DAView : 리눅스 WebDAV 클라이언트

황의윤[0] 신원준 안건태 정혜영 이명준
울산대학교 컴퓨터정보통신공학부
{heyoon[0], mathpf, java2u, hyjung, mjlee}@ulsan.ac.kr

DAView : a WebDAV Client for Linux

Euiyoon Hwang[0], Wonjoon Shin, Geontae Ahn, Hyeyoung Jung, Myungjoon Lee
School of Computer Engineering & Information Technology, University of Ulsan

요 약

WebDAV (Web-based Distributed Authoring and Versioning, RFC 2518)는 웹 통신 프로토콜인 HTTP/1.1의 확장으로 인터넷을 통하여 다수의 그룹간의 다양한 콘텐츠의 비동기적인 협업을 지원하기 위한 표준 하부구조를 제공한다. WebDAV 클라이언트는 이러한 명세를 지원하는 WebDAV 서버와 HTTP 요청을 통하여 서버 자원에 대한 변경 및 수정 작업을 수행하게 된다. 따라서, 클라이언트는 리소스 종류에 따른 저작 응용프로그램의 자동 구동과 속성 관리를 통하여 자원의 저작과 버전관리의 기능을 제공하는 것이 바람직하다.

본 논문에서는 기존의 WebDAV 클라이언트가 가지는 저작 응용프로그램의 구동과 속성 관리에 대한 문제점을 개선한 리눅스 기반의 WebDAV 클라이언트를 개발하였다. 개발된 클라이언트는 서버의 자원에 대한 저작 응용 프로그램의 실행과 적절한 잠금 제어를 지원한다. 또한, 프로세스 감시를 통하여 사용자에 의한 자원의 변경이 완료되었을 때 서버 자원에 대한 자동 갱신을 지원한다.

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. 그러나 HTTP 프로토콜에 각자의 고유 프로토콜의 추가를 통한 협업 지원은 공급업체들 간의 상호 운용성이 결여되는 문제가 발생하였다[1,2]. 따라서 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*(Internet Engineering Task Force) 산하의 작업 그룹에 의해서 WebDAV 명세가 1999년 2월에 발표되었다[3].

WebDAV는 HTTP/1.1[4]을 확장한 웹 통신 프로토콜로서 인터넷을 통하여 다양한 콘텐츠의 비동기적인 협업 제작을 지원하기 위한 표준 하부구조를 제공한다. WebDAV 명세를 지원하는 다양한 종류의 서버들이 상호간의 협업 작업이 가능하게 되었고 클라이언트 응용 프로그램들이 WebDAV를 지원함으로서 이기종의 WebDAV 서버들을 통하여 분산 저작을 수행할 수 있게 되었다.

WebDAV 클라이언트의 기능은 WebDAV 명세를 지원하는 서버의 자원에 대한 HTTP 요청을 통하여 자원의 속성과 동시성의 문제를 해결하면서 자원의 저작을 지원하는 것이다[1]. WebDAV를 지원하는 클라이언트 제품군은 크게 3가지 형태로 분류할 수 있는데, 직접 WebDAV 서버에 접속할 수 있도록 하는 저작 응용 프로그램(Authoring Application) 형태, 운영체제의 파일 관리자나 네트워크 연결 드라이브로 작동하는 형태 및 GUI 환경의 WebDAV 서버 탐색기 형태로 나눌 수 있다. GUI를 제공하는 WebDAV 탐색기 형태의 클라이언트는 필요시 클라이언트의 구동을 통하여 WebDAV 서버에 접속할 수 있어서 보안성을 효과적으로 유지할 수 있으며 작업 공간의 자원 관리와 속성 관리가 용이하다. 하지만 네트워크 연결 드라이브 형태의 WebDAV 클라이언트는 네트워크 연결을 위한 메모리 상주 프로그램의 구동으로 항상 연결된 상태로 유지하고 있기 때문에 사용자 시스템의 메모리 자원에 대한 불필요한 점유와 보안 문제를 야기할 수 있다. 대부분의 상업적인 제품들은 네트워크 연결 드라이브 형태의 클라이언트로 WebDAV 서버 자원의 공유를 목적으로 하고 있다.

