**시맨틱 웹 환경을 위한 RDF 메타데이터 편집기**

**2. 더블린코어, HTML 메타데이터, RDF**

**2.1 더블린 코어**

단순하기 때문에 메타데이터의 제작비용을 줄이고 상호운영성을 높일 수 있는 반면에 복잡한 메타데이터체계에서 지원되는 의미와 기능 측면에서의 풍부함은 제공되지 못하는 더블린 코어의 특징과 구성요소를 간략히 살펴보면 다음과 같다

1) 특징

- 데이터의 형식과 구조를 단순화하여 원문의 저자나 발행자가 메타데이터를 직접 작성할 수

가 있다.

- 트워크 출판을 위한 저작도구의 개발자가 정보에 대한 템플릿을 직접 해당 소프트웨어에

포함할 수 있도록 한다.

- 작성된 데이터를 기초로 특정 분야에서 요구되는 상세한 수준으로 확장하여 사용할 수 있

도록 한다.

2) 구성요소

더블린 코어의 데이터 요소는 정보 저장을 위하여 [표 1]과 같이 분류된다.

[표 1] 더블린 코어의 데이터 요소 분류

|  |  |
| --- | --- |
| 구분 | 더블린코어 데이터 요소 |
| 컨텐츠 관련요소 | Title, Subject, Description, Source, language, Relation, Coverage |
| 지적 속성요소 | Creator, Publisher, Contributor, Rights |
| 물리적 기술요소 | Data, Type, Format, Identifier |

**2.2 HTML 메타데이터**

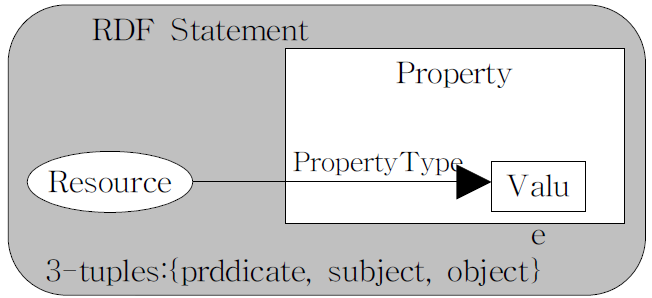
HTML 메타데이터는 HTML 문서안의 <head>부분에 <LINK>와 <META> 요소를 이용하여 기술한다. <LINK> 요소는 문서 외부의 다른 문서와의 관련성을 나타내며 메타데이터 스키마를 참조하는데 사용되고 REL이나 REV를 포함한다. <META> 요소는 HTML 문서에 메타데이터를 포함하는데 사용되며 속성 유형과 속성값을 기술한다[3,4]. HTML 메타데이터를 기술하기 위한 <LINK>와 <META> 태그의 형식은 [그림 1]과 같다.

|  |
| --- |
| <LINK REL = “SCHEMA.PREFIX”  HREF = “LOCATION\_OF\_DEFINITION”>  <META NAME = “PREFIX.ELEMENT\_NAME”  CONTENT = “ELEMENT\_VALUE”> |

[그림 1] HTML의 LINK와 META 태그 형식

**2.3 RDF(Resource Description FrameWork)**

RDF는 구조화된 메타데이터에 대한 부호화, 교환, 그리고 재사용을 가능하게 해주는 하부구조로 의미, 구문, 구조에 대한 공통적인 협약을 지원하는 매커니즘 설계를 통해 상호운영성을 가능하게 하며, 메타데이터 요소를 정의하는 도구를 제공한다. RDF의 데이터 모델은속성과 속성값을 표현하는 모형을 기초로 하고 있으며 자원(Resource), 속성(Properties), 문장(Statement)과 같은 세 개의 오브젝트 타입으로 구성된다. 자원은 RDF 형식으로 표현되는 모든 것을 말하여, 속성은 자원을 기술하기 위한 특정한 측면, 특징, 속성, 관계를 의미한다.  
특정 자원과 지정된 속성, 그리고 그 속성의 값으로 이루어진 자원을 문장이라고 하는데 자원을 의미하는 주어(subject)와 속성 유형을 의미하는 술어(predicate), 속성값을 의미하는 목적어(object)로 구성된다. RDF 데이터 모델에서 노드는 자원을 표현하는 것이고, 아크는 속성을 의미하며 문자열이나 리터럴을 표현한 노드는 직사각형으로 표현되는데 이것은 속성값이 된다. RDF 데이터 모델의 예는 [그림 2]와 같다.



