

Validation RDF 저작 툴의 설계 및 구현

조성훈^o 이무훈 손덕주* 조현규* 송병렬* 이찬섭 최의인

한남대학교 컴퓨터공학과

*한국전자통신연구원

{shcho^o, mhlee, cslee, eichoi}@dclab.hannam.ac.kr, {djson, hkcho, sby}@etri.re.kr*

The Design and Implementation of Validation RDF Authoring Tool

Sung-Hoon Cho^o Moo-Hoon Lee Duk-Joo Son* Hyun-Kyu Cho*

Byoung-Youl Song* Chan-Seob Lee Eui-In Choi

Dept. of Computer Engineering, Hannam University
Electronics and Telecommunications Research Institute*

요 약

웹의 이용이 보편화되면서 웹 상의 자원(Resource) 또한 급격히 증가하고 있다. 이러한 상황에서 웹 자원을 효율적으로 기술할 수 있는 RDF(Resource Description Framework)는 웹 자원을 체계적으로 관리할 수 있는 지평을 열어주었다. 그러나 현재 RDF 문서의 저작환경은 저차원적인 텍스트 편집 수준으로 파싱(parsing)과 유효성 검사(validation checking)를 지원하지 못하고 있는 실정이다. 따라서 본 논문에서는 저작환경을 개선함과 동시에 RDF 및 XML 문서에 대한 파싱 및 유효성 검사를 효율적으로 수행하고, RDF로 기술된 웹 자원을 n-triple 형식으로 생성할 수 있는 RDF 저작 툴을 설계 및 구현하여 유효성이 보장된 웹 자원을 효율적으로 생성 및 처리할 수 있도록 하였다.

1. 서론

인터넷의 확산과 웹 서비스 사용자의 증가로 정보의 교환과 재사용의 필요성이 급속히 증가하고 있다. 그러나 정보의 교환과 재사용에 앞서 해결해야 할 사항이 두 가지 있다. 첫째는 교환 및 재사용될 정보가 서로 다른 응용 분야에서 서로 다른 의미(semantic)를 가지고 사용될 경우에 이를 구별할 수 있어야 한다는 것이고, 둘째는 정보가 분산된 웹 환경에서 요청된 자원이 존재하는 사이트(site)를 명확히 기술할 수 있어야 한다는 것이다. 이 두 사항이 해결되지 않을 경우 자원의 낭비는 물론이고 효율성 역시 낮아진다. 이것에 대한 예로 사용자가 인터넷 검색기에서 검색할 때 일치하지 않는 결과를 다수 출력하는 것과 전자상거래 분야에서 구매자가 구매를 요청하기 위해 물품 카탈로그를 검색할 때 다른 제품의 카탈로그를 결과로 전달할 때와 같은 것이 있다. 이런 문제의 원인은 웹 자원 자체에 있지 않다. 즉 웹 자원을 기술하는 메타데이터가 적절하지 못하거나 사용되지 않는 것이 원인이다. 기존의 웹 환경에서 이루어지는 서비스나 자원에 대해서 메타데이터의 사용이 사실상 없다는 것은 웹 환경에 맞는 메타데이터 표준의 부재에서 비롯된 것이라 할 수 있다. 이에 W3C(World Wide Web Consortium)에서 메타데이터 기술 구조로 RDF(Resource Description Framework)를 제안하였다[1,2,3,4]. RDF 모델(model)과

문법(syntax)에서는 자원을 속성 중심으로 기술할 수 있는 구조를 제공한다. 이러한 구조는 웹 자원간의 관계와 웹 자원에 대한 기술에 적합하다.

본 논문은 RDF syntax와 RDF schema를 기초로 하여 자원을 효율적으로 기술할 수 있는 저작 환경과 DTD와 XML Schema에 기반하여 RDF와 XML 문서의 유효성을 검증하는 유효성 검증기 validator를 설계하고 개발하였다.

논문의 구성은 2장에서 구현 환경 및 RDF 문서 저작 툴의 구성을 알아보고, 3장에서 RDF 및 XML 문서의 처리 과정을 설명한 뒤 마지막으로 4장에서 결론과 향후 연구내용을 기술하도록 한다.

