

다중 도메인명을 지원하는 URL 전달 서버의 설계 및 구현

노상호* · 장세현** · 김상연** · 양희재*

* 경성대학교 컴퓨터공학과
** 경성대학교 멀티미디어 응용학과

Design and Implementation of a URL Forwarding Server for Providing Multiple Domain Names

Sang-Ho Noh*, Se-Hyun Jang**, Sang-Yon Kim**, and Heejae Yang*

* Department of Computer Engineering
** Department of Multimedia Applications
Kyungsung University

요 약

본 연구에서는 인터넷상에 존재하는 웹 사이트를 새로운 도메인 네임으로 연결시켜주는 URL forwarding 서버를 설계 및 구현하였다. URL forwarding 서버는 수많은 인터넷 사용자들의 웹 사이트를 간단하고 다양한 형태의 URL로 매핑 시켜준다. URL forwarding 서버는 HTTP 프로토콜에서 redirection 응답코드를 기반으로 구현되었으며 Linux 시스템에서 실험을 하였다. 본 논문에서는 HTTP와 forwarding 서버를 서로 비교 분석하여 고찰해보고 구현된 forwarding 서버의 동작에 대해서 알아본다.

1. 서론

최근 인터넷의 사용은 기업, 회사, 학교, 사회 등 생활 전 분야에 걸쳐 폭발적으로 확대되어, 1999년 말 현재 세계적으로는 2억 명 이상, 국내에서만 500만 명이상이 인터넷을 사용하는 것으로 알려져 있다 [2]. 인터넷 사용자들은 웹 상에서 자신들의 고유한 홈페이지를 가지는데 이런 홈페이지는 전 세계적으로 유일한 주소를 필요로 한다 [4].

홈페이지의 문자열 주소는 도메인을 관리하는 네임서버에 의해 숫자 표기의 IP 주소로 매핑이 되는데 현재 IPv4에서 32bit 크기를 가지는 이러한 IP 주소는 수요가 포화상태에 이르렀다 [5].

본 연구에서는 새로운 IP 주소를 부여받지 않고도 문자열 주소를 사용할 수 있고 ‘~’ 형식의 인터넷 주소가 아닌 ‘원하는 문자열+도메인’ 형식의 다양한

문자열 주소를 사용하여 기존에 존재하는 홈페이지 주소로 연결시켜주는 URL forwarding 서버를 설계 및 구현하였다.

본 논문에서는 네임서버와 HTTP, URL forwarding 프로그램에 대해 고찰해 본다. 본 논문의 구성은 다음과 같다. 2절에서는 인터넷에서 홈페이지를 찾아가기 위해 필요한 네임서버와 URL forwarding 서버를 구현하기 위해 사용된 HTTP 요청 및 응답코드에 대해 알아보고 3절에서는 본 연구에서 구현된 URL forwarding 서버에 대해 알아봄, 4절에서는 결론 및 향후 연구방향에 대해 알아본다.

2. 관련연구

2.1 네임서버 (Name Server)

인터넷상에서 클라이언트가 URL을 이용해 원하는 서버에 접속하고자 할 때 해당 서버의 IP 주소를 얻기 위해서 네임서버를 사용한다 [3]. 네임서버는 도메인 관리자로서 해당 도메인을 사용하는 모든 문자열 주소를 IP 주소로 혹은 IP 주소를 문자열 주소로 연결 시켜주는 역할을 한다. 클라이언트가 문자열 주소를 입력하면 원하는 서버로 곧바로 접속하는 것이 아니라 클라이언트의 네임서버로부터 원하는 서버의 IP 주소를 얻은 후 서버를 찾아서 접속한다. 네임서버에서 처리하는 문자열 주소는 문서주소 즉 '호스트네임'도메인'을 처리하며 도메인 뒤에 '/' 형태로 표기되는 서버 속의 정보는 접속할 웹 서버에서 처리한다.

