A WebDAV-based Collaborative System Supporting Individualized Services

Hong-Chang Lee *, Myung-Joon Lee **

Abstract

As internet users swiftly increased in number, both collaborations and communications on the Internet gradually increased. Accordingly, the importance of individualized service has been raised as user-centered service has been activated. As of now, there are many systems supporting collaboration based on the Web. Unfortunately, since the existing Web-based collaborative systems do not support individualized service, they do not provide effective collaboration for various kinds of users.

In this paper, we describe the development of a collaboration portlet for the CoSlide collaborative system. The collaboration portlet provides various methods which control the resources of each workspace on the CoSlide server. As a component of a portal, the developed portlet provides effective collaborative environment satisfying various user requirements. Users can collaborate through the individualized interfaces which are composed by portlets on a portal for their own working environment.

* 제1저자 : 이홍창  교신저자 : 이명준
* 울산대학교 컴퓨터정보통신공학부 박사과정  ** 울산대학교 컴퓨터정보통신공학부 교수
* 본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT 연구센터 육성지원사업의 연구결과로 수행되었습니다. (IITA-2008-(C1090-0801-0039))
I. 서론

인터넷을 통하여 다양한 활동을 수행하는 사용자들이 급속히 증가함에 따라 여러 그룹에 소속된 사용자들이 간의 인터넷을 통한 협업과 정보 교환이 점점 늘어나고 있다. 또한, 이와 더불어 최근 사용자 중심의 서비스가 활성화됨에 따라 개인화된 인터넷 서비스의 중요성이 대두되고 있지만 기존의 협업시스템은 사용자들 간의 사용자 중심의 서비스를 제공하지 않아서 효과적인 협업 환경을 제공하기 어렵다.

포탈(Portal)(1,2)은 여러 시스템의 정보를 포털(Portal)(3) 단위로 정의하여 웹 기반의 통합 서비스를 제공하는 게이트웨이이다. 포털은 다양한 포털들을 통하여 사용자별로 인터페이스와 어플리케이션을 지원함으로써 개인화된 서비스를 제공하며, 그에 따라 접속 그 사용 기반을 넓히고 있다. 포털들은 포털을 구성하는 웹포털을 개별적으로 정보를 처리하고 화면을 구성하여 포탈의 다양성을 지원해주는 핵심 기술이다. 포털은 포탈을 통하여 여러 시스템의 정보를 통합하고 다양한 인터페이스를 구성하여 사용자별로 제공한다.


본 논문에서는 CoSlide 협업시스템 서버를 이용하여 다양한 사용자들의 협업 수행을 지원하기 위한 협업지원 포털의 개발에 대하여 기술한다. 협업지원 포털을 효과적으로 개발하기 위하여 MVC 모델 기반의 구조적인 웹 어플리케이션 모델링을 지원하는 Struts(12) 프레임워크를 바탕으로 포털 구조를 설계하였다. 또한, 포탈에서 서비스될 수 있도록 JSR-168 표준에 근거하여 각 모듈을 구현하였다. 협업지원 포털은 CoSlide 협업시스템 서버에서 제공하는 다양한 가상공간을 지원하며, 각 가상공간의 자원을 제어할 수 있는 여러 기능들을 제공한다. 개발된 포털들은 포털을 구성하는 웹포털을 포함하여 다양한 사용자의 요구를 충족시킬 수 있는 협업 환경을 제공한다. 포탈은 포털들을 구성된 다양한 인터페이스를 제공하며, 사용자는 자신의 작업 환경을 고려한 맞춤형 인터페이스를 통하여 효과적으로 협업을 수행할 수 있다.

본 논문의 구성은 다음과 같다. 서론에 이어 2장에서는 관련 연구에 대하여 살펴보고 3장에서는 협업지원 포털의 설계와 설계를 바탕으로 한 구현을 살펴본다. 4장에서는 개발된 포털과 CoSlide 협업시스템에 대하여 평가하고 마지막으로 5장에서 결론을 살펴보고 한다.

