

OpenAPI를 이용한 실시간 자치도서관 도서검색 서비스구현

Implementation of Real-time Book Search Service of Library using OpenAPI

임대현*, 박진태*, 이해림*, 정재필**, 문일영*

Dae-Hyun Lim*, Jin-Tae Park*, Hye-Rim Lee*, Jae-Pil Jeong** and Il-Young Moon*

요 약

본 논문에서는 OpenAPI를 이용해 실시간 자치도서관 도서통합검색 서비스에 대해 구현하였다. 2000년도부터 독서 장려정책으로 각 시,도등 자치지역마다 지역민들을 위한 지역도서관이 지어지기 시작했다. 이를 통해 방대한량의 도서를 확보하고 각 자치도서관은 회원제를 통해 대여 및 열람을 서비스하고 있다. 일반적으로 각 자치도서관에 방문하여 대여/반납과 열람하고, 각 도서관별 홈페이지를 운영하여 도서현황을 검색하고, 예약서비스를 하고 있다. 하지만 방대한 도서를 검색하고 이용하기란 쉽지 않은 실정이다. 따라서 현재 각 도서관에서 구축되어있는 모든 시스템의 구성 그대로 활용하면서 각 도서관별 데이터베이스의 효율적 통합과 검색을 위한 방안으로 TCP/IP방식의 네트워크와 각 도서관의 Database를 연결하기 위한 어플리케이션, OpenAPI 이용한 실시간 Database접근을 통해 도서검색을 서비스하는 방안을 제시하고자 한다.

Abstract

In this paper, the performances of implementation real-time book search service of government library using OpenAPI. Since 2000, each self-government made a government library for local resident because of policies to encourage reading. Each self-government library services book perusal and lend free for local resident. In current system local residents visit to a library for lending, returning and reading books. And each library run by a self-governing administration provides their own homepage for searching service as well as booking. For each services are not unified, however, it is not easy for lenders to search and lend these enormous books which are spread each library. Accordingly, we present a plan which using organization of every system established now in each library as it is, for a scheme of database by libraries' efficient unification and search, a network of TCP/IP and application which connects each library's database, servicing which uses Open API a check books through real time accessing Database.

Key words : OpenAPI, Real-time Book Search Service, TCP/IP

* 한국기술교육대학교 인터넷미디어공학부(School of Internet Media Eng., Korea university of Technology and Education)

** 가천의과학대학교 IT학과(Department of IT eng., Gachon University of Medicine and Science)

· 제1저자 (First Author) : 임대현

· 투고일자 : 2009년 10월 5일

· 심사(수정)일자 : 2009년 10월 6일 (수정일자 : 2009년 10월 23일)

· 게재일자 : 2009년 10월 30일

I. 서 론

2000년도부터 도서관령정책으로 각 시,도등 자치 지역마다 지역민들을 위한 지역도서관이 지어지기 시작했다. 국립중앙도서관과 국회도서관을 중심으로 2008년 8월 현재 전국엔 총 931개의 도서관이 대한민국 국민이면 누구나 열람할 수 있도록 열려있다.

전국 931개 도서관에서 소장하고 있는 방대한 량의 책들을 대여 및 열람서비스를 하고 있지만 사용자가 필요한 책을 찾기 위해서는 노력이 필요한 실정이다. 그런 수고를 덜기 위해 각 도서관들은 각자의 도서관 DataBase를 구축하고 온라인으로 홈페이지를 구축하여 도서를 검색하고 그 책의 위치와 열람 / 대출이 가능한지 확인할 수 있도록 하였다. 하지만 내가 필요한 책이 어느 도서관에 있는지, 어느 도서관에서 책을 열람할 수 있고 대출할 수 있는지 각 도서관의 홈페이지들을 하나하나 방문하여 찾아야 하는 번거로움이 있었다. 그리고 각 도서관마다 홈페이지와 각 DataBase들의 관리가 잘 되지 않아 링크가 깨져있거나 잘못된 정보를 보여주는 오점들이 있다.

