

# 실시간 OS 용 웹 브라우저의 설계

조수선, 김경일, 신규상  
{scho, kki, shin}@etri.re.kr  
한국전자통신연구원 컴퓨터·소프트웨어기술연구소  
실시간컴퓨팅연구부

## A Design of Web Browser for Real Time OS

Soosun Cho, Kyungil Kim and Kyusang Shin  
ETRI-CSTL Real-time Computing Department

### 요약

실시간 OS 용 웹 브라우저는 TV 셋탑 박스를 실행 환경으로 하므로 일반 데스크 탑용 웹 브라우저와 비교할 때 몇몇 주요한 특징을 가진다. 우선 실행시 상주되는 메모리 크기가 작아야 하며 신속한 웹 사이트 접근 및 적절한 형태로의 HTML 문서 변환이 필요하다. 이 특징들은 신속, 정확, 경량으로 실행되어야 하는 실시간 OS 응용의 특징에서 비롯된 것이다. 이러한 실시간 OS 용 웹 브라우저를 효과적으로 구현하기 위해서는 프락시 서버를 도입하여 분산 구조로 설계하는 것이 바람직하다. 본 논문에서는 일반 웹 브라우저와는 구별되는 성격을 지닌 실시간 OS 용 웹 브라우저에 대한 제약 사항 및 기능적 요구 사항을 정리하고 분산 구조에 기초한 시스템의 설계를 소개하고자 한다.

### 1. 서론

최근, DTG 가 출현하고 휴대 전화가 이동형 정보 단말기로 자리잡아가면서 가전과 컴퓨터의 경계가 차츰 없어지고 있다. 한편 현대인의 인터넷에 대한 수요는 날로 높아져 이런 기기들을 이용하여 편리하게 정보를 검색하려는 욕구가 급증하고 있다. 웹은 인터넷 상의 정보 검색 방법으로 압도적인 위치를 차지하고 있고 이러한 웹 환경을 컴퓨터 환경이 아닌 TV 등 가전 제품을 이용하여 일반 가정에서 손쉽게 사용하고자 하는 시도는 국내외에서 빈번하게 이루어지고 있다[2][4].

본 논문에서는 국내 기술로 만들어진 실시간 OS 환경 위에서 작은 실행 메모리로, 간편하고 쉽게 사용될 수 있는 웹 브라우저의 설계에 대하여 다룬다<sup>1</sup>. 이 웹 브라우저는 대상 환경이 TV 이므로 데스크 탑용 일반 웹 브라우저에 비해 여러 가지 제약 사항을 가진다. 본 논문에서는 먼저 이러한 제약 사항과 그에 따른 TV 용 웹 브라우저 고유의 요구 사항을 정의하며, 이러한 요구 사항을 만족시키면서 실시간 OS 상에서 효율적으로 실행될 수 있는 웹 브라우저의 설계를 소개하고자 한다.

### 2. 제약 사항 및 요구 사항

#### 2.1 TV 환경에 따른 제약 사항

개발하고자 하는 웹 브라우저는 일반 아날로그 TV

모니터를 출력 장치로 하고 마우스, 키보드 대신 리모콘을 입력 장치로 사용한다. 실시간 OS 는 셋탑 박스에 설치되어 TV 와 연결된다. 이러한 실행 환경으로부터 아래와 같은 제약 사항이 따른다.

#### ● 화면 디스플레이 모드의 제약

일반 NTSC TV 화면은 640\*480 의 해상도가 지원된다. 이것은 컴퓨터 모니터에 비해 턱없이 낮은 해상도이며 TV 는 2 미터 이상의 거리에서 시청해야 하므로 한 화면에 표시되는 텍스트, 이미지 등의 정보량은 매우 제한될 수밖에 없다.

#### ● 리모콘을 사용한 입력 방식의 제약

TV 를 통해 웹 탐색을 할 때, 사용자는 가정에서 소파에 앉아 TV 프로그램을 시청하는 것과 같이 리모콘 하나만으로 원하는 기능이 모두 동작되기를 원한다.