2. RDF 저작 툴의 구성

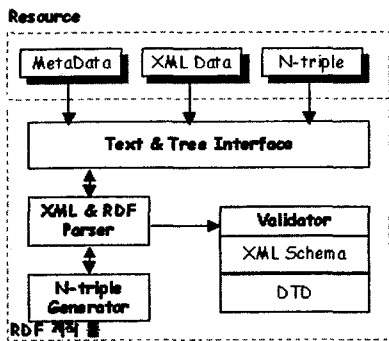
2.1 RDF 저작 툴의 구현 환경

본 논문에서는 RDF 저작 툴의 구현환경은 Java 기반의 J2SE v1.4이며 파서(parser) 구현을 위한 API로 Xerces 2 Java Parser 2.0.2 Release와 JAXP(Java API for XML Processing) v1.2를 사용하였다[5]. RDF 모델을 처리하기 위해 Jena v1.4를 사용하였다[6].

2.2 RDF 저작 툴의 구성

RDF 저작 툴의 구성은 기능적 측면에서 XML Schema 유효성 검증기, DTD 유효성 검증기, 텍스트 인터페이스(Text Interface), 트리 인터페이스(Tree Interface), XML 및 RDF 파서(Parser), n-triple 생성기(Generator) 등으로 분류할 수 있다. 그림 1은 기능에

* 본 연구는 한국전자통신연구원의 "ebXML 메타정보 저작 도구 개발"의 연구 결과임



[그림 1] RDF 저작 톨의 구성

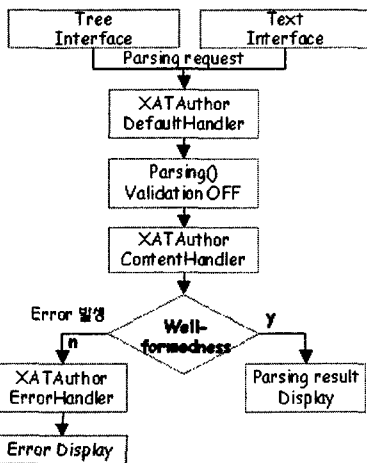
따른 RDF 저작 톨의 구성도이다. 그림 1에서 보이는 바와 같이 Text와 Tree인터페이스 모두 하나의 파서, 유효성 검증기, n-triple 생성기를 사용하는 것을 알 수 있다. 각각은 모듈 단위로 구분되어 있으며 저작 인터페이스에 독립적으로 저작 중인 콘텐츠를 처리한다. 이와 같은 구성을 통해 RDF에 제한적인 것이 아니라 XML 기반의 모든 자원을 저작할 수 있는 저작 환경이 제공되게 된다. 각각의 자세한 처리 루틴은 3장에서 설명하도록 하겠다.

3. RDF 저작 톨의 처리 루틴

RDF 저작 톨의 처리 루틴은 파싱, 유효성 검증, n-triple 생성에 기초한다.

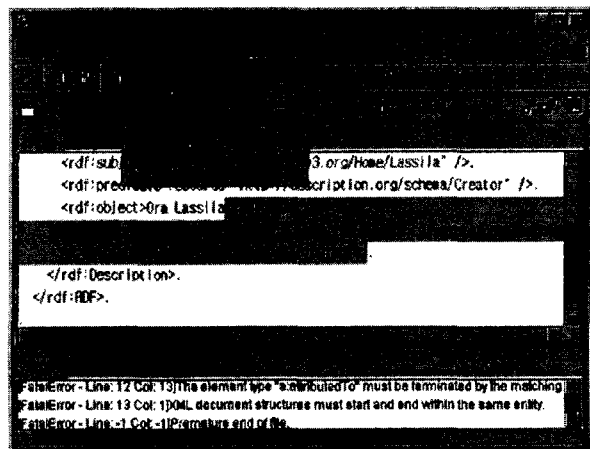
3.1 파싱 루틴

파싱은 XML 문서의 구문 검사를 의미한다. 즉 XML 문서가 문법에 맞게 구성되어 있는지를 검사하는 처리 과정을 말한다. 이때에 DTD나 XML Schema 문서에 대한 Location을 기술하는 문장이 명시되어 있어도 유효성 검사는 수행하지 않는다.



