메타데이터 환경과 과제

이재윤/ 연세대학교 문헌정보학과 강사

한국데이터베이스진흥센터 산하의 한국정보검색위원회에서는 위원간의 연구의욕 고취와 새로운 검색 및 데이터베이스 관련기술 보급을 위해 매월 연구발표회를 개최하고 있다. 본 코너는 매월 발표된 주제논문을 게재함으로써 정보검색과 관련된 정보를 제공하기 위해 마련된 것이다.
〈편집자〉

Ⅰ. 서론

인터넷의 확산과 함께 체계적인 정보유통을 위한 방안으로 메타데이터가 활발히 연구, 제안되고 있다. 메타데이터를 둘러싼 초기의, 가장 큰 문제라고 할 수 있는 형식의 표준화나 호환성은 NISO, ISO 등의 표준 기구에서 핵심 메타데이터의 표준화가 추진되고, 주요 메타데이터 사이의 매핑이나 연결 문제가 차츰 정리되면서 어느 정도 해결 방향이 드러나고 있다.

그러나 메타데이터를 기존의 인쇄자원에 대한 목록과 비교해볼 때, 이를 둘러싼 환경이 아직 수많은 문제와 발전가능성을 가지고 있음을 짐작할 수 있다. 자원 식별기호의 문제, MARC 형식과의 관계, 인쇄 자원에 대한 기술 문제, 자원 보존 문제, 메타데이터 저작 도구의 문제 등은 메타데이터를 둘러싼 환경으로서 각 분야에서 나름대로의 대안이 연구되고 있다.

이 글에서는 이와 같이 메타데이터를 둘러싼 환경과 그에 관한 몇 가지 문제에 대해서 현재 제안된 방안과 향후 과제를 살펴보기로 한다.

메타데이터와 식별기호

인쇄 출판물에 대하여 ISBN이나 ISSN과 같은 고유 기호가 부여되듯이 인터넷 자원의 경우에도 고유 식별기호가 필요하다. 출판물의 경우에 식별기호는 상이한 자원을 구별해주는 것을 핵심 역할로 한다. 그러나 인터넷 자원의 경우에는 식별기호가 제대로 기능하려면 서로 상이한 자원을 구분해줌과 동시에 특정 자원의 물리적 위치를 지정해주어야 한다. 현재 인터넷 자원의 식별기호로 널리 쓰이는 Uniform Resource Locator(URL)는 일시적인 소장 위치에 관한 정보만 제공하므로 지속적인 접근성이 보장되지 않으며 미러 사이트와 같이 상이한 URL로 표시되는 동일한 자원에 대해서는 아무런 제어가 이루어지지 않고 있다. 이와 같은 URL의 한계를 극복하기 위해서 Internet Engineering Task Force(IETF)에서는 Uniform Resource Identifier체계를 개발해오고 있다. URI체계 개발 과정은 다음과 같다(W3C).

1998년 8월에 제안된 RFC 2396은 이전에 제안된 RFC 1738과 RFC 1808의 개정판으로서 인터넷 상의 추상적, 물리적 자원을 식별하기 위한 간결한 문자열 체계를 제안하고 있다. URI 체계에는 식별기호 역할을 하는 Uniform Resource Name(URN)과 메타데이터 역할을 하는 Uniform Resource Characteristics (URC), 실제 자원의 접근 메카니즘을 지정하는 Uniform Resource Locator(URL)가 포함된다. 여기서는 URN과 그 구현 사례를 살펴보도록 한다.

〈표 1〉 URI 체계 개발 과정

시 기	URI Working Group 활동
1994년 3월	URI Working Group 결성
1994년 6월	RFC 1630 - Universal Resource Identifiers in WWW
1994년 12월	RFC 1737 - Functional Requirements for Uniform Resource Names RFC 1738 - Uniform Resource Locators (URL)
1995년 6월	RFC 1808 - Relative Uniform Resource Locators
1995년 7월	URI Working Group 폐쇄 - 세부 표준 단위로 확대 개편 예정
1998년 8월	RFC 2396 - Uniform Resource Identifiers (URI): Generic Syntax

URN

URN에 관해서는 현재 다음과 같은 RFC 문서가 제안되었다.

