컨텐츠 제공자 지정 웹 클리핑 방식의 이동 인터넷 컨텐츠 변환

양 서 민† · 이 혁 준††

요 약

작은 화면을 가진 이동 단말기에서 데스크탑 화면에 맞추어 제작된 웹 컨텐츠들을 브라우징 하는 것은 많은 어려움을 갖는다. 웹 페이지에는 이동 단말기용 브라우저의 제한된 기능으로 인해 표시할 수 없는 객체가 일부 포함되어 있는 경우도 있고, 브라우저에서 호환되지 않는 마크업 표준을 사용하여 브라우징 자체가 불가능할 수도 있다. 본 논문에서는 웹 페이지를 이동 단말기에 최적화된 형태로 변환할 수 있는 웹 클리핑 방식의 새로운 이동 인터넷 컨텐츠 적응 방법을 소개한다. 이 방법에서는 컨텐츠 제공자가 클립 편집기를 이용하여 설정한 클립 명세에 따라 원본 웹 문서가 자동으로 클리핑 되고 변환된다. 클립 편집기는 컨텐츠 제공자가 단일 클립, 그룹 클립, 다중 레벨 클립, 동적 클립을 설정하고, 문서의 레이아웃을 수정할 수 있도록 한다. 이렇게 설정된 클립 명세에 따라 원본 문서로부터 추출된 각 클립들은 먼저 중간 언어 형태의 문서로 변환되고, 이는 다시 이동 단말기를 위한 최종 마크업 문서로 변환된다. 또한 다양한 이미지 타입에 대한 변환기능을 제공한다.

A New Mobile Content Adaptation Based on Content Provider-Specified Web Clipping

Seomin Yang† · Hyukjoon Lee††

ABSTRACT

Web contents created for desktop screens give rise to problems when they are to be displayed on the small screens of mobile terminals. While in some cases some of the objects of a page may not be displayable due to the lack of browser capability, the entire page may not be displayable due to the incompatibility with the browser in other cases. In this paper, we introduce a new mobile content adaptation approach based on web clipping, which transforms an original page into one that is optimally displayed on a mobile terminal. In this method, a source page is automatically clipped and transformed according to the clip specification made by the content provider using a clip editing tool. The clip editing tool allows the user to specify group clips, multi-level clips and dynamic clips as well as simple clips, and the presentation layout through a graphic user interface. Based on the clip specifications, each clip is transformed into an intermediate meta-language document, which in turn is transformed into a presentation page in the target markup language. Transcoding of image objects in major image file formats is also supported.

키워드 : 웹 클리핑(Web Clipping), 무선 이동 인터넷(Wireless Mobile Internet), 변환코딩(Transcoding), XML, 이동 단말기(Mobile Device)

1. 서 론

최근 들어 무선 이동통신 기술과 이동단말기 기술의 발전으로 인해 인터넷 기술은 무선 이동영역으로 까지 확대되어 가고 있다. 이에 따른 이동 인터넷 서비스의 출현은 이동성과 휴대성이라는 두 가지 편리성 하에서 사용자들에게 언제 어디서나 인터넷 엑세스가 가능하도록 해주고 있다. 그러나 이동 인터넷 환경에서는 이동전화, 스마트폰, PDA(Personal Data Assistant), 웹 패드 등의 다양한 이동 단말기와 다수의 컨텐츠 표준이 공존하고 있기 때문에 일관된 인터페이스를 통해 다양한 컨텐츠를 제공하는 것이 어렵다. 또한 현재 유선 웹 상에서 제공되는 대부분의 컨텐츠들은 고해상도 화면을 가진 데스크탑 PC에서의 브라우징을 위해 작성되었기 때문에 작은 화면과 제한된 기능의 브라우저를 가진 이동 단말기에서는 정상적인 브라우징이 어렵다.

이를 해결하기 위해 이동 단말기의 디스플레이에 적합하도록 만들어진 이동 인터넷 전용 마크업 언어들이 제안되었다. 이러한 언어 표준으로는 WML(Wireless Markup Language)[1], CHTML(Compact Hyper-Text Markup Langu-

※ 본 연구는 한국과학재단 목적기초연구(R01-2001-00349) 지원으로 수행되었음.
† 준 회 원 : 광운대학교 대학원 컴퓨터공학과
†† 정 회 원 : 광운대학교 컴퓨터공학과 부교수
논문접수 : 2003년 11월 5일, 심사완료 : 2004년 2월 5일

age)[2], XHTML(eXtensible HTML) basic[3] 등이 있다. 그러나 이동 인터넷 마크업 언어를 사용하여 컨텐츠를 제공하기 위해서는 이동 컨텐츠 제공자들이 이동 단말기를 위한 별도의 컨텐츠를 제작해야 하고, 같은 내용의 컨텐츠에 대해 여러 형태의 마크업 언어를 사용하여 다수의 문서를 작성하고 유지해야 한다는 문제가 있다. 이를 보완할 수 있는 방법으로는 컨텐츠와 프리젠테이션(presentation)을 분리하여 작성하는 방식이 있다. 여기서는 XML(eXtensible Markup Language)에서 소개하는 스타일시트(stylesheet) 기술을 사용하여 사이트 구축 시에 컨텐츠 내용과 표현 양식을 분리하여 작성한다[4, 5]. 이러한 방법은 하나의 컨텐츠를 다양한 단말기 환경에 적합한 형태로 제공할 수 있다는 장점이 있으나 컨텐츠를 완전히 새로 구축해야 하기 때문에 신규 구축에 필요한 비용과 노력이 많이 든다는 단점이 있다.

이러한 문제를 해결하기 위해 다른 관점으로 제시된 것이 트랜스코딩(transcoding) 방식이다. 이는 이동 웹 액세스 시에 사용자 만족도를 향상시키기 위한 실용적인 방법으로 널리 사용되고 있다[6, 7]. 이 방식에서는 일반적으로 단말기와 컨텐츠 서버 사이에 게이트웨이를 이용하여 응용프로그램 계층에서 컨텐츠에 대한 변환 기능을 수행하기 때문에 기존의 웹 서비스에서 사용하고 있는 응용 프로그램들이나 컨텐츠를 수정하지 않고 활용할 수 있으며, 여러 압축 기법을 사용하여 무선 링크를 통해 전송되는 데이터의 크기를 감소시켜 결과적으로 사용자 입장에서의 전송속도가 향상되는 효과를 제공한다. 또한, 무선 이동 인터넷 환경상에 존재하는 다양한 단말기, 다양한 마크업 표준에 따라 별도의 컨텐츠를 개발해야 하는 필요성을 제거해 주기 때문에 컨텐츠 개발비용과 기간을 줄일 수 있으며, 단말기 기술의 변화에도 빠르게 대처할 수 있다는 장점을 가지고 있다. 이 방식은 크게 모든 페이지에 대해 같은 변환 규칙을 적용하는 자동 변환 방식[6, 7]과 각 페이지 별로 변환 규칙을 지정하는 변환 규칙 설정 방식[8-10]으로 나뉘어 진다. 하지만, 지금까지 개발된 변환기들은 다양한 표현 양식을 가진 유선 웹 컨텐츠에 대해 제한적인 변환기능만을 지원하고, 페이지 레이아웃에 대한 적응기능이 고려되지 않아 변환된 문서의 가독성이 낮아 사용자 입장에서 만족스러운 변환 결과를 얻기가 어렵다.

