

# 리눅스 기반의 WebDAV 클라이언트의 설계

황의운<sup>0</sup> 안건태 정혜영 김진홍 이명준  
울산대학교 컴퓨터정보통신공학부  
(heyoon<sup>0</sup>, java2u, hyjung, avenue, mjlee)<sup>0</sup>@ulsan.ac.kr

## Design of a WebDAV Client for Linux

Euiyoon Hwang<sup>0</sup>, Geontae Ahn, Hyeyoung Jung, Jinhong Kim, Myungjoon Lee  
School of Computer Engineering & Information Technology, University of Ulsan

### 요 약

WebDAV(Web-based Distributed Authoring and Versioning)는 웹상의 공동 저작활동을 지원하기 위한 IETF 표준으로서, 멀리 떨어져 있는 사용자들 간에 인터넷상에서 파일을 공동 편집하고 관리할 수 있도록 해주는 일련의 HTTP 확장이다. 이것은 웹상에서 가상의 작업공간을 구성함으로써, 원격 그룹들 간에 새로운 방식으로 공동 작업을 가능하게 한다. WebDAV 클라이언트는 이러한 명세를 지원하는 WebDAV 서버와 HTTP 요청을 통하여 서버 자원에 대한 변경 및 수정 작업을 수행하게 된다. 따라서 클라이언트는 뛰어난 동시성 제어와 속성 관리를 통하여 자원의 저작과 버전관리의 기능을 제공하는 것이 바람직하다.

본 논문에서는 기존의 몇 가지 WebDAV 클라이언트를 분석하여 리눅스 기반의 오픈소스 WebDAV 클라이언트를 설계하였다. 이것은 사용자의 수작업에 의한 문서 저작 절차를 단순화하고 저작 응용프로그램의 모니터링을 통하여 문서 변경에 대한 능동적인 서버 갱신을 지원하도록 설계하였다.

### 1. 서 론

인터넷의 발달로 인하여 지역적으로 원거리에 위치한 다수의 작업자 그룹들이 공동 작업을 수행할 수 있는 다양한 기술들이 개발되었다. 초기 공급업체들은 협업을 지원하기 위하여 HTTP(Hyper-Text Transfer Protocol) 프로토콜을 활용하거나 또는 각자의 고유 프로토콜들을 HTTP 프로토콜에 추가 정의하는 형태로 구현하였다. 하지만, HTTP 프로토콜에 각자의 고유 프로토콜의 추가를 통한 협업 지원은 공급업체들 간의 상호 운용성이 결여되는 문제가 발생하였다. 협업 시스템 공급업체들 간에 상호 운용성을 보장하기 위한 웹 기반의 분산 저작과 버전관리 표준의 필요성이 증가되었고, 이러한 요구를 충족시키기 위하여 W3C(World Wide Web Consortium)의 IETF 산하의 작업 그룹에 의해서 WebDAV 명세가 1999년 2월에 발표되었다[1].

WebDAV는 HTTP/1.1을 확장한 웹 통신 프로토콜로서 인터넷을 통하여 다양한 콘텐츠의 비동기적인 협업 제작을 지원하기 위한 표준 하부구조를 제공한다. WebDAV 명세를 지원하는 다양한 종류의 서버들은 상호간의 협업 작업이 가능하게 되었다. 클라이언트 응용 프로그램이 WebDAV를 지원함으로써 이기종의 WebDAV 서버들을 통하여 분산 저작을 수행할 수 있게 되었다.

WebDAV 클라이언트는 WebDAV 명세를 지원하는 서버의 자원을 HTTP(HyperText Transfer Protocol) 요청을 통하여 자원의 속성과 동시성의 문제를 해결하면서 자원의 저작(Authoring)을 지원하는 것이다. 현재 오픈소스 응용프로그램으로써 Cadaver[2], WebDAV Explorer[3] 등의 많은 프로그램들이 개발되어왔으나, 대부분은 무결성을 유지하면서 문서를 편집하는 절차에 대한 수작업적인 처리만을 지원한다.

본 논문에서는 사용자의 수작업에 의한 문서 저작 절차를 단순화하고 저작 응용프로그램의 모니터링을 통하여 문서 변경에 대한 능동적인 서버 갱신을 지원하는 리눅스 기반의 오픈소스 WebDAV 클라이언트를 설계하였다. 그리고 편리한 WebDAV 기

능 활용을 위하여 Drag & Drop을 통한 파일 이동 및 읽기 모드와 쓰기 모드의 파일 열기 등 다양한 추가 기능들을 설계하였다.