- RFC 1737 - Functional Requirements for Uniform Resource Names
- RFC 2141 - URN Syntax
- RFC 2168 - Resolution of Uniform Resource Identifiers using the Domain Name System
- RFC 2169 - A Trivial Convention for using HTTP in URN Resolution
- RFC 2276 - Architectural Principles of Uniform Resource Name Resolution

RFC 1737에 제시된 URN의 필수 조건을 살펴보면 다음과 같이 기능적 측면과 인코딩 측면으로 나누고 있다.

RFC 2141에 따르면 URN은 영구적이고 위치독립적인 자원 식별기호로 사용되는 문자열로서 다음과 같은 구조를 가진다.

〈URN〉 : : = "urn :" 〈NID〉 ":" 〈NSS〉

여기서 〈NID〉는 namespace 식별기호이고, 〈NSS〉는 namespace마다 고유한 문자열로서 실제 자원을 지칭한다. URN의 예를 들면 다음과 같다.

〈urn : x-dns-2 : physics.bigstate.edu : thesis12〉

〈urn : x-wink : physics.bigstate.edu : thesis12〉

urn : isbn : 0-262-12186-7

URN을 URL로 변환하는 과정은 〈그림 1〉과 같다.

NID 레지스트리는 기명 체계를 실제 변환 시스템 URS로 연결해주는 중계 시스템으로서 각 기명 전거 시스템에 대한 정보가 등록되어 있는 네트워크 상의 분산 디렉토리 시스템이다.

이와 같은 URN 체계는 아직 구체적으로 개발되지는 못했으며, 그 대신 영구적인 식별기호의 필요성을 인식한 도서관, 출판 등의 분야에서 URN 모형에 따라 몇 가지 식별기호 체계를 개발한 사례가 있다. 다음에 이런 사례인 PURL, Handle 시스템, DOI에 대해서 살펴본다.

PURL(Persistent URL)

PURL은 OCLC가 개발한 것으로서 URN의 표준 정착에 대비한 임시 체계이다. URN의 복잡한 기명 체계를 채택하는 대신 PURL서버라는 일종의 프록시 서버를 이용하여 영구 식별기호 체계를 구현하였다. PURL 시스템의 식별 체계는 〈그림 2〉와 같다.

PURL의 주소체계는 〈그림 3〉과 같이 변환기 역할을 하는 PURL서버의 주소와 자원 식별명으로 구성되어 있다.

PURL 서버는 OCLC의 InterCat 프로젝트, 호주국립도서관, 덴마크의 INDOREG프로젝트 등에서 구축한 사례가 있다.

Handle 시스템

Handle 시스템은 LC의 CNRI에서 디지털 도서관의 하부구조로 개발한 것으로서 디지털 객체의 주소체계를 지향한 것이다. 이 시스템은 IETF의 URN 모형을 그대로 적용하였으며 〈그림 4〉와 같은 체계로 운용된다. Handle 시스템을 적용한 사례는 LC의 National Digital Library Program, NCSTRL 등이 있다. 다음에 다룰 DOI도 Handle 시스템을 이용에서 출발한 식별기호 체계이다.

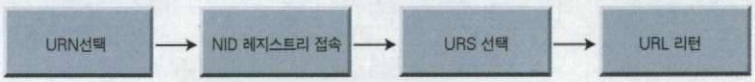
DOI(Digital Object Identifier)

DOI는 미국출판자협회(AAP)가 인터넷 출판물의 전자상거래와 저작권 관리를 위해 개발한 것으로서 유명 출판사와 출판 관련 단체가 협력하여 1997년 International DOI Foundation을 설립하고 활발히 보급하고 있다. 〈http : //www.doi.org〉

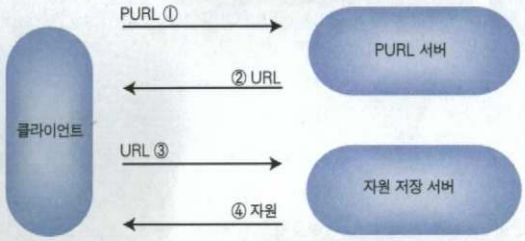
DOI는 초기에 Handle 시스템을 도입한 식별기호 체계로 출발했으나 점차 메타데이터까지 포함하는 개념으로 발전하고 있으며 Dublin Core에 기반한 메타데이터를 수용할 예정이다. DOI의 주소체계는 〈그림 5〉와 같이 prefix와 suffix로 구성된다.