최근에 제안된 웹 클리핑(Web clipping)은 PDA와 같은 작은 크기의 디스플레이 환경을 가진 기기를 위해 HTML 문서에서 특정부분을 발췌하고 표시하는 기술이다[11]. 웹 클리핑의 가장 큰 이점은 원본 페이지에 대한 정보의 손실 없이 이동 단말기에 적합한 레이아웃으로 효과적으로 적응할 수 있다는 것이다. 이에 따라, 원본 페이지가 트랜스코딩 단계 이전에 먼저 클리핑 된다면 더 나은 변환 결과를 얻을 수 있다.

본 논문에서는 웹 클리핑을 기반으로 하는 새로운 형태의 이동 컨텐츠 변환 방법을 제안한다. 이 방법에서는 원본 페이지에 대한 컨텐츠 제공자가 클립 편집기를 사용해 구축해 놓은 클리핑 규칙과 변환 규칙에 따라 원본 페이지의 일부, 즉, 클립이 자동으로 추출되고 변환된다. 클리핑 된 각 부분은 이동 단말기의 다양성으로 인한 혼란을 막고 일관된 형태의 변환을 위하여 먼저 XML 기반의 메타언어로 변환되고, 이를 다시 특정 단말기에 적합한 형태로 변환하여 다양한 마크업 언어들 간의 상호 변환이 가능하도록 한다. 이미지의 경우에는 이동 단말기의 제한된 환경에 최적화될 수 있도록 이미지 크기, 질, 포맷, 지원되는 컬러 수 등의 다양한 변환기능을 제공한다. 이를 통해 이동 인터넷 서비스를 위하여 기존 컨텐츠를 재작성해야 하는 필요성을 없애고, 새로운 단말기나 마크업 표준에 대한 확장을 쉽게 할 수 있다는 장점을 가지게 된다. 또한, 다양한 압축 기법을 통해 무선망을 통해 전달되는 컨텐츠의 양을 최적화할 수 있다.

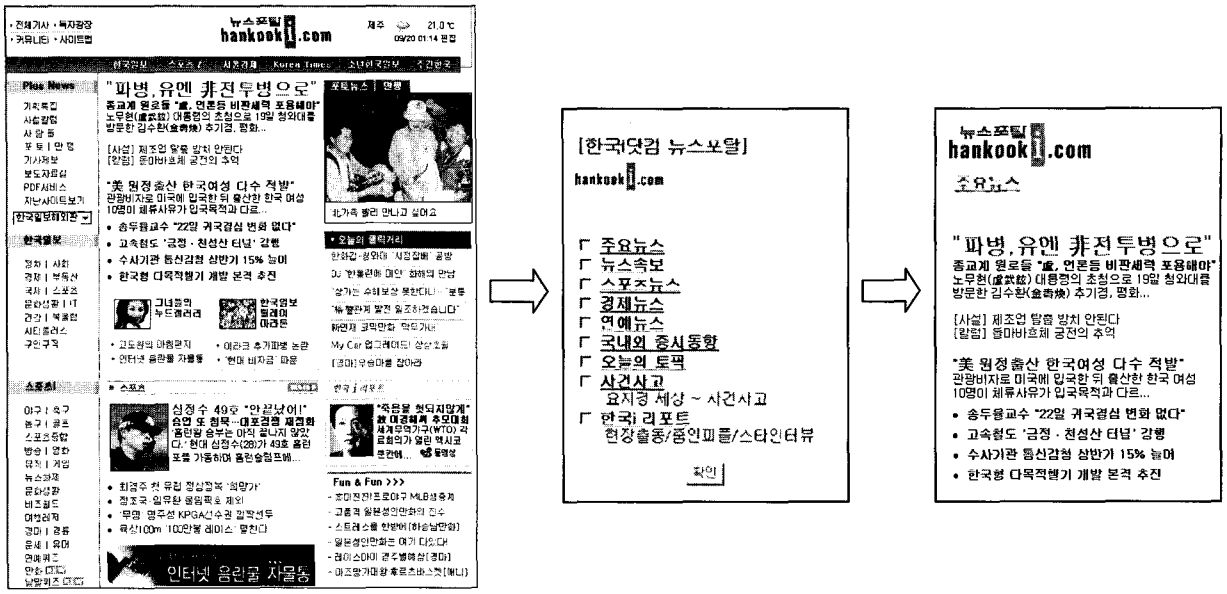
본 논문의 구성은 다음과 같다. 2장에서는 제안하는 웹 클리핑 시스템의 전반적인 구조에 대해서 설명하고, 3장에서는 웹 클리핑을 포함하는 컨텐츠 변환 방법에 대해서 자세히 기술하며, 4장은 클리핑 규칙 설정을 위한 클립 편집기에 대해 설명한다. 5장에서는 시스템의 구현내용을 설명하고, 6장에서 결론을 맺는다.

2. 유무선 통합을 위한 웹 클리핑 시스템

2.1 유무선 통합 웹 컨텐츠 서비스

우리가 제안하는 웹 클리핑 시스템을 사용하는 유무선 통합 웹 컨텐츠 서비스의 시나리오는 크게 3단계로 이루어져 있다. 각 단계별로 살펴보면

① 변환 규칙 설정 : 기존의 유선 웹을 통해 서비스되는 컨텐츠는 데스크탑 환경에 적합하도록 설계되었기 때문에 상대적으로 작은 화면과 다양한 표준을 따르는 무선 단말기에서 그대로 서비스되는 것은 부적합하다. 이에 따라 변환과정에서 필수적으로 요구되는 것이 웹 클리핑과 트랜스코딩을 통한 컨텐츠의 재구성이다. 이를 위한 규칙 설정은 GUI 기반의 WYSIWYG 인터페이스를 가지고 있는 클립 편집기를 사용한다. 원본 문서의 내용을 화면으로 보면서 추출될 영역을 지정하고, 문서를 어떻게 재구성할지를 설정한다. 이렇게 설정된 변환 규칙은 실제 사용자의 요청에 따라 컨텐츠 변환을 수행하는 웹



(a) 원본 웹 문서 (b) 메뉴페이지 (c) 웹 클리핑 문서

(그림 1) 유선 웹 페이지의 웹 클리핑 결과 예제(PDA의 경우)

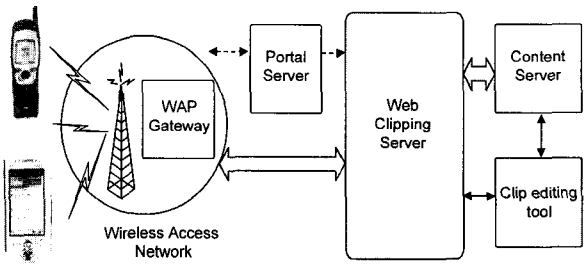
클리핑 서버내의 데이터베이스에 전달되어 변환동작에 바로 적용된다.