본 논문의 구성은 다음과 같다. 서론에 이어 2장에서 관련 연구를 통하여 WebDAV와 WebDAV 클라이언트 그리고 리눅스 기반의 응용프로그램 개발에 대하여 살펴보고 3장에서는 실제로 설계한 WebDAV 클라이언트에 대하여 기술하고, 끝으로 4장에서 결론과 향후 연구 방향에 대하여 기술한다.

### 2. 관련 연구

#### 2.1 WebDAV

WebDAV는 인터넷을 통하여 광범위하고 다양한 콘텐츠의 비동기적인 협업 제작을 지원하기 위한 표준 하부구조이다. 이는 HTTP/1.1 프로토콜의 확장을 통하여 사용자들에게 원거리 서버들의 파일들을 수정하고 관리할 수 있도록 한다[4].

WebDAV의 주요 기능은 덮어쓰기 방지(Overwrite prevention), 속성 관리(Property Management), 컬렉션(Collection), 이름 공간 관리(Namespace management) 등이 있다.

#### 2.2 WebDAV 클라이언트

WebDAV를 지원하는 클라이언트 제품군은 크게 3가지 형태로 분류할 수 있는데, 저작용 응용프로그램에서 직접 WebDAV 서버에 접속할 수 있도록 하는 형태와 운영체제의 파일시스템 드라이버처럼 작동하는 제품 그리고 WebDAV 서버 탐색기 형태로 나눌 수 있다. 제품들을 형태별로 분류하면 [표1]과 같이 분류할 수 있다.

#### 2.3 리눅스 기반의 응용프로그램 개발

X 윈도우 상에서 동작하는 응용프로그램을 만드는 방법은 GNOME 데스크탑에 사용하는 GTK+(Gimp Tool Kit)와 KDE 데스크탑에 사용하는 Qt로 나눌 수 있다. 가장 큰 차이는 Qt는 클래스화된 라이브러리고 GTK+는 C언어 중심의 개발 환경을 제공한다는 점이다. 차후에 Qt 라이브러리를 사용하여 클라이언트

\* 본 논문은 2003년 한국소프트웨어진흥원의 공개 S/W 프로젝트 활용 S/W 전문 인력 양성 사업의 지원으로 이루어졌음.

형태	관련 제품들
저작 응용 프로그램	-마이크로소프트사의 MS-Office 2000 -Adobe Photoshop 6, Acrobat 5, Go Live 5 -Micromedia Dreamweaver 4 -Excsoft Documentor
파일 시스템 드라이버	-SouthRiver Technology사의 WebDrive -Apple MacOS의 webdavfs -TeamStream사의 TeamDrive -마이크로소프트사의 WebDAV Redirector
탐색기	-마이크로소프트사의 Internet Explorer -U.C. Irvine의 DAV Explorer -Apple MacOS의 Goliath

[표1] 형태별 WebDAV 관련 제품들

를 구현할 것이므로 본 논문에서는 Qt를 사용하는 WebDAV 클라이언트를 설계하였다.

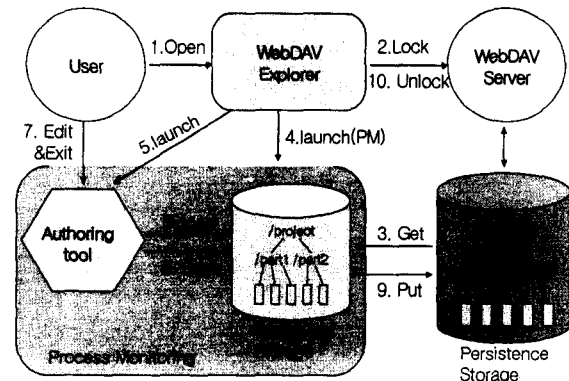
### 3. 리눅스 기반의 WebDAV 클라이언트 설계

WebDAV 클라이언트는 서버가 제공해주는 여러 가지 기능을 보다 잘 활용할 수 있도록 사용자에게 편리한 인터페이스를 제공해야 한다. 본 논문에서는 SkunkDAV[5] 와 WebDAV .Net Explorer[6]의 인터페이스를 분석하여 보다 편리한 구성을 가지도록 노력하였다. 그리고 리눅스 기반의 응용프로그램으로써 KDE 데스크탑에 사용하는 Qt 라이브러리를 사용하여 구현할 것이기 때문에 기능에 대하여 체계적으로 프로그램이 가능하도록 클래스 설계를 하였다.

#### 3.1 클라이언트 설계 모델

사용자가 WebDAV 클라이언트를 이용하여 서버의 문서를 자신의 컴퓨터에서 저작하기 위한 WebDAV 클라이언트의 내부 처리절차는 [그림1]과 같다.

