

모바일 신디케이션

왕결*, 최상민, 신민철, 박진원, 황기태
한성대학교 컴퓨터공학과
wangjie@hansung.ac.kr*

A Study on Mobile Syndication

Wang jie, Sang Min Choi, Min Cheul Shin, Jin Won Park, Kitae Hwang
Dept of Computer Engineering, Hansung University

요 약

새로운 웹 정보를 얻기 위해 웹 사이트에 빈번히 방문하는 것은 시간 낭비이며 비효율적이다. 이러한 문제를 해결하기 위해 사용자가 웹 사이트의 정보를 인터넷으로 정기 구독하는 웹 신디케이션 기술이 최근에 도입되었다. 이를 통해 사용자는 웹 사이트에 대한 낭비적인 네비게이션을 하지 않고도 비교적 빠른 시간 내에 최신 정보를 획득할 수 있다. 그러나 현재 웹 신디케이션 기술은 주로 PC 환경에서 구현 활용되고 있기 때문에 이동성에 대한 고려가 되어 있지 않다. 본 논문에서는 웹 신디케이션 기술을 모바일 단말기와 모바일 인터넷으로 확장하는 시스템을 제안한다.

1. 서론

웹에서 정보를 검색하는 전통적인 방식에서는 정보가 갱신되었는지 새로운 정보가 올라왔는지를 알기 위해 웹 사이트에 직접 방문하고 웹 사이트로 이동한다. 웹 신디케이션 기술을 이용하면 사용자는 웹 사이트들을 직접 방문하지 않고 클라이언트 컴퓨터에 채널 구독(subscribe)을 신청하면 클라이언트에 설치된 RSS(Really Simple Syndication) 구독기와 같은 응용 프로그램을 통해 갱신된 정보들을 주기적으로 획득할 수 있다. 이를 통해 사용자가 시간을 낭비하지 않고도 여러 사이트로부터 많은 정보를 수집하는 이점을 누릴 수 있다[1,4].

기존 웹 신디케이션 기술은 PC 환경에 사용될 수 있도록 개발되었으므로 이동성이 보장 되지 않는다. 또한 응용 프로그램 미실행이나 클라이언트 PC를 꺼 둔 상태에서는 갱신된 정보를 획득하지 못하고 지나치게 되는 단점이 있다[2,5].

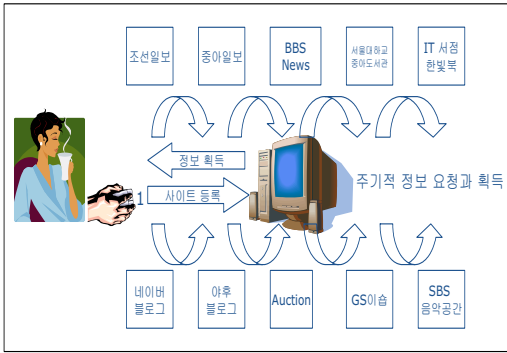
본 논문은 모바일 인터넷 사용자들이 늘어가고 있으며 언제 어디서나 정보를 획득하고자 하는 정보 시대에, 모바일 인터넷을 이용하여 모바일 단말기에서 신디케이션 기술을 구현하는데 초점을 맞추고 모바일 웹 신디케이션 시스템을 설계하였다. 모바일 단말기의 사용은 통신 비용을 초래하는 어려움이 있다. 너무 자주 혹은 불필요하는 RSS 채널을 받기 위해 무선 인터넷을 사용하게 되면 사용자의 비용을 초래하므로 단순히 PC 상에서 개발된 RSS 구독기를 모바일 용으로 포팅하는 것 외에 무선 인터넷 비용을 최소화하는 대비책이 필요하다. 본 논문에서는 모바일 웹 신디케이션 프록시 서버를 도입하였으며 모바일 단

말기가 직접 RSS 사이트에 접속하여 RSS 채널을 가져오는 부담을 줄이고 프록시 서버를 통해 간접적으로 RSS 채널을 받음으로써 통신 비용을 줄인다. 프록시 서버는 통신 비용을 줄이기 위해 불필요한 채널의 아이템을 줄이는 등의 지능적인 알고리즘을 구현한다. 본 논문에서는 이러한 지능적인 알고리즘에 대해서는 다루지 않으며 모바일 신디케이션 시스템을 제안하고 프록시와 단말기의 설계 및 구현 사례에 대해서만 다룬다.