② 사용자 요청 : 무선 단말기의 사용자는 무선 이동 망을 통해 무선 인터넷 컨텐츠의 관문서버에 연결하여 원하는 컨텐츠를 검색하고 선택한다. 컨텐츠에 대한 요청은 웹 클리핑 서비스를 받기 위한 변환 요청을 생성하게 되고, 이를 받은 웹 클리핑 서버는 단말기의 종류를 인지하고. 요청 내용을 파악한다. 원본 컨텐츠는 편집기에 의해서 설정된 변환규칙에 따라 변환된다.

③ 컨텐츠 변환 : 웹 클리핑 규칙에 따라 변환된 문서는 여러 개의 분할된 클립 형태로 구성된다. (그림 1)과 같이 사용자에게는 클립을 선택할 수 있도록 메뉴 페이지가 먼저 제공되고, 이 메뉴 페이지를 통해 사용자가 원하는 정보를 선택하면 특정 클립에 대한 웹 클리핑 요청이 수행된다.

2.2 웹 클리핑 시스템의 구조

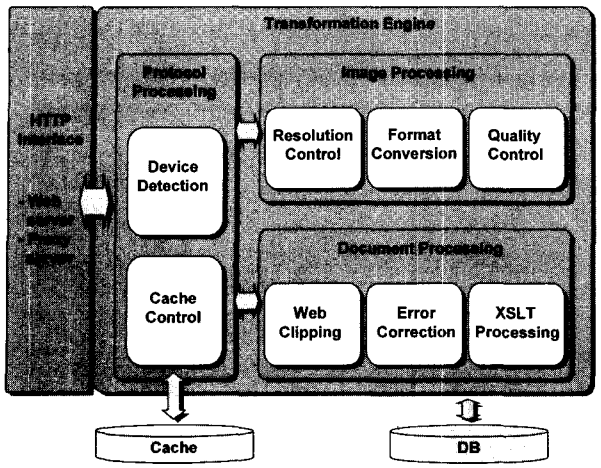
무선 인터넷 환경에서 이동단말기의 요청은 무선 이동 망을 거쳐 인터넷상의 컨텐츠 서버에게 전달되며, 이에 대한 응답은 반대경로를 통해 이동 단말기에 전달되게 된다. 제안하는 웹 클리핑 시스템은 (그림 2)와 같이 무선 이동 망과 인터넷상의 컨텐츠 서버들 간의 연결을 중재할 수 있는 위치에 존재하기 때문에 단말기 브라우저의 종류에 상관없이 웹 서버 혹은 프록시 서버의 형태로 변환 기능을 제공할 수 있다. 시스템의 구성을 살펴보면 크게 관문(portal) 서버, 클립 편집기, 웹 클리핑 서버로 구성되어 있다. 여기서 관문 서버는 컨텐츠 서비스에 접근하고자 사용자들을 위한 관문 역할을 수행하는 웹 서버로 웹 클리핑 서버의 전단에 위치한다. 클립 편집기는 컨텐츠 제공자 입장에서 클리핑 규칙과 변환 규칙을 자유롭게 편집할 수 있도록 해주는 GUI 기반 편집기이며, 웹 클리핑 서버와의 정보교환을 통해 설정된 규칙 정보들을 관리하는 기능을 수행한다.



(그림 2) 웹 클리핑 시스템

웹 클리핑 서버는 이동 단말기에게 직접 변환 기능을 제공하는 서버로서 (그림 3)과 같은 구조를 가지고 있으며, 내부적으로 크게 HTTP 인터페이스, 변환 엔진, 데이터베이스, 캐쉬로 구성되어 있다.

HTTP 인터페이스는 이동 단말기의 브라우저로부터 컨텐츠 변환 요청을 받아서 이로부터 원본 문서의 URL, 문서내의 클립 식별자, 이동 단말기의 종류를 추출한다. 또한 원본 문서 URL을 이용하여 원본문서를 가져온 후에 다른 정보들과 함께 변환 엔진에 전달하여 컨텐츠 변환 동작을 시작하도록 한다. 변환 엔진으로부터의 변환결과는 다시 HTTP 인터페이스를 통해 이동 단말기에 대한 응답으로 전송된다.



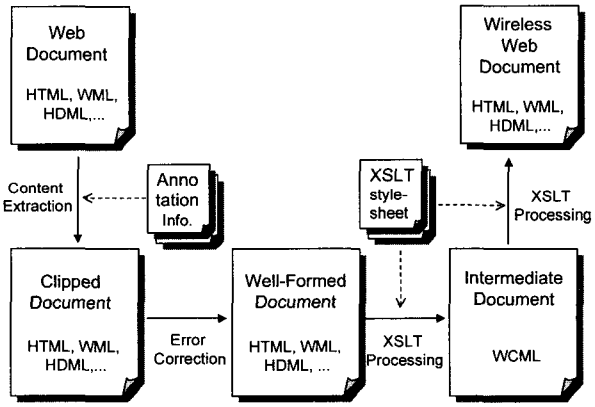
(그림 3) 웹 클리핑 서버의 구조

변환 엔진은 실제 변환 동작이 이루어지는 부분으로 이미지 처리부와 문서 처리부로 구성된 컨텐츠 처리부와 프로토콜 처리부로 나뉜다. 문서 처리 내용은 크게 웹 클리핑 기능, 에러 보정 기능, XSLT를 사용한 마크업 변환 기능이 있다. 클리핑 기능은 기존 웹 컨텐츠 내에서 원하는 부분들만을 추출하여 이를 재구성하는 기능이고, 에러보정과 XSLT 변환 기능은 유무선 웹에서 사용되는 다양한 마크업 문서간의 상호변환 기능을 제공한다. 이미지 처리 기능으로는 다양한 이미지 포맷들간의 상호 변환과 이미지 크기, 해상도, 표현 가능한 컬러 수와 같은 속성 변환 기능이 있다. 변환엔진의 자세한 기능은 다음 장에서 설명한다. 프로토콜 처리부는 이동 단말기와 컨텐츠 서버들간의 컨텐츠 요청과 응답을 제어하는 기능을 제공하며, 문서 요청 헤더정보를 활용하여 이동 단말기의 종류를 인식하는 기능, 캐쉬 제어 기능을 부가적으로 가지고 있다. 단말기 식별을 위해서는 단말기로부터의 HTTP 헤더 중에서 User Agent 필드의 분석을 통해 디바이스의 종류와 브라우저의 종류를 알아낸다. 또한 웹 클리핑 서버에서는 무선 웹 요청에 대한 응답시간 최소화를 위해 원본 컨텐츠 캐쉬와 변환 중간결과 캐쉬를 가지고 있다. 변환 중간결과 캐쉬는 변환 성능 향상을 목적으로 변환 시에 생성되는 중간 결과를 저장한다. 최종 변환 결과의 경우에는 하나의 원본 컨텐츠에 대해 단말기의 종류에 따라 각기 다른 형태로 변환된 다수의 컨텐츠가 생성되기 때문에 불필요하게 저장공간을 많이 차지하게 되고, 캐쉬 적중률도 낮아진다는 단점이 있기 때문에 캐쉬에 저장하지 않는다. 캐쉬에 실제 저장되는 중간결과 데이터는 컨텐츠 종류에 따라 구분된다. 문서의 경우에는 문서 변환 과정에서 공통으로 생성되는 중간 단계 문서를 저장하고, 이미지인 경우에는 비트맵 형태로 저장한다. 이렇게 저장된 데이터들은 캐쉬 적중 시에 변환 절차의 중간 단계부터 시작하여 원하는 최종 변환 결과를 얻을 수 있기 때문에 사용자 응답속도 뿐만 아니라 변환으로 인한 오버헤드도 줄일 수 있다.