〈표 2〉 URN의 필수 요건

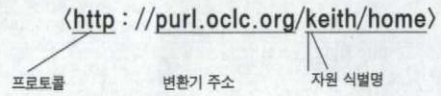
기능적 측면의 필수조건	포괄성	전체 네트워크에서 동일한 의미로 통용
	유일성	URN 중복 할당 금지
	영속성	영속적으로 사용 가능
	할당 가능성	네트워크의 어느 자원에나 할당 가능
	호환성	URN의 필수 조건을 만족하는 기존 기명 시스템과 호환
	확장성	미래의 표준 기명 체계로 통합 가능
	독립성	각 namespace마다 독립적으로 URN 관리
	변환성	URN을 URL로 해석할 수 있는 메카니즘
인코딩 측면의 필수조건	동일 인코딩	변환된 URL 전송시 WWW과 동일한 인코딩 방식 채택
	간단한 알고리즘	적합한 URN을 찾기 위한 비교 알고리즘이 간단하여 외부 서버에 접속할 필요가 없을 것
	가독성	사람이 해석할 수 있도록 대소문자 구별, 특수 문자 사용 배제 등으로 간결한 형식 체택
	전송 친화성	기존의 인터넷 프로토콜인 TCP, SMTP, FTP, Telnet 등에서 전송 가능
	기계 처리	컴퓨터로 구문 분석 처리
	텍스트 인식	본문에서 URN을 탐색, 구문 분석



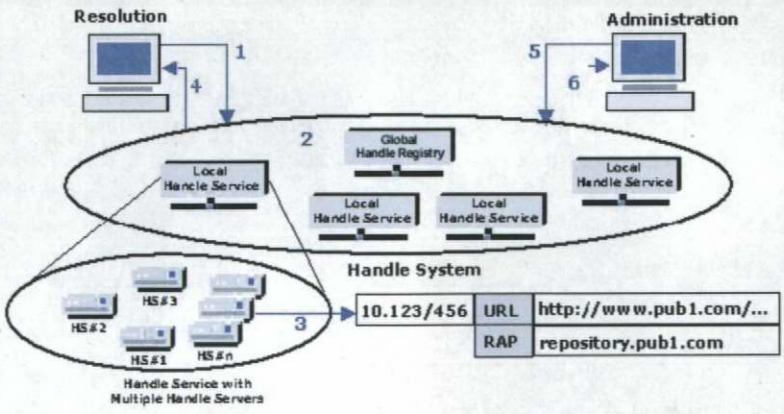
〈그림 1〉 URN -> URL 변환 과정



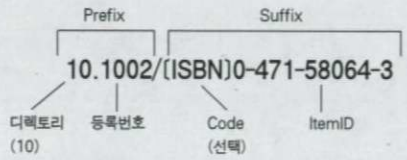
〈그림 2〉 PURL 시스템의 식별 체계



〈그림 3〉 PURL의 주소 체계



〈그림 4〉 Handle 시스템의 운용체계



〈그림 5〉 DOI의 주소 체계

여기서 디렉토리 번호는 해당 레코드를 관리하고 있는 곳이고 등록 번호는 개별 출판사의 번호이며, Code는 BICI, ISSN, ISAN, ISBN, SICI등과 같은 국제표준번호 체계이고 자료 ID는 최대 128자의 문자 및 숫자로 구성하고 있다. 식별 대상인 자료는 특정 호, 기사, 초록, 도표 등까지 검토하여 융통성있는 단위를 채택할 예정이다.

DOI를 이용해 자료에 접근하는 절차는 〈그림 6〉과 같다.