#### ● HTML 획득 및 표현 속도의 제약

TV 를 보고있는 사용자는 TV 화면이 표시되는 것과 같은 속도로 웹 화면이 나타나기를 기대한다. 실시간 OS 용 웹 브라우저에서는 TV 화면이 바뀌는 것과 같은 자연스러운 속도가 제공되어야 하는 제약이 따른다.

#### 2.2 기능적 요구 사항

실시간 OS 용 웹 브라우저는 일반 웹 브라우저와 거의 같은 기능적 요구 사항을 가진다.

#### ● 기존 웹 페이지의 브라우징 기능

● 대표적인 멀티미디어 데이터 지원 기능 (텍스트, 이미지, 오디오, 비디오 등)

<sup>1</sup> 본 연구는 "조립형 실시간 OS 개발 사업"의 세부과제로 이루어졌으며 정보통신부로부터 연구비 지원을 받고 있다.

- 항해를 위한 컨트롤 기능 (Home, Back, Forward, Go to, Reload, Bookmark, History list 등)
- 환경 설정 기능 (서버 설정, 보기 옵션 설정 등)
- 간편 접속 기능

이러한 일반적인 요구사항 외에 앞에서 살펴본 TV 환경에 따른 제약 사항 때문에 다음과 같은 요구 사항이 더 추가된다.

- TV-HTML 변환 기능  
기존의 웹 문서를 TV 화면에 적절히 표시하기 위해서는 일반 HTML 이 TV-HTML[3]로 변환되어야 한다.
- 리모콘으로 기본 항해할 수 있는 기능  
웹 검색의 기본적인 상호작용인 Home, Back, Forward 와 같은 기능은 리모콘 버튼 하나로 처리되어야 한다.
- 화면 키보드 지원 기능  
URL 주소를 입력하는 등 기본적인 텍스트 입력을 위해 화면 키보드를 제공하여야 한다.
- TV 로의 순간 화면 전환 기능  
웹 검색 도중 언제든지 간편하게 TV 화면으로 전환할 수 있어야 하고 그 반대도 가능해야 한다.
- 필터링 기능 (유해 정보, 불필요 정보 차단)  
유해 정보 또는 불필요 정보의 차단을 위해 미리 제어 및 필터링을 해 주어야 한다.

### 2.3 비기능적 요구사항

실시간 OS 는 빠르고 정확해야 하며 경량의 메모리로 운용되어야 한다. 웹 브라우저를 비롯한 실시간 OS 용 어플리케이션은 보통 OS 커널과 함께 컴파일되어 하나의 프로그램으로 메모리에 상주하게 되므로 그에 따른 요구 사항으로 빠른 성능과 효율적인 메모리 관리가 필요하다.

많은 경우 실시간 OS 가 탑재되는 셋탑 박스에는 하드디스크 등의 보조 메모리를 채용하지 않는다. 주 메모리와 함께 플래시 메모리가 사용되며 경우에 따라서 시스템이 부팅될 때 어플리케이션이 바로 다운로드되는 방식으로 사용되기도 한다. 이러한 환경에서 빠른 실행을 위해서는 셋탑 박스의 주 메모리에 상주하는 웹 브라우저의 크기는 수백 Kbyte 이내로 제한되어야 하며 보조적인 기능은 외부 장치에 분산 수행되어야 한다.

## 3 설계

### 3.1 분산 구조

빠른 성능 및 효율적인 메모리 관리에 대한 요구 사항을 수용하기 위하여 웹 브라우저 시스템을 'HTTP 프락시 서버', 'HTML 엔진', 'HTML 브라우저'의 3 가지 서브시스템으로 구성한다. HTTP 프락시 서버는 TV 셋탑과 네트워크를 통해 연결되어, 일반적인 프락시 서버의 역할을 모두 포함하고 TV 용 셋탑을 위한 특별한 기능이 추가된다. 셋탑 내에서는 HTML 브라우저와

HTML 엔진으로 기능 모듈이 구분되어 각각의 서브 시스템으로 구현된다.