데이터 베이스에서는 컨텐츠 변환을 위해 필요한 모든 정보들을 저장하고 있다. 컨텐츠에 따른 클리핑 및 변환 규칙 정보, 이동 단말기에 따른 변환 규칙 정보와 각 특성정보 등을 저장한다. 또한 원본 컨텐츠 URL과 변환 규칙 정보들 간의 연결 관계를 저장한다. 데이터 베이스내의 모든 변환 관련 데이터는 편집기에 의해서 추가 및 수정된다.

3. 변환 규칙에 의한 컨텐츠 변환

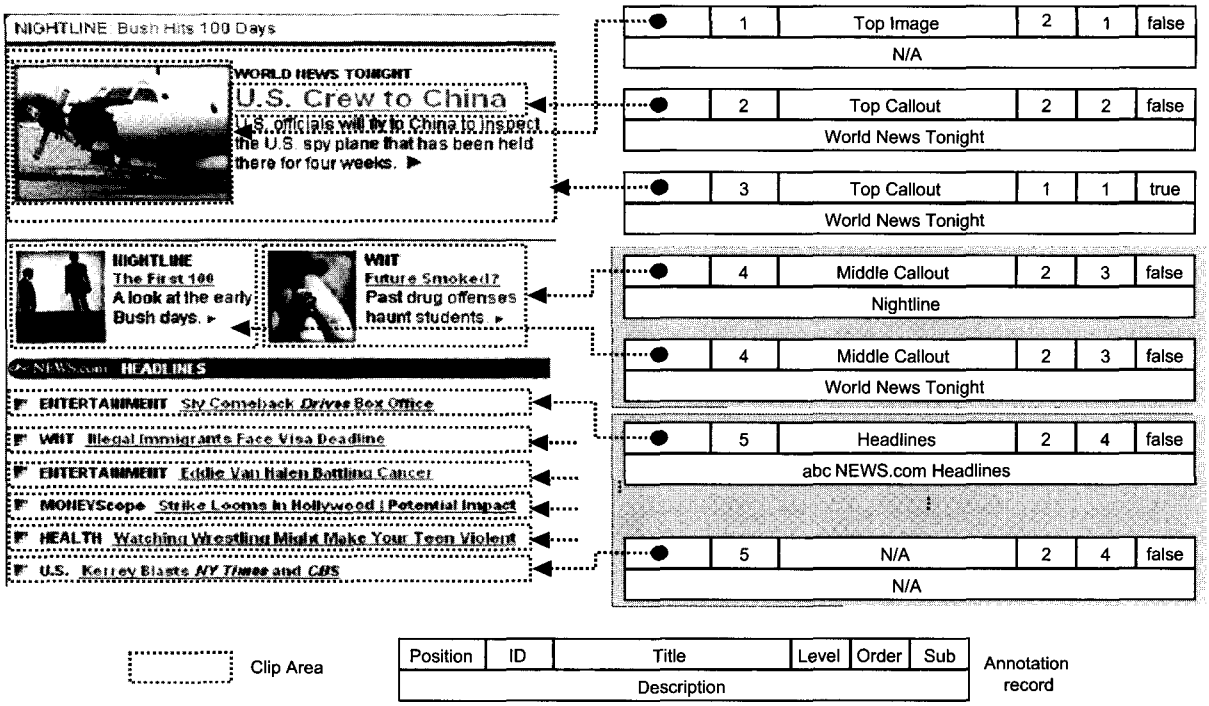
웹 클리핑 서버에서의 문서 변환 과정은 (그림 4)와 같이 컨텐츠 추출, 에러보정, 중간 단계 문서로의 변환, 최종 무선 웹 문서 변환의 4단계를 거쳐 수행된다. 우리가 제안하는 변환과정을 통해 현재 유선 인터넷상에서 서비스되고 있는 원본 컨텐츠는 컨텐츠 자체에 대한 수정과정 없이도 참조형태의 정보들에 의해 제어되어 무선 단말기의 특성에 적합한 형태로 재구성되고 단말기 브라우저에 적합한 포맷으로의 적응이 가능하게 된다. 각 단계별 동작에 대해서 자세히 살펴보겠다.



(그림 4) 문서 변환 과정

3.1 클리핑

유선 컨텐츠의 효율적인 무선화를 위해서는 컨텐츠 추출을 통해 정제된 내용만을 고르고, 불필요한 내용을 제거하여 이동 단말기에 적합한 단위로 브라우징 할 수 있도록 해주는 웹 클리핑이 필수적으로 요구된다[11, 12]. 추출을 통한 변환을 위해서는 컨텐츠의 내용과 레이아웃간의 관계를 파악하는 것이 필요하다. 하지만 현재 사용되고 있는 컨텐츠 표준들은 대부분 프리젠테이션을 목적으로 하고 있기 때문에 단순한 키워드 검색과 같은 기능은 가능하지만, 문서를 분할하거나 추출하기 위해 문서의 의미론적인(semantic) 구



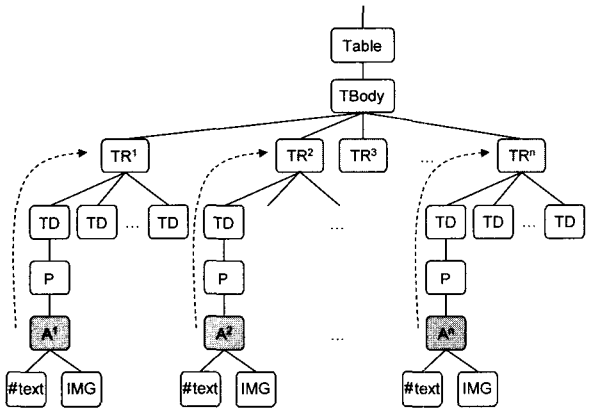
(그림 5) 컨텐츠 추출을 위한 클립 명세 정보

조를 문서처리 과정에서 얻어내는 것은 매우 어렵다. 이를 해결하기 위해서는 컨텐츠 관리자가 직접 컨텐츠의 문서구조나 의미에 대한 정보를 파악하고, 이를 기준으로 클리핑 및 변환 규칙을 설정하는 것이 필요하다. 하지만 복잡한 마크업으로 이루어진 문서를 내부적으로 살펴보면서 규칙을 설정하는 작업은 현실적으로 어려움이 많기 때문에 본 논문에서는 이러한 작업을 보다 편리하게 수행하기 위한 GUI 기반의 클립 편집도구(4장에서 설명)를 사용한다. 기본적인 컨텐츠 추출의 단위인 클립(clip)은 TABLE, P, IMG, TEXT 등과 같은 문서상의 마크업 요소(element)의 단위로 구성한다. 편집기에 의해 선정된 각 클립에는 부가 정보가 지정되며, 다양한 목적에 따라 그룹화 되어 관리될 수 있다. 모든 클립 정보는 웹 클리핑 서버로 전송되어 이 정보를 기준으로 요청된 페이지에 대한 클리핑 동작을 수행하게 된다.

