

## 모바일 환경을 위한 웹 서비스 전송 시스템

김진일\*, 김용태\*\*, 박길철\*\*\*

### A Web Services Delivery System for Mobile Environment

Jin-Il Kim\*, Yong-Tae Kim\*\*, Gil-Cheol Park\*\*\*

#### 요약

최근 무선 통신의 대중화로 인터넷 사용자들은 모바일 컴퓨팅 환경에서도 HTTP 기반의 웹 서비스를 요구하고 있다. 그러나 모바일 디바이스의 제한적인 자원의 특성으로 인해 전체 웹 서비스 성능이 저하될 수 있다. 그러므로 무선 인터넷 전송 시스템을 설계하는 데 가장 중요한 요소로 급속도로 증가하는 수요에 따라 대용량 트래픽을 처리할 수 있는 확장성을 고려해야 한다. 그러나, 현재까지 몇몇 선행 연구의 경우에도 단순한 압축, 캐싱 등을 이용하여 문제를 해결하고자 했을 뿐 근본적인 성능 개선책을 제시하지 못하고 있다.

따라서 본 논문에서는 모바일 환경에서 웹 서비스를 수행할 때 발생하는 여러 문제점들과 사용자의 요구 사항을 해결하기 위해 HTML/WML 변환기, SOAP 메시지 처리기 등으로 구성된 모바일 환경을 위한 웹 서비스 전송 시스템을 제안한다. 제안된 시스템을 평가하기 위해 기존의 시스템과 비교하여 분석한다.

#### Abstract

Recently, as wireless communication are becoming more and more popular, Internet users is becoming increasingly demand HTTP-based web service in mobile computing environment. But, overall performance of web services may slow down, because mobile device are generally equipped with lower capability. Therefore, the most important factor to design wireless internet delivery system should be scalable to handle a large scale traffic due to rapidly growing users. But, in case of some preceding research, they not present a basic efficiency reform measure until currently. Therefore, In this paper, We propose a Web Service Delivery System for mobile environment that alleviates problems that occurred while executing web services in mobile environment and are helpful to overcome the above needs. The system composed of HTML/WML Converter, SOAP message processor. We compare the existing system with the proposed system.

▶ Keyword : 웹 서비스(Web Services), 모바일(Mobile), 전송(Delivery), SOAP message(SOAP 메시지), HTML/WML converter(HTML/WML 변환기)

• 제1저자 : 김진일    교신저자 : 김용태

• 접수일 : 2008. 4. 7, 심사일 : 2008. 6. 5, 심사완료일 : 2008. 7. 25.

\* 배재대학교 교양교육 교수    \*\* 한남대학교 멀티미디어학부 강의전담교수    \*\*\* 한남대학교 멀티미디어학부 교수

※ 본 연구는 지식경제부 지역혁신센터사업인 민군겸용보안공학연구센터 지원으로 수행되었음

## I. 서론

최근 모바일 디바이스가 대중화되면서 시간과 장소에 무관하게 HTTP 기반의 웹 서비스를 이용하고자 하는 요구가 늘어나고 있다. 이러한 모바일 컴퓨팅 환경은 무선 네트워크가 사용되면서 초기의 단순한 작업에서 벗어나 기존의 인터넷 서비스를 사용할 수 있도록 요구하게 되었다.

모바일 디바이스는 기존의 일반적인 데스크톱과 비교하여 CPU, 메모리, 디스플레이, 대역폭 등의 측면에서 많은 제한성으로 인해 성능이 떨어진다. 그러나 사용자들은 변화하는 IT 환경에 따라 모바일 환경에서도 유선 인터넷과 유사한 서비스를 요구하고 있다. 따라서 기존의 비즈니스 관련 사이트에서 모바일 디바이스를 통해서도 접속할 수 있도록 지원하려면 기존에 구축된 정보와 동일한 정보를 가지는 별도의 WML 사이트를 구축해야 한다. 또한 별도의 WML 사이트를 구축하는 것은 비용과 시간 면에서 많은 부담이 발생한다. 더욱이 HTML 사이트의 정보가 변경될 경우에 발생하는 일관성 문제는 WML 사이트를 관리하는 일을 더욱 어렵게 만든다.