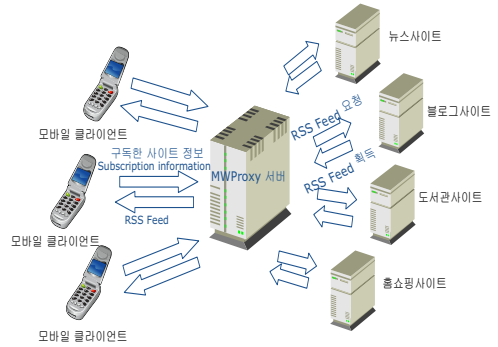
2. 웹 신디케이션

웹 신디케이션은 웹 사이트 내 요약된 정보인 메타데이터를 생성하고 이를 수집하는 응용 프로그램을 통해 실시간으로 사용자들에게 최신 정보를 알려주는 기술이며 그림 1와 같이 구성된다. 채널은 요약된 웹 사이트 내용을 사용자에게 전달하는 데이터이며 피드라고도 불리고 XML로 구성된다. 웹 신디케이션은 뉴스 서비스에서 시작되었지만 블로그나 뉴스, 날씨, 도서관, 교통정보, 홈쇼핑, 금융정보, 영화, 음악 등 활용 분야가 점점 늘어가고 있다.

RSS는 현재 웹 신디케이션에서 가장 많이 사용하고 있는 채널 형식이다[3]. 현재 주로 Userland사의 RSS 2.0이 사용되고 있으며 본 논문에서도 RSS 2.0 규격을 적용한다. 하나의 RSS 파일은 하나의 정보 채널(channel)로 구성되며 한 채널은 여러 개의 아이템(item)으로 구성된다. 그림 2는 조선일보의 웹 콘텐츠에 대한 RSS 채널의 예이다.



(그림 1) 웹 신디케이션 사용시 정보 수집 방식



(그림 3) 시스템 모델

```

<?xml version="1.0" encoding="euc-kr" ?>
<rss version="2.0" xmlns:dc="http://purl.org/dc/elements/1.1/">
  <channel>
    <title>english.chosun.com : Total</title>
    <link>http://english.chosun.com</link>
    <description>english.chosun.com RSS Service | Total</description>
    <lastBuildDate>2008.05.28 10:25:45.492</lastBuildDate>
    <copyright>Copyright (c) 2004 chosun.com All rights reserved</copyright>
    <webMaster>letters@chosun.com</webMaster>
    <language>ko</language>
  </channel>
  <item>
    <title>english.chosun.com</title>
    <url>http://image.chosun.com/img/200305/logo_rss.gif</url>
    <description>Chosun.com - English Edition</description>
  </item>
  <item>
    <title>
      <![CDATA[ Ties With China Require Delicate Handling ]]>
    </title>
    <link>http://english.chosun.com/w21data/html/news/200805/200805280026.html</link>
    <dc:date>2008-05-28 10:25:45</dc:date>
    <author />
  </item>
  <item>
    <title>
      <![CDATA[ Discontent Growing as Gov't Marks 100 Days in Office ]]>
    </title>
    <link>http://english.chosun.com/w21data/html/news/200805/200805280025.html</link>
    <dc:date>2008-05-28 09:57:29</dc:date>
    <author>(englishnews@chosun.com)</author>
  </item>
</rss>
    
```

(그림 2) 조선일보의 웹 콘텐츠의 RSS 채널의 예

3. 모바일 신디케이션 시스템

3.1 시스템 개요

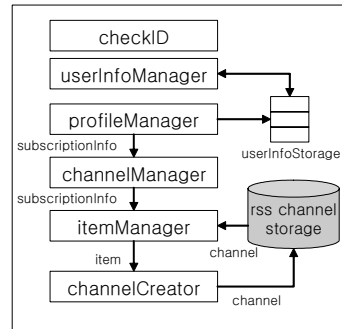
전체 시스템은 그림 3과 같이 모바일 단말기를 사용하는 모바일 클라이언트, RSS 채널 정보를 클라이언트 대신 관리하는 프록시 서버(Mobile Web Syndication Proxy : MWProxy)로 구성되며, 웹 콘텐츠와 RSS 채널을 제공하는 많은 사이트들이 존재한다. 모바일 사용자가 모바일 단말기에서 관심 있는 웹 사이트에 대한 정기 구독을 요청하면 모바일 단말기의 클라이언트 소프트웨어는 이 가입 정보(subscription)를 MWProxy에게 요청한다. MWProxy는 단말기와 정기 구독 사이트의 정보를 저장하여 기억하고, 해당하는 사이트에서 주기적으로 채널을 수집하여 사용자 설정에 따라서 최소화하여 모바일 장치로 전송한다.