각 클립을 묘사하기 위해 저장되는 명세 정보들은 (그림 5)의 예와 같이 문서내의 클립 위치정보(position), 식별자(id), 제목(title), 설명(description), 클립의 레벨정보(level), 표시순서(order), 부영역 포함여부(sub)로 구성되어 있다. 클립 위치정보는 문서 마크업상의 DOM(Document Object Model) 트리 구조에서 대상이 되는 노드에 대한 절대 경로 정보를 추적하여 저장한다. 클립의 위치와 범위를 모두 알아야 하기 때문에 클립의 시작 위치와 끝 위치가 함께 저장된다. 식별자는 모든 클립에 겹치지 않도록 부여되며, 하나 혹은 복수개의 클립들을 서로 구분한다. 클립 제목과 설명은 클립 내에 포함된 컨텐츠의 내용에 대한 부가 설명정보로서 활용된다. 한 페이지 내에 복수의 클립이 존재할 때는 사용자가 원하는 클립을 선택할 수 있도록 제목과 설명정보를 사용하여 메뉴 페이지를 구성하게 된다. 클립 레벨정보는 디스플레이 능력에 맞춰 구분된 이동단말기의 레벨에 따른 다중 레벨 클리핑을 위해 사용된다. 표시순서는 원본 문서상에서의 위치와 상관없이 변환 결과 문서상에 표시되길 원하는 위치를 나타낸다. 다른 영역 포함여부는 현재의 클립영역 내에 또 다른 클립이 포함되어 있는지를 나타낸다.

이렇게 정의되는 클립은 단일 클립, 그룹 클립, 동적 클립, 제거 클립, 사용자 클립으로 세분화된다. 단일 클립은 가장 일반적인 형태의 클립으로 단일 HTML 요소 단위로 지정되고, 하나의 유일한 식별자를 갖는 영역이다. 그룹 클립은 여러 개의 다른 영역들이 같은 클립 식별자를 가지면서 그룹으로 다루어지는 경우이다. 제거 클립은 이미 지정된 클립 영역 내에서 삭제하고자 하는 부분을 지정한다. 예를 들면 연속된 문장 사이에 삽입되어 있는 이미지를 이동단말기에서는 제거한 상태로 브라우징하고 싶은 경우에 사용한다. 사용자 클립은 원본 문서에 존재하지 않는 요소를 변환 결과 문서에 추가할 수 있는 규칙이다. 예를 들면 변환 결과 문서에 브라우저 히스토리 상의 뒤로(back) 이동하는 링크나 홈페이지의 시작위치(home)로 이동하는 링크 등을 추가할 수 있다.

ASP, JSP, CGI와 같은 동적 컨텐츠 생성 기술을 통해 만들어진 컨텐츠들은 같은 URL에 대해서도 다른 문서구조를 가질 수 있기 때문에 고정된 태그 정보만을 사용하는 클리핑 동작만으로는 한계가 많다. 특히 동적인 요소가 가장 많이 사용되는 웹 게시판이나 웹 메일과 같은 문서는 주로 행과 열로 구성된 테이블이나 리스트의 구조를 가지고 있다. 이러한 동적 영역의 경우에는 일반 영역과는 구분하여 별도의 추출 방법을 사용한다. 동적 클립으로 지정된 경우에는 먼저 일반 클립 추가 과정을 통해 문서 내에서 반복되는 요소를 선택하고, 이를 기준으로 패턴정보를 추출한다. 패턴 추출을 위해서는 먼저 DOM 트리 상에서 지정된 반복 노드로부터 부모 방향으로 모든 노드를 차례로 검색한다. 상위 단계 노드로 이동하면서 각 노드에 복수의 자식 노드가 있는지를 확인하고, 모든 자식 노드를 통해 현 노드와 반복 요소까지의 경로와 같은 패턴의 경로가 존재하는지를 확인한다. 같은 패턴의 경로가 모든 자식에 대해서 존재하면 현재 노드의 절대 경로와 반복 노드까지의 길이를 저장한다. (그림 6)은 <table>내의 반복 패턴을 추출하는 예를 보이고 있다. 여기서 반복 노드로 지정된 요소는 $<A^1>$이고, HTML 테이블에서 열 방향으로 반복되는 패턴을 갖는 경우이다. <TBODY>를 기준으로 모든 n에 대해 $<TR^n>$으로부터 $<A^n>$까지는 같은 패턴의 경로를 가지고 있다.



(그림 6) 동적 클립에서의 패턴 추출의 예

클립 편집기를 사용하여 원본문서에 대한 클리핑 및 변환 규칙이 입력되면 이 정보들은 원본 문서의 URL과 매핑되어 변환 서버의 데이터베이스에 저장된다. 데이터베이스에는 원본문서 URL과 규칙간의 연결 관계와 각 규칙 내용이 저장된다. 이중에 클리핑 정보의 경우에 위치 정보는 문서의 DOM 트리 상의 요소 위치를 기준으로 생성하기 때문에 다른 내용을 가진 문서라도 동일한 DOM 구조를 가지고 있다면 한번 설정된 클리핑 정보와 변환 규칙 정보는 그대로 재사용할 수 있다는 이점이 있다.

3.2 마크업 트랜스코딩

일반적인 HTML 문서의 경우는 다양한 마크업 오류들을 포함하고 있으며, 추출과정을 통해 오류가 추가될 수도 있다. 따라서 컨텐츠 추출 단계에서 추출된 컨텐츠들은 재구성된 후에 에러보정 단계를 거치게 된다. 에러보정은 추출된 내용에 대한 마크업 문법 오류를 수정하여 잘 구성된(well-formed) 문서로 만들어 주는 것으로 다음 단계의 XML 문서 처리 과정을 위해서 반드시 필요한 과정이다[13, 14].

이렇게 처리된 문서는 XSLT(eXtensible Style-sheet Language Transformation) 기술을 사용하여 중간 단계의 문서로 변환한다[5]. 중간 단계의 문서는 모든 이동 단말기용 마크업 문서로의 변환이 가능하도록 설계된 XML 기반의 WCML(Web Clipping Markup Language)로 정의된다.