그러므로 모바일 환경에서 기존의 유선 인터넷 서비스의 콘텐츠를 활용에 대한 요구가 증가하지만 기존의 서비스를 그대로 사용하기는 불가능하다. 따라서 이러한 문제를 해결하기 위해서는 유선 인터넷 서비스를 위해 제작된 콘텐츠를 모바일 환경에 맞도록 자동 변환하는 것이 필요하다. 즉, 모바일 디바이스에서 전송된 WML 문서를 HTML/XML 문서로 재구성하고 HTML/XML 형태로 기술된 웹서버에 존재하는 문서를 WML과 WML Script의 형태로 변환하여 모바일 디바이스로 전송하는 변환기가 필요하다. 그리고 무선 인터넷 전송 시스템을 설계할 때 중요한 점은 급속도로 증가하는 수요에 따라 대용량 트래픽을 처리할 수 있는 확장성을 고려해야 한다는 것이다. 그러나 모바일과 같은 자원의 특성으로 인한 제약점들이 전체 웹 서비스 성능을 저하시키는 문제점이 발생할 수 있다. 현재까지 이러한 제한적인 환경에서의 성능 문제에 대하여 많은 연구가 이루어지지 못하였고 몇몇 선행 연구의 경우에도 단순한 압축, 캐싱 등을 이용하여 문제를 해결하고자 했을 뿐 근본적인 성능 개선책을 제시하지 못하고 있다 [1,2,3].

따라서 본 논문에서는 무선 인터넷의 여러 문제점들과 요구 사항을 해결하기 위해 모바일 환경을 위한 웹 서비스 전송 시스템을 제안한다. 제안된 시스템은 HTML/WML 변환기, SOAP 메시지 처리기로 구성한다. 모바일 환경을 위한 웹 서비스 시스템 구현을 위한 HTML/WML 변환기는 먼저 무선

웹 서비스 통신을 위해서는 기존의 웹에 존재하는 HTML 데이터를 무선 통신 데이터 형식인 WSDL 데이터 형식으로 변환한 다음, HTML 문서 분석을 통해 불필요한 정보를 제거하고 필요한 정보를 추출하여 WML 문서로 변환하는 역할을 담당한다. SOAP 메시지 처리기는 부가적인 처리 시간을 요구하는 웹 서블릿 컨테이너를 제거하고 SOAP 메시지를 처리 시간을 최소화하도록 설계한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구로서 웹 서비스와 모바일 서비스에 관련된 기술을 요약·정리하고 3장에서는 제안된 웹 서비스 전송 시스템의 구현 방법을 기술한다. 4장에서는 실험 및 분석에 대하여 기술하고, 마지막으로 5장에서는 결론과 향후 연구 방향에 대하여 기술한다.

## II. 관련 연구

### 2.1 무선 인터넷

일반적으로 웹서비스를 위한 구성 요소는 웹 서비스 소비자, 웹 서비스 중개자, 웹 서비스 제공자로 구성된다. SOAP(Simple Object Access Protocol)은 웹 서비스 사용을 위한 메시지 전송 프로토콜이고 웹 서비스에 대한 요청과 응답에 사용되는 메시지 형식을 정의한다. WSDL(Web Services Description Language)은 웹 서비스의 인터페이스를 기술하고, UDDI(Universal Description, Discovery, and Integration)는 웹 서비스 게시 및 질차를 규정한다. 웹 서비스는 XML 기반의 세 가지 대표적인 표준을 이용한다. XML 메시지를 웹을 통해 전송함으로써 서로 다른 컴퓨팅 환경에서 사용되는 모든 애플리케이션들이 상호작용하여 실행하는 동적 소프트웨어 시스템으로, 모바일 웹 서비스로 영역이 확장되어 WAP/WML 등의 기반 기술과 연동된다.

무선 인터넷 프로토콜 WAP은 그림 1과 같이 무선망과 웹을 연동하기 위하여 클라이언트와 웹 서버 간에 WAP 프록시(Proxy)인 WAP 게이트웨이를 둔다. WAP 게이트웨이는 WAP 기반 무선 인터넷의 핵심 요소로써 WAP 프로토콜과 인터넷의 HTTP 프로토콜을 중간에서 메시지를 변환하는 역할을 수행한다. 즉, WAP 프로토콜(WSP, WTP, WTLS, WDP)과 IP 기반의 패킷 네트워크 사이에서 데이터를 변환하는 중개자 역할을 수행한다. WAP 게이트웨이의 구현은 단일 서버 형태로 될 수도 있고 여러 서버에 분산되어 돌아가도록 구현할 수도 있다. 게이트웨이의 기능은 프로토콜 변환이 대표적인 것이다[4]. 유선 인터넷의 경우 HTTP를 통해 브라우저

로부터의 요구(Request)와 서버로부터의 응답(Response) 그리고, 서버에서 브라우저에게 전해지는 HTML 파일 등이 전송됨으로써 브라우저를 이용한 인터넷 서핑이 이루어진다.