본 논문은 네트워크 드라이브 연결 없이도 클라이언트의 구동만으로 WebDAV 서버에 접속할 수 있으면서 사용자의 편리성을 위하여 리눅스의 KDE 윈도우즈 환경 기반의 탐색기 형태로 WebDAV 클라이언트를 개발하였다. 이것은 기존의 클라이언트가 가지는 수작업 처리에 의한 저작의 문제점을 개선하여 WebDAV 서버 문서에 대하여 마우스 더블 클릭 또는 문서 열기만으로 WebDAV 서버 문서에 대한 잠금 관리와 저작 응용 프로그램의 자동 구동을 제공한다. 클라이언트의 프로세스 감시를 통하여 문서 변경 상태에 대한 실시간 감시와 구동중인 저작 응용 프로그램에 의한 문서의 변경에 대하여 해당 서버로의 능동적인 갱신이 가능하도록 하였다.

본 논문의 구성은 다음과 같다. 서론에 이어 2장에서 관련 연구의 소개를 통하여 WebDAV에 대한 소개 및 KDE 라이브러리와 Qt 라이브러리에 대하여 간단히 살펴보고 3장에서는 리눅스 기반의 WebDAV 클라이언트의 개발에 대해서 기술한다. 끝으로 4장에서 결론과 향후 연구 방향에 대하여 기술한다.

2. 관련 연구

2.1. WebDAV

WebDAV는 인터넷을 통하여 광범위하고 다양한 콘텐츠의 비동기적인 협업 저작을 지원하기 위한 프로토콜이다. WebDAV는 HTTP/1.1 프로토콜의 확장을 통하여 사용자들에게 원거리 서버들의 파일들을 수정하고 관리할 수 있도록 한다. WebDAV 표준 명세는 W3C의 IETF WebDAV 작업 그룹에 의해서 1999년 2월

* 본 연구는 한국과학재단 목적기초연구(R05-2004-000-10662-0)의 지원으로 수행되었음.

에 발표되었다.

WebDAV의 주요 기능으로는 잠금 관리(Lock Management), 속성 관리(Property Management), 컬렉션(Collection), 이름 공간 관리(Namespace management) 등이 있다.

2.2. KDE와 Qt

리눅스의 X윈도우 상에서 동작하는 응용프로그램을 만드는 방법은 GNOME 데스크탑에 사용하는 GTK+(Gimp Tool Kit)]와 KDE 데스크탑에 사용하는 Qt로 나눌 수 있다. Qt는 객체지향 방식의 클래스화된 라이브러리이고 GTK+는 C언어 중심의 개발 환경을 제공한다.

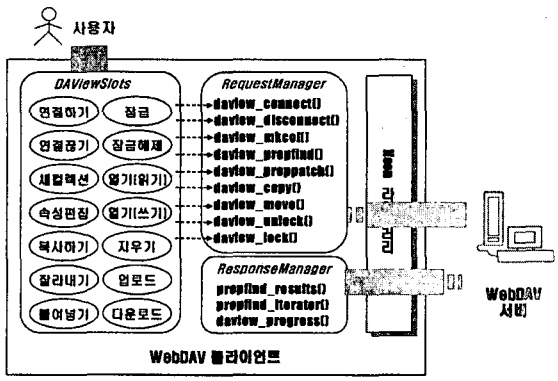
KDE는 유닉스 워크스테이션용 오픈소스 그래픽 데스크탑 환경이다. 이는 완전한 GUI와 함께, 파일관리자, 윈도우 관리자, 도움말 시스템, 구성관리 시스템, 도구 및 유틸리티, 그리고 여러 개의 응용프로그램 등을 포함하고 있다.

3. 리눅스 기반의 WebDAV 클라이언트의 개발

본 논문에서 개발한 WebDAV 클라이언트는 DAView 라고 명명하였다. DAView 프로그램은 기본적으로 Qt 라이브러리를 사용하여 사용자 인터페이스를 구성하였고, 저작 응용프로그램의 자동 구동과 Drag&Drop 기능을 구현하기 위하여 KDE 라이브러리를 사용하였다. 그리고 neon 라이브러리[5]를 사용하여 WebDAV서버와 통신을 하고 결과를 처리한다.

3.1. 클라이언트 구조

클라이언트 구조는 다음 [그림 1]과 같다. 메인 윈도우에서 사용자에 의해 명령이 내려졌을때 *DAViewSlots* 클래스는 사용자가 보낸 시그널을 받아들여 *RequestManager*에 전달한다. 이때 그 시그널에 해당하는 DAViewSlots 클래스의 슬롯이 동작한다. RequestManager는 전달되어진 명령과 요청의 내용에 따라서 작성한 neon 라이브러리의 함수를 호출한다.