II. 관련 연구

2.1 포탈(Portal)

포탈은 개인이나 그룹이 여러 시스템으로 나뉘서 관리하는 정보들을 통합하여 개인이나 그룹의 활동에 필요한 모든 자원을 하나의 화면으로 제공하고 관리, 분석할 수 있는 시스템이다. 포탈은 웹 기반의 게이트웨이로서 다양한 사용자와 그룹의 요구에 상응하는 맞춤형 인터페이스를 제공하고 그에 대응하는 어플리케이션을 지원하여 개인화된 맞춤 서비스를 제공한다. 포탈은 이러한 개인화된 서비스와 사용자 중심의 인터페이스 제공 기계화하여 포털이라는 웹포털을 사용한다. 포탈은 사용자들을 고려한 다양한 서비스를 제공하는 포탈 서비스는 Sun의 Portal Server 7(13), IBM의 WebSphere(14), Oracle의 OracleAS(15) 등과 함께 오픈소스 사용자 백서인 Liferay Portal(16)이 널리 사용되고 있다.
2.1.1 Liferay Portal

Liferay은 가장 널리 활용되고 있는 오픈 소스 엔터프라이즈 포털로서 Java, J2EE, Web2.0 등의 기술을 기반으로 개발되어 사용자들을 고려한 다양한 포털 기능을 제공하고 포털 컨테이너에서 포털을 활용하여 다양한 기능을 제공하기 위한 기반을 지원한다. 또한, 주요 웹 서버와 서플릿 컨테이너와 쉽게 통합되어 동작하며 다양한 데이터베이스와 운영 체제를 지원하여 플랫폼 독립적으로 구축할 수 있다. (표 1)은 Liferay 포털의 주요 특징을 나타낸다.

표 1. Liferay 포털의 주요 특징

<table>
<thead>
<tr>
<th>Table 1. Features of the Liferay Portal</th>
</tr>
</thead>
<tbody>
<tr>
<td>- JSR-168 표준 포털형 소프트 지원</td>
</tr>
<tr>
<td>- 60가지 이상 다양한 포털 형식 제공</td>
</tr>
<tr>
<td>- 보편적 기반 도구 제공</td>
</tr>
<tr>
<td>-웹사이트 관리 시스템(CMS) 지원</td>
</tr>
<tr>
<td>- AJAX를 지원하는 사용자 인터페이스</td>
</tr>
<tr>
<td>- 약 22가지 언어 지원</td>
</tr>
<tr>
<td>- 보편적인 L/DAP 통합 지원</td>
</tr>
<tr>
<td>- M/T 레이아웃 제공</td>
</tr>
<tr>
<td>- 화면 투명(look and feel) 원활한 적용</td>
</tr>
<tr>
<td>- 실시간 포털 형식 확장</td>
</tr>
<tr>
<td>- 합성된 레이아웃アイ(Layout) 기능</td>
</tr>
</tbody>
</table>

2.2 포털

2.2.1 포털

포털은 웹 화면에서 동적 콘텐츠 구성을 지원하는 환경으로 정적인 인터넷 환경에서 개인화된 서비스를 제공할 수 있는 핵심 기술이다. 초기 웹 환경에서는 정적인 인터넷을 기반으로 개인화된 서비스를 제공하기 위한 특별한 규약이 없어서 서비스 제공자별로 콘텐츠가 복잡하게 개발되어 제공되었으나, 2003년에 Sun을 주축으로 웹 기반의 포털을 정의하고 효과적인 포털을 개발하기 위하여 JSR-168라는 Java 기반의 표준이 마련되었다.

