

# 링킹시스템(Linking Systems)이란 무엇인가?



심 경

shim@irisnet.co.kr

정보학박사

한국도서관협회 평생회원

(주)아이리스넷 대표

**우리**가 자료를 찾을 때<sup>1)</sup> 색인, 초록지를 먼저 검색하고 그 검색결과를 가지고 다시 우리 도서관이 소장한 장서에서 원문을 복사하거나 외부기관에 DDS 신청을 하는 것이 일반적이었다. 이와 같은 절차는 시스템이 전산화되고 전자저널이 출현해서도 크게 변하지는 않았다. 그러나 도서관이용자가 데이터베이스에서 서지나 기사색인(article citations)을 검색하고 다시 다른 시스템을 거치지 않고 이미 찾아낸 개별 메타데이터<sup>2)</sup>에서 직접 자신이 필요로 하는 전자저널의 원문을 볼 수 있는 방법은 없을까? 또한 전자저널에 수록된 학술논문의 말미에 있는 참고문헌들(references) 가운데 원문을 보고 싶은 것이 있을 때 바로 그 화면에서 접근하는 방법은 없을까? 특히 근래 인터넷에서 발견되는 자료들은 상당 부분 하이퍼링크로 직접 연계가 가능한데 유독 도서관시스템에서는 이를 지원하지 않으니, 한번쯤은 생각해 볼만한 시나리오이다.

OpenURL기반의 링킹시스템<sup>3)</sup>이 바로 이와 같은 시나리오를 현실

1) 아는 자료 검색(known-item search) 과 반하여 특정 주제에 대한 검색(subject search)을 말한다

2) 이 글에서는 서지정보와 기사색인정보를 포괄하는 의미로 사용되었다.

3) 엄밀히 말하여 링킹시스템은 OpenURL을 전제로 하지 않는 다양한 종류가 있으나, 이 글에서는 별도 설명이 없는 한 링킹시스템은 OpenURL을 기반으로 한 시스템을 의미한다.

로 구현한 것이다. 링킹시스템은 검색된 개별 메타데이터나 참고문헌 리스트의 개별 항목 옆에 링크 버튼(link button)<sup>4)</sup>을 생성하여 단 한번의 클릭으로 원문에 연계하는 시스템이다(그림 2, 3 참조). 그래서 이것을 “기사색인 링킹(citation linking)” 또는 “참고문헌 링킹(reference linking)”이라고도 부른다. OpenURL 기반 링킹시스템은 또한 이용자로 하여금 오직 접근권한이 있는 원문으로만 연계되게 하므로 이용자가 접근을 거절 당하거나 복잡한 라이선싱 정보에 대해서 이해할 필요가 없다. 따라서 이를 흔히 “적합한 카피(appropriate copy)”로의 “맥락 감지 링킹(context-sensitive linking)”이라고 한다.

링킹시스템은 도대체 어떻게 생겼고, 무슨 원리로 그런 것이 가능할까? 그리고 시연을 본 경험이 있는 독자는 “시연에서 본 것이 ‘모든 경우’에 자동으로 그렇게 되는가?” 등의 의문이 생길 것이다. 이 글에서는 일단 링킹시스템의 원리 및 구조와 문제점을 살펴본다.

## 링킹시스템은 어떻게 작동되며 어떤 기능을 제공하나?

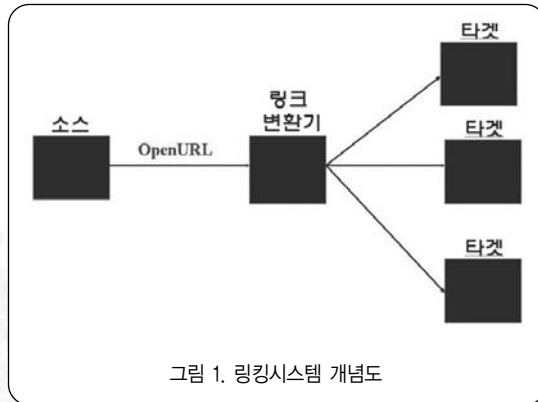
링킹시스템의 핵심은 링크변환기(link resolver 또는 link server)로서(그림 1 참조) 우리가 링킹시스템을 구매했을 때 제공되는 대상이다. 링크변환기는 기본적으로 링킹엔진(linking engine)과 지식베이스(knowledgebase)라는 두 가지 요소를 가진다. 링킹엔진의 역할은 소스에서 전송되는 OpenURL을 분석하여<sup>5)</sup> 에러를 점검하고 제목이나 ISSN과 같은 주요 정보의 누락여부를 점검하여 보완하는 전처리(preprocessing) 과정을 수행한다. 또한 이 과정에서 재구성된 메타데이터 요소를 활용하여 모든 타깃(target)으로 연계해 주는 역할을 한다.