[그림 1] 클라이언트의 구조

3.2. neon 라이브러리의 사용

neon은 C언어로 구성된 HTTP와 WebDAV 클라이언트를 위한 라이브러리이다. HTTP와 WebDAV 메소드를 구현하기 위해서 유연하고 비교적 간단한 API를 제공하고 있으며, cadaver, daview 등 몇 가지 클라이언트에서 WebDAV 메소드의 구현을 위해 사용하고 있다.

neon 라이브러리를 사용하기 위해서는 neon에서 요구하는 형식에 맞게 프로그램을 작성하여야 한다. 따라서 서버에 요청을 하기 전에 클라이언트에서 적절한 형식에 맞게 neon의 함수를 호출한다. neon API를 사용하는 메소드는 RequestManager 클래스에 정의되어 있다. 아래의 [그림 2]는 RequestManager의 daview_lock() 메소드에서 neon을 사용하여 서버의 특정 자원에 LOCK을 요청하는 예이다.

```
void RequestManager::daview_lock (DAVData *data) {

  // Lock이 수행되어지는 리소스에 대한 URI를 가져옴.
  char *res_uri = ne_strdup(data->getPath());

  // ne_lock 구조체의 포인터 생성 및 초기화
  struct ne_lock *lock;
  lock = ne_lock_create ();
  lock->owner = ne_strdup("daview");
  lock->depth = NE_DEPTH_INFINITE;
  ne_fill_server_uri(sess, &lock->uri);

  // 실제로 LOCK을 수행하는 URI를 구조체에 추가
  lock->uri.path = ne_strdup(res_uri);

  // LOCK이 성공적으로 수행되면 lockstore에 추가
  // 실패하면 lock destroy
  if ( ne_lock (sess, lock) == NE_OK ) {
    ne_lockstore_add (DAView::lock_store, lock);
  } else {
    ne_lock_destroy (lock);
  }
}
```

[그림 2] neon을 이용하여 LOCK메소드 호출

3.3. 기본기능

기본기능은 WebDAV의 기본 메소드를 구현한 것이다. 예를 들어, PUT, GET, LOCK, UNLOCK, PROPPATCH, PROPFIND 등의 기능이 있다. Qt기반의 GUI프로그램으로써 시그널/슬롯 방식과 neon 라이브러리를 적절하게 사용하여 구현하였다. 다음의 [그림 3]은 시그널/슬롯 관계를 설정하는 것이고, [그림4]는 neon라이브러리를 사용하여 WebDAV서버에 연결을 하고 첫 PROPFIND 요청을 하는 것이며, [그림 5]는 서버에 연결했을때 PROPFIND 요청에 대한 결과를 콜백 메소드를 사용하여 처리하는 것이다.

```
/* 시그널/슬롯 관계설정 */
connect (connectAction, SIGNAL(activated()),
                dvSlots, SLOT(openConnDialog()));
```

[그림 3] 시그널/슬롯 관계 설정

```
QString host_name = url.section( '/', 2, 2);
ne_sock_init();     // 소켓 초기화
/* neon 함수 사용 */
sess = ne_session_create("http", host_name, 80);
handler = ne_propfind_create(sess, ne_strdup(name), find_depth);
ne_propfind_allprop(handler,
                ResponseManager::propfind_results, NULL);
```

[그림 4] neon을 사용하여 서버에 연결

```
void ResponseManager::propfind_results(void *ud, const
                char *href, const ne_prop_result_set *rset) {
  // 결과를 분석
  // 메인 윈도우에서 콜렉션 트리 및 리소스 리스트 수정
}
```

[그림 5] 결과 처리 콜백메소드

3.4. 고급기능

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.

• **Drag & Drop**

Drag & Drop을 사용한 파일 이동은 사용자들에게 편리한 파일 업로드와 다운로드가 가능하게 한다. 이는 다른 KDE기반의 응용 프로그램에서 Drag 한 것을 본 논문에서 개발한 DAView 클라이언트에 Drop했을 때에도 유효한 것으로서 KDE의 K브라우저를 분석하여 개발하였다. Drag와 Drop 이벤트가 발생했을 때 일반적인 KDE 프로그램과 연동이 가능하도록 각각 이벤트에 맞는 자료구조와 그에 따른 행동을 결정하였다.

