 데이터 모델링 방법을 제안하였다. 아울러 변환 방법에 의해 객체지향 데이터베이스 스키마 스크립트를 구현했다.

```

class Catalog (extent Catalogs){
  attribute String name;
  attribute Date expirationDate;
  relationship set<CatalogItem> item;
};

class CatalogItem (extent CatalogItems) {
  attribute String name;
  attribute Money listPrice;
  relationship Catalog;
  relationship Product;
};

class Product : CatalogItem (extent Products) {
  attribute photoURL;
  attribute UnitOfMeasure units;
  relationship CatalogItem;
};

class UnitOfMeasure (extent UnitOfMeasures) {
  attribute each;
  attribute dozen;
};

```

그림 9. 객체지향 데이터베이스 스키마 스크립트

참고문헌

- [1] Andrew Davidson, Matthew Fuchs, Mette Hedin, Mudia Jain, Jari Koistinen, Chris Lloyd, Murray Maloney, Kelly Schwarzhof, "Schema for Object-Oriented XML 2.0", July, 1999. See <http://www.w3.org/TR/NOTE-SOX>
- [2] Boem, K., Aberer, K.: HyperStorM-Administering Structured Documents Using Object-Oriented Database Technology. Proc. of the ACM SIGMOD Int. Conf. on Management of Data, Montreal, Canada, June 1996.
- [3] Carey, M., Florescu, D., Ives, Z., Lu, Y., Shanmugasundaram, J., Shekita, E., Subramanian, S.: XPERANTO: Publishing Object Relational Data as XML. In Suci, D., Vossen(eds.), G., Proceedings of the Third International Workshop on the Web and Databases, WebDB 2000, Dallas, Texas, USA, May 18-19, 2000, 105-110.
- [4] Cheng, J., Xu, J.: XML and DB2, In Sixteenth International Conference on Data Engineering(ICDE'00), 28 February-3 March 2000, San Diego, IEEE Computer Society Press, Los Alamitos, CA, 2000, 569-576.
- [5] Florescu, D., Kossmann, D.: Storing and Querying XML Data using an RDBMS. Data Engineering 22:3 (1999), 27-34.
- [6] Kanne, C.-C., Moerkotte, G.: Efficient Storage of XML Data. In Sixteenth International Conference on Data Engineering(ICDE'00), 28 February-3 March 2000, San Diego, IEEE Computer Society Press, Los Alamitos, CA, 2000, 198.
- [7] Klettke, M., Meyer, H.: XML and Object-Relational Database Systems-Enhancing Structural Mappings Based on Statistics. In Suci, D., Vossen(eds.), G., Proceedings of the the Third International Workshop on the Web and Databases, WebDB 2000, Dallas, Texas, USA, May 18-19, 2000, 63-68.
- [8] Lee, D., Chu, W. W.: Constraints-Preserving Transformation from XML DTD to Relational Schema. In Laender, A., Liddle, S., Storey(eds.), V., Conceptual Modeling, Salt Lake City, Utah, USA, October, 9-12, 2000, LNCS 1920, Springer-Verlag, Berlin, 2000, 323-338.
- [10] Shanmugasundaram, J., Shekita, E. J., Bar, R., Carey, M. J., Lindsay, B. G., Pirahesh, H., Reinwald, B.: Efficiency Publishing Relational Data as XML Documents. In Abbadi, A. E., Brodie, M. L., Chakravarthy, S., Dayal, U., Kamel, N., Schlageter, G., Whang(eds.), K. Y., VLDB2000-Proceedings of the 26th International Conference on Very Large Data Bases, September 10-14, 2000, Cairo,

Egypt, Morgan Kaufmann Publishers 2000.

[11] Hiroshi Matuyama, Kent Tamura, Naohiko Uramoto, XML and Java Developing Web Applications, Addison Wesley, 1999, pp 20-21

[12] James Rumbaugh, Object-oriented modeling and design, Englewood Cliffs, N.J, Prentice Hall, 1991.

[13] Joo Kyung-soo, "A Design of Middleware Components for the Connection between XML and RDB", 2001 IEEE International Symposium on Industrial Electronics Proceedings, Pusan, Korea, June 2001

[14] Michel Goossens, Janne Saarela, A Practical Introduction to SGML, volume 16(3), In TUGboat, 1995

[15] World Wide Web Consortium, The XML Data Model. <http://www.w3.org/XML/Datamodel.html>, 2000

[16] R.G.G.Cattell, Douglas K. Barry, The Object Database Standard : ODMG 2.0, Morgan Kaufmann Publishers, Inc, 1997.

[17] Robert J. Muller, Database Design for Smarties: Using UML for Data Modeling, Morgan Kaufmann Publishers, Inc, 1999.

[18] 이상태, 주경수, "제약조건 유지를 위한 XML DTD의 관계 스키마로 변환 방법", 한국인터넷 정보학회, 제1권, 제2호, pp.189-196, 2000

[19] 이상태, 주경수, "UML 객체모델의 ORDB 객체 매핑", 한국인터넷 정보학회, 제2권, 제1호, pp.187-191, 2001

[20] 이상태, 주경수, "객체모델을 이용한 XML DTD의 ORDB 스키마로의 변환", 한국데이터베이스 학회 춘계 Conference, pp.303-310, 2001

[21] 이상태, 이정수, 주경수, "객체모델을 기반으로 한 XML DTD의 RDB 스키마로의 변환 방법", 대한전자공학회 하계종합학술대회 논문집 III, pp.113-116, 2001

[22] 방승윤, 주경수. "XML 스키마를 관계형 스키마로의 변환에 관한 연구", 한국정보처리학회 및 데이터 공학연구회 학술발표논문, 2001, 293-299.

[23] 최문영, 방승윤, 주경수, "객체모델을 이용한,

XML DTD의 OODBMS 스키마로의 매핑", 한국정보처리 지식 및 데이터 공학연구회 학술발표논문, 2001, 3001-307.

[24] Alexander & Tom 저 유진희, 박성준 역, Professional Java XML Programming, 정보문화사

[25] Simon North 저 노정운 역, 초보자를 위한 XML 21일 완성, 인포북.

저자 소개



김 경 수
1995 순천향대학교 전산학과 졸업(석사)
1997 순천향대학교 전산학과 수료(박사)
1998~현재 천안외국어대학 컴퓨터정보과 조교수
<관심분야>
데이터베이스, XML 모델링.



최 문 영
1998 청운대학교 전자계산학과 졸업(학사)
2000 청운대학교 정보산업대학원 전산전자정보학과 졸업(석사)
2000~현재 순천향대학교 전산학과 재학중(박사과정)
<관심분야>
Database Systems, Object-oriented, XML



주 경 수
1980 고려대학교 이과대학 수학과 졸업(학사)
1982 고려대학교 일반대학원 전산학과 졸업(석사)
1986 고려대학교 일반대학원 전산학과 졸업(박사)
1998년 University of North Carolina, visiting Professor
1986년~현재 순천향대학교 정보기술공학부 교수
<관심분야>
Database Systems, System Integration, object-oriented Systems