3.2 시스템 구조

3.2.1 MWProxy 구조

MWProxy 서버의 구조는 그림 4와 같다. checkID 모듈은 클라이언트 가입 요청 정보를 분석하고 처음 가입하는 경우 ID를 부여하여 이 클라이언트를 위한 채널 저장 공간을 생성한다. profileManager 모듈은 모바일 클라이언트로부터 받은 사용자의 정기 구독 정보를 분석하여 userInfoStorage에 저장한다. channelManager는 profileManager로부터 받은 채널 정기 구독 요청 정보를 채널 업데이트 주기에 따라 구분하여 저장한다. 채널의 주기는 시간별, 분별, 날짜별로 나누고 해당하는 각 주기 별로 하나씩 3 개의 스레드를 두어 정기 구독된 사이트로부터의 RSS 채널을 가지고 와서 갱신하도록 하였다.

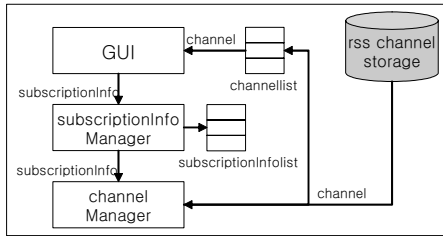
itemManager는 웹 사이트로부터 받은 RSS 채널을 아이템별로 파싱하고 channelManager로부터 받은 채널 설정 정보에 따른 맞는 아이템을 추출한다. 추출된 아이템과 MWProxy 서버에 저장 되어 있는 이전의 아이템과 비교하여 갱신된 아이템만을 추출하고 이들을 하나의 채널로 다시 묶는 channelCreator에게 전송한다. 그리고 새롭게 구성된 채널은 모바일 단말기의 클라이언트 소프트웨어로 전송된다.



(그림 4) MWProxy 서버 구조도

3.2.2 모바일 클라이언트 구조

모바일 클라이언트 소프트웨어는 모바일 단말기에서 실행되는 프로그램으로서 주요 기능은 사용자로부터 정기 구독 정보를 받아 MWProxy에게 등록하는 기능과 사용자가 설정된 주기로 MWProxy에게 채널을 받아오는 기능이다. 모바일 클라이언트 소프트웨어의 구조는 그림 5와 같다. UIManager는 사용자와의 입출력을 위한 인터페이스 모듈이며 사용자가 입력한 정기 구독 정보는 subscriptionInfoManager에서 관리된다. channelManager는 스크롤을 이용하여 정기 구독을 위해 주기적으로 MWProxy에게 채널을 요청하는 기능을 수행한다. 그리고 사용자가 채널 정보를 보고자 할 때 스토리지로부터 읽어서 전달한다.



(그림 5) 모바일 클라이언트 구조도

4. 테스트 시스템 구현 및 시연

4.1 구현

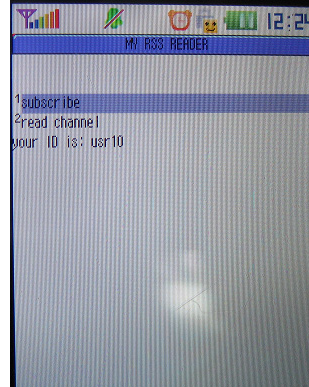
본 논문에서는 모바일 단말기 클라이언트로 위피 플랫폼 상에서 작동하는 위피 기반의 jlet 응용 프로그램으로 작성하였다. MWProxy는 리눅스 상에서 톰캣을 설치하고 자바 서버릿(servlet)으로 작성하였다. 클라이언트 소프트웨어는 아로마 위피 에뮬레이터를 이용하여 PC에서 테스트를 수행한 후, 위피 플랫폼을 탑재한 모토롤라 휴대폰으로 실제 테스트를 수행하였다.

4.2 시연

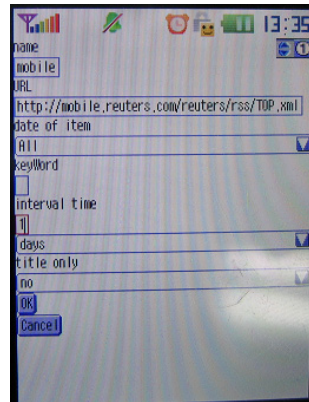
본 논문에서 테스트로 개발한 클라이언트 소프트웨어를 모토롤라 휴대폰에 탑재하여 실제 신디케이션을 수행을 테스트 하였다. 그림 6부터 그림 11까지는 작동되는 휴대폰의 화면을 캡처한 것이다.