WCML은 유무선 컨텐츠간의 일관된 변환을 위해 설계된 XML의 한 응용이다. WCML은 XML의 구조적인 특징을 그대로 가지면서, 별도의 구성 요소들로 정의된다. WCML은 HTML과 이동 단말기용 마크업 언어에서 사용하는 프리젠테이션 요소와 문서의 구조를 정의하기 위한 요소로 구성되어 있다. 각 요소들은 마크업 변환 시에 각기 다른 역할을 하게 된다. 프리젠테이션 요소들은 주로 변환 테이블에 의해 단순 변환되지만, 문서구조를 표현하는 요소는 무선 컨텐츠 변환과정에서 변환 결과의 단위나 순서를 결정하거나 부가설명을 추가하는데 사용된다. 이러한 마크업 변환 과정은 미리 정의된 스타일시트에 의해 이루어진다. 모든 컨텐츠들을 중간 단계 문서인 WCML 문서로 변환하는 것은 변환 과정을 모듈화 시켜주며, 이를 통해 새로운 단말기나 브라우저 혹은 마크업 언어가 출현하더라도 적합한 스타일시트와 간단한 설정만으로도 새로운 환경에 적응할 수 있게 된다.

<표 1>은 WCML을 구성하는 주요 요소들을 정리한 것이다. 여기서 wcml : clip-info 요소는 문서의 내용과 구조를 기술하는데 사용되며, wcml : content는 프리젠테이션을 목적으로 하는 요소로서 기존의 마크업 표준인 HTML, WML 등과 유사한 구성요소를 가진다.

이와 같은 WCML 문서는 원본 문서의 구조에는 거의 영향을 주지 않으면서 이동 단말기 측에 큰 처리능력을 요구하는 요소 및 속성을 제거하고, 문서가 단순화 된 후에도 다른 표준 문서로 변환했을 때 원본 문서의 문맥을 그대로 유지할 수 있도록 문서의 구조나 내용을 기술하는 요소나 속성은 유지한다.

다음 단계에서는 다시 한번 XSLT를 이용하여 WCML 문서를 각 이동단말기에서 인지할 수 있는 형태의 문서로

변환한다. 이러한 변환은 각 단말기에 따라 별도로 작성된 XSLT 스타일시트에 따라 수행된다. 단말기 브라우저의 특성에 따라 불필요한 요소나 속성들을 추가로 제거하고, 각 마크업 표준에서 요구하는 별도의 특성에 맞도록 결과 문서를 재구성한다. 예를 들면 WML이나 HDML의 경우는 card/deck 구조를 사용하기 때문에 이 과정에서 결과 페이지에 card와 deck을 생성하여 준다.

〈표 1〉 WCML의 주요요소

1-level 요소	2-level 요소	설 명
wcml : clip-info	wcml : clip-id	각 클립의 식별자
	wcml : clip-title	각 클립의 표제를 정의
	wcml : clip-desc	각 클립의 내용에 대한 간략한 요약 정보
wcml : content	wcml : clip-div	블록 단위 텍스트를 정의
	wcml : heading	표제를 정의
	wcml : para	문단을 정의
	wcml : anchor	하이퍼링크를 정의
	wcml : list	리스트 형태의 내용을 정의
	wcml : table	행과 열로 이루어진 표 형태의 내용을 정의
	wcml : object	이미지와 같은 내장 객체를 정의

3.3 이미지 트랜스코딩

인터넷상의 웹 페이지는 문서 데이터 뿐만 아니라 많은 수의 이미지 데이터를 포함하고 있다. 하지만, 무선 단말기는 화면 크기가 작고, 표현 가능한 컬러수가 적으며, 지원되는 이미지 포맷의 한계로 인해 유선상의 선명한 이미지를 제대로 표현할 수가 없는 경우가 많다. 이런 한계를 극복하면서 데이터 전송 부담을 줄이고, 무선 단말에 적합한 형태의 이미지로 변환하기 위한 이미지 트랜스코딩 처리내용은 크게 단말기에 대한 적응과 손실압축으로 나눌 수 있다. 여기서 단말기에 대한 적응은 원본 이미지를 단말기 화면 크기, 지원되는 이미지 포맷, 지원되는 컬러 수 등에 적합하도록 변환하는 것을 말하며, 손실압축은 이미지의 질(quality), 크기, 사용되는 컬러 수 등을 낮추어 손실을 통한 데이터 크기의 축소효과를 얻는 것을 말한다.

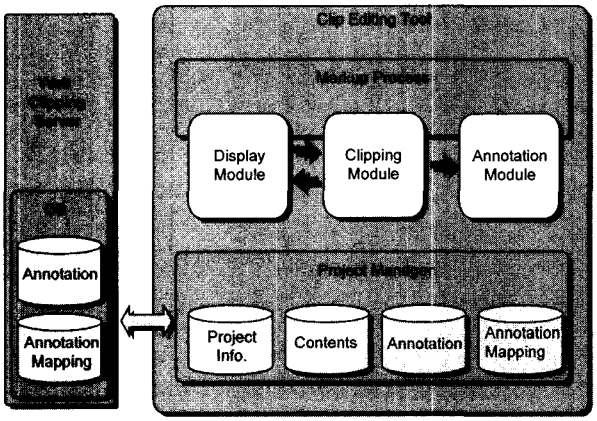
이미지 크기의 축소비율은 단말기 화면 크기와 미리 지정된 기본 이미지 축소비율을 사용하여 결정된다. 이미지 질은 이미지 데이터의 크기를 결정하는 중요한 변수로 이미지의 선명도와 표현 가능한 컬러 수에 따라 조정될 수 있다. 이미지 압축방식 중에 JPEG과 같이 양자화에 의한 손실압축을 사용하는 경우는 이미지 질 변수를 낮게 조정하여 이미지 데이터의 크기를 줄일 수 있다[15]. 비트맵 방식의 이미지의 경우에는 디더링을 이용하여 컬러 팔레트의 크기와 각 픽셀 당 데이터의 크기를 줄임으로써 단말기 환경에 대한 적응과 동시에 압축 효과를 얻을 수 있다. 무선 단말기의 브라우저는 유선 웹에서 사용할 수 있는 모든 이미지를 지원하지 못하고, 때로는 별도의 표준(예 : WAP의 WBMP)을 요구하는 경우가 많다. 본 시스템에서는 이미지 포맷간의 상호 변환을 위해서 모든 원본 이미지를 중간 단계의 순수 비트맵 형태로 변환하고, 이를 원하는 포맷의 이미지로 다시 변환하는 단계를 거치게 된다.

4. 클립 편집기

변환 규칙 편집 작업은 무선 컨텐츠 제작 작업과 같은 비중을 갖는 중요한 과정이다. 또한 사람에 의해서 이루어지는 과정이기 때문에 웹 문서 편집기와 같은 응용 프로그램 설계 시와 유사한 요구사항을 갖는다. 본 논문에서 개발한 클립 편집기의 핵심적인 특징은 다음과 같이 요약될 수 있다. 첫째, 마우스 클릭을 이용한 직관적인 조작을 지원하는 GUI 환경을 통해 HTML 문서 편집기와 같이 사용법이 쉽다. 둘째, 설정된 추출 및 변환 규칙에 따라 변환된 페이지를 편집기 상에서 미리 보기를 통해 확인할 수 있다. 이때 이동전화, PDA 등과 같이 다양한 이동 단말기 환경에 대한 결과를 별도로 확인할 수 있다. 셋째, 변환 결과에 대해 미리 보기 상태에서 추가로 레이아웃 조정과 다양한 사용자 정의 객체의 추가가 가능하다. 넷째, 웹 클리핑 서버와의 연동 기능을 통해 수정된 클리핑 규칙 내용과, 규칙과 원본 컨텐츠간 연결 관계를 서버의 데이터베이스에 전송하고 관리한다. 마지막으로 편집 대상 컨텐츠들은 사이트 단위의 프로젝트로 구성하여 관리하고, 이를 통해 한 사이트 내의 모든 문서들이 통합 관리된다.