DOI는 전자출판물의 인터넷 상거래를 위해 개발된 만큼 주요 출판사에서 적극적으로 구현하여 보급하고 있다. DOI의 구현사례는 DOI 요청에 대응하여 제공하는 수준에 따라서 세 가지로 나눌 수 있다.

자료의 원문에 접근하는 링크를 제공하는 Wiley DOI Server, 초록을 제공하는 Elsevier나 Springer-Verlag, 원문 및 관련 정보까지 제공하여 완전한 전자상거래 수준에 다다른 DOI Gallery (Academic Press와 John Wiley & Sons) 등이 대표적인 사례이다.

메타데이터와 MARC

각종 메타데이터 형식 중에서 가장 역사가 오래되고 널리 쓰이고 있는 것은 서지 데이터베이스에 쓰이고 있는 MARC 형식이다. 다양한 메타데이터의 연결 문제와 기존에 널리 쓰이던 MARC 형식과의 호환문제가 큰 과제로 대두되면서 MARC형식을 통합 메타데이터로 쓸 지 여부에 대한 상반된 주장이 나타나고 있다.

여러 연구에서 제시된 MARC 형식으로의 통합 근거는 다음과 같다.

- MARC에는 모든 유형의 자료를 수용할 수 있다.
- 다양한 접근점을 제공한다.
- 상이한 형식을 유지하는 것은 비경제적이다.
- MARC는 표준 커뮤니케이션 형식이다.
- 다양한 수준의 정보를 융통성있게 기술할 수 있다.
- 입수여부를 판단하기 위한 완전한 기술정보를 제공한다.
- 모든 이용자들이 컴퓨터와 인터넷계정을 가지는 것은 아니다

특히 미국 도서관협회 산하 ALCTS의 목록위원회는 1998년에 작성한 보고서에서 목록 규칙에 따라 작성된 메타데이터가 아니면 서지 데이터베이스에 수용하지 말 것을 권고하면서 다음과 같은 결론을 내리고 있다. (ALCTS Committee on Cataloging : Description and Access 1998)

- 메타데이터는 목록의 대체수단이 아니나 목록작성자에게 유용한 정보원이다.
- 대부분의 메타데이터는 탐색을 위한 도구이지 기술을 위한 도구가 아니다.
- 이름이나 주제전거 제어를 하지 않기 때문에 도서관목록에 메타데이터를 직접 사용하는 것은 부적절하다.
- MARC로 변환이 가능한 메타데이터라 하더라도 일관성이 부족하여 표준화에 적합하지 않다.
- 목록규칙에 기반하지 않은 메타데이터는 검증받아야 한다.
- 메타데이터의 등록과 유지관리를 위해서는 책임있는 기관의 지원이 있어야 한다.

한편 김태수(1998)에서는 다음과 같은 이유로 MARC 대체형식이 필요하다고 주장하고 있다.

- 인터넷 자원이 급증하여 현재의 목록방식으로는 수용하기가 어렵다.
- 새로운 목록방식과 지원도구가 필요하다.
- 궁극적으로는 자원의 기술방식은 최대한 간편하게 자동화하되 그대신 탐색방식을 지능화하는 지식베이스 기반의 접근도구가 필요하다.

MARC와 새롭게 대두되고 있는 메타데이터와의 관계는 여전히 큰 과제가 되고 있다. 실제로 최근 Dublin Core 개발 진영에서 MARC와의 매핑 문제 및 구분의 어려움 때문에 주요 15개 요소 중에서 Creator와 Contributor, Publisher의 세 요소를 합치거나 재검토하자는 주장까지 제기된 바 있다.

현재까지 제기되고 있는 주장은 MARC로의 통합, MARC의 대체 형식 개발, 새로운 메타데이터로의 통합, MARC와 새로운 메타데이터의 공존 등 다양한 양상을 보이고 있다.

인쇄 출판환경에 비유해보면 인터넷 메타데이터는 원 자료 생산자가 주로 작성한다는 측면에서 CIP 레코드나 판권지 정보와 유사하다고 볼 수 있다. 완전한 목록데이터 수준에 이르기에는 인터넷 메타데이터 형식이 부족하며, 저자가 작성하는 정보 이상의 것이 필요한 것은 분명하다.

