

DavSUDP: 웹데브 사용자 정의 속성의 활성화를 위한 단순 프로토콜

정 혜 영* · 안 건 태** · 유 양 우*** · 박 양 수**** · 이 명 준****

요 약

웹데브(WebDAV)는 HTTP/1.1의 확장된 프로토콜로서 인터넷을 통하여 분산된 저작과 버전관리를 지원한다. 웹데브의 주요 기능인 속성 관리는 저장소의 역할과 함께 자원의 추가 정보를 관리하고 활용할 수 있는 장점을 가지고 있다. 지금까지 대부분의 웹데브를 지원하는 응용 시스템은 속성 관리 기능의 활용보다는 단순 저장소의 역할로 웹데브를 사용하고 있다. 그러나 협업 지원 시스템과 같은 웹데브 기반의 고급 응용 프로그램을 개발하기 위해서는 웹데브 사용자 정의 속성의 설계와 활용이 매우 중요하다. 본 논문에서는 웹데브 사용자 정의 속성을 정의한 XML 형식의 환경 설정 속성과 그 처리 수행 절차를 기술한 *DavSUDP(WebDAV Simple User-defined Property Definition Protocol)* 프로토콜을 제안한다. DavSUDP는 웹데브 기반의 응용 시스템을 개발하는데 있어서 웹데브 서버가 사용자 정의 속성을 효과적으로 관리할 수 있게 하여준다. 이를 보이기 위하여 아파치의 mod_dav 모듈이 DavSUDP를 지원하도록 확장하였으며, 이를 이용하여 iPlace 협업 지원 시스템의 공개작업장을 개발하였다.

DavSUDP: A Simple Protocol for Utilizing WebDAV User-defined Properties

Hye-Young Jung* · Geon-Tae Ahn** · Yang-Woo Yu*** · Yang-Soo Park**** · Myung-Joon Lee****

ABSTRACT

WebDAV extends the web communication protocol HTTP/1.1 to provide a distributed authoring and versioning across the Internet. A property management, the principal ability of WebDAV, has many advantages for managing and utilizing additional information of server resources as well as supports a role of storage. Most WebDAV Application systems currently available are using WebDAV for providing simple storage space, not for the functionality of property management. However, to develop an advanced application such as a collaborative system based on WebDAV, it is very important to design and utilize user-defined WebDAV properties. In this paper, we propose *DavSUDP(WebDAV Simple User-defined Property Definition Protocol)* that describes XML-based configurations representing WebDAV user-defined properties and their processing steps. DavSUDP enables WebDAV servers to effectively manage user-defined properties in developing WebDAV-based applications. To show this, we extended the Apache mod_dav module to support DavSUDP and developed the OpenSpace in the iPlace collaborative system using the extended Apache web server.

키워드 : 웹데브(WebDAV), 속성 관리(Property Management), 사용자 정의 속성(User-defined Property), DavSUDP

1. 서 론

웹 서비스는 인터넷 사용자들에게 간결하고 동일한 사용자 인터페이스를 통하여 필요한 정보를 검색하고 얻기 위한 단순하고 편리한 수단을 제공한다. 이러한 웹의 장점을 이용하여 지역적으로 넓게 분산되어 있는 그룹의 구성원들 간에 정보공유 공간을 지원해 준다면 구성원간의 상호작용이 가능하다. 이를 위하여 보다 효과적인 iPlace[1], BSCW[2]과

같은 웹 기반 협업지원 시스템이 개발되어 왔다.

웹을 통하여 협업을 지원하기 위한 표준 프로토콜로는 기존의 HTTP(Hypertext Transfer Protocol) 프로토콜을 확장한 웹데브(WebDAV) 프로토콜이 있다. 웹데브는 HTTP /1.1을 확장한 웹 통신 프로토콜로서 인터넷을 통하여 다양한 콘텐츠의 비동기적인 협업 제작을 지원하기 위한 표준 하부 구조를 제공한다[3, 4, 5]. 웹데브를 지원하는 대표적인 서버 제품으로는 마이크로소프트사의 IIS 5.0[6], 아파치(Apache)의 mod_dav[7] 모듈, UCSC의 Catacomb[8], Software AG사의 Tamino Server[9] 등이 있으며, 클라이언트 제품으로는 마이크로소프트사의 MS-Office 2000[10], SouthRiver Technology사의 WebDrive[11], Xythos사의 WebFile Client [12], 닷넷기반 웹데브 탐색기[13] 등이 있다.

현재 기존의 다양한 웹 기반의 응용 시스템들(예를 들어,

※ 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT 연구센터 육성지원사업의 연구결과로 수행되었습니다.

교신처자 이명준(mjlee@ulsan.ac.kr)

*정 회 원 : (주)LG CNS S/W아키텍처센터 근무

**준 회 원 : 울산대학교 대학원 컴퓨터정보통신공학부

***준 회 원 : 울산과학기술대학교 컴퓨터정보학부 전임강사

****정 회 원 : 울산대학교 컴퓨터정보통신공학부 부교수

*****정 회 원 : 울산대학교 컴퓨터정보통신공학부 교수

논문접수 : 2004년 7월 20일, 심사완료 : 2004년 10월 15일

협업 지원 시스템, 문서관리 시스템, 그룹웨어, 인터넷 포털, 커뮤니티, 인터넷 저장소 등)이 웹데브 기술을 지원하도록 개발되고 있다. 만약 웹데브 기반의 협업 지원 시스템을 개발하려고 한다면 협업 시스템의 공동 작업 공간의 객체 모델을 웹데브의 이름 공간 구조로 변환하고 작업장에 대한 권한과 기능 등에 대한 주요 정보가 웹데브 속성으로 추가되어야 한다. 웹데브 속성은 시스템 정의 속성과 사용자 정의 속성으로 구분된다. 웹데브 명세에서는 이러한 속성을 라이브(live) 속성과 데드(dead) 속성으로 구분하여 표현하고 있다[5]. 클라이언트 기반의 응용 시스템에서 자원에 대한 특정 정보를 관리하기 위해서는 사용자 정의 속성을 이용하여야 한다.