이와 같이 소스에서 전송된 OpenURL에 포함된 메타데이터 요소를 활용하여 “적합한 카피”로 연계하기 위하여는, 즉 지난 호에서 설명한 바와 같이 “무엇(OpenURL)”을 “어디(URL)”라는 정보로 변환하기 위하여는 “중간매체”가 필요하다. 일상생활에 비유하자면 이름으로 전화번호나 주소를 찾는 전화번호부 같은 것을 말한다. 이 중간매체는 해당 도서관이 제공할 수 있는 “모든” 전자자원에 대한 정보를 가진 포괄적 참조 데이터베이스(comprehensive reference database)여야 하는데 이것이 바로 링크변환기의 두 번째 요소인 지식베이스이다. 지식베이스는 크게 “글로벌 지식베이스(global knowledgebase)”와 “로컬 지식베이스(local knowledgebase)”로 구성된다. 글로벌 지식베이스는 데이터베이스 공급자들이 제공하는 전자저널 타이틀, 그와 관련된 벤더, 수록기간 등의 정보를 가지고 있다. 이러한 글로벌 지식베이스에 저장되는

4) 링크 버튼이란 그림 2와 3의 좌측 화면에 나타난 ‘Open Resolver’ 라는 버튼과 같이 링킹시스템을 사용 시 메타데이터 옆에 생성되는 버튼이다.

5) 지난 호에서 OpenURL은 단순히 자원의 위치를 전달하는 것이 아니라 메타데이터 요소를 포함한다고 설명하였다.

정보를 개별 도서관에서 직접 추적하고 갱신하는 것은 엄청난 시간과 노력이 소요되므로 이 정보를 얼마나 정확히 폭넓게 제공하는가가 특정 링크 변환기 벤더의 우세를 결정한다.



로컬 지식베이스는 일단 링킹시스템을 구매하면 글로벌 지식베이스에 마련된 링크 가능성 중 자관이 구독하는(institution-specific) 서비스로의 링크만을 제공하도록 설정해 주는 작업을 수행하여 그에 따라 설정된 글로벌 지식베이스의 하위세트를 의미한다. 개별 도서관에 설치된 링킹엔진이 소스에서 전송된 OpenURL에 포함된 메타데이터 요소를 활용하여 적합한 카피(타겟)로 이용자를 연계해 주기 위하여 검색하는 대상이 바로 로컬 지식베이스이다.

원래 링킹시스템의 고전적 개념은 메타데이터에서 그 데이터가 표현하는 자료의 원문 카피 중 접근권한이 있는 카피로 연계해 주는 역할을 하는데 왜 그림 1에는 타겟이 여러 개가 있는 것일까? 이는 링킹시스템의 고전적 개념을 확장한 것으로 ‘확장 서비스 링킹(extended-services linking)’이라고 한다<sup>6)</sup>. 이 확장개념을 적용하여 일반적으로 링킹시스템은 이용자가 소스에서 링크 버튼을 클릭하면 직접 원문으로 연계하기 보다는 “인터미디어리 메뉴(intermediary menu)”를 제공하여 선택하도록 하는 방식을 취하므로 복수 타겟으로 제시된 것이다(그림 2와 3의 우측 화면 참조). 하지만 이 인터미디어리 메뉴를 사용하지 않고 직접 원문에 연계하는 방법도 있는데 이를 “원문 불러오기(invoking the full text)”라고 하며, 이러한 기능의 제공여부도 링킹시스템을 구매 시 고려할 사항이다.