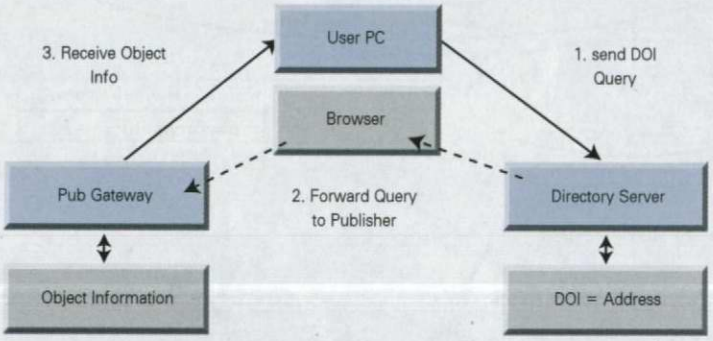
다만, 정보 생산자와 이용자가 뚜렷하게 구분되지 않으며 유통환경이 인터넷은 인쇄출판계와 상당히 다르다는 측면에서 새로운 관계 정립을 모색해야 할 것이다.

메타데이터와 인쇄 자원

비록 메타데이터가 데이터에 관한 데이터라는 느슨한 정의로 인식되고 있긴 하지만, Dublin Core가 '인터넷 자원의 간단한 발견'을 기본 목표로 하고 있는 것에서 알 수 있듯이 메타데이터의 적용 대상은 아직까지 인터넷 자원에 한하고 있는 것이 보편적인 인식이다. 그러나 최근 메타데이터의 적용이 점차 활발해지면서 인쇄자원에까지 MARC가 아닌 메타데이터 형식을 적용하려는 시도가 나타나고 있다.

호주판 GILS 프로젝트라고 할 수 있는 Australian Government Locator Service (AGLS) 프로젝트에서는 DC를 확장한 형식을 사용하고 있으면서, 최근에 전자 자원과 인쇄 자원을 동일한 AGLS 메타데이터로 기술하려고 시도하고 있다.

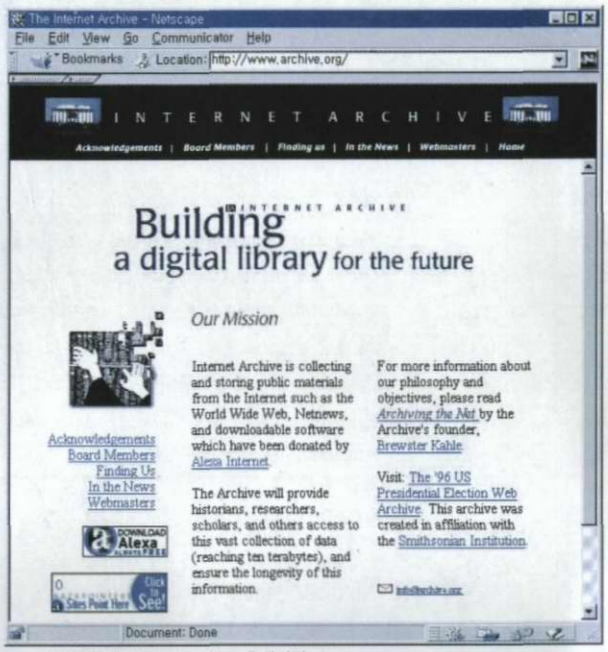
여기서는 기존의 요소에 availability 요소를 추가하여 인쇄 자원을 포함할 예정이다 (Australian Government Locator Service Working Group. 1997. Australian Government Locator Service Implementation Plan. 〈http : //www.aa.gov.au/AA_WWW/



〈그림 6〉 DOI를 이용한 자원 접근 절차

AGLSfinal.html)).

한편 국내에서도 첨단학술정보센터(KRIC)가 1998년 하반기에 진행중인 연구 프로젝트인 '메타데이타를 이용한 도서 종합목록 구축 연구'에서는 네트워크 자원을 포함한 모든 유형의 자료를 대상으로 하면서, MARC를 탈피하고 종합목록용 메타데이터를 새로 제정할 것을 검토하고 있다. 이 연구에서는 새로 제안된 메타데이터와 기존 MARC DB와의 연계방안까지 연구할 계획이다(첨단학술정보센터 1998).