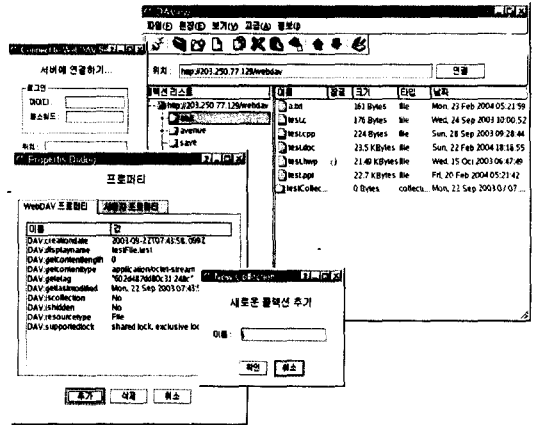
사용자는 WebDAV 클라이언트를 통하여 WebDAV 서버의 이름 공간 정보를 확인하고 저작을 원하는 자원을 열기 명령을 수행하면 클라이언트는 서버의 자원에 대하여 잠금 처리를 하고 로컬의 임시 저장소에 받기를 하고 프로세스 모니터링 기능을 구동한 후, 해당 문서의 저작 도구를 구동하게 된다.



[그림1] WebDAV 클라이언트의 내부 처리절차

### 3.2. 인터페이스 설계

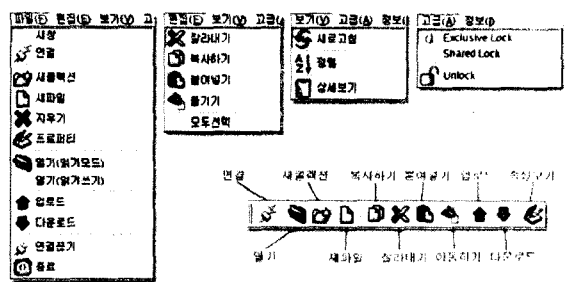
WebDAV 클라이언트의 인터페이스는 Java기반의 SkunkDAV 와 .Net기반의 WebDAV .Net Explorer를 참고하여 보다 사용하기 편리하도록 설계하였다. [그림2]는 Qt designer를 사용하여 설계한 인터페이스를 보여준다. 메인화면은 메뉴바, 툴바, 주소 연결 패널, 콜렉션 트리, 선택된 콜렉션의 목록 리스트로 구성된다.



[그림2] 설계된 클라이언트 인터페이스

#### 3.2.1 메뉴바와 툴바

메뉴바는 클라이언트의 기능을 모두 알 수 있는 구성요소이고, 툴바는 보다 편리하게 메뉴를 사용할 수 있도록 도와주는 구성요소이다. 메뉴는 파일, 편집, 보기, 고급 등의 메뉴 아이템으로 구성된다. [그림3]은 설계된 메뉴와 툴바의 자세한 사항을 보여준다.



[그림3] 메뉴바와 툴바

### 3.3. 기본기능 설계

기본기능은 WebDAV 서버가 기본적으로 제공하는 기능을 클라이언트가 잘 활용할 수 있도록 설계하였다. [그림4]에 나타나 있듯이, 파일에 대한 속성 편집 기능, 파일 편집 기능, 그리고 WebDAV 리소스에 대한 잠금 기능 등 기본적으로 수행해야 할 기능들을 포함한다. 그리고 추가적으로 콜렉션 및 개체 리소스에 대한 복사, 이동 및 삭제 기능과 업로드, 다운로드 기능도 있다. 속성 편집은 새로운 다이얼로그 박스를 생성하여

속성을 추가, 삭제 및 수정이 가능하도록 하였고([그림4]), 파일 편집은 쓰기 모드로 파일 열기를 하면 수행되도록 하였다. 그 외 새로운 콜렉션 추가 다이얼로그 박스, WebDAV 서버 연결 다이얼로그 박스 등의 추가적인 인터페이스도 설계하였다.



[그림4] 속성(Properties) 다이얼로그

3.4. 고급기능 설계

클라이언트의 고급 기능은 편리한 파일 업로드와 다운로드를 위하여 Drag & Drop을 사용한 파일 이동을 가능하게 하는 기능과 파일을 편집할 때 자동으로 응용프로그램에 연결하는 기능, 프로세스 모니터링을 통한 자동 저장, 잠금, 해제를 하는 기능 등으로 분류할 수 있다.