포털은 서블릿과 같은 웹 화면을 구성하는 콘텐츠이지만 서블릿이 하나의 페이지를 정의하는 단위임에 반해, 포털은 하나의 페이지를 구성하는 여러 요소 중의 요소를 정의하는 마크업 단위이다. 서블릿으로 구성된 웹 화면은 사용자들에게 동일한 서비스를 제공하지만 포털으로 구성된 웹 화면은 사용자의 요구에 따라 다양한 포털들을 조합하여 차별화된 서비스를 제공할 수 있다. 또한, 포털은 서버나 웹 서버 같은 웹 어플리케이션에 쉽게 결합되어 사용될 수 있는 고유의 속성을 가지고 있다. 이러한 고유의 속성을 포털을 관리하는 포털 컨테이너에 의하여 관리되고 제어된다. (그림 1)은 하나의 웹 페이지를 구성하는 다양한 포털의 조합을 보여준다.[17]

그림 1. 포털 페이지를 구성하는 포털

Fig 1: Portlets on a Portal Page

2.2.2 포털 컨테이너

포털 컨테이너는 포털에 관한 클라이언트, 혹은 서버의 요청을 내는 메시지를 포뮬러와 그 포털의 포트바이어를 실행하고, 요청하는 곳에 포털 서비스를 제공한다. 포름 컨테이너는 포털을 실행할 수 있는 기본 환경을 제공하며 포털이 실행되고 종료될 때까지의 생명주기에 관여하며, 포털의 경우의 속성을 제공하거나 관리한다.

포털 컨테이너에서 포털이 실행될 때, 포털 컨테이너는 포털을 위의 콘텐츠와 정보를 전달받고 이를 요청한 곳으로 전달하는 중계 역할을 한다. 포털 서버는 포털 컨테이너로부터 전달받은 콘텐츠와 정보를 바탕으로 사용자의 요청에 부응하는 포털 페이지를 생성하여 제공한다.

그림 2. 포털 페이지 생성 과정

Fig 2: Portal Page Creation

(그림 2)는 포털이 생성되어 제공되는 과정을 보여준다.[18] 포털 컨테이너는 요청된 여러 포털들을 실행하고 그 결과를 포털 서버에게 전달한다. 포털 컨테이너로부터

![그림 3. CoSlide 협업시스템의 동작](CoSpace/CoDAView 클라우드먼트)


### III. 협업 수행을 지원하는 포털 개발

본 장에서는 효과적인 협업 환경을 제공하기 위하여 사용자 중심의 인터페이스를 지원하는 협업지원 포털의 개발에 대하여 기술한다. 협업지원 포털은 포털에 포함되어 효과적으로 협업 정보를 제공하여야 하며, WebDAV 프로토콜을 이용하여 CoSlide 협업시스템 서버에 접근하고 협업 정보를 처리하기 위한 요청을 처리해야 한다. 협업지원 포털들은 CoSlide 협업시스템 서버와 연동하여 다양한 가상공간과 이를 제공하는 기능을 제공하는 효과적으로 합동 수행을 지원한다. 이와 같은 다양한 기능을 지원하기 위한 여러 모듈을 효과적으 로 구성하기 위하여 Struts 프레임워크를 사용한다. 협업지원 포털은 Struts 프레임워크를 사용하여 프레임워크 모듈과 비즈니스 로직으로 구성되어 각 영역의 유기적인
3.1 협업지원 포털의 설계

협업지원 포털은 Struts 프레임워크의 MVC 모델링에 따라 모델, 뷰, 컨트롤러 영역을 구현한다. 모델 영역은 WebDAV API를 사용하여 협업 정보를 처리하는 비즈니스 로직을 정의하며, 뷰 영역은 Struts 레라이브러리를 이용하여 협업 정보를 표현하는 프레젠테이션 로직을 정의한다. 또한, 컨트롤러 영역은 모델과 뷰 영역의 연결과 데이터 훼름을 정의한다.

3.1.1 협업지원 포털의 모델 영역

협업지원 포털의 모델은 CoSlide 협업시스템 서버와 연관하고 협업 정보를 처리하는 비즈니스 로직이 정의되어 있는 영역으로 뷰 영역에서 전달된 요청을 처리하는 역할을 하는데 주로 서버에 접근하고 기상공간의 자원을 제어하기 위한 WebDAV 메서드를 호출한다. CoSlide 협업시스템 서버로부터 전달받은 요청 정보는 컨트롤러 영역에 정의된 훼름에 따라 새로운 뷰 영역으로 전달된다.