본 논문에서는 웹데브 기반의 응용 시스템을 개발하는데 있어서 클라이언트 측에 의해서 관리되는 사용자 정의 속성을 효과적으로 설정하고 관리할 수 있는 XML 형식의 환경 설정 속성과 그 처리 수행 절차를 기술한 *DasSUDP (WebDAV Simple User-defined Property Definition Protocol)* 프로토콜을 제안한다. 또한 웹데브 서버의 컬렉션과 자원에 대한 추가된 환경 설정 속성을 이용하도록 웹데브 서버의 기능을 확장하였다. 확장된 웹데브 서버는 클라이언트에 의해서 관리되어야 할 사용자 정의 속성들의 관리 기능을 웹데브 서버 측에서 수행하도록 하였다. 결과적으로 사용자 정의 속성을 활용하려는 웹데브 기반의 고급의 응용 시스템들의 개발을 서버차원에서 체계적으로 지원할 수 있도록 하여 준다.

본 논문의 구성은 다음과 같다. 서론에 이어 2장에서는 관련 연구를 통하여 웹데브와 웹데브 속성에 대해서 살펴보고 실험 모델인 iPlace 시스템을 소개한다. 3장에서는 새롭게 정의된 환경 설정 속성의 정의와 이러한 속성을 처리하는 확장된 웹데브 서버 모델에 대하여 기술한다. 끝으로 4장에서 결론과 향후 연구 방향에 대하여 기술한다.

2. 관련 연구

본 장에서는 웹데브 프로토콜에 대한 소개와 웹데브의 주요 기능인 속성 관리에 의해 관리되는 속성에 대하여 자세히 기술한다. 그리고 본 연구에서 제안하는 웹데브 사용자 정의 속성의 실험을 위한 관련 연구로 웹 기반의 협업 지원 시스템인 iPlace 시스템에 대해 소개한다. 구현 결과에 대한 실험으로 iPlace 시스템의 공개작업장을 웹데브 서버로 적용하여 확장된 웹데브 서버의 기능을 검증한다.

2.1 웹데브(WebDAV)

웹데브는 인터넷을 통하여 광범위하고 다양한 콘텐츠의 비동기적인 협업 저작을 지원하기 위한 프로토콜이다. 웹데브는 HTTP/1.1 프로토콜의 확장을 통하여 사용자들에게 원거리 서버들의 파일들을 수정하고 관리할 수 있도록 한다. 웹

데브 표준 명세는 W3C의 IETF 웹데브 작업 그룹에 의해서 1999년 2월에 발표되었다[5]. 웹데브의 주요 기능으로는 잠금 관리(Lock Management), 속성 관리(Property Management), 이름 공간 관리(Namespace management) 등이 있으며 이들에 대한 개략적인 설명은 다음과 같다.

(1) 잠금 관리

웹데브에서는 한 문서에 대하여 한명 이상의 작업자가 동시에 작업할 수 없게 하는 기능을 제공한다. 이것은 한 저작자가 자신의 변경 내용을 서버에 반영한 것을 다른 저작자가 자신의 수정 내용에 반영하지 않고 저장시킴으로 인하여 변경 내용을 잊어버리는 갱신 분실 문제를 방지한다. 웹데브는 잠금 소유자만이 잠금된 자원에 대한 갱신을 허용하는 배타적 쓰기 잠금(exclusive write lock)과 자원에 대한 협업 작업자 그룹에게 공동 작업을 허용하는 공유된 쓰기 잠금(shared write lock)을 제공한다. 웹데브 잠금은 TCP 연결과 독립적으로 존재하기 때문에 잠금 토큰을 가지고 네트워크로부터 연결이 끊어진 상태로 작업을 하고난 후, 다시 연결하여 수정내용을 갱신할 수 있다.

(2) 속성 관리

웹데브 속성 관리는 웹 자원에 관한 저자, 마지막 수정 일자 등과 같은 정보에 대한 생성, 삭제 그리고 검색을 할 수 있도록 한다. 웹데브 속성은 이름/값의 쌍으로 이루어져 있다. 속성 이름은 URL이며, 속성 값은 잘 정의된 XML(eXtensible Markup Language) 문서이기 때문에 광범위한 XML 형태의 데이터 저장이 가능하다. 웹데브 속성은 라이브 속성과 데드 속성으로 나뉘어 진다. 본 논문에서는 이러한 속성을 시스템 정의 속성과 사용자 정의 속성으로 표현하고 있다. 웹데브 속성에 대한 자세한 설명은 2.2장에서 기술하도록 한다.

(3) 이름 공간 관리

이름 공간 관리는 서버의 이름 공간 내에 있는 웹 문서를 복사하고 이동할 수 있는 능력과 자원을 위한 저장 공간인 컬렉션의 콘텐츠를 생성하고 내용을 보여준다. HTTP는 개별 자원에 대한 직접 접근만을 처리하지만 웹데브는 데이터를 보다 효율적으로 구성하는 수단을 제공한다. 웹데브를 통한 자원 관리에는 컬렉션 자체와 컬렉션 내에 있는 자원이나 파일에 대한 생성, 이동, 복사 그리고 삭제 기능을 포함한다.

웹데브 프로토콜의 핵심 기능을 강화하기 위하여, 여러 가지 연관되는 프로토콜들이 정의되고 있다. DeltaV는 웹데브에 대형의 소프트웨어 프로젝트 또는 문서들의 컬렉션에 대한 원격 협업을 지원할 수 있는 버전과 환경 관리 형태를 추가하고 있다[14]. DASL(DAV searching and locating)은 속성 값에 대한 질의를 지원하여 광범위한 웹데브 메타데

이터를 검색할 수 있다[15]. 접근 제어 프로토콜(Access Control Protocol)은 특정 연산처리를 수행할 수 있는 지를 제어하기 위한 자원에 대한 접근 제어 리스트를 설정하도록 허가하고 있다[16]. 이러한 프로토콜은 최근에 승인되기도 하였고 아직 개발단계에 있는 것도 있다.