[그림 2] 파싱 루틴의 구성 및 흐름도

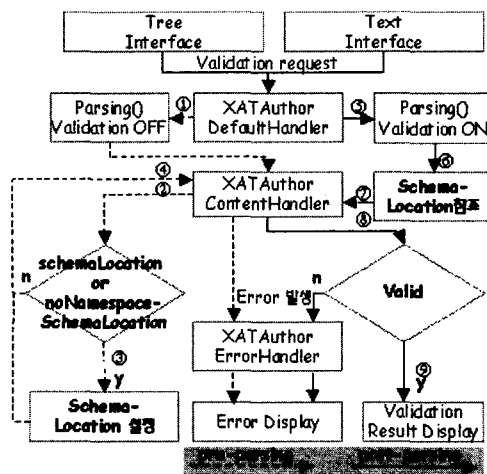
본 논문의 파싱 루틴은 두개의 인터페이스인 Text와 Tree에서 저작되거나 저작중인 RDF 및 XML 문서에 대해 Parsing을 지원한다. 처리 루틴은 처리 속도를 고려하여 SAX를 기반으로 하였다. 파싱 처리는 그림 2와 같이 텍스트 인터페이스와 트리 인터페이스에 있는 문자열을 판독한 뒤 well-formedness한지를 검사한다. 이 때 토큰이 Preprocess Instruction, ELEMENT, ATTRIBUTE, CHARACTER인지에 대한 유형(type) 판별 및 처리는 XATAuthorContentHandler에서 처리하며 유효하지 못해 오류가 발생했을 경우에는 XATAuthorErrorHandler에서 오류 처리를 수행한다. 그림 3은 RDF 문서의 파싱 결과이다.



[그림 3] RDF 문서 파싱 결과

3.2 유효성 검증 루틴

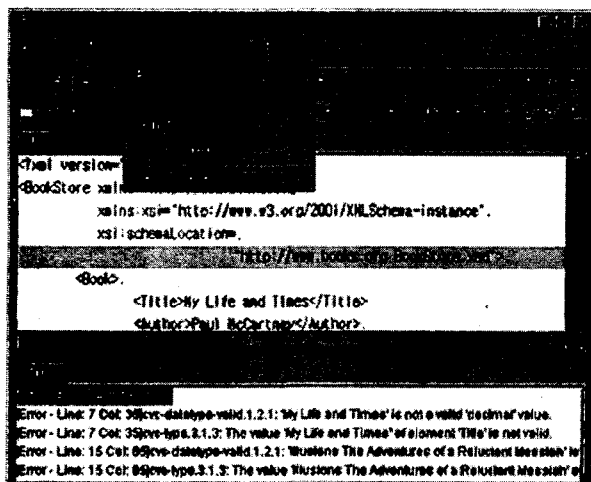
유효성 검증 수행 루틴을 살펴보면 파싱 루틴과는 다르게 2회에 걸쳐 수행된다.



[그림 4] 유효성 검증 루틴의 구성 및 흐름도

첫 번째 파싱 단계는 pre-parsing이라고 하고 두 번째는 post-parsing이라 한다. pre-parsing 단계에서는 유효성 검증은 수행하지 않고 well-formedness만을 검사한 뒤 DTD 및 XML Schema의 위치 정보를 얻게 되는데 이 정보는 URI 형태로 획득되고 상대(relative) 및 절대(absolute) 경로 두 가지 형식 모두를 포함할 수 있다. post-parsing 단계에서는 유효성 검증을 위한 설정을 한 뒤에 pre-parsing 단계에서 설정된 schema-Location과 noNamespaceSchemaLocation를 참조하여 스키마를 판독하고 스키마를 기초로 유효성 검증을 수행한다. 이와 같은 처리 과정에서 각각의 처리기(Handler)가 수행하는 기능은 파싱 루틴과 같다.