판매목적의 새 책을 사기 위해서는 네이버, 다음 등의 각종 포털사이트에서 데이터들을 한데 모아 각 서점과 연결해주고, 가격비교까지 해주고 있다. 하지만 각 자치 도, 시, 구등의 도서관 및 대학 도서관에서는 그런 서비스를 하고 있지 않아 많은 어려움을 겪는 것이 현실이다. 이러한 불편함과 오점들을 해소하기 위한 노력으로 각 자치 구와 시의 관점에서 통합검색을 할 수 있도록 많은 자치구에서는 관할지역 내 도서관검색 홈페이지를 통합 운영하고 있다.

하지만 이러한 노력을 하는 자치도는 많지 않을 뿐 아니라 각 대학 도서관들은 통합되지 않고 있는 실정이다[1].

또한 각 도서관 홈페이지의 관리 및 유지보수가 제대로 이루어지지 않아 링크가 깨져있거나 쿼리결과가 실제 검색한 내용과 다르게 출력되는 경우 또한 많이 발생하고 있다. 하지만 각 도서관별로 모든 소장도서에 관한 DataBase를 효율적으로 구축해 놓았고, 검색의 용의성 및 신속 / 편리성을 위해 제목, ISBN, 도서의 종류에 따른 분류번호등 비교적 충분한 검색조건을 갖추고 있다. 현재 구축된 각 도서관

들의 Database와 홈페이지들을 그대로 유지하면서 통합된 도서검색을 위한 방법으로 실시간 쿼리검색을 이용하여 구현하였다.

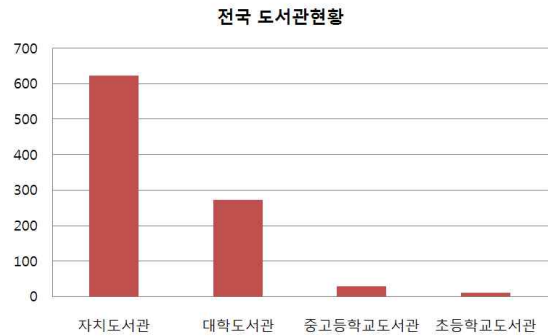


그림 1. 전국도서관 현황 (2008년 8월 현재, 네이버)

Fig. 1. Presentation of self-government library (August. 2008, Naver)

II. 시스템의 구성

구현한 시스템의 소프트웨어는 크게 세가지로 나누어진다. 사용자가 접속하여 검색하는 첫 번째 클라이언트 역할이 되는 웹페이지와 각 도서관의 메인서버에 설치되어있는 DataBase에 접근하는 두 번째 클라이언트 역할을 하는 소프트웨어, 마지막으로 사용자와 도서관 DataBase를 연결시켜줄 중앙서버로 나뉜다. 먼저 사용자가 웹페이지에 접속하고 원하는 도서관의 지역을 설정하면 해당 구의 도서관 데이터가 데이터그램형태로 저장된다. 원하는 도서관을 체크하고 검색하고자하는 도서키워드를 입력하면 NaverAPI 도서검색을 이용한 검색결과가 출력된다. 검색결과에서 해당 책을 찾아 검색버튼을 누르게 되면 Java소켓이 생성되어 ISBN코드를 도서관으로 전송하게 된다.

NaverAPI를 이용한 이유는 검색의 신속/정확성을 기하기 위함이다. 사용자가 직접 검색한 문자열 키워드를 전송하게 되면 포함된 키워드의 책들이 모두 검색이 되어 전송되게 된다. 그렇게 되면 정확성이 떨어질 뿐 아니라 호스트와 클라이언트간의 통신량이 증가하게 되어 트래픽이 발생하게 되고 이는 웹페이지의 속도와 직결되기 때문이다. 이미 사용자의 선택으로 저장되어있는 도서관정보와 ISBN정보를 중앙서버로 전송하게 되고, 중앙서버는 도서관정보를

식별하여 미리 입력되어있는 IP주소로 라우팅하여 검색요청을 하게된다. 검색요청을 받은 도서관배포용 소프트웨어는 도서관 서버의 DataBase에 접근하여 검색을 하고 그 결과를 중앙서버를 통해 다시 사용자가 확인할 수 있도록 웹페이지로 전송해준다.