3.2 협업체인 포털의 구현

포털이 제공하는 사용자 중심의 인터페이스에 포함되며 개인화된 서비스를 제공하고 효과적으로 CoSlide 협업시스템과 연동하기 위하여 Struts 구조로 설계한 협업지원 포털의 각 영역별 구현이 필요하다. 그림 5은 포털에 포함되어 사용자의 요청을 처리하는 협업체인 포털의 내부 동작을 보여준다.

3.2.1 협업체인 포털의 모델 영역 구현

협업체인 포털에서 비즈니스 로직을 담당하는 모델 영역은 CoSlide 협업시스템 서버에 접근하여 협업 정보를 주고받는 서블릿 클래스로 구성된다.

(1) 서블릿 클래스의 구현

협업체인 포털의 모델 영역을 구성하는 서블릿 클래스는 뷰 영역에서 전달받은 정보를 바탕으로 CoSlide 협업시스템 서버에 접근하고 해당 정보를 요청하고 그 결과를 처리한다.
음. 블 영역으로 전달하는 역할을 한다.

1. CoSlide 협업시스템 사용자 인증

```java
// WebDAV API 포함
import edu.uiuc.cs.DAVExplorer;*

public String getServerInfom(String ...)
    {DefaultAuthHandler.setAuthorizationProvider(...);
      CookieModule.setCookiePolicyHandler(null);
      // 협업시스템 서버에 접속하는 클레스 생성
      ws = new WebDAVConnection(serverAdd, 8080);
      // 사용자 정보를 통해 인증
      ws.addBasicAuthorization(..., name, pass);
      // 접속 및 요청에 필요한 정보
      NVPair[] Headers = new NVPair(3);
      Headers[0] = new NVPair("Host", serverAdd);
      Headers[1] = new NVPair("Depth", "infinity");
      ....
    }
```

그림 6. CoSlide 협업시스템 서버에 접근
Fig 6. Connection to the CoSlide Collaborative Server

(그림 6)은 서블릿 클래스에서 WebDAV API를 사용하여 CoSlide 협업시스템 서버에 접근을 시도하는 코드를 보여준다. 이 코드는 블 영역에서 전달받은 사용자의 정보를 바탕으로 WebDAV API에서 제공하는 WebDAVConnection 클래스의 객체를 통하여 연결을 시도하고 사용자 인증을 요청하는 내용을 보여준다.

2. 접속 가능한 작업장의 정보 요청

```java
String s = ("<D:propfind xmlns:D=""DAV:"">""
    + ... ("<D:propfind>
    byte[] c = s.getBytes();
    // WebDAVConnection 클래스 객체의 메서드를 통한 서비스 요청
    HTTPResponse h =
        ws.PropFind("/slide/OpenWorkspace", c, Headers);
    // 요청 결과를 XML 형태로 전달 받음
    String result =
        getData(new ByteArrayInputStream(h.getData()));
```

그림 7. WebDAV를 이용한 협업 정보 요청
Fig 7. Request the Information using WebDAV

(그림 7)은 CoSlide 협업시스템 서버와의 연결 정보가 있는 WebDAVConnection 클래스 객체의 PropFind 메서드를 통하여 특정 작업장에 접근하여 협업 정보를 요청하는 코드를 보여준다.

3. XML 정보의 변환

```java
// 전달받은 XML 형태의 정보를 XML 파서를 통하여 변환, 변환 후 작업장 정보를 분류
NodeList nodeResponse =
   doc.getElementsByTagName("D:response");
for(int i=0;i<nodeResponse.getLength();i++) {
  Node nodeResponseChild =
      nodeResponse.item(i).getFirstChild();
  uridata = new URIData();
  // 작업 정보 중에서 작업장 목록을 추출
  for (int j=0;j<nodeResponseChild.getLength();j++) {
    String name =
        nodeResponseChild.item(j).getLocalName();
    if (name.equals("D:href")) {
      NodeList nodeRef =
          nodeResponseChild.item(j).getChildNodes();
      for (int k=0;k<nodeRef.getLength();k++) {
        // 작업장 목록에서 작업장 이름 추출
        int type = nodeRef.item(k).getNodeType();
        if (type == Node.TEXT_NODE) {
          uridata.setURL(
              nodeRef.item(k).getNodeValue());
        }
      }
    }
  }
}
```