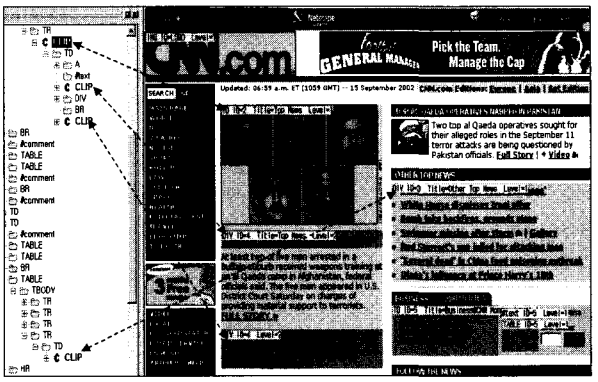
클립 편집기는 (그림 7)과 같이 내부적으로 화면 처리 모듈, 클리핑 모듈, 참조(annotation) 모듈, 프로젝트 관리자로 구성되어 있다. 여기서 화면 처리 모듈은 웹 컨텐츠와 설정된 클리핑 규칙에 대한 화면 표시 기능을 처리한다. 기본적으로 웹 브라우저와 같은 형태의 화면을 제공하며, 이에 겹치는 형태로 변환 규칙을 설정하고, 현재의 편집 상태를 직관적으로 확인할 수 있도록 해준다. 클리핑 모듈은 원본 문서상에서 사용자가 원하는 위치에 추출 영역을 지정하기 위하여 HTML 문서의 구조를 검색하고, 해당 위치를 참조 정보에 저장할 수 있도록 해준다. 이 모듈에서 지원하는 클립 지정 방법은 크게 브라우징 화면상 좌표로부터의 지정과 문서 마크업 트리 구조로부터의 지정으로 나뉜다. 전자는 웹 브라우징 화면 상태에서 사용자의 마우스 클릭 좌표로부터 문서 트리 구조상의 위치를 검색하는 방식이고, 후자는 웹 페이지의 내용을 DOM 트리 형태로 보면서 사용자가 원하는 위치를 직접 검색하여 지정하는 방식이다. 참

조 모듈에서는 클리핑 과정을 통해 설정된 클립 명세 정보를 외부에서 참조할 수 있는 파일 형태로 저장하고 관리한다. 저장되는 정보는 클립 명세 정보와 같다. 마크업 처리부는 화면 처리 모듈, 클리핑 모듈, 참조 모듈에서의 다양한 문서 처리를 위한 기본적인 마크업 문서 처리 기능들을 제공한다. 프로젝트 관리자는 하나의 웹사이트 전체를 통합된 프로젝트 형태로 관리하여 일관된 변환 규칙의 설정 및 유지, 관리가 가능하도록 한다. 프로젝트 내에서는 사이트 내의 모든 웹 문서와 이 문서를 통해 설정된 클리핑 및 변환 규칙 정보, 그리고 원본 컨텐츠와 규칙간의 연결 관계를 저장하고 관리한다. 또한 변환 서버와의 연동 기능을 제공한다.

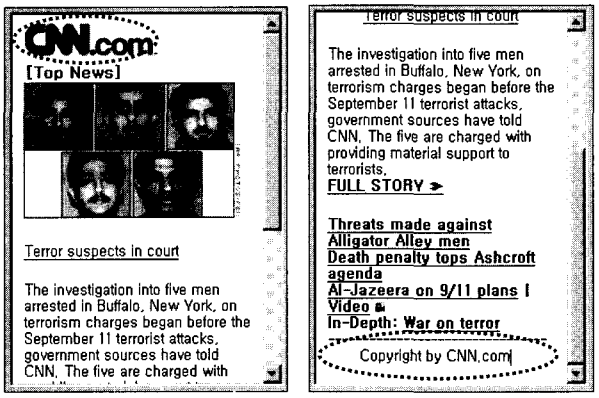


(그림 7) 클립 편집기의 구조

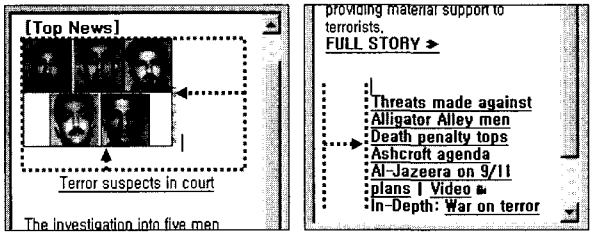
편집기에서는 (그림 8)과 같이 원본 컨텐츠에 대한 클리핑 규칙과 변환 규칙을 지정하기 위해 기본적으로 브라우징 화면과 문서의 DOM 트리 화면이 동시에 제공된다. 브라우징 화면에서는 웹 브라우저 상에서 문서를 보는 상태로 마우스를 이용하여 클립을 지정할 수 있으며, 문서 트리 화면에서는 세부적인 문서 구조를 보면서 보다 정교한 클립의 지정이 가능하다. 이렇게 클립이 지정된 후에는 각 클립에 대한 제목이나 설명과 같은 부가정보를 추가하고 편집할 수 있다. 각 클립은 화면상에 박스 형태로 표시되며, 속성 값들은 웹 클리핑 서버에서의 클리핑 단계에서 쓰일 수 있도록 참조 정보로 저장된다. 이 정보는 기존에 설정된 규칙을 수정하거나 같은 문서 구조를 가진 다른 컨텐츠를 위해서도 사용된다. 기본적인 클리핑 규칙은 원본 문서의 내용을 기준으로 구성되지만, 이동 단말기에서의 서비스 편의를 위하여 원본 웹 문서에는 존재하지 않는 내용이 추가되어야 할 필요가 있다. 이를 위해 편집기에서는 변환 결과를 미리 보기 상태로 보면서 레이아웃을 조정하고, 다양한 사용자 정의 객체를 추가하는 것이 가능하다. 예를 들면 (그림 9)에서와 같이 사이트의 로고를 추가하거나 각 변환 결과마다 꼬리말을 삽입하는 기능 등이 있다. 또한 특정 문서 객체의 위치나 크기 조정, 문단 모양 조정과 같은 정교한 규칙을 설정할 수 있다.