[그림 2] RDF 데이터 모델

**3. RDF Tools**

**3.1 DC-dot**

DC-dot[5]는 더블린 코어의 기본 요소를 사용하며 웹상에서 URL를 입력하면 해당 웹페이지를 검색하여 더블린 코어 요소에 해당되는 메타값을 자동 생성하고 추출되지 않은 요소들은 수동으로 입력하여 XHTML, HTML, RDF, XML로 표현할 수 있다. [그림 3]은 DC-dot에서 “www.semantic.or.kr”이라는 사이트를 입력하여 얻은 RDF 형식의 결과물이다.



[그림 3] DC-dot 의 출력 화면

**3.2 Reggie**

Reggie[6]는 서로 다른 구성요소나 언어를 가진 메타데이터를 쉽게 작성하기 위해서 스키마 파일을 이용하여 메타데이터를 작성할 수 있지만 자바 애플릿 방식으로 실행 속도가 늦고 생성된 구문을 변경하기가 불편하다. 변환형식은 HTML3.2와 HTML4.0, RDF이고 스키마를 더블린 코어 외에 GILS, ANZLIC, AGLS, EdNa, IMS, GEM, vCARD 등으로 변경할 수 있다. [그림 4]는 Reggie에서 메타데이터를 입력하는 화면이다.



[그림 4] Reggie의 메타데이터 입력 화면

**3.3 Dublin Core Metadata Template**

Dublin Core Metadat Template[7]는 Nordic Metadata Project에서 개발하였으며 더블린 코어의 데이터 요소와 한정어를 사용하고 환형식은 HTML과 HTML4.0이며 HTML 플릿 방식이다. 특징은 생성 후 편집이 가능하고 수많은 나라의 언어를 지원한다. [그림 5]는 Dublin Core Metadata Template를 이용하여 생성된 HTML 메타데이터이다.



**[그림 5] HTML4.0 형식의 메타데이터**

이상으로 3종류의 대표적인 메타데이터 생성 Tool들을 살펴보았다. 하지만 이 Tool들은 모두 웹상에서 운영되기에 인터넷이 되지 않으면 사용을 할 수가 없다. 이 점에 착안하여 URL을 입력하면 해당 웹 페이지를 분석하여 메타데이터 요소를 추출할 수도 있고 파일로 저장된 HTML 문서를 Load하여 편집할 수 있는 RDF 메타데이터 편집기를 제작하였고 이를 통해 구문 오류 방지 및 일관성을 유지할수 있도록 하여 상호운영성을 제공하고 시맨틱웹을 위한 최소한의 환경을 이루었다.

**4. 편집기의 설계 및 구현**

**4.1 RDF 메타데이터 편집기 설계**

본 편집기는 크게 소스 분석 과정과 메타데이터 생성 과정으로 나누어진다. 먼저, 웹 사이트의 URL이나 HTML 문서, RDF 문서가 Load되면 더블린 코어의 15가지 요소에 해당되는 속성값을 화면에 표시해 주는데 이 때, HTML의 메타데이터를 RDF 메타데이터 모델로 모델링 하기 위해 웹 문서의 태그들을 분석하여 추출하는 규칙은 다음과 같다[8].

1) URL은 하나의 자원으로 Identifier 요소의 값이 공백일 때 URL이 그대로 입력된다.

2) Title 태그는 데이터 모델의 속성유형으로 요소 내용은 속상값으로 <Title>과 </Title>태그 사이의 내용을 Title 요소의 값으로 표시한다.

3) Format 요소와 Language 요소의 값은 HTML의 Content-Type 메타의 값을 이용하는데 앞부분을 Format 요소의 값으로 추출하고 charset의 값을 Language의 값으로 이용한다.

4) meta 태그는 자원에 대한 정보 및 검색엔진, 에이전트 탐색 시 대상이 되는 태그로 name이나 http-equive 속성은 RDF 데이터 모델의 속성 유형이 되고, content 속성은 속성값이 된다.

두 번째로 RDF 메타 데이터의 생성은 필수적인 선언부를 기술하고 앞의 분석 규칙에 의거 생성하게 되는데 선언부는 RDF 문서가 XML 기반 문서이고 이름 공간(Name Space)를 이용하므로 반드시 표현되어야 한다.더블린 코어의 이름 공간을 이용하므로 다음과 같이 표현되어야 할 것이다.

<?xml version="1.0"?>

<rdf:RDF xmlns:rdf="http://www.w3.org/1999 /02/22-RDF-syntax-ns#" xmlns:dc="http://prul.org/dc/elements/1.1/" >

</rdf:RDF>

선언부를 표시한 뒤 자원을 표시하고 설명하기 위하여 <rdf:Description>과 about 속성을 사용하여 자원을 기술한다.

<rdf:Description about=" URI-reference" >

</rdf:Description>