2.2 HTTP

HTTP는 분산환경 및 공동작업 환경에 이용할 하이퍼미디어 정보시스템의 개발을 목적으로 설계된 응용 계층의 프로토콜로서 WWW에서의 하이퍼텍스트 문서의 전송을 위해 사용된다 [1][6]. HTTP에서 메시지의 요구는 3가지 영역으로 구성되어 있다. 요구라인(request-line)과 헤더 그리고 body이다. 헤더는 반드시 필요한 것이 아니기 때문에 요구 메시지를 보낼 때 포함되지 않아도 되지만 일반적으로 클라이언트가 웹 브라우저를 사용해 서버로 접속할 때 몇 가지 헤더가 전송된다. 그 중에서 'Host' 필드는 클라이언트가 접속하고자 하는 서버의 주소가 명시되는 헤더 필드를 나타낸다.

메시지의 응답도 3가지 영역으로 구성되어 있다. 상태라인(status-line)과 헤더와 body이다. 상태라인은 'HTTP-version response-code response-phrase'의 형식으로 구성된다. 응답코드는 클라이언트에서의 요청에 대한 서버의 응답 결과를 숫자로 나타낸 것으로 본 연구에서 사용한 300번대는 요청한 문서가 다른 주소로 옮겨졌음을 의미하며 문서가 옮겨진 URL을 나타내기 위한 'Location' 헤더 필드가 포함되어야 한다 [1].

3. URL Forwarding

3.1 URL Forwarding 원리

일반적으로 클라이언트가 원하는 웹 서버에 인터넷 문자열 주소를 사용하여 접속할 때 먼저 네임서버로 접

속해 문자열 주소와 일치하는 IP 주소를 응답 받은 다음 웹 서버로 접속한다. URL Forwarding의 경우 클라이언트가 요청한 문자열 주소의 IP 주소는 모두 URL Forwarding 서버의 주소를 가리키고 있다. URL Forwarding 서버가 클라이언트의 요청을 받으면 메시지에서 'Host' 헤더 필드를 분석하여 요청한 메시지가 자신의 서버에 있는 웹 문서인지 다른 곳에 있는 웹 문서인지를 판단한다. 자신의 서버로의 요청인 경우는 80번이 아닌 다른 포트로 동작하고 있는 웹 서버로 요청을 넘기며, 다른 서버에 있는 문서를 요청한 경우는 클라이언트에게 요청한 문서가 위치한 문자열 인터넷 주소를 응답하고 클라이언트는 요청한 문서가 실제로 위치한 서버에게 다시 문서를 요청한다 (그림 1). 그러므로, URL Forwarding 프로그램은 인터넷 문자열 주소를 가지고 있는 웹 문서를 클라이언트가 원하는 간단하고 명확한 인터넷 문자열 주소로도 접근이 가능하게 한다

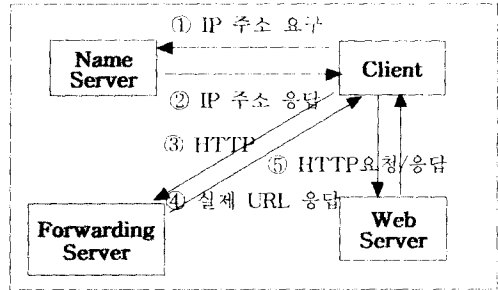


그림1. URL forwarding 원리

3.2 구현방법

URL Forwarding을 하기 위해서 클라이언트의 HTTP 요청을 분석하고 문서의 위치를 알려주는 프로그램을 구현하였다. URL forwarding 프로그램은 서버의 80번 포트를 사용하고 웹 서버는 다른 포트를 사용하여 동작하며, URL forwarding 프로그램이 시작할 때 URL forwarding을 하기 위한 인터넷 문자열 주소 데이터를 데이터베이스로부터 읽어서 메모리에 적재한다. HTTP 요청이 들어오면 프로세스를 생성시켜 요청을 처리하는데 자신의 서버에 있는 문서를 요청한 경우는 80번이 아닌 다른 포트로 동작하고 있는 웹 서버에게 요청을 넘기며, 메모리에 적재되어 있는 인터넷 문자열 주소인 경우는 HTTP를 이용해 클라이언트에게 응답을 하고 생성된 프로세스는

소멸된다.