그림 2. 구현 시스템 구성도

Fig. 2. Construction of System implementation



그림 3. 웹페이지 구현부

Fig. 3. Implementation of web

III. 도서관 배포용 소프트웨어

각 도서관에 배포되는 소프트웨어는 MFC, TCP/IP 기반의 소켓프로그램과 도서관 DataBase에 접근하여 쿼리문을 만들고 검색하여 시스템의 서버로 책의 정보들을 전송해준다.

각 프로그램의 통신을 가능하게 해주는 것이 바로 소켓이다. 소켓에는 TCP/IP, UDP등과 같은 종류가 있지만 본 논문에서 구현한 소프트웨어에서는 TCP/IP를 이용하여 서버를 구축하였다.

이 소프트웨어가 하는 역할은 다음과 같다. 첫째, 서버로부터 전송 되어진 정보를 받는다. 둘째, 웹으로부터 전송받은 키워드를 이용, 해당 도서관의

DataBase에 접속하여 관련 정보를 얻는다. 셋째, 검색 결과를 프로토콜을 이용하여 시스템의 서버로 전송한다[2].

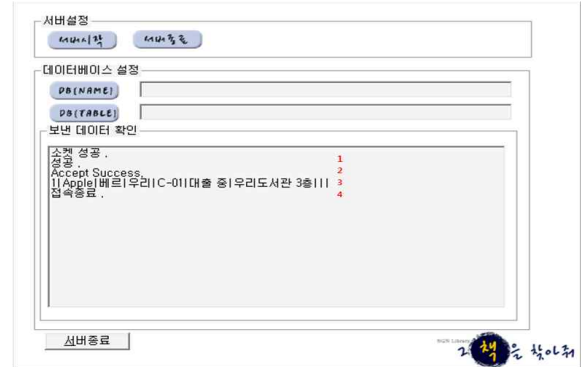


그림 4. 소프트웨어 구현부 I

Fig. 4. Implementation of S/W I

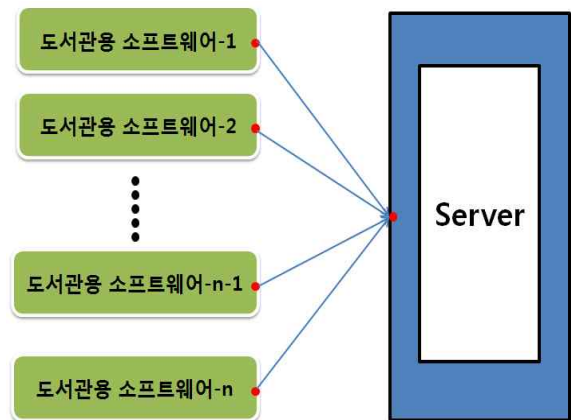


그림 5. 소프트웨어 개념도

Fig. 5. Implementation concept of S/W

표 1. Socket전송을 위한 스트링변환[4]

Table 1. String change for Socket transmission

```

void CClientSocket::OnReceive(int nErrorCode)
{
    ...
    if (!mysql_query(&conn, k))
    {
        sql_result = mysql_store_result(&conn);
        field_num = mysql_num_fields(sql_result);
    } else {
        pDlg->GetDlgItemText(IDC_DB_VALUE, temp);
        temp += "실패.\r\n";
        pDlg->SetDlgItemText(IDC_DB_VALUE, temp);
    }
    if(sql_row=
mysql_fetch_row(sql_result))!=NULL) {
        for(int i=1; i<10; i++)
        {
            str_list += sql_row[i];
            str_list += "|";
        }
    }
}
    
```

```
CString temp;
pDlg->GetDlgItemText(IDC_DB_VALUE, temp);
temp += str_list;
pDlg->SetDlgItemText(IDC_DB_VALUE, temp);
...
```

