분산통합검색을 위한 시스템 개발

정창후* 이중현* 이현숙* 김평* 양명석* 맹성현* 서정현** 김현**
충남대학교 컴퓨터과학과*, 한국과학기술정보연구원**
{seal, jhlee, hslee, pyung, msyang, shmyaeng}@cs.cnu.ac.kr,
{jerry, hkim}@kisti.re.kr

Development of an Integration System for Distributed Retrieval

Chang-Hoo Jeong* Jung-Hyun Lee* Hyun-Suk Lee* Pyung Kim*
Myung-Seok Yang* Sung-Hyon Myaeng* Jeong-Hyun Seo** Hyun Kim**
Dept. of Computer Science, Chungnam National University*, KISTI**

요 약

사용자들은 기하급수적으로 증가하는 정보 중 원하는 정보를 보다 빠르고 정확하게 제공받기를 원한다. 웹 환경에서 분산되어 있는 검색서버의 특성을 고려한 통합검색 서비스를 통해 사용자는 다수의 분산 서버에서 원하는 정보를 손쉽게 찾을 수 있다. 본 연구에서는 웹 상에 물리적으로 분산된 서버, 즉 이질적인 환경을 가지는 다수의 분산서버를 대상으로 논리적으로 묶어서 하나의 통합검색환경을 제공하기 위한 시스템을 설계하고 구현하였다. XML 기반으로 정의된 프로토콜을 사용해서 각각의 분산된 서버는 쉽게 통합검색에 참여할 수 있으며, 통합서버는 분산서버의 메타 데이터 정보를 이용해서 사용자의 질의 및 결과를 효율적으로 처리하도록 한다.

1. 서론

인터넷의 급속한 발전으로 인해 정보가 기하급수적으로 증가하고 있으며 이와 더불어 정보검색 엔진의 보급이 날로 증대되고 있다. 또한 정보 집합들은 거대화, 집적화되고 있으며, 효율적인 정보 제공을 위해 여러 개의 서버에 정보를 분산시키고 있다. 따라서 유사한 정보들이 여러 서버에서 서비스됨에 따라 적은 비용으로 이들을 통합하여 검색하려는 시도가 진행되고 있다.

그 동안 정보검색 시스템은 정보 집합의 크기 변화가 거의 없는 정적환경과 비분산 환경에서 연구되고 개발되어 왔다. 하지만, 분산되어 있고 각각 다른 정보들을 저장하고 있는 검색 시스템이 늘어나게 되면서 이런 정보들을 통합하여 검색하고자 하는 필요성이 대두되고 있으며, 정보집합은 계속하여 새로운 정보가 추가되는 동적인 상황이 되어가고 있다. 이러한 동적인 분산환경에서 기존의 검색엔진이 갖는 한계를 극복하고자 하는 것이 분산통합검색이다.

분산통합검색은 분산되어 있는 검색서버들을 하나로 통합함으로써 사용자가 원하는 정보를 좀 더 쉽게 찾을 수 있도록 도와준다. 사용자는 각각의 검색엔진들의 특성을 알지 못하더라도 원하는 결과를 얻을 수 있는 것이다. 인터넷의 급속한 발전과 검색엔진의 보급에 따라 분산통합 검색 서비스에 대한 요구가 계속적으로 증대하고 있고 국가의 지식기반 연계사업에서도 통합검색을 필수적으로 제공하도록 요구하고 있어 분산통합검색기의 개발이 꼭 필요한 시점이다.

2. 관련연구

정보검색이라는 학문 분야가 국내에 비해 훨씬 먼저 개척된 외국의 경우에도 통합검색과 직접 관련되어 있는연구는 제한적으로 수행되고 있으며 크게 세가지 방법이 적용되고 있다.

첫 째는 새로운 분산 검색 아키텍처를 개발하여 모든 검색기가 동일한 검색환경을 가지도록 접근하는 방법인데 대표적인 예로 미국 Colorado 대학의 Harvest시스템[1]을 들 수 있다.