URL forwarding 서버는 HTTP 요청을 처리한 후 forwarding을 요구하는 경우 클라이언트에게 HTTP의 301번 상태 코드를 응답한다. 301번은 요청한 웹 문서가 영구히 다른 곳에 존재한다는 뜻을 의미하는 코드로서 응답 메시지의 헤더 필드에 'Location'이 포함된다. 'Location'은 웹 문서가 존재하는 위치를 가리키는데 웹 문서의 위치는 URL forwarding 프로그램이 시작될 때 메모리에 적재된다. 그림 2는 클라이언트가 요청한 인터넷 문자열 주소가 'aaa.bbb.com'이고 실제 주소가 'ccc.ddd.com/~foo'일 때 URL Forwarding 프로그램이 클라이언트에게 응답하는 메시지의 예를 보여준다.

```

HTTP/1.0 301(or 302) Redirection
Location: http://ccc.ddd.com/~foo
(blank line)
    
```

그림2. URL Forwarding 응답 메시지의 예

3.4 구현 환경

URL Forwarding 서버는 유닉스 운영체제를 기반으로 하는 인터넷 서버 컴퓨터에서 사용 되도록 설계 되었다. 현재 URL Forwarding 서버는 유닉스 운영체제로 널리 사용되고 있는 솔라리스와 Linux 운영체제에서 각각 구현, 작동되었으며 실제 URL 주소는 MySQL을 사용하여 데이터베이스에 저장되어 있으며 시스템 부팅시 URL forwarding 프로그램이 적당한 절의를 사용하여 메모리에 적재시킨다. 본 서버 구현에 있어 URL Forwarding 서버와 네임 서버는 표 1의 환경에서 개발, 구현되었다.

구분	O.S	program	memory
DNS 서버	Redhat Linux 6.1	BIND	128MB
Forwarding 서버1	Solaris 2.4	GNU C	32MB
Forwarding 서버2	Redhat Linux 6.1	GNU C	128MB

표 1. 개발 환경

4. 결론 및 향후 연구 과제

본 논문에서는 웹서버에서 가상 호스팅과 같은 기능을 하면서 웹 서버에 전혀 부담은 주지 않고 실제 존재하는 웹 주소로 연결시켜주는 URL Forwarding 서버의 구조 및 설계에 대해서 고찰하고 평가하였다.

URL Forwarding 서버는 가상 호스팅 기법과 달리 별도의 프로그램이 실행되어 네임 서버와 HTTP 요청을 하는 클라이언트 사이에 위치하여 클라이언트에서 요청한 HTTP 문자열 주소를 분석하여 실제 주소를 클라이언트에게 알려 준다. 이러한 방식은 첫째, 새로운 IP 주소 부여 없이 새로운 문자열 주소를 사용할 수 있으며 둘째, 하위 디렉토리 개념의 복잡한 주소 구조를 가지는 개인의 홈페이지 주소를 외우기 쉽고 간단명료한 주소로 변경할 수 있는 기능을 가지고 있다.

본 논문에서 구현된 URL Forwarding 서버는 새로운 IP 주소를 사용하지 않고도 이미 존재하는 웹 문서를 다른 웹 주소로 쉽게 변환할 수 있지만, 네임 서버와 사용자 사이에 URL Forwarding 서버가 위치하여 웹 문서를 접속하는 과정에서 지연 시간이 발생하며 실제 문자열 주소를 찾는 과정에서 검증되지 않는 탐색 알고리즘으로 오버헤드가 발생한다.

향후 연구 과제로서는 URL Forwarding 서버에서 클라이언트로부터의 문자열 주소 요구와 사상되는 실제적 주소를 찾는 알고리즘에서 해쉬와 같은 보다 효율적 알고리즘을 개발 적용하는 것과 URL forwarding서버를 프록시 서버로 사용하여 캐시를 이용한 구현에 대해 연구가 필요하다.

참고 문헌

- [1] Irvine H. Frystyk MIT/LCS, HyperText Transfer Protocol(HTTP), 1996, RFC 1945
- [2] Korea Network Information Center (KRNIC), Domestic Internet Users('99. 11), 1999. 11. 30. Available from: <http://www.nic.or.kr/net/net.html>
- [3] Cricket Liu & Paul Albitz, *DNS and BIND 2nd ed*, O'Reilly, 1997
- [4] Xerox Corporation M. McCahill, Uniform Resource Locators (URL),1994, RFC 1738
- [5] W. Richard Stevens, *TCP/IP Illustrated, Volume I*, Wesley, 1993
- [6] W. Richard Stevens, *TCP/IP Illustrated, Volume III*, Wesley, 1993