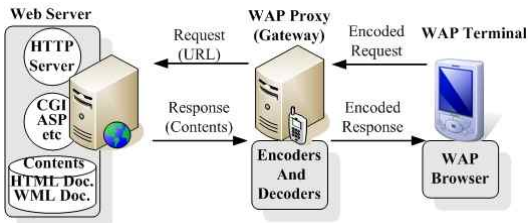


그림 1. WAP을 통한 무선 인터넷 연결 개념도  
Figure 1. Structure of wireless Internet

WAP 단말기에서 웹 연결을 요청하면 WAP 프로토콜을 사용하여 메시지를 압축하여 WAP 프록시로 보내지고 WAP 프록시는 메시지를 디코딩하고 URL을 사용하여 해당 웹 서버로 서비스 요청을 한다. 서비스 요청을 받은 웹 서버는 WML과 WML Script의 형태로 콘텐츠를 WAP 프록시에 전송하고 다시 WAP 프록시는 WAP 프로토콜에 맞게 메시지를 변환하여 WAP 단말기로 콘텐츠를 인코딩해서 전송한다.

WAP(wireless application protocol)에서 작동하는 무선 프로토콜 마크업(markup) 언어인 WML(wireless markup language)은 낮은 대역폭, 제한된 메모리와 저성능의 CPU를 가진 개인 이동 단말기에서 인터넷 서비스를 제공할 수 있도록 단말기 화면의 표현과 사용자와의 상호 작용을 표현하기 위해 XML(eXtensible Markup Language)의 문법을 따른다. 개인 이동 단말기와 서버 간에 정보를 전달하기 위해서 WAP은 WML을 이용하는데 이 언어는 XML 형식으로써 HTML과 많은 요소들을 공유한다. 그러나 출력 형태에 관하여 다소의 차이점들이 있는데 가장 중요한 차이점은 WML이 웹 내용을 설명하는 페이지를 한 벌의 카드들로 구조화한다는 것이다. 여기서 한 벌은 하나의 HTML 문서에 대응되는 개념으로 하나의 WML 문서를 말하며 한번에 전송 가능한 크기를 넘을 수 없다. 한 벌은 여러 개의 카드로 이루어져 있으며 각 카드는 보통 무선 단말기에 나타나는 하나의 화면 정보를 의미한다.

한 벌에 있는 여러 카드들을 보낼 때 클라이언트 측에서 전체 카탈로그를 다운로드함으로써 프로세싱 속도를 높이도록 한다. 일시에 많은 카드들을 보내느냐 아니면 적게 보내느냐는 것은 보내는 내용의 크기에 의해 결정된다. 즉 초기에 여러 카드들을 보내면 로딩 시간이 길어지고 보다 적게 보낼 때는 검색 시간이 길어지기 때문에 상황에 따라 균형을 맞추도록 한다.

## 2.2 기존의 무선 인터넷 프록시 서버

무선 프록시는 캐싱(Caching) 압축, 대용량 트래픽에 대한 확장성을 고려해야 한다. 기존의 프록시들은 기본적으로 캐싱과 압축 기능을 제공하고 있다. 그러나 확장성의 관점과 구조적인 관점에서 무선 프록시 간의 개별적 특성에 따라 차이점을 가지고 있다. 먼저 확장성의 관점에서는 일부는 고려하고 있는 반면에 대부분은 고려하고 있지 않다. 구조적인 관점에서도 캐싱과 압축기능이 대부분 하나의 호스트에 통합되어 있지만 일부 연구에서는 다른 호스트로 분리되어있다[1].

WebExpress[8]는 일반적으로 사용자 요청에 대한 응답이 많은 부분에서 비슷하다는 점을 이용하여 차이가 나는 부분만을 전송하는 방식을 사용하였고 TranSend[9]는 대용량 트래픽에 대한 확장성을 고려하여 클러스터링으로 구현하였다. Class-based Proxy[10]는 사용자 이동 각각의 화면에 따라 요청 이미지의 압축 정도를 조절하는 방식을 사용하였다. 또한, 표준 웹 서비스나 웹 어플리케이션 서버들의 경우에 SOAP 메시지를 처리하기 위해서는 웹 서블릿 컨테이너를 필요로 하는 데, 이것은 부가적인 작업이 필요로 한다는 문제점과 기존의 아파치 웹 서버의 경우에 동적 문서가 많아 질수록 성능이 저하되는 문제점을 해결하기 위해서 모바일 웹 서비스를 위한 SOAP 메시지 처리가 연구되었다[11]. 따라서 본 논문에서는 SOAP 메시지 처리를 기반으로 동작하는 HTML/WML 변환기를 구현함으로써 모바일 환경에 적합한 웹 서비스 전송 시스템을 구축하고자 한다.