2.2 웹데브 속성(WebDAV Property)

웹데브 프로토콜은 모든 웹 자원과 관련하여 기본적으로 몇 개의 속성을 가질 수 있다. 웹데브 속성은 이름/값 쌍으로 구성되어 있다. 이름은 이름 공간 이름(XML 네임스페이스 이름과 동일한 URL or URI)과 속성 이름으로 구성되고, 값은 잘 정형화된 XML 형태이다. 예를 들어, 웹데브 서버에 저장된 문서의 제목이 “WebDAV Property”일 경우 다음의 트리플(triple)로 표현할 수 있다.

```
( http://www.webdav.or.kr/props/, title, WebDAV Property )
```

이것을 XML로 표현한 것은 다음과 같다.

```
<ns:title xmlns:ns="http://www.webdav.or.kr/props/">WebDAV Property</ns:title>
```

웹데브 속성들은 시스템 정의 속성과 사용자 정의 속성으로 구분된다. 웹데브 명세에서는 이러한 속성을 *라이브*(live) 속성과 *데드*(dead) 속성으로 구분하여 표현하고 있다. 사용자 정의 속성은 문법, 의미, 그리고 일관성이 클라이언트에 의해서 유지되어 지는 것이다. 그리고 서버는 데이터에 대한 처리를 거의 수행하지 않는다. 이러한 속성들은 클라이언트 응용 프로그램에 의해서 설정되고 갱신되어진다. 예를 들어 협업 지원 시스템의 공동 작업장의 경우, 공동 작업 공간에 등록된 모든 문서의 설명이나 주석 등이 사용자 정의 속성이 된다. 이러한 속성들이 해당 응용 시스템에서 특정 형식을 가지고 있다면 클라이언트 응용 프로그램에서는 지정된 형식을 일관되게 보장해주도록 해야 한다.

반대로, 시스템 정의 속성은 서버가 속성의 값을 제공하는 것이다. 한 예로서 웹데브의 *getcontentlength* 속성의 값은 자원의 길이를 계산한 것이다. 시스템 정의 속성에 대하여 웹데브 서버는 속성의 값을 설정하고 검색하는 것과 관련한 모든 처리를 수행하는 반면 사용자 정의 속성에 대해서는 XML 형식을 따르는지를 검사하는 것 이외의 어떠한 처리도 하지 않는다. 시스템 정의 속성들의 중요한 역할은 프로토콜의 연산처리를 위해 필요한 정보를 포함하고 있다. 예를 들어, 자원의 잠금 정보와 같은 것은 웹데브 서버마다 지원 여부가 다를 수 있다. 따라서 웹데브 클라이언트는 항상 주어진 자원에 현재 적용되어 있고 지원되는 잠금 종류가 무엇인지를 시스템 정의 속성으로 서버에서 제공되는

정보를 확인한다.

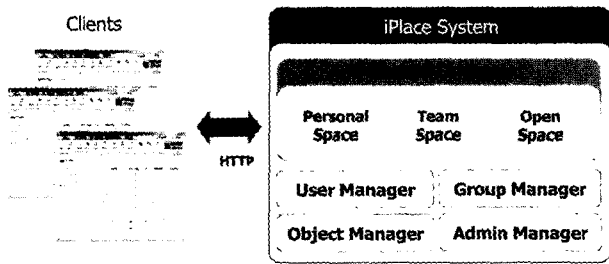
웹데브는 속성에 대한 처리를 위하여 *PROPFIND*와 *PROPPATCH*라는 두 개의 메서드를 제공한다. *PROPFIND* 메서드는 자원에 정의된 속성들을 검색하기 위하여 3가지 조건, 즉 모든 속성 조회, 특정 이름을 가진 속성 조회, 속성 이름 조회를 지원한다. 모든 속성 조회는 서버에 설정된 완전한 속성에 대한 정보를 모르는 탐색기 클라이언트에게 유용하다. 특정 이름을 가진 속성 조회는 특정 사용자 정의 속성을 사용하는 응용 프로그램에 유용하다. 끝으로 속성 이름 조회는 자원에 어떤 이름의 속성이 정의되어져 있는 지를 검색하려고 할 때 사용한다. *PROPPATCH* 메서드는 자원에 속성들을 설정하고 제거하기 위한 요청을 처리한다. 하나의 *PROPPATCH*와 관련하여 제출되어지는 요청의 전체 집합은 트랜잭션처럼 순서적으로 수행되어진다. 만약 하나의 설정이나 제거가 실패한다면 연산의 모든 영향 받은 속성은 *PROPPATCH*의 시작시점의 상태로 되돌려진다. *PROPFIND*와 *PROPPATCH* 모두 하나의 자원('depth 0'), 컬렉션과 그 바로 밑의 자식('depth 1'), 컬렉션의 자식 전체 계층('depth infinity')에 영향을 줄 수 있다[17].

2.3 iPlace 협업시스템

iPlace 시스템은 웹 기반 협업 지원 시스템으로 본 연구에서 정의한 웹데브 사용자 정의 속성의 환경 설정을 지원하는 웹데브 프로토타입 서버의 적용 실험을 위한 응용 시스템이다. 웹데브의 컬렉션 구조와 사용자 정의 속성을 이용하여 iPlace의 주요 기능인 공개작업장 공간의 구성과 적용을 실험한다.

iPlace 시스템은 웹상의 가상공간을 통하여 그룹 구성원들 간의 효과적이고 체계적인 공동작업을 지원하는 시스템이다. iPlace 시스템은 자바 컴포넌트 아키텍처인 EJB (Enterprise JavaBean) 기술[18]을 기반으로 설계되어 시스템의 확장성을 지원하고 있으며, 다중 사용자 환경에 적합한 트랜잭션 처리를 통하여 안정적인 서비스를 제공한다. 또한, 순수 자바 기술과 무료 데이터 관리시스템인 PostgreSQL을 이용하여 개발함으로써 저렴한 비용으로 서비스를 제공할 수 있다. iPlace 시스템은 공동작업을 필요로 하는 그룹간의 공유 작업공간을 지원함으로써 여러 작업그룹 내의 구성원이 이러한 환경에서 효과적으로 서로의 정보를 교환하고 공유할 수 있도록 지원하여 준다.

iPlace 시스템의 전체 구조는 (그림 1)과 같다. iPlace 시스템의 작업장은 개인작업장, 팀작업장, 공개작업장으로 구성되어 있다. 개인작업장은 개인의 작업 공간 및 자료저장을 위한 목적으로 사용되며, 팀작업장은 부서 그룹이나 공동작업을 필요로 하는 구성원들 사이의 자료 공유를 위한 장소로 사용된다. 공개작업장은 다수의 일반 사용자를 대상으로 공개된 자료의 공유 및 교환 기능을 제공한다. 또한 공개작업장의 성격을 설정해 줌으로써 자료의 공유와 관리를 용이하게 할 수 있도록 지원하고 있다[19].



