

모바일 응용 서버에서의 다중 단말 장치간 세션 전이 구현

장철수⁰, 김성훈, 노명찬, 김중배

한국전자통신연구원 인터넷컴퓨팅연구부
{jangcs⁰, saint, mcroh, jjkim}@etri.re.kr

An Implementation of the Session Migration between Multiple Devices on Mobile Application Server

Choulsoo Jang⁰, Seonghoon Kim, Myeongchan Roh, Joongbae Kim

Internet Computing Department,
Electronics and Telecommunications Research Institute

요 약

유무선 통합의 인터넷 환경을 지원하는 모바일 응용 서버는 휴대폰, PDA, PC 등 다양한 단말 종류를 타겟으로 한번 저장된 동일한 콘텐츠를 단말기 종류에 상관없이 지원할 수 있다. 모바일 응용 서버로 구축된 비즈니스 업무를 이용하는 사용자는 무선 인터넷의 간헐적 단절성, 업무의 긴급성, 표현 장치 및 입력 장치의 적절성, 업무의 이동성 등을 고려하여 적절한 단말 장치를 임의로 선택하여 비즈니스 업무를 수행할 수 있다. 또한, 이러한 비즈니스 업무들은 단말기 종류가 변하더라도 이전의 단말기에서 처리하였던 내용을 계속 이어 받아 업무를 진행시켜야 한다. 단일 단말기를 이용할 때에 페이지 간의 작업 내용을 저장하는 방식으로 세션 객체를 이용하는 방식이 널리 사용된다. 그러나 세션 객체는 유선 인터넷상의 단일 단말기만을 대상으로 만들어진 것이기 때문에 단말기를 변경하는 경우에는 세션을 이용할 수 없는 단점을 갖고 있다. 본 논문에서는 다중 단말기간에 세션 전이를 가능하게 하여 세션 객체에 저장되어 있는 정보를 다중 단말기간에 공유하는 방법에 대한 설계 내용 및 구현 상황에 대해 논하겠다.

1. 서론

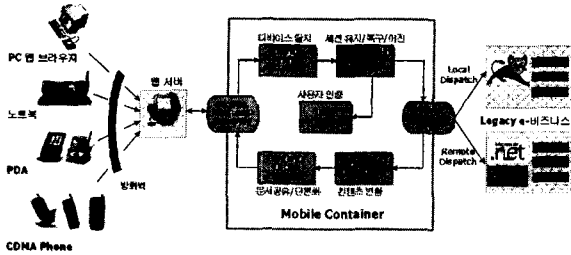
최근 이동통신 기술의 발전에 따라 개인용 컴퓨터뿐만 아니라 다양한 유형의 모바일 단말기까지 포함하는 유무선 통합 인터넷 서비스 환경으로의 변화가 급격히 일어나고 있다. 이러한 유무선 통합의 인터넷 환경을 지원하는 모바일 응용 서버는 휴대폰, PDA, PC 등 다양한 단말 종류를 타겟으로 한번 저장된 동일한 콘텐츠를 단말기 종류에 상관없이 지원할 수 있다. 또한, 모바일 응용 서버로 구축된 비즈니스 업무들은 무선 인터넷의 간헐적 단절성, 업무의 긴급성, 표현 및 입력 장치의 적절성, 이동성 등을 고려해 사용자에게 의해 단말기 종류를 변경하더라도 이전의 단말기에서 처리하였던 내용을 계속 이어 받아 업무를 진행시켜야 한다. 이를 위해 본 논문에서는 JSP 및 서블릿 등에서 단일 단말기 전용으로 페이지 간에 데이터 공유 기술에 사용되던 세션 객체를 모바일 응용 서버에서 다중 단말기간의 세션 전이가 가능하도록 확장하여 별도의 추가적인 프로그래밍 작업을 요구하지 않으면서도 다중 단말기간의 데이터를 공유하도록 하여, 자유롭게 단말기를 변경하여도 이전 단말기에서 완료되지 않은 업무 내용을 계속 전달 받아 처리할 수 있는 방법을 설명한다. 이 논문의 구성은 다음과 같다. 2 장에서는 유무선 통합의 모바일 응용 서버의 개념에 대해서 설명하고, 3

장에서는 세션 객체에 대해 살펴보고, 4 장에서는 모바일 응용 서버에서 다중 단말기간의 세션 전이에 대한 설계 내용을 설명하고, 5 장에서 결론을 맺는다.