## III. 제안된 웹 서비스 전송 시스템

본 논문에서는 제안된 모바일 디바이스를 위한 웹 서비스 전송 시스템은 HTML/WML 변환기, WSDL 변환기, SOAP 메시지 처리기로 구성된다. 그림 2는 제안한 모바일 웹서비스 시스템의 구성을 나타낸다.

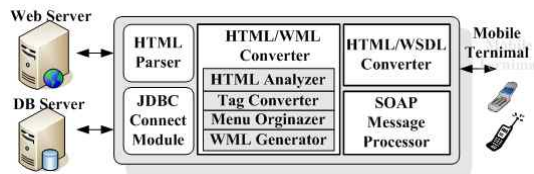


그림 2 제안된 모바일 웹 서비스 시스템  
Figure 2. A Proposed Mobile Web Service System

모바일 디바이스를 위한 웹 서비스 시스템 구현을 위한 HTML/WML 변환기는 HTML 문서 분석을 통해 불필요한 정보를 제거하고 필요한 정보를 추출하여 WML 문서로 변환하는 역할을 담당한다. 그리고 WSDL 변환기는 HTML/XML 메시지를 문법적으로 분석하여 WSDL 파일로 자동으로 변환함으로써 시스템의 부하를 감소시키는 역할을 담당한다. 마지막으로 SOAP 메시지 처리기는 부가적인 처리 시간을 요구하는 웹 서블릿 컨테이너를 제거하고 SOAP 메시지를 처리 시간을 최소화하도록 설계한다.

### 3.1 HTML/WML 변환기

기존의 유선 인터넷 환경에서 구축한 방대한 HTML 정보를 별도의 WML 사이트를 구축하지 않고 모바일 인터넷 서비스를 제공하도록 WML 문서로 자동으로 변환하는 HTML/WML 변환기를 설계한다. HTML/WML 변환기는 무선 웹 서비스 통신을 위해서는 기존의 웹에 존재하는 HTML 데이터를 무선 통신 데이터 형식인 WSDL 데이터 형식으로 변환한다. 이 변환기는 세 개의 하위 모듈로 구성되는데 재구성하기 위한 규칙을 저장하는 규칙 스크립트(rule script), 스크립트 규칙과 클라이언트 정보를 사용하여 콘텐츠를 재구성하는 스크립트 엔진(script engine), HTML 분석 및 태그 변환기, 메뉴 구성기, WML 문서 재구성기 구성한다.

그림 3는 클라이언트가 서버에 WSDL을 요청하는 과정을 나타낸다. 클라이언트가 HTTP를 통해서 웹서버에 접속하면 클라이언트의 요청이 WSDL 요청으로 확인되면 클라이언트의 요청을 분석하고 WSDL을 생성하기 위하여 HTML/XML 파서를 이용한다.

HTTP를 사용하는 클라이언트의 XML 웹서비스 메시지에 대한 WSDL 정보를 생성하는 과정은 다음과 같다.

먼저 응답 헤더 정보를 생성하고 WSDL 정보를 얻도록 한다. 그리고 새로운 XML 문서 정보를 생성한다. WSDL에 필요한 엘리먼트를 생성하고 WSDL 모든 서비스, 바인딩에 필요한 엘리먼트를 추가한다. 바인딩의 VARIABLE에 필요한 complexType를 추가하고, 생성한 XML 문서에 element를 추가한다. 이렇게 WSDL을 생성하고 WSDL이 요구하는 정보를 서버로부터 전송받아서 클라이언트에게 생성된 결과를 전송한 다음 종료한다.

HTML 분석 및 태그 변환기는 HTML로 작성된 정보들 중에서 필요한 정보만을 선택하여 WML 문서로 재구성하는 역할을 담당한다. HTML로 작성된 정보를 WML 문서로 변환하여 재구성을 위하여 고려해야할 사항은 WML 문서는 시

작 및 종료 태그가 반드시 존재하지만 HTML 문서는 시작 및 종료 태그가 생략되는 경우도 존재하기 때문에 HTML 문서를 WML 문서로 변환할 때 고려한다.