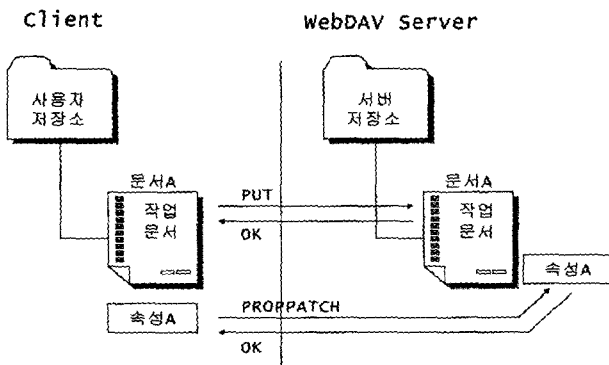
(그림 1) iPlace 시스템의 구성

3. DavSUDP(WebDAV Simple User-defined Property Definition Protocol)

본 논문에서 제시하고 있는 *DavSUDP(WebDAV Simple User-defined Property Definition Protocol)*는 웹데브 서버에 설정되는 클라이언트 측의 사용자 정의 속성을 표현하기 위한 환경 설정 속성의 정의와 웹데브 서버에 의해서 수행되어야 할 절차를 정의하고 있는 프로토콜이다.

3.1 사용자 정의 속성의 활용

사용자 정의 속성의 관리는 웹데브 서버를 사용하는 클라이언트 측에서 속성에 대한 추가, 갱신, 삭제에 대한 모든 관리 책임을 가진다. 즉, 클라이언트는 웹데브에 새로운 자원을 추가할 경우, 해당 자원별로 적용되어야 할 사용자 정의 속성을 정의하고 PROPPATCH 메시지를 이용하여 사용자 정의 속성을 설정하는 작업을 수행하여야 한다. 또한 웹데브 서버에 저장된 자원에 설정된 사용자 정의 속성에 대한 정보를 확인하기 위하여 클라이언트는 PROPFIND 메시지를 이용하여 사용자 정의 속성의 종류와 설정 상황을 확인하게 된다. 클라이언트 입장에서는 하나의 자원에 대하여 내용과 속성을 웹데브 서버에 등록하는 것은 단일 처리에 해당하지만, 실제 웹데브 서버에 등록하는 절차는 (그림 2)에서와 같이 PUT과 PROPPATCH 메시지의 2단계의 요청 수행을 통해서만 가능하다. 이러한 불합리성은 웹데브의 사용자 정의 속성을 활용하는데 있어서 장애 요인으로 작용하고 있다.

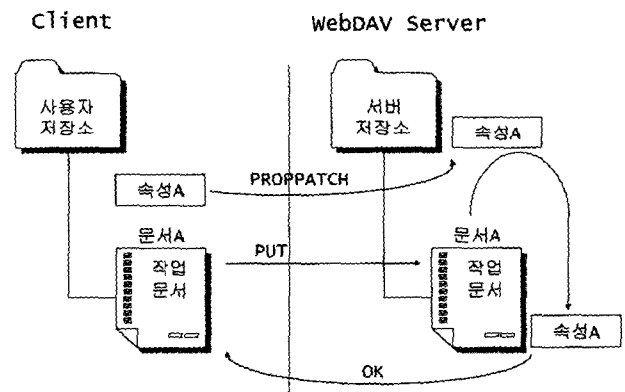


(그림 2) 문서 등록 절차

예를 들어, iPlace 시스템의 공개작업장을 웹데브 서버를 저장소로 구현할 경우, 각 공개작업장에 등록되는 문서에는 소속 작업장 이름, 작성자, 접근 권한 등의 여러 가지 속성들이 함께 설정되어야 한다. 따라서 사용자는 작업장 서버에 문서를 등록(PUT)하고 문서에 공개작업장 성격에 맞는 속성들을 설정(PROPPATCH)하여야 한다. 만약, 속성 설정에 실패한다면 문서의 등록 자체가 취소되어야 할 것이다.

3.2 사용자 정의 속성을 위한 설정 속성

본 논문에서는 웹데브 사용자 정의 속성의 정의는 클라이언트에 의해서 수행되는 반면, 자원의 등록이나 변경에 따른 속성의 설정과 변경을 웹데브 서버에 의해서 수행하도록 하기 위한 방안을 제시하고자 한다. 먼저, 일반적으로 웹데브를 활용하는 응용 시스템의 경우, 서버의 모든 자원에 대하여 기본적으로 설정되어야 하는 속성이 사전에 정의되어 있어서 클라이언트 응용 프로그램에서는 자원에 설정된 속성 값을 이용하여 자원의 소유자, 내용, 접근 권한 등을 확인할 수 있다. 따라서 (그림 3)과 같이 사용자 정의 속성의 정보를 서버의 상위 컬렉션에 정의하고 클라이언트에 의해서 자원이 등록될 때 사용자 정의 속성 또한 서버에 의해서 설정되도록 한다. 이렇게 하기 위해서는 서버에 설정할 사용자 정의 속성 정의를 위한 프로토콜이 필요하고, 웹데브 서버는 이 정의 내용을 검사하여 등록되는 자원에 맞는 사용자 정의 속성을 설정하는 기능의 추가가 필요하다.



(그림 3) 서버에 정의된 사용자 정의 속성

본 논문에서 제안하는 사용자 정의 속성 활성화 모델은 다음과 같다.

- 클라이언트는 웹데브 서버의 자원에 대하여 응용 프로그램에서 사용하고자 하는 사용자 정의 속성에 대한 환경 설정 속성을 XML 형식으로 작성하고 이 환경 설정 속성을 적용하고자 하는 웹데브 서버에 설정한다.
- 웹데브 서버는 정의된 환경 설정 속성을 검사하여 자원이 서버에 추가 및 변경되는 메소드(PUT, MKCOL, COPY, MOVE, ...)를 처리하는 과정에서 자원에 대한 사

용자 정의 속성의 추가 및 변경을 서버에서 처리하고 수행 결과를 돌려준다.