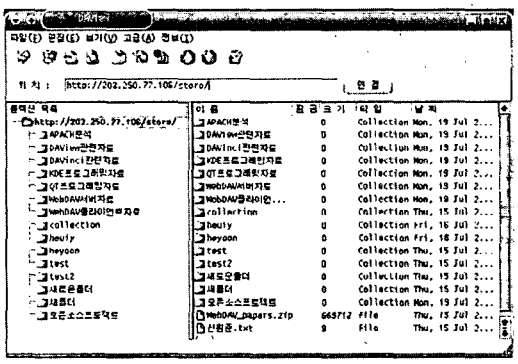
• **응용프로그램 자동 구동**

이 기능은 파일을 편집하기 위하여 열기를 수행할 시 적합한 응용프로그램으로 연결하고 자동 실행을 하는 기능이다. 사용자에게 빠르고 정확하게 파일을 수정할 수 있도록 해주며, 내부적으로 프로세스 모니터링을 통하여 자동 저장과 리소스 잠금 및 해제를 통하여 편리하게 파일 편집이 가능하도록 지원한다.

3.5. 사용자 인터페이스

• **메인 인터페이스**

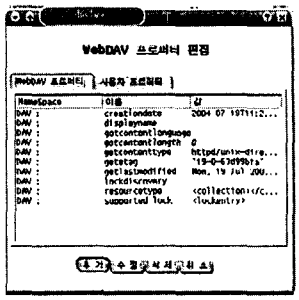
메인 인터페이스는 [그림 6]과 같이, 메뉴, 툴바, 연결바, 콜렉션 트리, 리소스 리스트로 구성된다.



[그림 6] 메인 인터페이스

• **속성 편집 다이얼로그**

WebDAV의 중요한 특성중에 하나인 속성에 대하여 추가, 삭제, 수정의 기능을 위해 편리한 인터페이스를 제공하며 속성을 WebDAV 속성과 사용자 정의 속성으로 나누어 관리하고 있다. [그림 7]은 속성편집 다이얼로그를 나타낸다.



[그림 7] 속성편집 다이얼로그

• **기타 다이얼로그**

여러가지 다른 명령을 수행하기 위한 다이얼로그가 있으며, 각각 해당하는 메뉴를 구동시켰을 때 동작한다. 예를 들어, 서버에 컬렉션을 생성하기 위해 컬렉션 이름을 입력받는 다이얼로그와 자원의 다운로드와 업로드를 돕기위한 파일 다이얼로그가 있다.

4. 결론 및 향후연구

본 논문에서 개발한 DAView 클라이언트는 기존의 클라이언트가 가지는 수작업 처리에 의한 저작의 문제점을 개선하여 WebDAV 서버 문서에 대하여 마우스 더블 클릭 또는 문서 열기만으로 WebDAV 서버 문서에 대한 잠금 관리와 저작 응용 프로그램의 자동화된 구동을 제공한다. 클라이언트의 프로세스 감시를 통하여 문서의 변경 상태에 대한 실시간 감시와 구동중인 저작 응용 프로그램에 의한 문서의 변경에 대하여 해당 서버로의 능동적인 갱신이 가능하도록 하였다. 또한 편리한 사용자 인터페이스를 통하여 다양한 보기 기능, WebDAV 서버 자원의 속성 관리, 잠금 관리, 복사 및 붙여넣기, 문서와 컬렉션 생성 및 문서 열기 등의 기능을 제공한다. 본 논문에서 개발한 WebDAV 탐색기와 타 WebDAV 클라이언트의 비교 결과, 사용자 인터페이스 부분과 프로세스 감시를 이용한 저작 응용 프로그램의 자동 구동을 지원하는 기능이 우수하다고 판단된다.

향후 연구과제로는 WebDAV 서버와의 통신 상황을 확인하기 위한 로깅 기능을 지원하고 WebDAV 클라이언트의 자원관리를 위한 임시 영역의 보안성 문제를 해결하는 연구와 확장된 버전관리의 지원 및 접근 제어(Access Control)를 지원하는 연구를 수행할 계획이다.

5. 참고 문헌

[1] E. James Whitehead, Jr., Meredith Wiggins, "WEBDAV: IETF Standard for Collaborative Authoring on the Web," IEEE Internet Computing, pp. 34-40, September/October 1998

[2] Jim Whitehead, Yaron Y. Goland, "WebDAV: A network protocol for remote collaborative authoring on the Web," ECSCW'99, 1999

[3] Y. Goland, E. Whitehead, A. Faizi, S. Carter, D. Jensen, "HTTP Extensions for Distributed Authoring - WEBDAV," RFC 2518, Standards Track, February, 1999

[4] R. Fielding, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1," RFC 2616, Standards Track, June, 1999

[5] Joe Orton, "neon HTTP/WebDAV client library", http://www.webdav.org/neon, March 2000