그림 6은 사용자가 모바일 클라이언트에게 정기 구독 가입 요청 메뉴를 선택하는 화면이며, 그림 7은 이 메뉴가 작동하여 사용자로부터 정기 구독 사이트의 채널을 등록 하도록 하는 화면이다. 그림 8은 클라이언트가 MWProxy에 정기 구독 신청을 하고 곧 바로 채널을 획득한 후 지금까지 정기 구독 신청을 한 채널의 목록을 출력한 화면이다. 그림 9는 사용자가 보고자 하는 채널을 선택하여 채널의 아이템을 열람하는 화면이며 그림 10은 이 아이템 중에서 한 아이템을 선택하여 아이템에 관련된 간단한 정

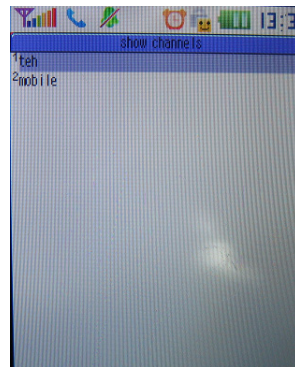
보를 얻는 화면이며 그림 11은 이 아이템을 사이트로부터 가지고 와서 웹 브라우저를 통해 실제 뉴스 정보를 읽고 있는 화면이다.



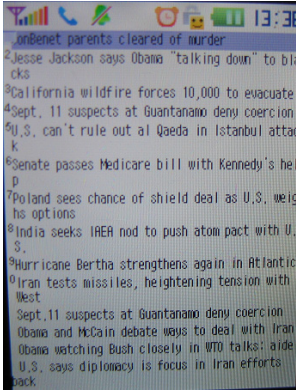
(그림 6) 가입 요청



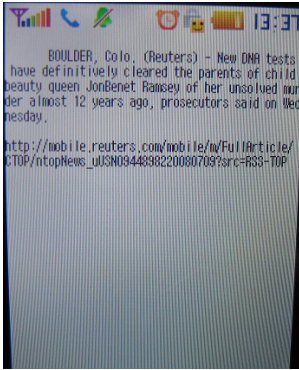
(그림 7) 채널 등록



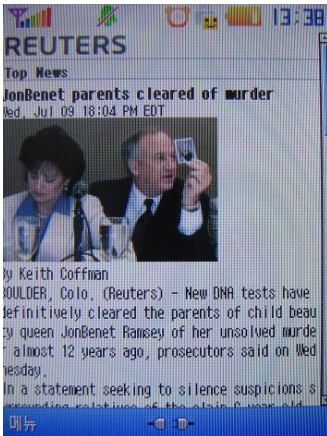
(그림 8) 획득한 채널 목록



(그림 9) 선택한 채널 열람



(그림 10) 선택한 아이템 열람



(그림 11) 선택한 아이템 해당 사이트

5 결론

본 논문에서는 PC 기반에서 주로 사용되는 웹 신디케이션을 모바일 인터넷에서도 사용할 수 있도록 하기 위해 모바일 신디케이션 모델을 설계하고 테스트한 사례를 소개하였다. 모바일 단말기에서 신디케이션의 대상 사이트를 직접 방문하여 채널 정보를 가져오는 방식은 모바일 통신 비용에 대한 큰 부담이 있기 때문에 프록시를 두어 이것을 대행하게 하고 채널의 정보량을 줄여서 모바일 단말기에 전송함으로써 보다 효율적인 상황을 만들어 낸다. 본 논문에서는 위피 기반의 모바일 클라이언트 응용 프로그램을 작성하고 MWProxy라고 명명한 프록시 서버를 구현하여 신디케이션을 테스트하였다. 현재는 성능을 평가하고 있으며, 단말기와 프록시 서버 사이의 통신량을 줄이는 방법들을 테스트하고 있다.

참고문헌

- [1] Reusch P.J.A., Stoll, B., Schulwandt T., Serwatowski P., "New Communication Concepts Based upon Advanced RSS Feeds", The Third Workshop 2005 IEEE Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, pages 699-702, Sept. 2005
- [2] Chmielewski, D., Gongzhu Hu, "A distributed platform for archiving and retrieving RSS feeds", Computer and Information Science, 2005
- [3] <http://validator.w3.org/feed/docs/rss2.html>
- [4] Baumgart A.S., Knapp H.; Suetterlin P., Schader M., "A Profile-Based Peer-to-Peer RSS Information Distribution", Wireless Pervasive Computing, 2007
- [5] De Sutter R., Lerouge S., De Schrijver D., Van de Walle R., "Enhancing RSS Feeds: Eliminating Overhead through Binary Encoding", Information Technology and Applications, 2005