두 번째 방법은 새로운 통신 프로토콜을 구현하여 분산되어 있는 모든 검색기들이 이러한 표준을 따르도록 유도하는 연구가 진행되고 있다. 대표적인 예로 미국 Stanford대학에서 개발한 STARTS[2] 및 *SDLIP* 프로토콜[3]을 예로 들 수 있는데, 효과적인 통합검색을 제공하기 위한 각종 정보를 서로 교환할 수 있도록 프로토콜을 설계 구현하고 상용시스템을 포함한 참여 시스템들이 이를 지원할 수 있도록 한다. 이러한 시도가 향후 통합검색 기능의 제공에 어느 정도 영향을 미칠 지 아직은 시기적으로 확실치 않다.

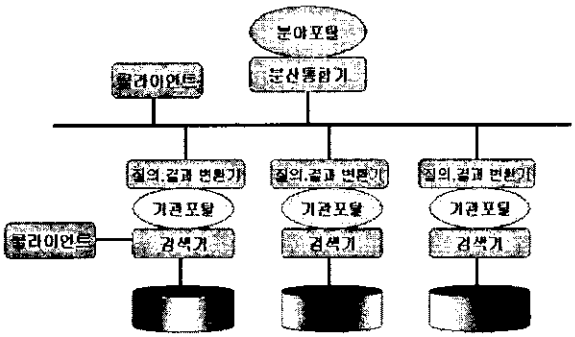
세 번째 방법은 HTTP나 Z39.50[4]과 같은 기존 통신 프로토콜을 사용하고 있는 시스템간의 통합검색을 제공하는 방법으로 가장 보편화될 수 있는 방법이다. 이는 WWW 환경에서 메타검색의 형태로 구현되거나 기존의 Z39.50 프로토콜의 WWW에서의 한계를 극복하려는 형태로 나타나고 있으며 현재 가장 널리 이용되고 있다.

3. 분산통합검색 시스템

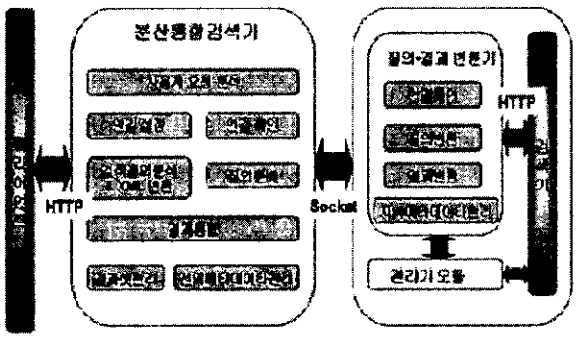
3.1 분산통합검색기 시스템의 구조

분산통합검색기의 전체적인 시스템 구조는 [그림 3-1]과 같은 외부구조와 [그림 3-2]와 같은 내부구조로 이루어져 있다. 분산 모델로서

모든 원천 시스템에는 각 시스템에 대한 질의 및 결과 변환기가 설치된다. 이것을 XML 형태로 분산통합기와 상호 통신을 이루도록 함으로써 임의의 원천 서버와 통합 서버가 쉽게 상호작용할 수 있도록 하였다.



[그림 3-1] 분산통합 검색기의 외부구조



[그림 3-2] 분산통합 검색기의 내부구조

각각의 원천 시스템은 기존과 같이 자신들만의 고유한 서비스를 제공할 수 있으며, 질의 및 결과 변환 모듈을 통해서 분산 통합 검색에 참여할 수 있게 된다. 이것을 가능하게 하기 위해서 각 원천 시스템에 검색엔진의 특성에 맞는 질의 및 결과 변환 모듈이 설치되어야 한다. 완전분산 통합검색에 대한 개념적 및 물리적인 부담을 줄이기 위해서 일단은 질의 및 결과 변환 모듈만을 원천 시스템에 설치하고, 분산검색 통합모듈은 통합검색기에만 존재하도록 한다. 모든 원천 시스템에 분산 통합검색 모듈을 설치하는 것은 향후 연구과제로 남겨둔다.