(그림 8) 변환 규칙 편집의 예(CNN.com)



(a) 이미지의 추가 (b) 꼬리말의 추가



(c) 객체의 배치 및 크기의 수정 (d) 문단모양의 수정

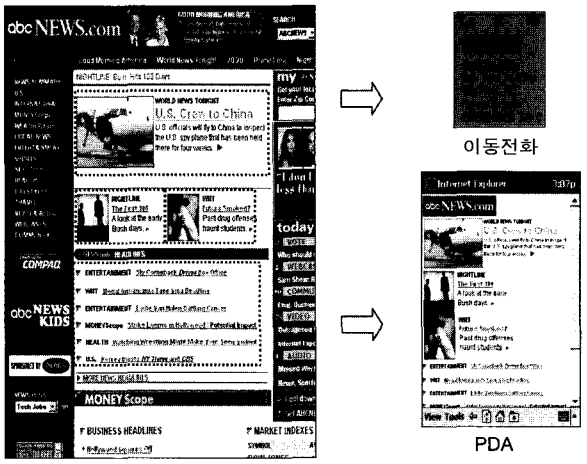
(그림 9) 사용자 정의 편집 기능의 예(CNN.com)

5. 구현 및 변환 결과

변환 서버의 구현은 Microsoft사의 IIS(Internet Information Server)를 기반으로 한다. 서버내의 주요기능에 대한 제어 및 접근 관리자는 확장성과 유연성을 높이기 위하여 ASP 형태로 구현되었다. 컨텐츠 추출, 마크업 변환, 이미지 변환과 같은 변환 엔진 기능은 성능 최적화를 위하여 Microsoft의 COM 컴포넌트 구조로 구성되어 있다. 본 논문에서 제안하는 무선 웹 변환 서버는 기본적으로 웹 서버 형태로

구현되어 있으나 변환 엔진이 컴포넌트 구조를 가지기 때문에 프록시 서버 형태로의 변형도 가능하도록 설계되었다. 변환 규칙 편집기는 Windows 환경의 데스크탑 PC에서 동작할 수 있도록 구현되었다. 편집기의 브라우징 기능과 문서 처리 기능은 WYSIWYG을 지원한다.

본 논문에서 제안하는 변환 시스템의 변환 기능을 테스트하기 위해서 국내외 다양한 사이트들에 대한 변환 규칙을 설정하고 결과를 확인하였다. (그림 10)은 본 논문에서 제안하는 웹 클리핑 시스템을 사용하여 미국의 대표적인 뉴스 웹사이트인 abcnews.com 사이트에 변환 규칙을 설정하고, 이동 전화와 PDA상에서의 변환 결과를 각각 나타내고 있다. 여기서 이동 전화와 PDA에서 서로 다른 결과가 나오는 것은 단말기의 디스플레이 수준에 따라 각기 다른 변환 규칙이 적용되었기 때문이다. 무선 단말기에 나타나는 변환 결과는 클리핑 규칙을 어떻게 설정했느냐에 따라 많은 차이를 가질 수 있다.



(그림 10) abcnews.com 원본 문서와 변환 결과

6. 결 론

이동 인터넷 환경에서는 이동 단말기의 제한된 기능과 다양성으로 인해 유선상의 웹 컨텐츠를 직접 액세스하는 것은 현실적으로 문제가 많다. 본 논문에서는 이미 구축되어 있는 유무선상의 웹 컨텐츠를 이동 단말기의 특성에 따라 가장 적합한 형태로 적응할 수 있는 새로운 웹 클리핑 시스템을 제안한다. 제안하는 방법에서는, 컨텐츠 제공자가 클립 편집기를 사용하여 원본 컨텐츠에 대한 클리핑 및 변환 규칙을 설정하고, 서버에서는 이에 따라 변환 동작을 수행한다. 원본 컨텐츠는 먼저 클리핑 된 후에 일관된 형태의 변환을 위하여 XML 기반의 메타언어인 WCML 문서로 변환된다. 이는 최종적으로 특정 이동 단말기에서 인지될 수 있는 형태의 문서로 변환된다. 이미지의 경우에는 다양한 트랜스코딩 기술을 통해 변환된다. 이러한 변환 동작을 통하여 이동 인터넷 서비스를 위해 컨텐츠를 재작성해야 하는 필요성을 없애고, 새로운 단말기나 새로운 마크업 표준이 등장하더라도 컨텐츠 제공자 입장에서는 확장을 쉽게 할 수 있으며, 다양한 압축 기법을 사용하여 무선망을 통해 전달되는 컨텐츠의 양을 최적화할 수 있다.

참 고 문 헌

[1] WAP Forum, "Wireless Markup Language version 2 Specification," http://www.wapforum.org/, 2001.

[2] Tomihisa Kamada, "Compact HTML for Small Information Appliances," http://www.w3.org/TR/1998/NOTE-compactHTML-19980209/, 1998.

[3] W3C, "XHTML basic," http://www.w3.org/TR/xhtml-basic/, 2000.

[4] W3C, "eXtensible Markup Language(XML) 1.0," http://www.w3.org/TR/REC-xml/, 2000.

[5] W3C, "XSL Transformation(XSLT) version 1.0," http://www.w3.org/TR/xslt/, 1999.

[6] A. Fox, S. D. Gribble, E. A. Brewer and E. Amir, "Adapting to network and client variability via on-demand dynamic distillation," Operating Systems Review, Vol.30, No.5, pp. 160-170, 1996.

[7] K. Ham, S. Jung, S. Yang, H. Lee and K. Chung, "Wireless-adaptation of WWW Content over CDMA," MOMUC '99 San Diego, pp.368-372, 1999.

[8] Oracle Co., "Oracle 9i Application Server Wireless," http://otn.oracle.com/products/iaswe/, 2003.

[9] IBM Co., "WebSphere Transcoding Publisher," http://www-4.ibm.com/software/webservers/transcoding/, 2000.

[10] OpenTV Inc., "Spyglass Prism," http://www.opentv.com/utilities/product-sheets/prism.pdf, 2002.

[11] P. Gomes, S. Tostao, D. Goncalves and J. Jorge, "Web Clipping : Compression Heuristics for Displaying Text on a PDA," Mobile HCI '01, France, Sept., 2001.

[12] Palm, Inc., "Web Clipping Developer's Guide," http://www.palmos.com/dev/support/docs/webclipping/, 2001.

[13] W3C, "HTML 4.01 Specification," http://www.w3.org/TR/html4/, 1999.

[14] W3C, "HTML Tidy," http://www.w3.org/People/Raggett/tidy/, 2002.

[15] G. K. Wallace, "The JPEG Still Picture Compression Standard," Communications of the ACM, Vol.34, Issue.4, pp.30-44, April, 1991.

양 서 민

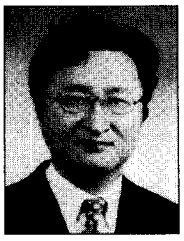
e-mail : uniload@kw.ac.kr

1997년 광운대학교 컴퓨터공학과(학사)

1999년 광운대학교 대학원 컴퓨터공학과 (공학석사)

1999년~현재 광운대학교 대학원 컴퓨터 공학과 박사과정

관심분야 : 유비쿼터스 컴퓨팅, 이동 인터넷, 이동 애드혹 네트워크 등



이 혁 준

e-mail : hlee@daisy.kw.ac.kr

1987년 University of Michigan, Computer Science(학사)

1989년 Syracuse University, Computer Science(석사)

1993년 Syracuse University, Computer Science(박사)

1994년~1996년 삼성전자(주) 멀티미디어 연구소 선임연구원

1996년~현재 광운대학교 컴퓨터공학과 부교수

관심분야 : 무선네트워크, 이동 인터넷, 이동 애드 혹 네트워크 등