〈그림 7〉 Internet Archive 홈페이지

비록 ALA와 같은 도서관계에서는 메타데이터를 인쇄 매체에 적용하는 것을 거부함은 물론 인터넷 자원에 대해서도 목록 작성은 MARC를 적용하는 것을 권장하고 있는 실정이지만, 인터넷 자원에 대해 메타데이터를 이용하는 측에서 편이성과 일관성 문제로 인쇄자원도 동일하게 다루고자 하는 경향은 점차 확산될 것으로 보인다.

〈표 3〉 인터넷 자원 보존 프로젝트 주요 사례

프로젝트	국 가	주 체	시 기	수집방법	수집범위
Kulturarw3	스웨덴	왕립도서관	1996.9~	로봇을 이용한 자동 수집	스웨덴의 모든 웹사이트
	전자출판물 장서의 기초, 능동적 자료 수집				
INDOREG	덴마크	덴마크도서관센터	1996~	수집. 후에 출판사의 직접 등록으로 바뀔 계획.	몇 가지사항을 제외한 모든 웹사이트
	국가서지등록 시스템 구축. 전자문헌등록과 전자문헌 제어 중심.				
EVA	핀란드	헬싱키대학도서관	1997.1 ~1998.6	로봇에 의한 자동 수집	모든 웹사이트
	문헌관리를 위한 기술적 문제 고려(파일 포맷 등)				
PANDORA	호주	국가도서관	1997~	SCOAP 지침에 따른 수집	강력한 선정기준에 따른 전자출판물
	전자 보존소 구축. 선정과 접근 중심의 프로젝트.				
NEDLIB	EU	유럽 8개국 국가도서관과 출판사	1998.1~ 2000.12	검토 중	검토 중
	전자출판물 보존 시스템을 위한 국가간 협력, 표준 정의				
CEDARS	영국	eLib의 일부	1998.4 ~2000.4	DESIRE 선정원칙에 따른 수집	선정기준에 부합한 전자출판물
	eLib의 3단계 프로젝트. eLib, JISC의 다른 프로젝트와 연결.				

메타데이터와 인터넷 자원 보존

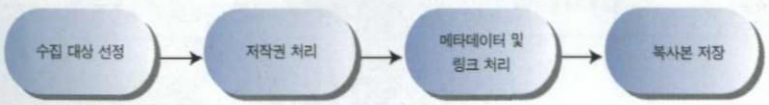
많은 인터넷 자원은 시간이 지나면 사라지는 휘발성이 있기 때문에 인터넷 자원의 보존 및 장기적 이용이 큰 과제로 대두되고 있다. 이런 문제를 해결하려면 자원의 고유 식별기호와 메타데이터를 표준화하고 이를 등록하여 이용시킬 수 있는 체계를 개발해야 한다.

1996년 4월에 WAIS의 개발자인 Brewster Kahle이 동료와 함께 설립한 Internet Archive는 인터넷상의 공공자료를 수집하여 보존하고 역사가나 연구자, 학자 등에게 장기적으로 이용시키는 디지털도서관을 표방하고 있다.

그러나 이곳은 어디까지나 미국 중심이기 때문에 전 세계를 대상으로 하지는 않는다. 한편, 미국의 LC는 최근 Alexa Internet으로부터 2테라바이트에 달하는 분량의 1997년 1,2월치 웹 저장 자료를 기증받아서 관련 사업을 준비하고 있다.

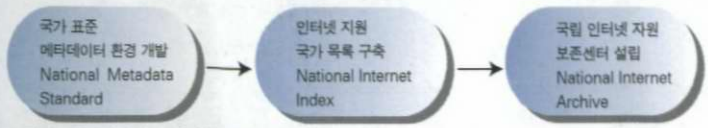
미국 이외의 다른 나라에서는 자국의 자원에 대한 책임을 지는 프로젝트가 제각기 시도되고 있다. 주요 사례는 〈표 3〉과 같다(이재윤, 조현주 1998, 38).