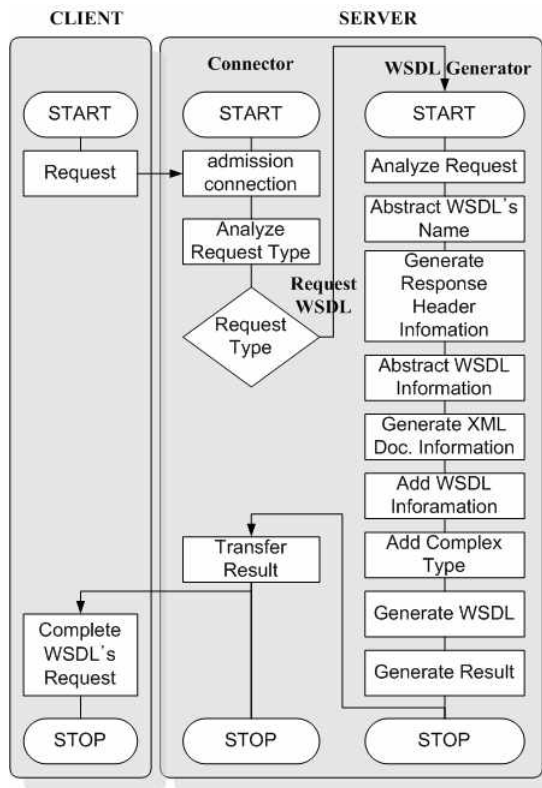


그림 3. WSDL 요청 과정  
Figure 3. WSDL request process

둘째, HTML 문서에서 사용되는 태그들 중에서 <HTML></HTML>, <HEAD></HEAD>, <BODY></BODY>, <FRAMESET></FRAMESET> 태그만을 고려한다. HTML 문서가 프레임들로 구성된 경우에는 각각의 프레임을 모두 병합하여 하나의 문서로 생성한다. 또한, HTML 문서는 플래시 파일, 자바스크립트, 애플릿, 주석, 스타일 등과 같이 WML 문서로 변환할 때 처리하기 어렵거나 불필요한 사항들은 제거한다.

셋째, HTML에서 사용하는 태그를 WML에서 모두 지원하지 않는다. 조사한 결과를 보면 공통적으로 존재하는 태그는 <head>를 비롯하여 22개, WML에만 존재하는 태그는 <wml>를 비롯하여 14개, HTML의 태그는 <body>를 포함

하여 69개가 존재한다. 태그뿐만 아니라 속성을 비교한 결과도 동일한 결과를 나타낸다.

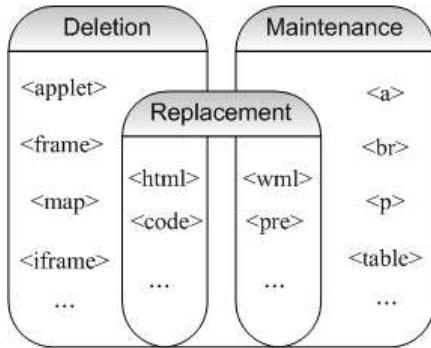


그림 4. HTML-WML 변환 규칙  
Fig 4. HTML-WML translation rules

삭제(deletion)에 해당하는 태그들은 HTML에 존재하는 태그를 WML에서 지원하지 않거나 모바일 디바이스에서 지원하지 않는 태그들로 모바일 디바이스에서 지원하지 않는 <applet> 및 <object>이다. 또한 WML 문서로 변환할 때 태그가 일치하지는 않지만 유사한 기능을 가진 WML 태그로 대체(replacement)하는 태그들로 <form>과 관련된 태그가 존재한다. HTML 태그와 WML 태그가 일치하면 그대로 사용(maintenance)하도록 한다. 그러므로 이러한 분석 결과를 기반으로 HTML 태그가 WML 태그로 변환될 수 있도록 적절한 변환 규칙을 정의한다. 그림 4는 태그 변환기에서 사용하는 변환 규칙의 일부분을 나타낸다.

메뉴 구성기는 태그 분석 및 변환기를 거친 문서를 메뉴, 내용과 관련된 사용자 입력 부분으로 분리한다. 이것은 모바일 디바이스를 이용하여 사용자가 원하는 정보를 보다 빠르게 이용할 수 있도록 한다. 사용자가 HTML 문서의 정보를 요구했을 때 변환되어 가장 먼저 제공되는 WML 문서에서 메뉴, 내용, 사용자 입력 부분 중에서 사용자가 선택할 수 있도록 한다. 메뉴로 구성된 것은 링크로 설정되고 링크가 존재하는 것들을 모아서 메뉴로 구성한다.

WML 재구성기는 모바일 디바이스의 저장 용량의 한계와 작은 화면 크기를 고려하여 WML 문서를 작성해야 한다.

### 3.2 SOAP 메시지 처리기

일반적으로 웹 서비스 구현은 아파치 톱캣과 AXIS로 구성된다. 그러나 서블릿 엔진은 추가의 통신 포트를 사용하고 부가적으로 처리 시간을 요구한다. 따라서 본 논문에서는 서블릿 엔진을 사용하지 않고 직접 SOAP 요구 메시지와 응답 메시지를 처리하는 SOAP 메시지 처리기를 사용한다. 이 시스

템의 가장 큰 특징은 기존의 시스템과는 달리 톱캣을 사용하지 않는다는 점이다.