인터미디어리 메뉴에 포함되는 요소는 주로 소스에서 전송된 서지 또는 기사색인정보, 이용자가 메타데이터를 수정할 수 있는 방법<sup>7)</sup>, 적합한 원문 링크들, 자관 OPAC 링크, 도서관상호

6) 이 개념도 역시 van de Sompel과 그의 동료에 의하여 1999년에 제시되었다

7) 이는 소스에서 전송된 원래 메타데이터 정보가 잘못되었거나 정확하지 않을 때 이용자가 정보를 보완하도록 하는 기능이며, 결국 이 과정은 링크변환기에 의하여 OpenURL로 변환되며 타겟으로 보내지는 정보를 수정해 주는 것이다.



## 소스에서 링크 버튼과 OpenURL은 어떤 방법으로 생성되나?

앞에서 링킹시스템의 기능을 설명하였으나, 짚고 넘어가야 할 부분이 있다. 링킹시스템을 구매한다는 것은 그림 1에 제시된 링크변환기를 구매하는 것과 동일하다고 하였는데, 그렇다면 어떻게 그림 2와 3에 보인 것과 같이 링크변환기와는 다른 업체의 제품인 데이터베이스 검색시스템에서 동적으로 디스플레이되는 검색결과 화면에 링크 버튼을 생성하고 그 버튼의 클릭으로 OpenURL이 생성되어 링크변환기로 전달되는 것일까? 즉, 링크변환기의 설치가 소스에서 일어나는 모든 변화를 제어하는가 하는 궁금증이 생긴다.

이에 대한 답변은 링킹시스템 자체보다 그 기법을 구성하는 주요 요소가 무엇인가를 살펴봄으로서 찾을 수 있다. OpenURL기반 링킹 기법을 구성하는 요소는 저작(works)을 표현하는 식별자(identifiers), 메타데이터로부터 식별자를 생성하는 방법(mechanism), 그리고 한 식별자를 가지고 특정 카피(item)를 찾는 방법의 세 가지로 구성된다고 할 수 있다. 첫째 요소는 저작을 표현하는 식별자인 OpenURL로서 특정 카피의 위치를 미리 고정적(statically)으로 표현하지 않고 상황에 따라 유동적(dynamically)으로 자원의 적합한 카피들을 “검색”하여 제공할 수 있게 해 주는 구문이며, 이는 링킹시스템 외부에서 정의된 하나의 표준이다. 세 번째 요소인 식별자로 특정 카피를 찾는 방법은 링크변환기에서 일어나는 작업이라는 것을 이미 “링킹시스템의 구조”에서 설명하였다. 두 번째 요소인 식별자 생성방법은 OpenURL이라는 구조를 준수하여 개별 메타데이터마다 OpenURL을 생성하는 방법을 말한다. 바로 이 요소가 우리가 궁금해하는 검색결과 화면에 나열된 메타데이터 옆에 링크 버튼을 생성하고 그를 클릭하면 OpenURL을 생성하여 링크변환기에 전송하도록 하는 기능을 누가 어떻게 만드는데에 해당되는 것이다.

그 답을 그림 2에서 찾아보자.

그림 2의 좌측 화면은 Biological Abstracts에서 이용자가 한 논문정보를 검색한 결과이다. 화면 하단에 “Open Resolver” 라는 버튼이 있음을 주목하라. 이 화면은 데이터베이스 제공자의 시스템 영역이며 이곳에서 생성된 링크 버튼이나 OpenURL의 생성은 링킹시스템(즉, 링크변환기나 그 벤더) 자체와는 관련이 없다. 이는 소스 즉, BIOSIS 측에서 해당 메타데이터 옆에 버튼을 생성하고 그 메타데이터 요소를 OpenURL 표준에 따라 생성하여 노낼 준비를 해야 하는 것이다. 이와 같이 OpenURL을 지원하도록 준비된 소스를 흔히 “OpenURL 사용가능(enabled)”, “OpenURL 지원(aware)”, 또는 “OpenURL 준수(compliance)” 자원이라고 칭한다. 즉, 특정 데이터베이스 검색결과 화면에서 링크 버튼과 그 속에 내장된 OpenURL의 생성은 우리가 구매하는 링크변환기나 그 벤더의 역할이 아닌 해당 화면의 생

산자에게 달려있다<sup>8)</sup>. 물론 링크 버튼은 링크변환기 구매 후 설정하도록 되어있어 그 이전에는 보이지 않으며 활성화시킬 때 아이콘의 선정은 도서관이 결정한다. 이와 같이 소스가 OpenURL 사용가능 자원이 되도록 준비하는 것은 상당 규모의 업무를 필요로 한다.