이러한 모델의 설계를 통하여, 웹데브 서버는 기존의 웹데브 명세에 정의된 표준을 준수하면서 추가된 사용자 정의 속성을 위한 환경 설정 속성의 존재 여부에 따라 추가적인 작업만을 수행하게 되고 이것은 클라이언트 측에서 처리해야 할 많은 작업을 줄일 수 있다.

3.3 DavSUDP의 설계

(그림 4)는 환경 설정 속성에 대한 XML DTD 정의 모델로서 환경 설정 이름과 설명, 컬렉션과 자원에 대하여 각각 설정되어야 할 사용자 정의 속성을 지정할 수 있는 구조로 되어 있다. DavSUDP에서 정의한 추가 요소는 웹데브 서버 측의 관점에서 보면 클라이언트에 의해서 정의된 하나의 새로운 사용자 정의 속성이다. DavSUDP의 정의 요소를 위한 유일한 네임스페이스 이름으로 "http://www.webdav.or.kr/davsudp/"를 사용하고 있다.

```

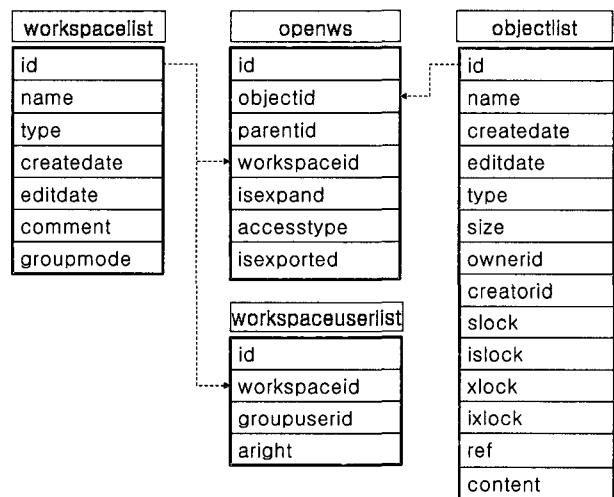
<!ELEMENT Configuration ( Name, Description, Collection?, Resource? ) >
<!ELEMENT Name ( #PCDATA ) >
<!ELEMENT Description ( #PCDATA ) >
<!ELEMENT Collection ( UserType+ ) >
<!ELEMENT Resource ( UserType ) >
<!ELEMENT UserType ( Name, Depth?, Property ) >
<!ELEMENT Depth ( Depth0 ! Depth1 ) >
<!ELEMENT Depth0 EMPTY >
<!ELEMENT Depth1 EMPTY >
<!ELEMENT Property ANY >
    
```

Configuration 요소는 웹데브 서버의 상위 컬렉션에 적용되는 환경 설정 정보를 포함하고 있는 사용자 정의 속성이다. *Name* 요소는 상위 요소의 이름을 정의하고 있고 *Description* 요소는 상위 요소의 용도와 설명을 정의하고 있다. *Collection* 요소는 MKCOL 메서드와 같이 컬렉션에 영향을 주는 메서드의 요청에 의해서 추가되거나 변경되는 컬렉션에 설정될 사용자 정의 속성을 정의하고 있다. 컬렉션은 하위 컬렉션을 생성할 수 있기 때문에 적용되어야 할 사용자 정의 속성을 세분화할 필요가 있다. 따라서 사용자 정의 속성을 *UserType* 요소로 설정하고 이러한 UserType 요소 내의 *Depth* 요소를 이용하여 하위 요소에 적용할 사용자 정의 속성을 결정한다. *Depth* 요소는 *EMPTY* 형의 *Depth0*과 *Depth1* 요소를 가진다. *Depth0* 요소가 *Depth* 요소의 하위 요소로 존재하면, UserType 요소 내의 사용자 정의 속성이 Configuration 요소가 정의된 바로 하위의 컬렉션에 적용됨을 의미한다. 반면에 *Depth1* 요소는 바로 하위의 컬렉션을 제외한 하위의 모든 컬렉션에 적용된다. *Depth* 요소의 옵션 값은 필요에 따라 좀 더 세분화하거나 조건 설정의 지정 등의 확장이 가능할 것이다. *Resource* 요소는

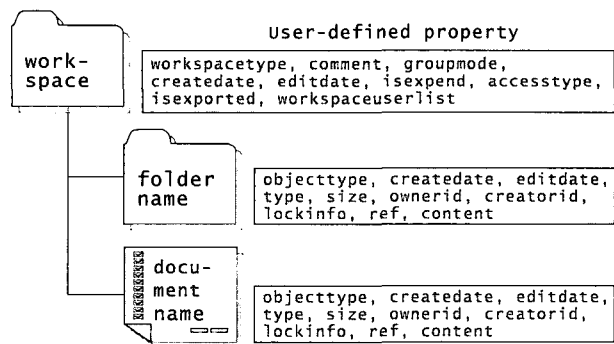
PUT 메서드에 의해서 추가되는 자원에 대하여 설정될 사용자 정의 속성을 정의한 UserType 요소를 가질 수 있다. Resource 요소 내의 UserType 요소의 경우, 모든 자원에 동일하게 설정되기 때문에 Depth 요소를 필요로 하지 않는다. Property 요소는 모든 컬렉션과 자원에 추가될 사용자 정의 속성을 정의하는 요소로 클라이언트마다 각각의 사용자 정의 속성을 정의할 수 있기 때문에 ANY 형으로 정의되어 있다.

3.4 적용 사례

DavSUDP에서 정의한 Configuration 속성의 적용 사례를 위하여 본 논문에서는 iPlace 시스템의 공개작업장 공간을 웹데브 서버의 컬렉션 구조로 표현하고자 한다. 먼저 iPlace 시스템의 공개작업장의 데이터 모델을 보면 (그림 5)와 같다.