그림 5은 SOAP 메시지 처리 시스템의 실행 흐름도를 나타낸다. 클라이언트가 접근하는 URL은 WSDL을 통해 얻어진다. 클라이언트가 SOAP 형태의 XML 문서로 HTTP를 통해서 웹 서비스를 요청하기 위해서는 SOAP 메시지를 서버로 전달하면 서버는 클라이언트의 접속을 허락하고 클라이언트의 요구 사항을 분석한다.

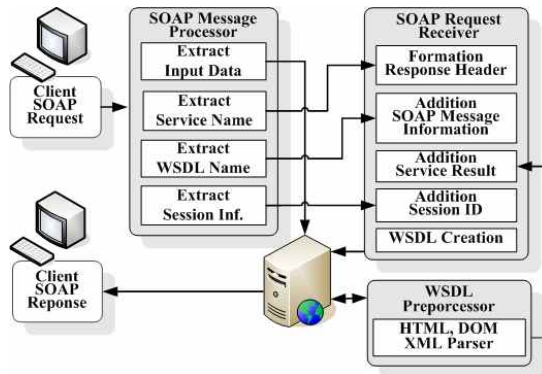


그림 5. SOAP 메시지 처리기의 실행 흐름도  
Figure 5. Execution flow of SOAP Message Processing

클라이언트가 웹 서비스를 요구하는 경우에는 SOAP 메시지 분석부로 SOAP 메시지를 전달한다. SOAP 메시지 분석부는 클라이언트의 요구를 분석하여 클라이언트가 요구한 서비스의 WSDL명을 추출한 후 전달된 SOAP 메시지를 분석한다. 분석한 정보에서 서비스명과 입력 값 등을 추출한 다음 서비스 처리부에 서비스 처리를 요청한다. 서비스 처리부에서 처리한 결과는 SOAP 메시지 생성부로 전달되어 응답 헤더 정보와 응답 SOAP 메시지에 대한 정보를 이용하여 응답 SOAP 메시지를 생성하고 서비스 처리 결과가 정상인지 확인한다. 만약 정상이면 서비스 결과 값과 세션 아이디를 추가하고 그렇지 않으면 실패 정보를 추가하고 SOAP 메시지 정보에서 SOAP 메시지를 생성하여 접속부로 전송하면 접속부는 다시 클라이언트로 전달한다.

## IV. 실험 및 성능 평가

### 4.1 실험 방법

본 논문에서 제안한 모바일 웹 서비스 시스템은 콘텐츠 웹

서버, 웹 서비스 클라이언트 그리고 모바일 클라이언트로 구성되고, SOAP 기반의 상호 작동에 의해 메시지 교환이 가능하게 구현하였다. 본 논문의 실험을 위해서 각각의 시스템을 구현하기 위해 환경은 다음과 같다. 서버는 CPU는 셀러론 1.2 GHz \* 2, Memory는 512 MByte, HDD : 100 GByte, NIC은 100 Mbps급이고 OS는 Windows 2000 서버 운영체제, JAVA JDK 1.4, IE 이다.

본 논문에서 제안한 모바일 환경을 위한 웹 서비스 시스템의 성능 평가를 위해서 톱캣 서블릿 컨테이너를 제거하고 JAVA 기반의 Apache-AXIS를 설치한 웹 서버(제안 시스템)와 AXIS 기반의 톱캣 서블릿 컨테이너를 포함하는 표준 웹 서비스 구조의 웹 서버(표준 시스템)를 비교한다. 두 시스템의 성능을 비교하기 위한 과정은 그림 6과 같다.

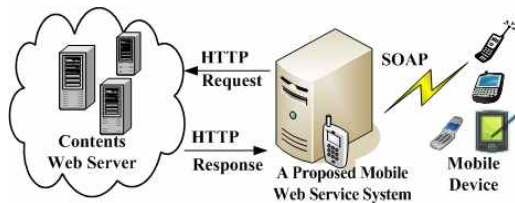


그림 6. 실험 과정  
Figure 6. Experiment Process

본 논문의 실험을 위하여 모바일 클라이언트의 서비스 요청은 시뮬레이션 프로그램에 의해 생성되고, 여러 개의 SOAP 요구 메시지를 전송한다. 모바일 클라이언트는 1초에서 10초까지 사이에 시간 간격으로 SOAP 요구를 반복적으로 요청한다. 연결이 종료되면 시뮬레이션 프로그램은 실험 시스템에 연결하기 위해 반복적인 연결 시도를 한다. 실험을 위한 동시 사용자는 200명으로 가정하고 모바일 클라이언트 요구는 4개의 클라이언트 프로그램에 의해 생성되고 각각 50개의 쓰레드를 생성한다.