그렇다면 데이터베이스를 제공하는 출판사나 벤더들이 이러한 비용이 발생하는 수고를 하는 이유는 무엇일까? 이는 오늘날 도서관 이용자들이 OpenURL 기술은 필수적인 기능이라고 생각하며 데이터베이스 벤더들은 이러한 시장요구에 부응해야 자신들의 데이터베이스가 편리한 OpenURL 링크를 통하여 보다 많이 그리고 자주 이용됨으로써 제품가치를 높일 수 있다는 정책적 판단을 하기 때문이다. 또한 도서관 컨소시엄이나 대학도서관 위원회 등에서 OpenURL을 지원하지 않는 출판사나 데이터베이스 공급자를 상대로 OpenURL 준수여부를 구독 평가 항목으로 포함시키겠다는 압력을 행사하는 것도 OpenURL 적용을 촉진시키는 주요 추진력이다<sup>9)</sup>.

## 결언

OpenURL 기반 링킹시스템의 구조와 원리를 살펴보는 과정에서 링킹시스템이 우리나라에는 보급 초기 단계이나 외국에서는 이미 일반화되어 있음을 알 수 있다. 앞선 내용에서는 효과적인 개념 전달을 위하여 설명의 간략성을 추구하다 보니 링킹시스템의 대상을 색인, 초록 데이터베이스와 전자저널 위주로 설명하였으나 실제 링킹시스템이 적용될 수 있는 대상은 풀 텍스트 리포지토리 같은 전자 집서(digital collection)와 온라인 목록은 물론 Google Scholar와 같은 웹 검색엔진 등 다양하다.

이와 관련하여 OpenURL 링킹을 무료 웹 사이트에 적용하려는 시도인 COinS(ContextObject in Span)와 같은 움직임을 주의 깊게 관찰할 필요가 있다.

이 글의 초두에서 링킹시스템의 시연을 본 경험이 있는 독자는 “시연에서 본 것이 ‘모든 경우’에 자동으로 그렇게 되는가?”라는 의문을 가질 것이라고 하였다. 그렇다면 어떤 경우에 완벽해 보이는 링킹시스템이 제대로 작동하지 않을까?

이는 성공적인 링킹이 어떠한 경우에 보장되는가를 살펴보면 쉽게 역추론이 가능하다. 한

8) 즉, 데이터베이스를 제공하는 벤더가 지원해야 하는 작업이다. 좀 더 상세한 예를 보려면 Enabling OpenURL: Technical Bulletin, no. 002를 참조하라.

(출처: [http://gale.cengage.com/pdf/customer\\_service/bulletins/openURL.pdf](http://gale.cengage.com/pdf/customer_service/bulletins/openURL.pdf))

9) 한 예로 호주의 Council of Australian University Librarians(CAUL)에 소속된 CAUL Electronic Information Resources Committee(CEIRO)의 활동을 들 수 있다.

(출처: <http://www.caul.edu.au/surveys/OpenURLcompliance2006.doc>)

문헌에 보면 성공적인 링킹은 다음 요소에 의존한다고 한다<sup>10)</sup>:

- 링크서버에 저장된 구독정보의 정확도
- OpenURL 소스에 의하여 생성되는 OpenURL에 포함된 메타데이터의 품질

이 두 가지 요소 중 전자는 앞서 설명한 바와 같이 링킹시스템 벤더가 OpenURL을 지원하는 모든 전자저널 공급자로부터 입수하는 정보로서 지식베이스에 저장되는 정보의 정확성을 의미한다. 그리고 후자는 개별 소스(즉, 출판사와 데이터베이스 벤더)마다 자신들의 메타데이터에서 추출하여 생성하는 OpenURL에 포함된 데이터 요소의 품질을 말한다. 사실 우리가 벤더로부터 데이터베이스를 구독할 때 제공받는 구독대상자료 리스트는 표준을 준수하는 메타데이터가 아닌 개별 벤더 고유의 포맷을 가지며 종종 정확도 문제도 있다. 따라서 이러한 메타데이터에서 추출한 OpenURL에 포함된 메타데이터 요소는 그 품질을 보장할 수 없으며 벤더마다 메타데이터 요소가 다양하여 추출되는 요소도 동일하지 않을 것이라는 짐작은 쉽게 할 수 있다. 그러므로 이들을 기반으로 생성된 OpenURL은 링크변환기가 해석하기에 적절하지 않을 수 있다는 것이다.