(그림 4) iPlace 시스템의 공개작업장의 데이터 모델



(그림 5) 공개작업장의 웹데브 매핑 구조

*workspace*는 생성되어 있는 공개작업장의 목록을 가지고 있으며 *workspaceuser*에는 공개작업장을 생성한 관리자의 정보를 가지고 있다. 그리고 *objectlist*는 실제 저

장되는 자원에 대한 정보를 가지고 있으며 *openws*는 *objectlist*에 저장된 자료가 어느 공개작업장의 정보인지와 그 자료에 대한 정보를 가진다. *iPlace* 시스템의 공개작업장의 모델을 웹데브 서버의 컬렉션과 문서, 그리고 속성 정보로 매핑을 시킨다면 (그림 6)과 같이 표현할 수 있을 것이다.

```

<sudp:Configuration xmlns:sudp="http://www.webdav.or.kr/davsudp/">
  <sudp:Name>iPlace 공개작업장</sudp:Name>
  <sudp:Description>정보를 공유하고 교환할 수 있는 작업장</sudp:Description>
  <sudp:Collection>
    <sudp:UserType>
      <sudp:Name>Workspace</sudp:Name>
      <sudp:Depth><sudp:Depth0/></sudp:Depth>
      <sudp:Property>
        <wms:workspace xmlns:wms="http://iplace.ulsan.ac.kr/workspace/">
          <wms:workspace/><wms:comment/><wms:groupmode/>
          <wms:createdate/><wms:editdate/><wms:isexpend/>
          <wms:accesstype/><wms:isexported/><wms:workspaceuserlist/>
        </wms:workspace>
      </sudp:Property>
    </sudp:UserType>
    <sudp:UserType>
      <sudp:Name>Folder</sudp:Name>
      <sudp:Depth><sudp:Depth1/></sudp:Depth>
      <sudp:Property>
        <fns:folder xmlns:fns="http://iplace.ulsan.ac.kr/folder/">
          <fns:objecttype/><fns:createdate/><fns:editdate/><fns:type/>
          <fns:size/><fns:ownerid/><fns:creatorid/><fns:lockinfo/>
          <fns:ref/><fns:content/>
        </fns:folder>
      </sudp:Property>
    </sudp:UserType>
  </sudp:Collection>
  <sudp:Resource>
    <sudp:UserType>
      <sudp:Name>Document</sudp:Name>
      <sudp:Property>
        <docns:document xmlns:docns="http://iplace.ulsan.ac.kr/document/">
          <docns:objecttype/><docns:createdate/><docns:editdate/>
          <docns:type/><docns:size/><docns:ownerid/><docns:creatorid/>
          <docns:lockinfo/><docns:ref/><docns:content/>
        </docns:document>
      </sudp:Property>
    </sudp:UserType>
  </sudp:Resource>
</sudp:Configuration>

```

(그림 6) 공개작업장을 위한 Configuration 속성

공개작업장의 웹데브 매핑을 통하여 새로운 공개작업장이 생성될 경우, 최상위의 컬렉션의 생성과 함께 사용자 정의 속성을 설정하여야 하고 그 하위의 폴더나 파일의 경우는 추가되는 폴더나 파일에 따른 사용자 정의 속성의 설정이 필요할 것이다. 이와 같이 정의된 사용자 정의 속성을 본 논문에서 정의한 *DavSUDP* 프로토콜의 *Configuration* 속성으로 정의한다면 (그림 7)과 같이 설정할 수 있다.

(그림 7)의 문서 내용을 살펴보면, 먼저 *Name*과 *Description* 요소를 통하여 *Configuration* 속성에서 정의하고 있는 사용자 정의 속성의 용도를 기술한다. 공개작업장과 공개작업장 내에 생성되는 작업 폴더는 (그림 6)과 같이 웹데브의 컬렉션으로 설정하였다. 그리고 공개작업장을 위한

데이터 모델의 데이터 필드는 사용자 정의 속성으로 설정하였다.

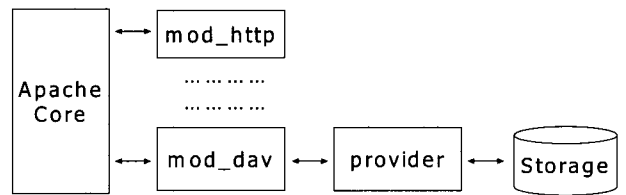
컬렉션에 설정될 사용자 정의 속성은 *Collection* 요소 내의 *UserType* 요소의 하위 요소로서 공개작업장에 정의될 속성과 폴더에 정의될 속성을 XML 구조로 구분하여 표현하고 있다. 새로운 공개작업장이 생성될 때 적용될 사용자 정의 속성은 *Depth* 요소의 하위 요소 값으로 *Depth0* 요소를 가지는 *UserType* 요소 내의 *Property* 요소이다. 또한 공개작업장 내에 생성되는 작업 폴더 컬렉션은 *Depth* 요소가 *Depth1* 요소를 가진다. 웹데브 서버는 새로운 컬렉션이 생성될 때 상위의 *Configuration* 요소를 분석하여 적절한 *Property* 요소 내의 속성 정보를 컬렉션에 설정한다.

공개작업장 내에 등록되는 문서는 (그림 6)과 같이 웹데브의 자원으로 설정하고 있다. 자원에 설정될 사용자 정의 속성은 *Resource* 요소 내의 *Property* 요소의 하위 요소로 정의하고 있다. 컬렉션과 마찬가지로 클라이언트는 자원에 등록된 속성을 읽어서 *Document*라는 이름을 가진 *UserType* 요소를 활용하게 된다. 이 *UserType* 요소에는 공개작업장에 등록되는 문서의 객체 유형, 문서 유형, 크기, 소유자 등의 사용자 정의 속성을 정의하였다.

3.5 DavSUDP 지원을 위한 웹데브 서버의 확장

기존의 RFC-2518을 지원하는 웹데브 서버는 시스템 정의 속성에 대한 관리 책임을 가지고 사용자 정의 속성은 필요로 하는 클라이언트에 의해서 *PROPFIND*와 *PROPPATCH* 메소드를 이용하여 웹데브 서버의 자원에 설정된 속성 정보를 처리하도록 되어있다.