분산 통합 프로토콜의 장점은 완전 분산 모델에서의 개념적인 복잡성을 줄일 수 있으며, XML을 사용함으로써 임의의 원천서버나 통합서버로의 확장을 쉽게 할 수 있고, 질의 및 결과 변환을 분산시켜 수행함으로써 통합서버의 시간적 부담을 줄여줄 수 있다. 하지만, 변환기 모듈을 각 원천 시스템에 설치해야 하고, 새로운 버전일 경우 각각의 원천 시스템에 설치된 모듈의 수정을 요구해야하는 단점이 있다.

3.2 분산통합검색기의 특징

분산통합서버는 메타데이타를 통해 여러 개의 검색 서버들에 대한 리소스를 효율적으로 공유할 수 있다. 또한 질의 포맷이나 결과 포맷, 메타데이타 포맷을 XML 형식으로 정의하여 표준화를 지향하고 데이터 해석을 쉽게 하도록 했다.

1) 분산통합 검색기

하나의 독립된 시스템으로 존재하며, 사용자와 각 원천 시스템을 연결해주는 미들웨어의 역할을 수행한다. 사용자의 요청을 분석하고 검색을 수행하여 통합된 검색 결과를 보여주고 각 검색기의 메타데이터를 관리하고 연결을 유지하며 사용자에 대한 통합 검색에 대한 투명성을 제공한다.

2) 질의 및 결과 변환기

분산통합검색기와 각 원천 시스템 사이에 존재하여, 분산통합검색기의 XML 형태의 표준질의를 각 검색기에 맞는 질의 형태로 변환하고 각 검색기에서 제공하는 검색 결과를 XML 형태의 표준결과 형태로 변환해 주는 역할을 수행한다. 질의 및 결과 변환기와 원천 시스템 검색기의 통신이 HTTP 위에서 돌아가도록 함으로써 기존의 검색 시스템과의 쉬운 인터페이스를 제공받는다.

3) 원천시스템

분산 통합 검색에 참여하는 검색 시스템을 말한다. 기존에 이미 서비스하던 검색 시스템일 수도 있고, 분산 통합 검색을 위하여 새롭게 개발된 시스템일 수도 있다.

3.3 프로토콜 서비스

프로토콜은 정확하고 효율적인 정보 전송을 하기 위해 필요한 각종 규약, 절차, 규격 등을 각각의 통신 방식에 맞춰 정의하고 정리해 놓은 것이다. 프로토콜은 컴퓨터와 통신 기기의 발전에 따라 정보 전송을 정확하고 효율적으로 수행하기 위해 점점 복잡해지고 다기능화 되었으며, 이에 따라 프로토콜의 계층화 개념이 나타나게 되었다.

이번 연구에서 설계 및 구현한 프로토콜은 응용 계층에서 작동하는 정보검색용 프로토콜이다. 서로 다른 시스템들이 통합검색에 참여하기 위해서는 통합검색 시스템과의 상호 인터페이스가 중요한데, 이것을 위하여 정보검색용 프로토콜을 새롭게 설계 및 구현하였다. 프로토콜의 정의에서 보았듯이, 프로토콜은 "서로 통신하기 위한 규약"이므로 서로 상이한 두 개 이상의 시스템 사이에서 각각의 프로토콜이 다르면 서로 간에 통신이 불가능하게 된다. 역으로 서로 다른 기종의 시스템이더라도 프로토콜만 일치된다면 서로 자유로이 통신할 수 있는 것이다. 통합검색에 참여하고자 하는 시스템들이 직접 이 프로토콜을 구현할 필요는 없고, 단지 질의결과 변환기만을 자신의 시스템에 설치해 주면 된다. 그러면 질의결과 변환기가 시스템 관리자가 작성한 XML 형식의 메타데이타만을 보고, 검색 시스템을 자동으로 통합검색 시스템에 참여시키게 된다.

