가변적 웹 데이터 추출을 위한 재사용 가능한 모바일 애플리케이션 개발에 관한 연구

임성호[0] 박상원
한국외국어대학교 정보통신공학과
shlim2000@naver.com, swpark@hufs.ac.kr

The study of development of reusable mobile application to extract variable web data

Seongho Lim[0] Sangwon Park
Department of Information and Communication Engineering,
Hankuk University of Foreign Studies

1. 서 론

현대 사회의 필수품으로 자리 잡은 스마트폰은 생활필수품으로 자리 잡고 있다. 최근 스마트 폰에는 본래 목적인 통화기능을 넘어서 음악, TV 및 동영상 감상, 게임 등의 부가적인 기능들이 탑재되고 있다. 스마트폰에서 구동되는 모바일 애플리케이션은 모바일 환경이라는 특성으로 단말의 배터리, 메모리, 성능 등의 자원 이용에 제한이 있다. 제한을 극복하기 위하여 처리는 서버에 위임하여 결과를 XML 또는 HTML 데이터로 가공하고 스마트폰은 가공된 데이터의 결과 값만을 취하는 방법을 사용한다. 또는 서비스 중인 기존 웹 페이지에 존재하는 태그의 특정 값을 추출하여 사용한다. 결과 값만을 추출해 사용하면 스마트폰 자체의 정보 처리 능력을 사용하는 것이 아니기 때문에 서버 접속과 파싱에 필요한 자원 외에 추가적인 자원이 필요하지 않아 효율적인 자원관리를 할 수 있다. 따라서 현재 배포되고 있는 많은 모바일 애플리케이션들은 웹에서 실시간으로 정보를 스크래핑(Scraping)하는 방식을 사용한다. 하지만 일반적으로 사용되는 웹 페이지에서 정보를 추출하는 웹 스크래핑의 경우 서버의 주소가 변경되거나 웹 페이지 내부에 정보 위치가 변경되었을 경우 애플리케이션을 다시 작성해야 하는 문제가 있다. 또한 원하는 정보가 다른 도메인에 산재해 있는 경우 추출할 정보에 대한 위치정보가 모두 다르기 때문에 도메인별로 파서의 동작을 다르게 구현하여야 한다. 웹 스크래핑은 위 두 가지의 문제점 때문에 애플리케이션 개발과 유지보수에 어려움이 따른다. 첫 번째 문제점의 예로 서울 버스 어플리케이션을 들 수 있다. 현재 앱 스토어에서 판매되고 있는 서울 버스 애플리케이션의 경우 현재 버스의 위치정보를 획득하기 위하여 버스 정보 시스템을 사용한다. 만약 버스 정보 시스템의 웹 페이지 형태가 바뀐다면, 웹 페이지 내의 추출하고자 하는 버스 정보의 위치가 변한다. 따라서 기존의 파서를 그대로 사용 할 경우 원치 않은 정보를 추출하게 되므로 파서를 수정하여 재배포 하여야 한다. 두 번째 문제점의 예로는 본 논문에서 구현 할 도서관 잔여좌석 조회 애플리케이션을 들 수 있다. 도서관 잔여좌석 정보는 도서관을 운영하는 기관마다 따로 제공하고 있기 때문에 하나의 웹 사이트에 존재하지 않고 분산되어 있으며 웹 페이지의 형태도 모두 다르다. 이러한 이유로 도서관 잔여좌석 조회 서비스 애플리케이션을 개발한다면, 도서관 별로 모두 다른 웹 페이지의 각각 다른 위치에 있는 잔여좌석 정보를 파싱하기 위해 도서관 별 파서를 모두 따로 설계해야 한다. 따라서 본 논문에서는 첫 번째 문제점인 분산정보의 추출을 위한 경로식 사용과, 두 번째 문제점인 정보 위치 변화에 대한 대응으로 메타 데이터를 서버에 저장하고 애플리케이션이 메타데이터 서버에 접속하여 원하는 정보를 추출하는 방법을 제안한다.

2.1 분산 정보의 추출

그림 1 웹에 분산되어 있는 도서관 정보

웹의 많은 정보가 그림 1의 도서관 잔여좌석 정보처럼 웹 페이지의 한 곳에 모여있는 경우 보다는 도메인이 다른 분산된 곳에 각기 다른 형태로 존재하는 경우가 많다. 정보가 여러 웹 페이지에 분산되어 있는 경우 각 정보를 추출하기 위해 파서를 웹 페이지 수만큼 구현하여야 한다. 이는 매우 비효율적이며 정보를 추출해야 할 웹 페이지가 많다면 모든 웹 페이지에 대한 파서를 따로 구현한다는 것은 현실적으로 불가능하다. 따라서 그림 2와 같이 경로식을 이용하여 각 페이지의 정보의 위치만을 기억해 놓으면, 각각의 경로를 파서의 입력으로 주고, 파서는 입력으로 들어온 경로에 대해 파싱하면, 하나의 파서만을 이용하여 분산된 위치의 정보를 가져올 수 있다.