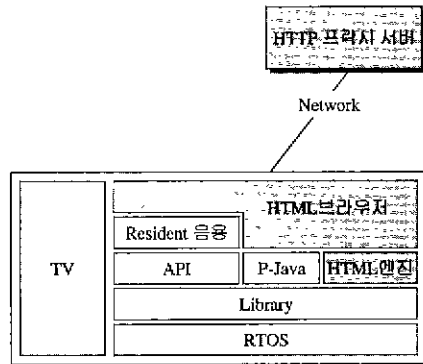


그림 1. 웹 브라우저 시스템의 개념적 구조

### 3.2 프락시 서버

웹 브라우저 시스템의 분산 구조하에서 기본 구성요소인 프락시 서버는 아래와 같은 기능을 수행한다.

- 일차적인 지역 콘텐츠를 위한 정보 저장소
- 외부 웹사이트와 TV 셋탑간의 버퍼
- 획득된 콘텐츠의 제어 및 필터링
- TV 화면에 적합하도록 콘텐츠 변환
- TV-HTML 변환

이 기능들 중 첫번째, 두번째는 일반적인 프락시 서버의 기능으로서 특별할 게 없다[1]. 그러나 나머지 세가지 기능은 TV 환경의 웹 브라우저에서 특별히 필요한 프락시 서버의 기능이다.

획득된 콘텐츠의 제어 및 필터링은 유해 정보를 위한 기능이다. HTTP 프락시 서버 레벨에서 유해 정보를 차단할 수 있는 필터링 방법은 주소 차단, 단어 차단, Whitelist 를 제외한 차단, 카테고리 차단 등 여러가지가 있다. 이러한 방법 중에서 본 시스템에서는 가장 보편적으로 사용되고 있는 주소 차단을 사용한다. 유해 정보가 포함되어 있는 사이트들의 주소를 HTTP 프락시 서버에 보관하고 있다가 외부 서버의 주소가 이곳에 포함되어 있으면 접근을 중단하는 것이다.

콘텐츠 변환 기능은 기존의 웹 문서 내용을 TV 화면에 적합한 형태로 바꾸어주는 기능이다. 출력을 위한 TV 화면이 640\*480 의 해상도에 256 칼라만을 지원하므로 이미지 처리에서 많은 부분이 고려되어야 한다. 이미지를 TV 화면에 적합한 형태로 변환하기 위해서는 먼저 256칼라가 넘는 이미지를 256칼라에 맞추어야 하고 낮은 해상도에서 볼 수 있도록 이미지 면적(픽셀 수)을 줄여야 한다. 경우에 따라서 이미지의 크기(Byte 수)를 줄이기 위한 포맷 변환도 필요하다.

TV-HTML 변환에서는 먼저 지원되지 않는 태그가 제

거되고 일반 HTML에서 TV-HTML로의 태그 변환이 이루어진다. 표준화 단계가 진행중인 TV-HTML 태그들은 WebTV, NetChannel, Sun CTG와 NCI에서 사용하고 있는 것들이 주류를 이루고 있다[3]. 이중 가장 범용적이고, 많은 태그들을 지원하는 WebTV에서 사용하고 있는 태그 리스트를 위주로 정하고 이 포맷에 맞는 형태로 변환을 수행한다.

이상과 같이 TV 화면에 웹 페이지를 출력하기 위한 많은 준비 작업이 프락시 서버를 통해 이루어짐으로써 셋탑에서 실행되는 웹 브라우저는 실행시 메모리 부담을 줄이면서 보다 빠른 성능으로 동작할 수 있다. 이 부분은 웹 브라우저의 분산 구조가 얻을 수 있는 가장 큰 효과이기도 하다.

### 3.3 시스템의 작동 모형

3가지 서브 시스템으로 이루어진 웹 브라우저 시스템이 작동하는 개략적인 순서는 다음과 같다. 그림 2는 작동 모형을 동작 흐름도로 표현한 것이다.