- 연결 서비스
  원천시스템에 대한 전역메타데이터를 이용하여 질의결과변환기와의 통신을 통해 연결가능 여부를 판단한다.
- Search & Retrieval 서비스
  가장 기본이 되는 서비스로 검색을 요청하고 그에 대한 결과를 가져오는 기능을 수행한다.
- Presentation 서비스
  검색된 문서의 전체 내용을 가져온다.
- 메타데이터 요구 서비스
  원천시스템에 대한 특성을 기술해 놓은 데이터로써 원천시스템의 전역 메타데이터가 갱신될 경우, 이 서비스를 통하여 전역메타데이터를 새롭게 갱신한다.
- 종료 서비스
  원천시스템과의 연결을 종료하고 검색 세션을 중지한다.

디지털도서관 표준 프로토콜로 불리는 Z39.50과 비교를 해보면

Z39.50은 레코드의 공유를 통한 효율적인 자료관리가 목적이기 때문에 통합에 관련된 내용을 다루고 있지는 않다. 또한 사용하기가 어렵고 복잡해서 웬만큼 큰 기관이 아니고서는 실제로 적용하기가 힘들다. 그러나 분산통합을 위해 개발된 프로토콜을 사용할 경우 관리자는 질의결과 변환기를 설치한 후에 시스템에 대한 메타데이타만을 작성하면 프로그램 수정없이 자동으로 통합검색에 참여할 수 있게 된다.

3.4 메타데이타

분산통합기가 각각의 원천 시스템들과 효율적으로 상호작용을 하기 위해서는 각각의 시스템들에 대한 특성, 상태, 성격을 표현하고 있는 메타데이타가 반드시 필요하다. 이 정보들을 이용해서 질의 분배나 결과 통합을 효과적으로 수행할 수 있다.

3.4.1 전역메타데이터

전역 메타데이터는 원천서버에 연결하기 위해 필요한 정보 및 원천서버에 대한 일반적인 정보들의 집합이다.

```
class MetadataInfo() {
    String    guid = "";                  // 전역메타데이타 Unique ID
    long      version = 0;                // 전역메타데이타 버전
    String    sourcename = "";            //  원천시스템 이름
    String    url = "";                   //  원천시스템 URL
    String    ip = "";                    //  원천시스템의 질의결과변환기 IP
    int       port = 0;                   //  원천시스템의 질의결과변환기 포트
    String    category = "";              //  원천시스템의 DB에 대한 분야
    boolean   IsOperating = false;        //  원천시스템의 운영여부
    boolean   IsAlive = false;            //  원천시스템 연결(검색) 가능 여부
    boolean   metadataupdate = false;     // 원천시스템 메타데이타 갱신 여부
    Socket    socket = null;              //  질의결과변환기와의 통신 정보
    DataInputStream    dis = null;        //  질의결과변환기에 대한 입력 정보
    DataOutputStream dos = null;          //  질의결과변환기에 대한 출력 정보
}
```

[그림 3-3] 전역메타데이타의 클래스 표현

각 원천시스템들에 대한 전역메타데이터 정보는 class로 표현되고 list로 관리된다. 한번 메모리에 적재된 메타데이터 정보들은 분산통합 검색기의 수행이 종료될 때까지 계속 유지되고 관리된다.

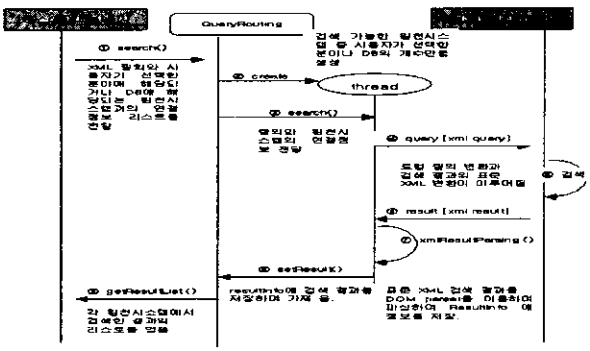
3.4.2 지역메타데이터

통합검색기로부터 요청이 들어 온 표준질의를 원천서버에 요청하기 위해서 표준질의를 원천서버에 맞는 질의어로 변환시켜야 한다. 또한 원천서버에서 반환된 결과는 표준결과 형태로 변환되어 통합검색기로 전달되어야 한다. 지역 메타데이터는 이러한 질의·결과의 변환을 위해 필요한 정보들의 집합이다.