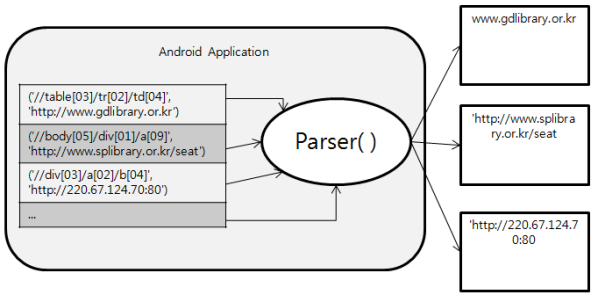

그림 2 경로식을 이용한 파싱

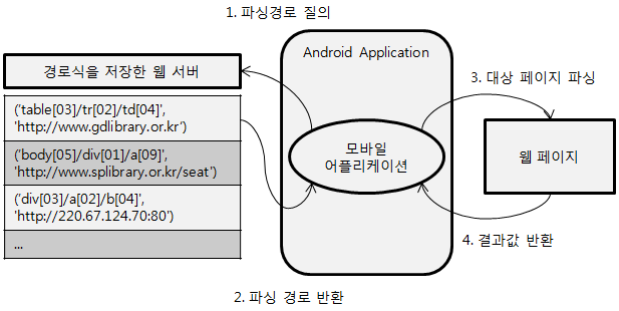

그림 3 위치 변화에 대한 대응

3.2 정보 위치 변화에 대한 대응

두 번째 문제점인 정보 위치 변화에 대한 대응으로는 그림 3과 같이 메타 데이터를 저장하는 서버를 둠으로써 해결할 수 있다. 웹 페이지가 개편되거나 웹 서버의 주소가 변경되면 추출 대상 정보의 위치가 변하게 된다. 정보의 위치가 변하게 되면 파서의 동작을 재설계해야 하고, 이는 애플리케이션의 재작성과 재배포로 이어진다. 이를 피하기 위해 파싱할 웹 페이지들의 경로식만을 가지고 있는 웹 서버를 둔다. 모바일 어플리케이션은 실행 시 또는 사용자가 요구 시 웹 서버에 저장된 경로식을 가져와 저장한 후 필요한 웹 페이지의 해당 경로에 접근하여 정보를 추출하면 웹 페이지가 변경 되더라도 웹 서버에 저장되어있는 파싱 경로만을 변경하면 되므로 애플리케이션을 재작성, 재배포할 필요가 없다. 개발할 모바일 애플리케이션의 특성에 따라 작성 시간 등의 메타 정보를 추가적으로 저장하므로 써 일정한 시간이 지나면 자동으로 메타데이터를 갱신하는 방법을 통해 유지보수를 쉽게 할 수 있다.

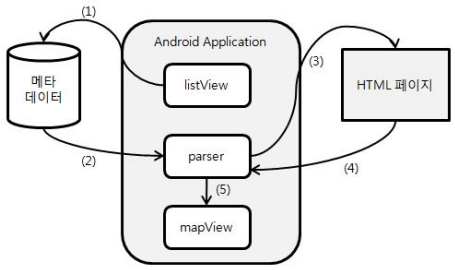

그림 4 프로그램 아키텍처

제안 사항을 이용하여 웹 스크래핑을 사용하는 모바일 애플리케이션 개발 시 발생하는 두 가지 문제점에 유연하게 대처할 수 있는 모바일 도서관 잔여좌석 조회 애플리케이션을 구현해 본다. 첫 번째로 모든 도서관의 잔여좌석 정보가 존재하는 웹 페이지 URL을 조사한 후 해당 웹 페이지 내의 잔여좌석 정보가 존재하는 태그의 경로식을 만든다. 두 번째로 만들어진 경로식들을 저장하는 메타데이터 서버를 만든다. 마지막으로 도서관 잔여좌석 조회 애플리케이션은 구동 시 메타데이터 서버에 연결 해 URL과 경로식을 가져와 가져온 URL에 접속한 후 경로식을 따라가 해당 위치의 잔여좌석 정보를 추출 후 화면에 출력한다.

5. 결 론

그림 5 모바일 도서관 잔여좌석 조회 애플리케이션

그림 5와 같은 도서관 잔여좌석 조회 애플리케이션의 경우 유지보수가 개발 프로세스의 대부분을 차지한다. 사용자의 요구에 맞추어 도서관을 수시로 추가, 제거하여야 하며 도서관 홈페이지 개편 등으로 웹 페이지의 구조 변경이 일어나면 정보를 다시 찾을 수 있게끔 수정하여야 한다. 제안한 방법을 사용하였을 경우 메타데이터 서버의 XML파일의 수정만으로 간단하게 유지보수를 할 수 있으므로 생산성을 크게 향상시켰다. 예를 들어 한국외국어대학교의 도서관 잔여좌석 페이지의 주소가 'http://203.232.237.169/dom-ian5.asp'에서 'http://203.232.237.169/lib' 으로 바뀐다면 파서가 URL을 얻기 위해 사용하는 데이터베이스나 파일, 또는 변수를 바뀐 URL로 수정한 후 빌드하여 재배포해야하는 번거로움이 있다. 제안 방법을 사용하면 모바일 애플리케이션의 소스코드를 직접 수정하지 않고 웹 서버의 메타데이터의 <name>태그의 내용이 한국외국어대학교인 <item>의 <url>을 바뀐 URL로 수정해주기만 하면 되므로 유지보수가 편리해 진다. 제안한 방법은 도서관 잔여좌석 조회 애플리케이션 외에 웹에 분산된 정보를 처리해야 하는 모든 애플리케이션에 응용할 수 있다.

참고문헌

[1] R.Allen Wyke, Sultan Rehman, Brad Leupen. "Programming XML" Original English Edition, Microsoft Press. 2002
[2] Quanzhong Li, Bongki Moon. "Indexing and Querying XML Data for Regular Path Expressions", Dept. of Computer Science University of Arizona, Tuscon, AZ 85721
[3] Mark Murphy. "Beginning Android" Original English language edition, Apress. 2009.
[4] Jericho HTML parser. available at http://jerichohtml.sourceforge.net/docs/index.html
[5] David Flanagan. "JavaScript: The Definitive Guide 5/E", O'REILLY. 2009