그림 4는 유효성 검증 루틴의 구성 및 흐름도이며, 그림 5는 본 논문에서 구현한 유효성 검증기를 통해 schemaLocation을 포함한 XML 문서를 검증한 결과이다.



[그림 5] schemaLocation을 포함한 XML 문서의 유효성 검증 결과

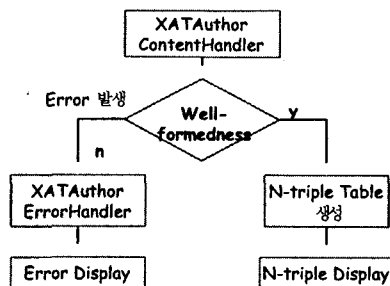
3.3 N-Triple 생성 루틴

RDF는 속성 중심으로 데이터를 표현한다. 즉, RDF syntax를 통해 데이터간의 관계와 의미해석이 가능하다는 것이다. 표현 방법은 RDF Schema를 따라 표현되는데, 보다 쉽고 가시적으로 표현하는 방법으로 n-triple이 있다. n-triple은 object-attribute-value의 구조를 가지며 A(O,V)와 같이 표현할 수 있다[7].

n-triple 생성 루틴은 파싱을 수행하면서 RDF에서 정의한 요소(ELEMENT)가 발생하면 자원(Resource)형태로 처리한다. 이때 RDF 문서에 대해 validation은 검사되지 않는다.

루틴의 세부적인 처리 과정을 살펴보면 우선 그림 2의 파싱 루틴과 동일하며 다른 점은 XATAuthorContentHandler에서 RDF 문서에 대해 well-formedness를 검사한 뒤 n-triple 테이블을 생성한다는 것이다. XATAuthorContentHandler는 RDF에서 정의된 요소가 발생하게 되면 요소의 형태, 즉 Subject,

Predicate, Object에 따라 n-triple 테이블에 저장한다. 에러 처리의 경우 파싱 루틴, 유효성 검증 루틴, n-triple 생성 루틴 모두 동일하다. 아래의 그림 6은 n-triple 생성 루틴의 구성 및 흐름도이다.



[그림 6] N-triple 생성 루틴의 구성 및 흐름도

4장 결론 및 향후 과제

본 연구는 RDF 및 XML 문서를 저작 단계에서 유효성 검증할 수 있도록 구현하여 유효한 데이터 집합을 효율적으로 생성할 수 있고, ebXML(electronic business eXtensible Markup Language)과 같은 전자상거래 프레임워크에서 사용되는 전자카탈로그 레지스트리(Registry)에서 RDF로 기술되는 전자카탈로그의 메타데이터를 효율적으로 관리할 수 있을 것이다.

그러나 XML Schema에 기술된 모든 사항을 처리할 수 없다. 따라서 본 연구의 추후 연구과제로서 XML Schema의 처리를 W3C의 지침을 기반으로 개선하고 RDF에서 자원간의 관계를 효율적으로 나타낼 수 있는 Node and Arc Diagram 인터페이스를 개발하는 것이다.

참고 문헌

- [1] W3C, Resource Description Framework(RDF), February, 1999, <http://www.w3.org/RDF/>
- [2] RDF Vocabulary Description Language 1.0, April, 2002, <http://www.w3.org/TR/rdf-schema/>
- [3] RDF: Understanding the Striped RDF/XML Syntax, Dan Brickley, October, 2001, <http://www.w3.org/2001/10/stripes/>
- [4] Jeen Broekstra, Michel Klein, and Stefan Decker, Enabling knowledge representation on the Web by extending RDF Schema, In Proceedings of the 10th World Wide Web conference, pg. 467-478, Hong Kong, China, May 1-5, 2001.
- [5] Apache XML Project, August, 2002, <http://xml.apache.org/xerces2-j/>
- [6] Hewlett-Packard, The jena semantic web toolkit, July, 2001, <http://www-uk.hpl.hp.com/people/bwm/rdf/jena/index.htm>.
- [7] Olivier Corby, Rose Dieng, and Cedric Hebert, A Conceptual Graph Model for W3C Resource Description Framework, ICCS2000, Darmstadt, Germany, August 14-18, 2000.