본 논문에서 제안한 시스템은 SOAP 메시지로 구성된 클라이언트의 웹 서비스를 요구를 분석하고 콘텐츠 웹 서버에게 HTTP를 통하여 요구를 전송한다. 콘텐츠 웹 서버로부터 응답 메시지를 수신하면 WSDL를 생성하고 클라이언트 모바일 장치에 SOAP 응답 메시지를 전송한다. 그리고 실험에 사용할 콘텐츠 서버는 다음 사이트와 야후 코리아를 선택하고 사전과 주식 서비스를 이용한다. 각각의 서버의 타임아웃은 30초이다. 각각의 벤치마킹은 30분 동안 지속적으로 실행했다.

4.2 실험 결과

본 논문의 실험을 위하여 구현한 2개의 시스템을 비교하기 위하여 실험 시간 동안에 발생한 클라이언트의 전체 요구의 개수, 타임아웃 기간에 연결을 실패한 요구의 개수, 서버의 busy 상태로 인한 연결이 실패한 개수, 세션 연결 오류 개수, 서버 연결을 실패한 클라이언트 개수 그리고 클라이언트의 요청에 대하여 타임아웃이 초과된 횟수와 같은 정보를 수집한다. 본 논문의 실험은 모바일 웹서비스 시스템의 실행 평가에 초점을 맞췄다.

<표 1>의 실험은 서버와 모바일 디바이스 사이에 하나의 연결이 개설되면 서버에게 다수의 문자를 전송한다. 100개의 문자를 전송할 경우는 서버와 클라이언트 사이에 초기 TCP, HTTP 연결을 설정하고 그리고 SOAP 메시지 형태로 문자를 전송한다. 수신측은 네트워크 버퍼를 사용한다. 제안 시스템의 지연 시간이 적은 이유는 서버의 처리 시간 단축으로 인한 것이다.

<표 1> 문자 전송 지연 시간(ms)  
Table 1. Character Transfer Delay Time

구분	100문자	300문자	500문자
표준 시스템	168.2	168.3	168.5
제안 시스템	152.1	152.1	152.2

<표 2>의 실험 결과를 보면, 제안된 시스템은 모든 항목에서 성능 향상이 이루어졌다. 비록 제안된 시스템의 시험 시간이 표준 시스템의 시험 시간보다 짧았지만 전체 요청 횟수는 표준 시스템보다 많았고 타임아웃 개수는 표준 시스템보다 적었다. 제안 시스템에서의 초당 평균 요구 수는 17.89이지만 표준 시스템에서는 9.65이다. 특히, 그림 7에서 보는 바와 같이, 접속 실패 횟수, 세션 오류 횟수, 접속실패(연결 실패) 횟수 등을 비교해 보면 표준 시스템에 비해서 연결 에러의 발생의 훨씬 적은 것으로 나타났다.

<표 2> 동시 요청의 실험 결과  
Table 2. Experiment Results

테스트 정보	표준 시스템	제안 시스템
실험 시간	3,600	2,300
총 요구 횟수	34,752	41,138
접속 실패 횟수(서버 busy)	1,251	82
세션 오류 횟수	1,460	165
접속 실패연결 실패 횟수	278	0
접속 제한 시간 초과 횟수	7,854	576

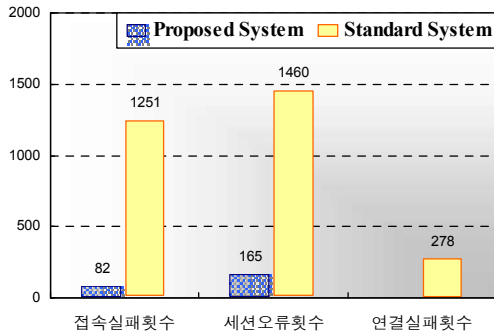


그림 7. 접속실패 횟수, 세션오류 횟수, 연결실패 횟수  
Figure 7. Connection Resuse, Connection Handshake Error, Connetion Trials

결과적으로, 본 논문에서 제안한 시스템이 서버릿 컨테이너를 사용한 표준 시스템보다 효율이 높다는 것을 의미한다. 그리고 웹서비스의 성능 향상은 전체 시스템의 성능 향상을 가져와 누적되는 연결 신호 처리의 향상을 의미한다. 결국 연결 요청 횟수가 많아질수록 기존의 웹서비스 시스템의 오류 횟수는 기하급수적으로 증가한다.