3.5 세부 구현사항

1) 검색요청 처리

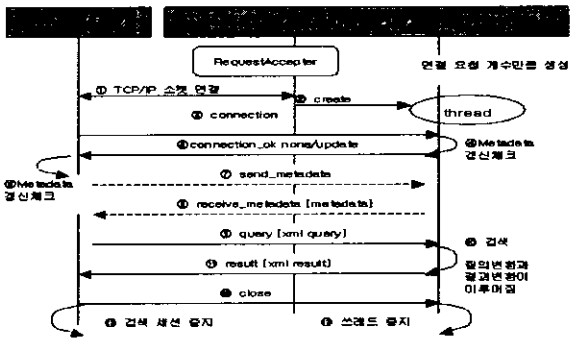
연결 요청을 수행하면 현재 가능한 원천시스템의 리스트를 얻어낼 수 있으며 이렇게 얻어진 원천시스템 리스트는 검색 요청에 사용된다. 검색 방법은 일반검색과 상세검색으로 나뉜다.



[그림 3-4] 검색요청 처리

2) 분산통합검색기와 질의결과변환기 사이의 통신

분산 통합 검색기와 질의 결과 변환기 사이의 상호 통신은 연결확인, 검색요청 그리고 종료 등으로 크게 세가지로 나눌 수 있다. 질의 결과 변환기는 분산 통합 검색기의 연결 요청에 따라 각각의 개별 요청과의 실제적인 서비스 처리를 담당하는 쓰레드를 생성시키고 관리한다. 이러한 쓰레드는 분산 통합 검색기로부터의 연결요청이나 검색요청, 메타데이터 요청 등을 분석하여 실제적인 작업을 처리하는 역할을 수행하며 분산 통합 검색기와의 통신이 중지될 때까지 수행된다.



[그림 3-5] 통신 과정

4. 결론 및 향후 연구

본 연구에서는 분산 통합 검색을 위한 프로토콜을 설계하였으며, 설계된 프로토콜의 기능 및 서비스 등을 실제로 구현하였다. 분산 통합 프로토콜은 디지털 도서관 표준 프로토콜인 Z39.50에서 제공하는 서비스 기능들을 포함하고 있으며, GILS[5]와 STARTS에서 제시하고 있는 메타데이터 기술에 관한 속성집합(attribute set)이나 분산 통합의 개념 등을 담고 있으며, SDLIP에서 제공하는 통신 방법이나 DIENST[6]에서 이용하는 HTTP 프로토콜을 활용하고 있다.

향후 연구는 컬렉션 통합 알고리즘을 적용하여 좀더 정확한 랭킹을 갖는 결과 리스트를 생성하도록 한다. 변환기가 로컬에서 전문정보를 받아서 색인하고 색인 용어들만을 통합기에 보내서 그것을 가지고 통합하도록 한다. 또한 Result Set을 여러 개 관리하면서 결과 셋 내에서 재검색을 수행할 수 있는 기능 등을 추가로 제공할 수 있도록 한다.

5. 참고문헌

[1] "Harvest Project"
http://harvest.transarc.com/

[2] "Stanford Protocol Proposal for Internet Retrieval and Search," January 1997. http://www-db.stanford.edu/~gravano/starts.html

[3] "Search Middleware and the Simple Digital Library Interoperability Protocol," D-Lib Magazine, March 2000. http://www.dlib.org/dlib/march00/paepcke/03paepcke.html

[4] 전우직, "디지털도서관 표준 프로토콜 Z39.50의 개요 및 구현 방안," KOSTI96, 1996.

[5] Government Information Locator Service.
http://www.gils.net/

[6] "Dienst Protocol Specification"
http://www.broadcatch.com/dienst.html