2. 유무선 통합 모바일 응용 서버의 개념

이동통신 기술의 발전에 따라 비교적 넓은 대역폭을 갖는 셀룰러 폰, PDA, Post PC 등과 같은 모바일 단말기들이 광범위하게 보급되어 활용되고 있다. 이러한, 환경 변화에 따른 새로운 형태의 모바일 서비스의 등장은 필연적이다. 이를 반영이라도 하듯, 국내 무선 인터넷 서비스 업체의 수는 2001년 2400 개에서 2003년 13만 개로 기하급수적으로 증가하는 추세에 있으며, 이처럼 모바일 서비스에 대한 요구가 증가함에 따라 특정 기능 위주의 모바일 지원 기술이 개발되고 있지만, 무선 인터넷 서비스를 효과적으로 제공하려면 기존의 유선 인터넷을 기반으로 구축되어 운영중인 서비스를 재개발하지 않더라도 모바일 환경에 쉽게 적용되도록 하기 위한 유무선 통합 모바일 지원 기술 개발이 절대적으로 필요하다. 모바일 비즈니스 서비스는 기존의 인터넷 환경에서 기업 내부 비즈니스나 기업간 비즈니스 등 유선 인터넷 서비스를 포함하며, 무선 환경에서의 효과적인 서비스를 위해 무선 단말기의 제약성과 이동 통신망의 간헐적 단절성을 극복할 수 있는 비즈니스 서비스를 제공할 수 있다[1].

모바일 응용 서버는 기존 e-비즈니스 응용을 m-비즈니스 환경으로 쉽게 연동 확장할 수 있는 유무선 통합의 웹 컨테이너 시스템이다. 모바일 응용 서버는 클라이언트와 기존 웹 컨테이너 사이의 중간 Tier 역할을 하여 선언적 방법으로 JSP, 서블릿 등의 웹 컴포넌트들을 모바일 환경에 적응시키는 기능을 제공하고 있다.



<그림 1> 유무선 통합 모바일 응용 서버 구성도

이를 위해 모바일 응용 서버는 <그림 1>과 같이 국제 표준의 프로파일 정보접근 방법인 W3C의 CC/PP (Composite Capability/Preference Profiles)[2]를 지원하는 디바이스 프로파일 관리 모듈, 어노테이션에 의한 의미 기반 변환 및 프로파일에 의한 규칙 기반 변환을 제공하는 콘텐츠 변환 모듈, 콘텐츠의 공유 및 문서 단편화 기능을 제공하는 콘텐츠 캐쉬 모듈, 레거시 유저 관리자와 통합 API를 제공하며 휴대폰과 같은 사용자가 인증된 단말기의 경우 자동 로그인 기능을 제공하는 사용자 관리 모듈, 모바일 환경의 간헐적 단절 투명성을 보장하며 다중 단말기간 세션 정보 자동 이전 기능을 제공하는 세션 관리 모듈 등으로 구성되어 있다.

3. 세션 객체

HTTP(HyperText Transfer Protocol)는 처음 제정될 때부터 connection의 지속성이 없는 stateless protocol 이었다[3]. 즉, 연결이 끊어지면 이전의 클라이언트의 상태를 기억하는 방법이 불가능하다. 따라서 새로운 연결이 맺어지면 이 연결이 기존의 연결이 재 생성된 것인지, 새로운 클라이언트가 새로 맺은 연결인지를 확인할 길이 없다. 이것이 HTTP의 가장 큰 단점인데 이 단점을 극복하기 위해서 URL rewriting, Hidden form field 사용, Cookie 사용, 세션 객체의 방법이 고안되었다. 이중 세션 객체는 응용 서버에서 제공하는 메커니즘으로서 웹 어플리케이션에서 사용자의 상태를 손쉽게 확인할 수 있는 방법이다. 모든 클라이언트가 서비스에 접속하면, 각각의 클라이언트에게 유일한 세션 ID를 발급한다. 클라이언트는 발급받은 세션 ID를 cookie에 저장하거나, URL rewriting 기법을 사용하여 세션을 추적하도록 한다. 그렇게 되면 매 요청마다 서버에서는 세션 ID를 알 수 있고 이 세션 ID를 이용하여 사용자가 저장해놓은 data들을 찾게 된다.

Java의 서블릿 API에서는 이러한 세션의 개념을 javax.servlet.http.HttpSession 인터페이스를 이용하여 제공하고 있다[4]. 세션의 개념은 클라이언트의 요청에 연관되어 있는 것이므로, HttpServletRequest 인터페이스에서 getSession()이라는 메소드를 제공하고 있으며, 이를 통해 요청을 보낸 클라이언트와 연관되어 있는 세션 객체를 얻어올 수 있다. 얻어온 HttpSession 객체로부터 (name, value) 쌍으로 구성된 data들을 추출하거나 나중의 사용을 위해 새로운 data들을 추가할 수 있다. 물론, 이전에 처리하였던 업무의 내용을 데이터베이스등과 같이 영속적인 저장장치를 이용하여 저장할 수도 있지만 이러한 방법은 응용 개발자들에게 JDBC등을 이용한 추가적인 프로그래밍을 요구한다. 그러나 세션을 사용하는 경우에는 세션객체의 생성과정이 응용 개발자에게는 투명하게 이루어지므로 별다른 추가적인 프로그래밍 작업이 없이 간단히 세션에 data를 임시로 저장하여 사용하도록 할 수 있다. 즉, 세션의 추적을 위해서 URL rewriting이나 cookie의 생성, 설정, 삭제등과 같은 명시적인 작업을 전혀 할 필요가 없다.