인터넷 자원의 보존 및 장기적 이용을 위해서 〈그림 8〉과 같은 처리 과정을 거쳐야 한다.



〈그림 8〉 인터넷 자원의 보존 및 장기적 이용을 위한 처리 단계

이와 같은 처리 과정이 원활히 이루어지기 위해서는 메타데이터를 이용하는 국가적인 표준 환경의 개발이 전제되어야 하며, 그 다음 국가 차원의 인터넷 자원 목록 시스템이 구축되어 메타데이터의 이용이 확산되어야 한다. 그 이후에야 비로소 인터넷 자원 보존센터의 설립이 〈그림 9〉와 같이 이루어질 수 있을 것이다.



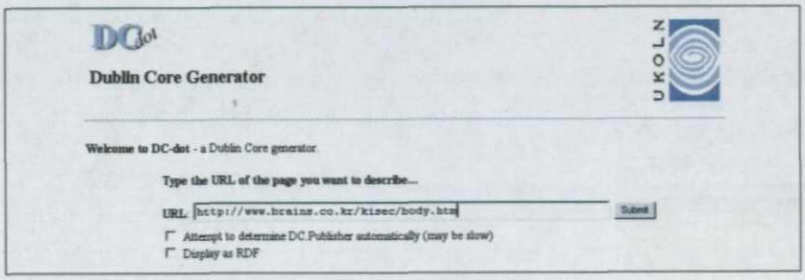
〈그림 9〉 인터넷 자원 보존 센터의 추진 단계

국가 표준 메타데이터 환경이 구축되기 위해서는 표준 메타데이터 형식을 지원하는 메타데이터 저작 도구의 개발 및 보급이 필수적이다.

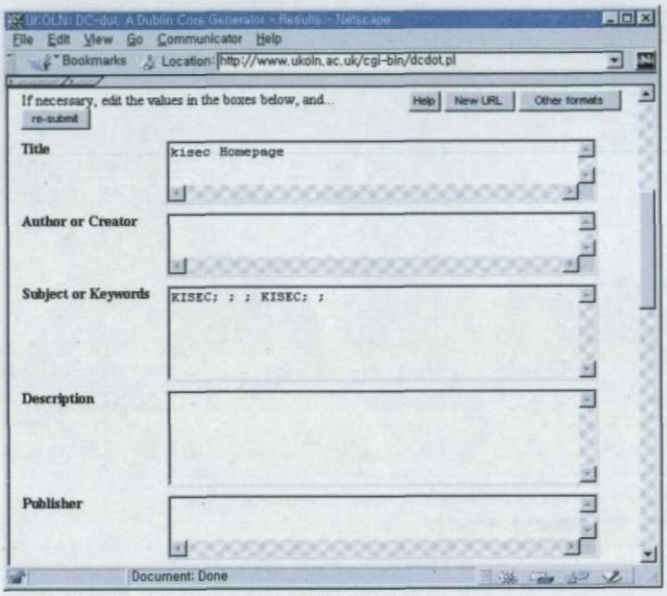
메타데이터 저작 도구

메타데이터의 표준 저작도구가 개발되면 다음과 같은 이점이 있다.