### V. 결론

최근 무선 통신이 대중화되면서 모바일 환경의 제약 사항에도 불구하고 사용자들은 이동 컴퓨팅 환경에서도 유선 인터넷과 유사한 서비스를 요구하고 있다. 그러나 기존에 구축된 정보와 동일한 정보를 모바일 디바이스를 통하여 서비스하기 위해서는 기존의 유선 인터넷 서비스를 기준으로 만들어진 콘텐츠를 그대로 활용하기가 어려움으로 모바일 환경에 맞도록 자동 변환하는 것이 필요하다. 또한, 무선 인터넷 전송 시스템을 설계할 때 중요한 점은 급속도로 증가하는 수요에 따라 대용량 트래픽을 처리할 수 있는 확장성을 고려해야 하는 데, 모바일 디바이스가 가지는 특성으로 인한 전체 웹 서비스 성능을 저하시키는 문제점이 발생할 수 있다는 점이다. 따라서 이러한 여러 문제점들과 요구 사항을 해결하기 위해서 본 논문에서는 모바일 디바이스를 위한 웹 서비스 전송 시스템을 제안하였다.

실험 결과 제안된 시스템은 기존의 표준 시스템에 비해 시스템의 지연시간이 줄어들음을 볼 수 있고, 총 요구 횟수, 접속 실패 횟수, 세션 오류 횟수, 접속 제한 시간 초과 횟수 등을 비교한 결과 모든 항목에서 우수한 결과를 얻었다. 그러므로 제안된 시스템은 새로운 비즈니스 패러다임의 변화에 따라 기

존에 구축된 많은 웹 서버 혹은 웹 애플리케이션 서버들을 이용하여 유·무선 통합 환경을 구축하는 데 유용하게 활용될 수 있으리라 기대된다.

### 참고문헌

- [1] 광후근, 정규식, "무선 인터넷 서버 클러스터 성능 개선", 한국정보과학회논문지 : 정보통신, Vol. 32, No. 3, pp. 415-426, 2005.
- [2] A. Fox, "A Framework For Separating Server Scalability and Availability From Internet Application Functionality," Ph. D. dissertation, U. C. Berkeley, 1998.
- [3] A. Maheshwari, A. Sharma, K. Ramamritham and P. Shenoy, "TranSquid: transcoding and caching proxy for heterogeneous e-commerce environments," Proceedings of 12th International Workshop on RIDE-2EC, IEEE, pp. 50-59, 2002.
- [4] W3C, Simple Object Access Protocol (SOAP) 1.1 (XML Protocol Working Group), <http://www.w3.org/2000/xp/Group/>
- [5] Patric Cauldwell and et al., Professional XML Web services, wrox, 2002.
- [6] W3C, WSDL Version 2.0(Web Services Description Working Group), <http://www.w3.org/2002/ws/desc>
- [7] 김인희, 모바일 분산 인터미디어리 기반 웹서비스를 위한 클라이언트/서버 메시지 처리 모델 설계, 서울여자대학교, 석사학위, 2005.
- [8] B. Housel, G. Samaras and D. Lindquist, "Web-Express: A client/intercept based system for optimizing web browsing in a wireless environment," Mobile Networks and Applications, ACM, pp. 419~431, 1998.
- [9] A. Fox, "A Framework For Separating Server Scalability and Availability From Internet Application Functionality," Ph. D. dissertation, U. C. Berkeley, 1998.
- [10] J. Lee, M. Kim, H. Youn, Y. Hahm and D. Lee, "Class-based proxy server for mobile computers," Proceedings of International Workshops on Parallel Processing, IEEE, pp. 559~566, 2000.

- [11] 김진일, 김용태, 박길철, “모바일 웹 서비스를 위한 고속 메시지 처리 시스템”, 한국컴퓨터정보학회 논문지, 제 3권 제 3호, 2008.5.

## 저 자 소 개



### 김진일

2000년 한남대 컴퓨터공학과 박사  
2006년 - 현재, 배재대 교양교육지원  
센터 교수  
<관심분야> 교육 공학, 모바일 웹 서  
비스, 네트워크, 퍼지이론,



### 김용태

1984년 한남대학교 계산통계학과  
1988년 숭실대학교 대학원 전자계산  
학과 공학석사  
2008년 2월 충북대학교 대학원 전자  
계산학과 이학박사  
2006.3 ~ 현재 한남대학교 멀티미  
디어학부 강의전담교수  
관심분야: 모바일 웹서비스, 정  
보보안, 센서 웹, 모바  
일 통신보안, 멀티미  
디어



### 박길철

1983년 한남대학교 계산통계학과  
1986년 숭실대학교 대학원 전자계산  
학과 공학석사  
1998년 성균관대학교 대학원 정보통  
신공학과 공학박사  
2006년 UTAS, Australia교환교수  
1998년 8월~ 현재 한남대학교 멀티  
미디어학부 교수  
(관심분야) multimedia and mobile  
communication, network security