표 2. 사용 함수와 역할 정리

Table 2. definitions of used Function

<p>MFC의 CSocket 클래스 재정의</p> <ul style="list-style-type: none"> - Listen 함수 (클라이언트의 접속을 기다림) - Accept 함수 (클라이언트의 접속을 처리) - Receive 함수 (클라이언트가 보낸 데이터를 받아 들임) - Send 함수 (클라이언트에게 Database 검색 결과를 전송) - Thread (다수의 클라이언트가 접속을 했을 때를 대비 작업을 처리하기 위한 함수) - CriticalSection (각기 다른 클라이언트로부터보내진 데이터의 Mixing을 방지하기 위한 임계지역 설정) <p>CDialog 사용자를 위한 어플리케이션 디자인</p> <ul style="list-style-type: none"> - Bitmap (다이얼로그의 버튼 이미지를 설정하기 위한 함수) - AlphaSplashDlg (프로그램 실행시 로그 화면 생성)
--

표 3. 사용 헤더 파일 및 라이브러리

Table 3. used header file and Library

<p>헤더</p> <ul style="list-style-type: none"> - wingdi.h : GDI procedure declarations, constant definitions and macros, 다이얼로그를 이용한 기본 어플리케이션을 디자인하기 위한 헤더파일 - winsock2.h : 기본적인 소켓 헤더 파일 - mysql.h : DataBase 접속을 하여 검색을 하기위한 헤더파일 <p>라이브러리</p> <ul style="list-style-type: none"> - libmysql.lib : mysql.h과 마찬가지로 Database에 접속하여 검색을 하기위한 라이브러리 - ws2_32.lib : 기본 소켓 통신을 위한 라이브러리
--

IV. 웹 페이지

사용자가 웹 페이지에 접속하게 되면 검색하고자 하는 도서관의 지역을 선택하게 된다.

검색된 도서의 ISBN값과 함께 Database에서 라우팅된 도서관의 IP주소를 php소켓을 생성하여 스트링의 형태로 해당 도서관 호스트로 전송하게 된다.[5]

호스트에게 요청을 한 뒤 최대지연시간은 5초로 설정하였다. 이는 사용자가 인터넷을 사용할 때 심리적으로 안정되게 대기할 수 있는 최대시간이 7초이기 때문이다.

최대 지연시간인 5초동안 호스트로부터 요청에 대

한 답변이 없을 경우 검색을 다시 해보도록 유도한다.



검색어	회원 출력 개수
소프트웨어	100

NGN 도서관 지역정보

지역 선택

지역을 찾고 싶은 지역을 설정해 보세요!

지역선택 >> > [시/구/군] >> > 선택

검색어 : 소프트웨어

소프트웨어 누가 이렇게 개역같이 만든 거야 >> 도서관 검색

"그동안 사용자는 원래 사용자는 그대로입니다." 오늘날 사용하고 있는 소프트웨어는 개역 같은 게 많습니다. 물론 뭐라 표현할 말이 없을 정도로 아주 개역 같은 것들도 있습니다. 온갖 별리적인 프로그램들...

조물 은 소프트웨어 >> 도서관 검색

2005년 15회 JOLT상 수상작! 이만준 선정 10대 도서 컴퓨터 인터넷 부문! Java.net 선정 개발자 #1! #222222 권위자! #1000000 세계를 돌릴게 안 조물 은 소프트웨어 불 불크 베스트 화음! 인디어 한국어판 출간!...

소프트웨어 아키텍처 >> 도서관 검색

소프트웨어 아키텍처는 물론, 아키텍처를 꿈꾸는 개발자, 대학생도 꼭 읽어야 할 아키텍처 바이블! 소프트웨어 엔지니어들의 패러다임을 바꾸고 있는 소프트웨어 아키텍처의 이론과 개념, 풍부한 예제와 프로그래밍...

Professional 소프트웨어 개발 >> 도서관 검색

그림 6. 웹페이지 구현부

Fig. 6. Implementation of web

호스트로의 답변을 미리 정의해 놓은 프로토콜을 중심으로 파싱하여 사용자의 화면에 띄워주게 된다.