그림 8. XML 형태의 작업 정보 변환
Fig 8. Parsing the Information in XML Format

(그림 8)은 CoSlide 협업시스템 서버로부터 받은 XML 형태의 작업 정보를 블 영역에서 처리할 수 있는 클래스 형태로 변환하는 코드를 보여준다. XML 형태의 전달받은 데이터는 트리 형태로 정보를 표현하며, 이 정보는 DOM 파서를 이용하여 변환된 후 블 영역에서 처리할 수 있는 JavaBean 형태의 클래스 객체에 저장되어 전달된다.

4. 처리된 정보를 블 영역으로 전달

```java
// 모델의 로직을 실행하는 메서드
public void processAction(...) throws Exception {
  String address = serverAddress;
  String result = "";
  if (null == address || address.equals(address)) {
    setForward(req, "portlet.ext.coportlet.error");
  } else {
    // JavaBean에 처리 결과를 저장
    result = getServerInfo(address, name, pass);
    req.setAttribute("result", result);
    setForward(req, "portlet.ext.coportlet.success");
  }
```

(그림 7)은 CoSlide 협업시스템 서버와의 연결 정보가 있는 WebDAVConnection 클래스 객체의 PropFind 메서드를 통하여 특정 작업장에 접근하여 협업 정보를 요청하는 코드를 보여준다.
(9)는 뷰 영역에서 전달받은 요청에 따라 모델 영역의 비즈니스 로직을 실행하는 processAction 메서드와 모델 영역에서 처리된 데이터를 뷰 영역으로 전달해주는 render 메서드를 보여준다. 비즈니스 로직을 처리하는 서블릿 클래스들 중에서 뷰 영역의 요청에 의해 실행되는 클래스는 PortletAction 클래스이다. PortletAction 클래스가 실행되면 processAction 메서드가 우선적으로 호출되고 전달 받은 데이터를 바탕으로 요청을 처리하게 된다. 요청된 데이터가 처리되고 나면 render 메서드를 통하여 뷰 영역으로 데이터를 전달하게 된다. render 메서드는 Struts 컨트롤러에서 정의된 경로를 참조하기 위하여 ActionMapping 클래스 객체의 findForward 메서드를 사용한다.

3.2.2 협업지원 포털리프의 뷰 영역 구현

협업지원 포털리프에서 프레젠테이션 로직을 담당하는 뷰 영역은 처리된 작업 정보를 사용자의 요구에 맞도록 화면을 표현하는 JSP 페이지로 구성된다. 뷰 영역은 작업 정보를 효과적으로 표현하기 위하여 사용자 정보를 나타내는 페이지 영역, 사용자가 접속한 서버의 접근 가능한 작업정보를 표현하는 페이지 영역, 그리고 다른 사용자들의 작업 정보를 합리적으로 나타내는 페이지 영역, 그리고 선택한 작업의 하위 리소스 정보와 해당 리소스를 제어할 수 있는 기능을 제공해주는 페이지 영역으로 구성된다.