4. 다중 단말기간 세션 전이

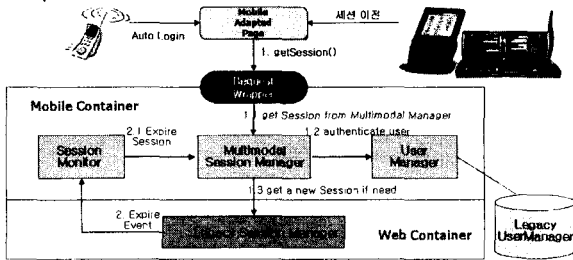
이와 같이 유용한 세션 객체는 유선상에서의 단일 단말기를 고려하여 만들어진 것이기 때문에 단말기를 변경한다거나 하는 경우에는 세션을 이용할 수 없는 단점을 갖고 있다. 이번 장에서는 이러한 세션 객체의 단점을 극복하여 다중 단말기간 세션 전이가 가능하도록 서블릿 컨테이너를 확장한 모바일 컨테이너의 내용을 살펴보겠다.

모바일 응용 서버를 이용하여 비즈니스를 처리하는 사용자는 특정 단말에 구매받지 않으면서 원하는 단말기 종류를 이용하여 비즈니스 서비스를 지원받을 수 있다. 즉, 긴급하고 이동성이 강조되는 업무에서는 화면표시 능력이 떨어지고 불편한 입력 수단을 갖고 있는 휴대폰으로도 간단한 메시지 확인이나 적은 양의 입력이 요구되는 단순 업무를 처리할 수 있도록 하고, 좀 더 세밀한 화면 조희가 요구되거나 많은 정보의 입력 작업이 요구되는 복잡한 비즈니스 업무를 위해서는 PC와 같이 풍부한 표시 및 입력 능력을 갖고 있는 단말기를 이용하여 업무를 처리한다. 뿐만 아니라, 이동중이라도 복잡한 업무이며 긴급히 처리하여야 할 업무라면 CDMA 모델이 내장된 PDA 등으로 업무를 처리할 수 있다.

이와 같이 다양한 단말기를 사용하는 경우 비즈니스 업무의 연속성을 위해서 사용자는 휴대폰이나 CDMA 모델이 내장된 PDA 등으로 긴급한 업무를 미리 처리한 후, 이전에 처리했던 업무 내용을 이어받아 시간적 여유가 있는 환경에서 PC등을 이용해 업무를 처리할 수 있거나 혹은 유선 PC 환경에서 미완의 처리 업무를 이어 받아서 이동 중에 처리할 수 있어야 한다. 이처럼 무선 인터넷의 간헐적인 단절성을 극복하고 비즈니스 업무의 연속성을 지원하기 위해서는 세션의 전이

방법이 필수적으로 요구된다.

모바일 응용 서버에서는 유무선 통합 인터넷 환경에서 필수적으로 요구되는 다중 단말간 세션 전이를 위해 <그림 2>와 같은 멀티모달 세션 관리 모듈을 지원한다.



<그림 2> 세션 전이가 가능한 세션 관리자 모듈

멀티 모달 세션 관리 모듈 중 대표적인 블록은 다음과 같다.

- **Request Wrapper** : `HttpServletRequest` 를 wrapping 하여 `HttpServletRequest` 로부터 세션 객체를 얻고자 할 때, 인터셉터 역할을 하여 Multi-modal Session Manager 로부터 세션을 얻어오도록 함.
- **Multi-modal Session Manager** : Multi-modal 세션에 참여하고 있는 사용자 ID 와 세션 ID 를 테이블 형태로 관리함.
- **User Manager** : Multi-modal 세션에 참여할 사용자에 대해 사용자 ID 와 암호를 이용한 인증 기능을 수행할 뿐만 아니라, 등록된 휴대전화번호에 대하여 자동적인 인증기능을 수행함.
- **Session Monitor** : 세션 객체에 저장되는 data 들이 클라이언트 대신 서버에 저장되기 때문에 일정 시간 이상 계속 유지하고 있으면 서버의 자원을 많이 소모하게 되므로 일정 시간이 지나면 세션을 자동적으로 소멸되도록 지원하고 있으며, 이러한 경우 Multi-modal 세션에 참여하고 있는 세션에 대한 Expire Event 를 받아, Multi-modal Session Manager 가 관리하는 세션 ID 를 삭제하도록 함