- 네트워크를 통하여 HTTP 프락시 서버에 on-demand 세션을 생성한다. 이 때, 사용자 인증이 필요하다.
- 브라우저가 HTML 엔진을 통하여 원하는 오브젝트를 요청한다. 요청된 오브젝트가 셋탑에 없으면 HTTP 프락시 서버로 요청이 넘어간다.
- HTTP 프락시 서버는 올바른 요청이면 요청을 인증하고 오브젝트를 로컬 캐쉬에서 찾는다. 로컬 캐쉬에서 찾을 수 없으면 외부 웹 서버를 통하여 오브젝트를 검색한다.
- 외부에서 얻어진 오브젝트는 포맷 변환하여 캐쉬에 저장한 후 HTML 엔진으로 되돌려 준다.
- 오브젝트는 HTML 브라우저로 전달되고 브라우저는 다시 HTML 엔진을 호출하여 전달된 오브젝트를 사용자에게 전시한다.

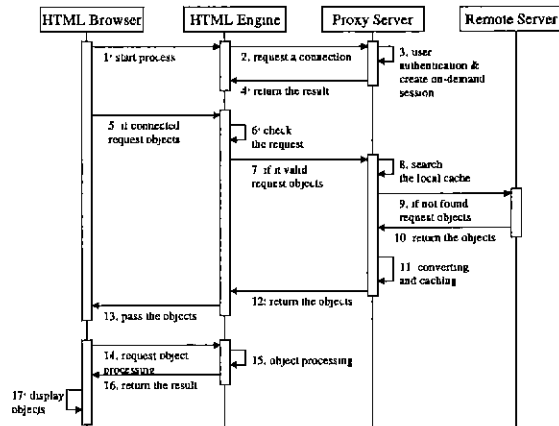


그림 2. 웹 브라우저 작동 모형

### 3.4 인터페이스

실시간 OS 용 웹 브라우저는 기본적으로 리모콘만으로 표준 한해 기능을 수행해야 하므로 무엇보다 직관적이고 편리한 인터페이스가 제공되어야 한다[5]. 인터페이스 설계는 리모콘의 설계와 밀접한 관련이 있다. 따라서 리모콘의 기능 버튼과 웹 브라우저 화면 상의 버튼을 정의해 보면 아래 표와 같다.

표 1. 리모콘과 웹 브라우저 화면의 버튼 정의

기능 정의	리모콘 버튼	화면 버튼
다른 응용로의 이동	TV, 메일, 부가 정보 이동	없음
설정	TV 환경 설정	웹 환경 설정
웹 브라우저 고유 기능	홈, 최근 사이트, 뒤로, 앞으로, 방향(상,하,좌,우), 확인	북마크, 재접속, 검색, 접속중단

표 1에서처럼 웹 브라우저 화면에서는 리모콘 버튼에 포함되어 있는 기능은 제외하여 낮은 해상도의 TV 화면에서 가장 단순하고 직관적인 화면을 디자인해야 하는 필요성을 만족시킨다.

### 4 결론

본 논문에서는 TV 셋탑 박스를 실행 환경으로 하는 실시간 OS 용 웹 브라우저의 개략적인 설계에 대하여 소개하였다. TV 환경의 웹 브라우저는 일반 웹 브라우저와 비교할 때 많은 부분이 서로 다르고 여러 가지 제약 사항도 있음을 감안하여 이러한 제약 사항 내에서 가장 효율적인 아키텍처로 시스템을 설계하고자 하였다.

이를 위해 프락시 서버를 도입한 분산 구조를 채택하였으며 프락시 서버의 기능에 유해 정보 차단, 콘텐츠 변환, HTML 변환 등을 포함시킴으로써 TV 환경의 웹 브라우저를 위한 고유의 기능을 제공함은 물론 프로세스를 적절히 배치하여 전체 시스템의 성능 향상을 위해 노력하였다.

### 5 참고문헌

- [1] Ari Luotonen, "Web Proxy Servers", Prentice Hall, 1998.
- [2] PowerTV Inc., "Product Overview - Web Browser Revision 1.04", Jan. 6. 1998.
- [3] Joseph T. Sinclair, "Developing Web Pages with TV-HTML", 1998.
- [4] Spyglass, "Spyglass Device Mosaic 3.1", <http://www.spyglass.com/solutions/technologies/devicemosaic/>
- [5] WebTV, "WebTV Design Guide", Feb. 26. 1999, [http://developer.webtv.net/design/designguide/1\\_Overview\\_fm.html](http://developer.webtv.net/design/designguide/1_Overview_fm.html)