표 4. 라우팅된 도서관의 IP주소를 스트링 형태로 파싱한 부분

Table 4. Parsing to string for IP address of library

```
function book(value) {
    if(!document.MyForm){
        alert('Error!');
        return false;
    }
    var form = document.MyForm;
    for(var i=0;i<form.list.length; i++) {
        if(form.list[i].checked == true) {
            checkedButton++;
            ip_value = ip_value + form.list[i].value;
            ip_value = ip_value + "|";
        }
    }
}
```

클라이언트에서 가장 중요한 부분은 소켓의 응답에 따른 지연시간과 정의된 프로토콜을 파싱하여 사용자에게 화면에 출력해 주는 부분이다[6],[7].

V. 서 버

서버는 웹 페이지와 도서관 배포용 소프트웨어를 연결해주는 역할을 수행한다. 웹 페이지로 각 사용자의 검색 요청을 받으면 이 요청을 서버로 전송하고 서버는 이 요청을 각 도서관 소프트웨어로 보내준다.

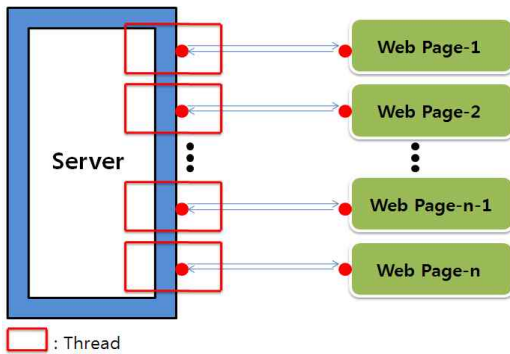


그림7. 서버와 웹페이지의 정보 교환
Fig. 7. data exchange between server and web

이러한 기능을 위한 소켓을 조금 더 편리하게 사용하기 위하여 소켓 함수들을 재정의하였다.

또한 서버는 각각의 쓰레드를 이용하여 여러 웹페이지의 요청을 처리할 수 있게 구성이 되어 있고 또한 여러 웹페이지의 정보 요청 시 각 데이터 간 간섭을 피하기 위하여 각 쓰레드 마다 임계지역을 설정하였다.

VI. 결 론

현재 네이버와 다음을 비롯한 포털사이트와 인터넷 서점들이 많은 사용자들을 불러모으고 있다. 특히, 네이버의 경우 각종 인터넷 서점과 교보문고등의 대형서점과의 제휴로 사용자가 원하는 책을 찾아주고, 직접 연계 시켜주고 있으며 가격비교까지 서비스를 하고 있다. 또한 인터넷으로 중고서적을 판매, 연결시켜주고 있는 사이트 또한 좋은 반응으로 사용자들에게 서비스를 하고 있다. 하지만 각종 자치도서관 및 대학도서관들을 연결해주고 정보를 제공해주는 사이트는 없다. 사용자들의 불편함을 자치도서관이라는 이유로 외면하고 있는 실정이다. 특히 900여개가 넘는 각종 도서관들을 한데로 묶어주고, 도서관장려정책에 발맞춰 나아갈 수 있는 단순 검색만이

아닌 포털사이트로의 확대 또한 고려할 수 있는 사항이다. 또한 많은 도서관서버들의 DataBase를 모두 통합하는 의미의 포털사이트가 아닌, 현재 운영중인 사이트는 그대로 운영하면서, 우리 사이트와 협약만 맺게 되면 개발한 S/W의 설치만으로도 모든 DataBase를 사용할 수 있다. 즉, 다른 포털사이트 및 검색사이트와는 다르게 사이트 구축의 가장 큰 어려움인 DataBase의 구축이 필요없게 된다. 그리고 현재 각 자치도서관의 문제점인 검색엔진의 노후화와 관리부실로 인한 불편함을 직접 DataBase에 실시간으로 접근하게 되어 유지/보수가 매우 쉬운 장점이 있다.