추가적으로 위 문제점들을 악화시키는 요소는 이 데이터 수집과 관련된 분산된 연결고리의 복잡성에도 존재한다. 링크변환기 벤더는 정보를 출판사나 구독증개자로부터 입수하는데 일반적으로 이 전달과정은 정규화와 품질제어 확인과정을 포함하지만 오히려 그 과정의 복잡성으로 오류를 유발하기도 한다.

결과적으로 링킹시스템은 보이는 것처럼 완벽하지는 않으며 그 원인은 시스템 자체의 구조적 결함보다는 소스에서 생성되는 OpenURL의 품질과 다양성 그리고 데이터베이스 벤더들로부터 지식베이스 구축을 위하여 제공되는 데이터의 품질 문제로 인하여 비효율성이나 부정확성이 발생한다고 하겠다. 더욱이 이 상황의 심각성은 링킹시스템의 오류가 동적으로 생성되는 정보(즉, OpenURL)에 의존하므로 그 원인을 찾아 보완하기가 다른 일반 시스템보다 어렵다는데 있다.

따라서 링킹시스템이 원래 고안된 목적을 원만하게 수행하기 위하여는 무엇보다도 데이터 품질에 대하여 데이터베이스 벤더와 링킹시스템 벤더들이 책임을 공동으로 자각할 필요가 있다. 이와 같은 데이터 품질 문제는 링킹시스템을 도서관자동화시스템에 적용하였을 때 각 도

10) 즉, 데이터베이스를 제공하는 벤더가 지원해야 하는 작업이다. 좀더 상세한 예를 보려면 Boyle, F., Humphreys, L. & Powell, A. (2003, October 30). The JIBS-UKOLN OpenURL Meeting. Ariadne, 37. Retrieved March 8, 2008, from <http://www.ariadne.ac.uk/issue637/openurl-rpt/intro.html>

서관 서지데이터베이스에 저장된 레코드의 품질에 따라 성능이 달라질 수 있음을 시사한다.

결국 도서관 분야에서는 아무리 좋은 시스템을 구축해도 데이터의 품질이 지원되지 않으면 원하는 결과를 얻을 수 없다는 것을 다시 한번 깨닫게 해 준다.

어쨌든 링킹시스템의 데이터 품질문제를 해결하기 위하여 Todd<sup>11)</sup>는 첫째, 지속되는 심각한 문제점의 인식 및 문제해결을 위한 협력, 둘째, 부정확하고 불안정한 데이터의 보완, 셋째, 서지데이터 전달 절차의 확립, 넷째, 데이터 포맷과 전송 표준 수립과 같은 공동 노력의 필요성을 주장하며 이러한 노력의 실현을 위하여 COUNTER 프로젝트<sup>12)</sup>와 유사하게 무슨 정보가 어떤 포맷으로 제공되어야 하는가 하는 지식베이스 구축을 위한 실무규칙 제정의 필요성을 제안하였다.

11) Todd (2007, April 20). Framework for improving link resolver systems. LiveSerials. Retrieved March 20, 2008, from <http://liveserials.blogspot.com/2007/04/framework-for-improving-link-resolver.html>

12) COUNTER란 Counting Online Usage of Networked Electronic Resources의 두문자어로서 전자자원의 온라인 이용 통계 기록 및 교환을 원활히 하여 사서, 출판사와 중개자 모두에게 편의성을 제공할 목적으로 2002년 3월에 시작된 국제적 움직임이다. 이는 특히 측정대상이 되는 데이터 요소(guidance on data elements to be measured), 이들 요소에 대한 정의(definitions of these data elements), 이용보고 내용과 포맷(usage report content and formats) 그리고 데이터 처리(data processing)의 표준화를 다룬다.