3.4.1. Drag & Drop을 사용한 파일 이동

Drag & Drop을 사용한 파일 이동은 사용자들에게 편리한 파일 업로드와 다운로드가 가능하게 한다. 따라서, 본 논문에서 설계한 리눅스 기반의 WebDAV 클라이언트에서도 가능할 수 있도록 클래스를 설계하였다. KDE 기반의 Qt라이브러리와 각종 이벤트 핸들러를 통해서 구현이 가능하다. [표2]는 설계한 클래스의 세부사항을 보여준다.

메소드 명	설명
contentsDragEnterEvent	Contents가 응용프로그램에 드래그 되어 들어올 때 처리
contentsDragMoveEvent	Contents가 응용프로그램에서 드래그 되어 움직일 때 대상 위치 확인
contentsDragLeaveEvent	Contents가 응용프로그램에서 드래그 되어 영역 밖으로 나갈 경우 발생 이벤트 처리
contentsDropEvent	Contents가 응용프로그램에 드롭 되었을 때 ActionType에 따른 작업 수행(이동 또는 업로드)
contentsURL	Contents의 URL정보를 가지고 있음
- QTLIB_EVENT, KDELib를 이용 - QDragEnterEvent, QDragMoveEvent, QDragLeaveEvent, QDropEvent 처리 - KIO::Job, KURL::List, KOperation	

[표2] Drag & Drop을 사용한 파일 이동의 클래스 설계

3.4.2. 응용프로그램 연결

파일을 편집하기 위하여 열기를 수행할 시 적합한 응용프로그램으로 연결하고 자동 실행을 하기 위해서 필요한 기능이다. 이 기능은 사용자에게 빠르고 정확하게 파일을 수정할 수 있도록 해주며, 내부적으로 프로세스 모니터링을 통하여 자동 저장과 리소스 잠금 및 해제를 통하여 편리하게 파일 편집이 가능하도록 지원한다. [표3]는 이러한 클래스에 대해 설계한 것을 보여준다.

메소드 명	설명
isLocalFile	해당 리소스의 URL이 로컬인지 URL인지 위치가 로컬인지 검사
setPreferredApp	URL을 열기위한 우선적인 App등록
setRunExecutables	리소스가 실행 가능한지 여부를 설정
runURL	해당 URL에 대한 열기 수행
IsExecutable	데이터 파일인지 실행가능 파일인지를 검사
displayOpenWith	Dialog박스를 통하여 응용프로그램을 설정
foundMimeType	해당 MimeType이 존재하는지 검사
hasError	에러발생시 True값 리턴
-KDELib, KIO Lib 및 LibKonq	

[표3] 응용프로그램 연결을 위한 클래스 설계

4. 결론 및 추후연구

본 논문에서는 기존의 WebDAV 클라이언트가 가지는 문제점을 개선하면서 유용성을 높이기 위하여 리눅스 기반의 클라이언트를 설계하였다. 사용자의 수작업에 의한 잠금 처리와 같은 문서 저작 절차를 단순화하고 저작 응용프로그램의 모니터링을 통하여 문서 변경에 대한 능동적인 서버 갱신을 지원하도록 설계 되었다. 그리고 편리한 파일 업로드와 다운로드를 위하여 Drag & Drop을 사용한 파일 이동에 대한 클래스와 파일을 편집하기 위하여 열기를 수행할 시 적합한 응용프로그램으로 연결하고 자동 실행을 하기 위한 클래스 설계를 하였다.

향후 연구과제로 본 논문에서 설계한 리눅스 기반의 WebDAV 클라이언트의 구현과 임시 영역에 의한 자원관리에 대한 보안성 문제를 해결하는 연구, 버전관리에 관한 지원, 다양한 WebDAV 서버군에서의 적용 시험 등을 수행할 계획이다.

5. 참고문헌

[1] Y. Goland, E. Whitehead, A. Faizi, S. Carter, D. Jensen, "HTTP Extensions for Distributed Authoring - WEBDAV", RFC 2518, Standards Track, February, 1999  
 [2] GNU GPL, <http://www.webdav.org/cadaver/>  
 [3] Gerair Balian, Mirza Baig, Robert Emmery, Thai Le, Tu Le, DAV Explorer project, <http://www.ics.uci.edu/~webdav/>  
 [4] Jim Whitehead, Yaron Y. Goland, "WebDAV: A network protocol for remote collaborative authoring on the Web", ECSCW'99, 1999  
 [5] Jacob Smullyan, "SkunkDAV DAV Client", <http://skunkdav.sourceforge.net/>,  
 [6] 정혜영, 송동호, 윤병훈, 안건태, 이명준, "닷넷 기반의 WebDAV 탐색기의 개발", "2003년 한국정보과학회 가을 학술발표논문집, Vol. 30, No. 2, p43-45"