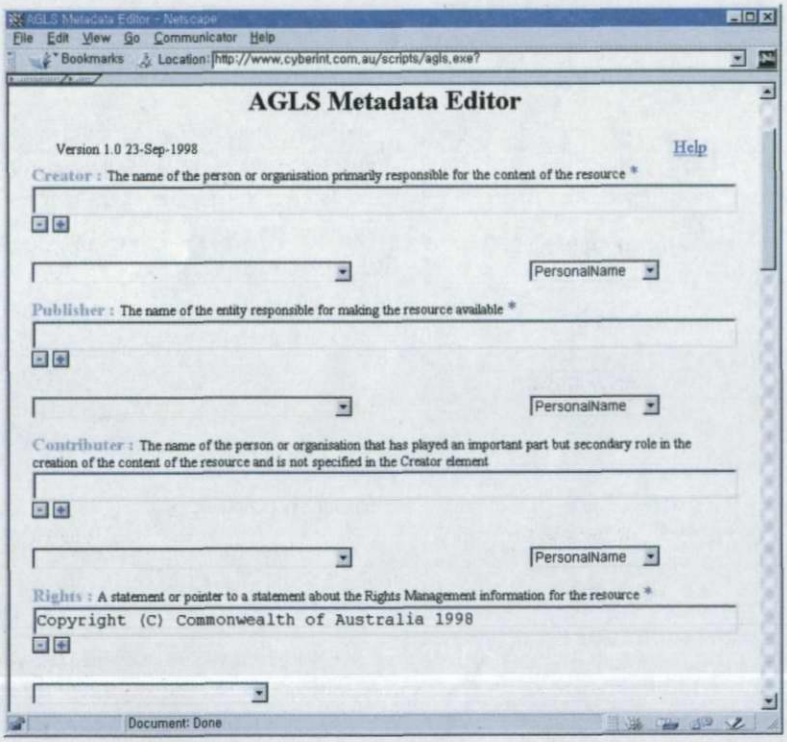
- 저작자의 간편한 메타데이터 생성 지원 - 메타데이터는 자원의 저작자가 작성하는 목록정보인 만큼 메타데이터 형식에 대한 지식이 없더라도 필요한 요소를 간단히 입력할 수 있다.



〈그림 10〉 DC.dot Dublin Core Editor의 URL 입력 화면



〈그림 11〉 DC.dot Dublin Core Editor의 추가 편집화면



〈그림 12〉 AGLS Metadata Editor

- 메타데이터 구문의 일관성 유지 - SGML이나 HTML 또는 XML로 메타데이터 구문을 자동 생성함으로써 구문오류를 방지하고 일관성을 유지한다.
- 메타데이터 수집의 용이성 - 표준 저작도구가 DMA나 WebDAV프로토콜과 같은 표준 저작 프로토콜을 이용하게 되면 메타데이터 생성 단계에서 직접 메타데이터를 중앙 데이터베이스에 등록할 수 있다.

Dublin Core를 비롯한 여러 메타데이터 형식에 대한 저작도구가 이미 개발되어 일부는 사용중이다. 개발된 저작도구는 웹브라우저로 사용하는 자바 애플릿이나 HTML 템플릿 방식이 많으며 별도 프로그램 형식으로 개발된 것도 있다. 다음에 그 사례 중 일부를 살펴보기로 한다. 메타데이터 관련 도구에 관한 정보는 Dublin Core Metadata Initiative에서 찾아볼 수 있다.

HTML 템플릿 방식

Nordic Metadata Creation Tool

- 〈http : //www.lub.lu.se/cgi-bin/nmdc.pl〉
- HTML 템플릿 방식
- Nordic 메타데이터 프로젝트에서 개발, 이후 여러 국가의 프로젝트에서 변형하여 사용함
- HTML 4.0과 HTML 2.0/3.2 구문 지원
- DC 형식만 지원
- 7개 요소만 지원하는 간략 템플릿도 제공

DC.dot Dublin Core Generator

- 〈http : //www.ukoln.ac.uk/metadata/dcdot/〉
- HTML 템플릿 방식
- URL을 지정하면 일부 요소를 자동 생성
- 생성 예 : 〈그림 10〉과 같은 입력창에 처리 대상 웹페이지의 URL을 입력하고 생성 구문을 HTML로 지정하면 다음과 같은 메타데이터가 자동으로 생성된다.

〈META NAME="DC.Title" CONTENT="kisec Homepage"〉
〈META NAME="DC.Subject" CONTENT="KISEC; ; ; KISEC; ;"〉
〈META NAME="DC.Date" CONTENT="1997-06-20"〉
〈META NAME="DC.Type" CONTENT="Text"〉
〈META NAME="DC.Format" CONTENT="text/html - 4296 bytes"〉
〈META NAME="DC.Identifier" CONTENT="http : //www.brains.co.kr/kisec/body.htm"〉

자동으로 생성되는 메타데이터는 15개 요소 중에서 5~6가지에 불과하므로 keyword를 비롯