본 논문에서 제안하는 *DavSUDP* 프로토콜은 웹데브 서버에서의 사용자 정의 속성 처리를 위한 절차를 확장하는 형태로 구현할 수 있다. *DavSUDP*의 *Configuration* 속성 또한 하나의 사용자 정의 속성으로 기존의 *DavSUDP*를 지원하지 않는 웹데브 서버의 경우에는 일반적인 사용자 정의 속성으로써의 역할만을 수행하게 된다. 웹데브 서버의 확장을 통하여 기존의 웹데브 서버의 기능을 그대로 유지하면서 추가된 속성을 검사하고 확장된 기능의 구현을 통하여 *Configuration* 속성에 정의된 기능을 추가적으로 웹데브 서버에서 수행하도록 처리하면 된다.



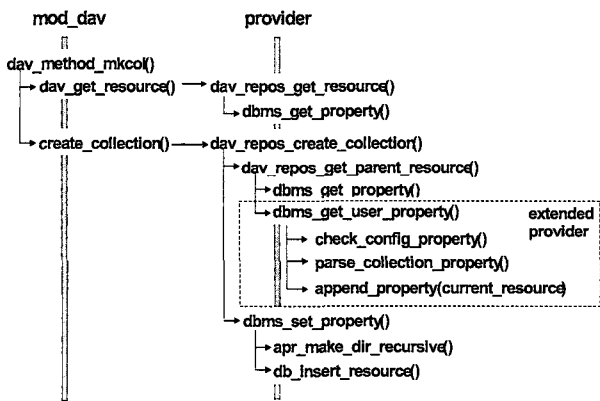
(그림 7) 아파치 mod_dav 모듈의 구조

공개 소프트웨어 기반의 웹데브를 지원하는 아파치 웹 서버를 확장하여 *DavSUDP*를 지원할 수 있도록 코드를 확장

하였다. (그림 8)과 같이 아파치 웹 서버는 클라이언트의 요청을 처리하기 위한 다양한 모듈들로 구성되어 있다.

아파치 웹 서버는 사용자가 서버의 환경 설정을 어떻게 하느냐에 따라 다양한 방법으로 작동된다. 이러한 결과로 아파치는 매우 유연하고 확장 가능한 형태를 가진다. 즉, 모듈화된 접근을 사용하여 아파치의 주 소스 코드를 변경하지 않고 서버 측의 기능을 구현하는 것이 가능하다. 아파치는 요청을 처리하기 위하여 초기화, 메서드 가져오기, 요청 읽기 등의 여러 가지 구분된 단계를 가진다[20, 21].

웹데브 프로토콜을 처리하는 모듈은 아파치 서버 내에서 (그림 8)과 같이 2개의 모듈로 나누어진다. 주 모듈인 mod_dav 모듈은 메서드 요청들의 처리를 다루며 provider 모듈은 자원의 저장소 기능을 다루기 위한 인터페이스를 처리한다[22]. 저장소를 위한 provider 모듈의 분리는 다양한 저장소 형태의 지원과 저장 방법의 다양성을 제공하고 있다. (그림 9)는 아파치 mod_dav 모듈을 DavSUDP를 지원할 수 있도록 확장한 웹데브 MKCOL 메소드의 의사 코드 (psuodo code)의 일부분이다. MKCOL 메소드는 웹데브 서버에 새로운 폴더를 생성하는 역할을 수행한다. MKCOL 메소드를 수행하면서 Configuration 속성의 설정 여부를 검사하여 사용자 정의 속성을 시스템 정의 속성 설정하는 단계 이전에 추가함으로써 그 기능을 수행하도록 구현하였다.



(그림 8) DavSUDP를 지원하는 확장된 MKCOL 메소드

4. 결 론

본 논문에서는 웹데브 기반의 고급의 응용 시스템을 개발 하는데 있어서 클라이언트 측에 의해서 관리되어야 하는 사용자 정의 속성을 효과적으로 설정하고 관리할 수 있도록 XML 형식의 Configuration 속성을 새롭게 정의하고, 이러한 Configuration 속성의 이름과 값을 웹데브 서버에 설정하고 처리 절차를 기술한 DavSUDP 프로토콜을 새롭게 제시 하였다. 웹데브 서버의 확장을 통하여 추가된 속성 정보를 이용하여 웹데브 서버에 새롭게 추가되는 자원과 컬렉션에 대하여 클라이언트에 의해서 관리되어야 할 사용자 정의 속

성들의 관리 기능을 웹데브 서버 측에서 수행하도록 개선하였다. 결과적으로 사용자 정의 속성을 활용하려는 웹데브 기반의 다양하고 고급의 응용 시스템들이 사용자 정의 속성들을 관리하기 위한 클라이언트 측의 부담을 줄일 수 있어서 새로운 응용 시스템 개발을 활성화하게 될 것이다.

적용 사례를 실험하기 위하여 웹 기반 협업 지원 시스템인 iPlace 협업시스템의 공개작업장 공간의 데이터 모델을 웹데브 서버의 구조에 매핑하였다. 공개작업장의 데이터 모델을 웹데브의 사용자 속성으로 정의하고 DavSUDP를 지원하는 확장된 웹데브 서버에 Configuration 속성으로 설정하였다. 확장된 웹데브 서버는 새로운 컬렉션과 자원의 추가 시에 각각에 설정된 사용자 정의 속성을 확인하고 적용한다. 실험 결과에서 보는 바와 같이 DavSUDP는 클라이언트의 사용자 정의 속성이 계층적으로 동일한 단순한 응용 시스템에서의 활성화에 적합하다. 응용 시스템에 따른 다양하고 계층적인 속성 구조를 지원하는 데는 한계가 있다.

향후 연구과제로는 계층적인 속성 구조의 지원과 다양한 사용자 정의 속성의 집합을 선택적으로 적용할 수 있도록 하는 일반화된 프로토콜에 대한 연구를 수행할 계획이다. 또한 DavSUDP 프로토콜에서 제한한 환경 설정 속성의 용이한 설계와 적용을 지원하는 웹데브 속성 설계 도구를 개발할 계획이다. 이러한 일반화된 프로토콜의 연구와 웹데브 속성 설계 도구의 개발은 웹데브를 이용한 다양한 응용 시스템의 개발에 많은 도움을 줄 수 있을 것이다.