그림 8. 구현 S/W의 장점
Fig. 8. Advantages of impletation system

자치도서관 및 대학도서관에서 책을 검색하는 이유는 정확한 제목으로 책을 찾기 위함이다. 하지만 현재 모든 도서관들의 책검색은 키워드를 중심으로 하기 때문에 내가 찾는 책을 찾는데 더 많은 책들이 검색이 되고 찾고자 하는 책을 찾기에 더욱 어려움이 생기게 된다. 하지만 본 논문에서 제시한 시스템은 검색시 ISBN을 사용하기 때문에 정확한 책을 더욱 빠르게 찾을 수 있는 장점이 있다. 각 도서관별로 DataBase를 가지고 있는데, 그 환경들이 많이 상이하다. 이를 통합한다는 것은 매우 번거로운 일이다. 하지만 직접 접근해서 데이터를 가져오는 것은 쉬운일이다. Socket부분과 DataBase의 처음 설정부분만 바꿔주게 되면 Mysql, Mssql, 오라클등 모두 접근이 가능하기 때문이다. 또한 특정 Database환경에 맞춰 S/W를 배포하게 되면 쉽게 해결될 수 있는 문제일 것이라 생각된다.

참 고 문 헌

- [1] 이영희, “공공도서관의 학교도서관지원팀의 활성화방안”, 독서문화연구, Vol.7, No.0, 2008
- [2] 이성근, “TCP와 UDP 흐름의 공평성 향상을 위한 트래픽 조절 메커니즘”, 컴퓨터교육학회논문지, Vol.7, No.1, 2004
- [3] 채덕진, 김룡, 이용미, 황부현, 류근호, “한 번의 데이터베이스 탐색에 의한 빈발항목집합 탐색”, 정보처리학회논문지D, Vol.15, No.1, 2008
- [4] 강은호, 윤석우, 김경창, “실시간 모바일 GIS 응용 구축을 위한 주기억장치 데이터베이스 시스템 설계 및 구현”, 정보처리학회논문지, Vol.11, No.1, 2004
- [5] 전용태, 정상화, 윤인수, “고성능 경량 TCP/IP를 이용한 소프트웨어 기반 TCP/IP 오프로드 엔진 구현”, 정보과학회논문지, 컴퓨팅의 실제 및 레터 제14권 제4호, 2008. 6
- [6] 이종서, 김유두, 문일영, “TCP/IP 소켓을 이용한 실시간 감시 시스템 설계 및 구현”, 한국항행학회논문지, Vol.12, No.1, 2008
- [7] 곽현민, 김남희, 이상태, 전병실, “정보통신:TCP/IP 트래픽의 공평성 향상을 위한 GFR 서비스의 버퍼관리기법”, 정보처리학회논문지, Vol.10, No.5, 2003

임 대 현 (林大鉉)



2003년 2월~현재 : 한국기술교육대학교 인터넷미디어공학부 재학
관심분야 : 멀티미디어, 영상처리

박 진 태 (朴珍太)



2005년 2월~현재 : 한국기술교육대학교 인터넷미디어공학부 재학
관심분야 : 클라우드 컴퓨팅

이 혜 림 (李德琳)



2008년 8월 : 한국기술교육대학교 인터넷미디어공학부 졸업 (공학사)
2008년 9월 ~ 현재 : 한국기술교육대학교 대학원 정보미디어공학과 재학 (공학석사)
관심분야 : 무선 TCP, 무선 Mesh 네트워크, 라우팅 프로토콜

정 재 필 (鄭在弼)



1985년 2월 : 단국대학교 전자공학과 (공학사)
1989년 8월 : 단국대학교 대학원 전자공학과 (공학석사)
2000년 8월 : 한국항공대학교 대학원 항공통신정보공학과 (공학박사)
1989년 8월 - 1991년 12월 : (주)동양 전자통신 중앙연구소
1994년 2월 - 현재 : 가천의과대학교 IT학과 교수
관심분야 : 이동통신, 신호처리, 유비쿼터스

문 일 영 (文日永)

2000년 2월 : 한국항공대학교



항공통신정보공학과 (공학사)
2002년 2월 : 한국항공대학교 대학원 항공통신정보공학과 (공학석사)
2005년 2월 : 한국항공대학교 대학원 정보통신공학과 졸업(공학박사)
2004년 ~2005년 : 한국정보문화진흥원 선임연구원

2005년 3월~현재 : 한국기술교육대학교 인터넷미디어공학부 조교수

관심분야:무선 인터넷 응용, 무선 인터넷, 모바일 IP