<그림 2>에서 보듯이 유무선 통합의 모바일 컨테이너는 기존의 서블릿 컨테이너를 확장하는 구조이다. 클라이언트는 기존의 JSP 나 서블릿 등의 페이지를 직접 호출하는 대신 Mobile Adapted Page 라는 가상의 페이지를 호출하고, 서블릿 컨테이너의 환경 설정파일에서 설정된 서블릿 매핑대로 클라이언트의 요청은 모바일 컨테이너 쪽으로 전달된다. 모바일 컨테이너는 클라이언트가 호출하는 실제의 페이지에 요청을 디스패치하기 전에, `HttpServletRequest` 객체의 형태로 전달된 원래의 클라이언트 요청을 wrapping 하는 객체를 Servlet API Version 2.3[5]에 추가된 `HttpServletRequestWrapper` 를 이용하여 만들고, 실제의 페이지를 서비스하는 서블릿 컨테이너에게는 원래의 request 객체대신에 wrapper 객체를

전달한다. 서블릿 컨테이너가 클라이언트가 요청한 실제 서비스를 위해 호출하는 JSP 및 서블릿은 전달받은 request 객체에게 `getSession()` 메소드를 호출하지만, 실제로는 wrapper 객체가 메소드를 가로채게 된다. `getSession()` 메소드를 호출받은 wrapper 객체는 로그인된 사용자와 연관된 세션이 있는 경우에는 해당 사용자가 이미 사용중인 세션 객체를 멀티모달 세션 관리자로부터 얻어서 JSP 및 서블릿에게 전달하여 사용할 수 있도록 한다. 만약, 아직 해당 사용자의 세션 객체가 들어가지 않은 경우에는 원래의 request 객체를 통해서 세션을 생성하도록 하고 생성된 세션 객체를 JSP 및 서블릿에 전달하여 사용하도록 하며, 생성된 세션 ID 와 세션 객체에 대한 참조를 멀티모달 세션 관리자에게 등록하여 관리한다.

세션을 wrapping 하는 `HttpServletRequestWrapper` 객체 또한 `HttpServletRequest` 를 구현한 객체이므로 서블릿 컨테이너가 실행하는 JSP 및 서블릿 입장에서는 어떠한 코드상의 변화도 없으며 따라서, 응용 개발자는 세션 전이에 필요한 추가적인 프로그래밍 작업 없이, 다중 단말기 간의 세션 전이를 제공받을 수 있다.

5. 결론

모바일 환경과 유선 인터넷 환경의 가장 큰 차이점은 연결의 불안정성이다. 사용자가 전화 음영 구역에 들어가거나, 네트워크의 불안정성 때문에 연결이 끊긴 경우에 재접속을 하다보면 이전에 web server 와 통신하며 생성된 session 정보들이 없어지는 경우가 빈번하다. 이를 위해 사용자마다 데이터를 저장하는 세션 객체를 전달시켜서 사용자와의 연결이 끊기더라도 연결라도 다시 접속을 하면 이전에 저장한 정보를 되살릴 수 있도록 하였으며 특히, 특정 단말기에서 비즈니스 업무를 처리하는 도중에 해당 네트워크의 이상 등으로 다른 네트워크를 지원하는 다른 단말기로 업무를 이관하여 처리하여도 이전까지 처리 내용이 세션에 그대로 보관되어 처리가 가능하도록 모바일 응용 서버를 구축하였다.

차후에는 클러스터드 환경으로 확장되는 모바일 응용서버에서 멀티 모달 세션 정보의 복제를 통한 클러스터링 지원 과제가 남아있다.

참고문헌

- [1] 김성훈, 장철수, 정승욱, 서범수, 노명찬, 박중기, 이경호, 김중배, 유무선 통합 모바일 응용 서버에 관한 연구, 정보과학회학술지, 제 20 권 1 호, 2002
- [2] <http://www.w3.org/Mobile/CCPP>, "Composite Capabilities / Preferences Profile Working Group Public Home Page".
- [3] T. Berners Lee, R. T. Fielding and H. Frystyk, "Hypertext Transfer Protocol-HTTP/1.1", 1997.
- [4] Subrahmanyam 외 17 인, "Professional Java Server Programming J2EE Edition", Wrox, 2000.
- [5] Danny Coward, "Java Servlet Specification Version 2.3", Sun Microsystems, 2001.