참 고 문 헌

- [1] 안건태, 정명희, 이근웅, 문남두, 이명준, “iPlace: EJB 기술을 이용한 웹 기반 협업시스템”, 정보처리학회논문지 D, 제8-D권, pp.735-746, 2001.
- [2] Bentley, R., Horstmann, T., Trevor, J., “The World Wide Web as enabling technology for CSCW: The case of BSCW,” Computer Supported Cooperative Work: The Journal of Collaborative Computing, Vol.6, pp.111-134, 1997.
- [3] E. James Whitehead, Jr., Meredith Wiggins, “WEBDAV: IETF Standard for Collaborative Authoring on the Web,” IEEE Internet Computing, pp.34-40, September/October, 1998.
- [4] Jim Whitehead, Yaron Y. Goland, “WebDAV: A network protocol for remote collaborative authoring on the Web,” ECSCW’99, 1999
- [5] Y. Goland, E. Whitehead, A. Faizi, S. Carter, D. Jensen, “HTTP Extensions for Distributed Authoring -WEBDAV,” RFC 2518, Standards Track, February, 1999.
- [6] Microsoft Corporation, “Internet Information Services 5.0 Technical Overview: White Paper,” Microsoft Corporation, October, 2001.

- [7] Greg Stein, "mod_dav: a DAV module for Apache," http://www.webdav.org/mod_dav/, November, 2001.
- [8] Sung Kim, Kai Pan, Elias Sinderson, "mod_dav_dbms: A Database Backed DASL Module for Apache," Department of Computer Science, University of California at Santa Cruz, March, 2002.
- [9] Software AG, "Tamino WebDAV Server White Paper," Software AG, November 2001.
- [10] Microsoft Corporation, "Microsoft Office Server Extensions: White Paper," Microsoft Corporation, March, 1999.
- [11] South River Technologies, "WebDrive," <http://www.webdrive.com/products/webdrive/>
- [12] Xythos Software Inc., "XYTHOS WFC DATA SHEET," Xythos Software Inc., 2002.
- [13] 정혜영, 송동호, 윤병훈, 안건태, 이명준, "닷넷 기반의 WebDAV Client의 개발", 한국정보과학회, 가을학술발표논문집(III), 제30권 2호, pp.43-45, 2003.
- [14] C. Kaler, J. Amsden, G. Celmm, B. Cragen, D. Durand, B. Sergeant, E. Whitehead, "Versioning extensions to WebDAV," IETF Internet Draft, January, 1999.
- [15] Reddy, S., Lowry, D., Reddy, S., Henderson, R., Davis, J. and A. Babich, "DAV Searching & Locating," ID draft-dasl-protocol-00, July, 1999.
- [16] G. Clemm, J. Reschke, E. Sedlar, J. Whitehead, "WebDAV Access Control Protocol," RFC 3744, May, 2004.
- [17] E. James Whitehead Jr. and Yaron Y. Goland, "The WebDAV property Design," Software:Practice and Experience, Vol.34, issue 2, pp.135-161, February, 2004.
- [18] Tyler Jewell, "EJB 2.0 specification release review," OnJava.com, May, 2001.
- [19] 박희중, 김진홍, 신원준, 박양수, 이명준, "iPlace 시스템에서의 공개 작업공간 지원", 한국정보과학회, 봄학술발표논문집(A), 제31권 1호, pp.478-480, 2004.
- [21] Lincoln Stein, Doug MacEachern, "Writing Apache Modules with Perl and C," O'Reilly, 1999.
- [22] 정혜영, 안건태, 박희중, 문납두, 이명준, "Apache와 PostgreSQL을 이용한 WebDAV 서버의 설계", 한국정보과학회, 봄학술발표논문집(A), 제31권 1호, pp.397-399, 2004.



정혜영

e-mail : hyjung@lgcns.com
 1994년 울산대학교 전자계산학과(공학사)
 1998년 울산대학교 대학원 전자계학학과(공학석사)
 2005년 울산대학교 대학원 컴퓨터정보통신공학부(공학박사)

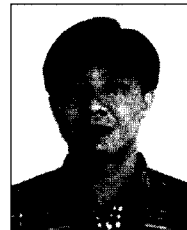
1993년~1995년 (주)현대중공업 조선사업부 근무
 2003년~2004년 울산대학교 컴퓨터정보통신공학부 객원교수
 1997년~현재 (주)LG CNS S/W아키텍처센터 근무
 관심분야 : WebDAV, 협업지원시스템, 분산 컴퓨팅 등



안건태

e-mail : java2u@ulsan.ac.kr
 1999년 울산대학교 전자계산학과(공학사)
 2001년 울산대학교 대학원 컴퓨터정보통신공학부(공학석사)
 2005년 울산대학교 대학원 컴퓨터정보통신공학부(공학박사)

관심분야 : 협업지원시스템, 생물정보학, 웹 프로그래밍, 프로그래밍 언어 등



유양우

e-mail : soft@mail.uc.ac.kr
 1995년 경일대학교 전자계산학과(공학사)
 1997년 울산대학교 대학원 전자계산학과(공학석사)
 2005년 울산대학교 대학원 컴퓨터정보통신공학부(공학박사)

2000년~현재 울산과학기술대학교 컴퓨터정보학부 전임강사
 관심분야 : 분산객체 시스템, 이동에이전트 시스템, 웹기반 정보시스템 등



박양수

e-mail : yspk56@ulsan.ac.kr
 1978년 울산대학교 전자계산학과(학사)
 1981년 서울대학교 계산통계학과(석사)
 1986년~현재 서울대학교 계산통계학과(박사과정)
 1980년~현재 울산대학교 컴퓨터정보통신공학부(부교수)

관심분야 : 분산처리, 컴퓨터알고리즘 등.



이명준

e-mail : mjlee@ulsan.ac.kr
 1980년 서울대학교 수학과(학사)
 1982년 한국과학기술원 전산학과(석사)
 1991년 한국과학기술원 전산학과(박사)
 1982년~현재 울산대학교 컴퓨터정보통신공학부(교수)

1993년~1994년 미국 버지니아대학 교환교수
 관심분야 : 웹기반 정보시스템, 프로그래밍언어, 분산 프로그래밍, 생물정보학