(1) 작업장 트리 페이지 구현

작업장 트리 페이지는 포털리프의 원쪽 프레임에서 사용자가 CoSide 작업시스템 사이에 접속한 뒤 접근할 수 있는 각 작업장들의 정보를 트리 형태로 보여주는 역할을 한다. 사용자가 접근할 수 있는 작업장의 크기, 작업장과 그 내부 작업장의 리소스를 각각 트리 형태로 표현하여 원하는 작업장을 순차적으로 선택할 수 있다.

```java
while(p ST.hasMoreTokens())
    StringToken p String = new
    StringToken((String)p ST.nextToken(), "/")
    ...
    // 트리 노드에 추가될 작업장 정보 추가
    while(p ST.hasMoreTokens())
        String p temp = p String.nextToken();
        resource = resource + " + p temp;
        Enumeration p e = p h.keys();
        while(p e.hasMoreElements())
            String p temp1 = (String)p e.nextElement();
            if(resource.equals(p temp1)){
                i = ((Integer)p h.get(p temp1)).intValue();
                p tok = false;
            }
    }
    if(p tok)
    %
    // 추출한 작업장 정보를 트리 노드에 추가
    d.add(("%" = j %, "%" = i %, "%" = p temp %));
    %
    }
```

(10)는 자바스크립트로 구현된 작업장 트리 페이지에서 작업장의 이름을 바탕으로 트리 노드를 구성하는 부분을 보여준다. 모델 영역에서 문자열 형태로 전달받은 작업장 리스토리를 이름으로 분류하여 트리의 노드로 추가한다. 사용자가 작업장 트리 페이지에서 노드를 선택하면 선택된 노드의 작업장 정보를 바탕으로 작업장 페이지를 호출하게 된다.

(2) 작업장 페이지 구현

작업장 페이지는 포털리프의 오른쪽 프레임에서 작업장의 세부 정보와 하위 리소스들을 제어할 수 있는 기능을 표현하는 영역이다.

```xml
<tr>
    <logic:iterate name="RL property=\"data\" id=\"ud\">
        <tr>
            <td>
            ...
            </td>
        </tr>
    </logic:iterate>
</tr>
```

// 리스토리의 태그 값을 따른 컨텍스트/파일 정보
```xml
<logic:equal name=\"ud\" property=\"type\" value=\"F\">
    <bean:define id=\"filedown\"/>
    <a href="/\"bean:write name=\"filedown\"/>
</logic:equal>
```
그림 11. 작업장 페이지 구성
Fig 11. Composing the Workspace Page

(그림 11)은 뷰 영역의 작업장 페이지에서 Struts 테그 라이브러리와ls JavaBean 클래스를 이용하여 작업장의 정보를 표시하는 프레젠테이션 로직을 보여준다. 작업장 페이지에서는 모델 영역에서 전달받은 JavaBean 클래스를 바탕으로 작업장의 정보를 불러오고 이 정보를 바탕으로 Struts 테그 라이브러리를 이용하여 화면을 구성한다. Struts 테그 라이브러리는 복잡한 코드의 기술을 줄이고 HTML 테그와 조화롭게 사용하여 효과적으로 프레젠테이션 로직을 구현하고 비즈니스 로직과 분리하는 역할을 한다.

3.2.3 협업지원 포털의 모델 영역 구현

협업지원 포털에서 컨트롤러 영역은 Struts 구조에 따라 모델과 뷰 영역을 구성하는 파일들의 연결과 데이터의 흐름을 정의하는 역할을 한다. Struts에서 제공하는 ModuleConfig 클래스 객체는 외부에 정의된 XML 문서의 정보를 바탕으로 생성되며, 이 객체는 포털이 실행될 때 요청에 따라 정해진 흐름으로 데이터를 전달하거나 페이지를 호출하는 것을 지원한다. 이러한 구조는 새로운 파일이나 서비스를 추가할 때 외부의 XML을 통하여 새롭게 정의함으로써 모델 영역이나 뷰 영역을 구성하는 파일의 내용이나 구조를 변경시키지 않고 적용할 수 있도록 도와준다.

(1) Struts-Config.xml 파일

Struts-Config.xml 파일은 Struts 구조에 따라 모델 영역과 뷰 영역을 구성하는 파일들과 데이터의 흐름을 정의한다.

```
<forward name="portlet.ext.copportlet.success" path="portlet.ext.copportlet.success"/>

...

<action>
</action>

<action-mappings>

그림 12. 영역 간 연결과 데이터 흐름의 정의
Fig 12. Definition Links and Data Flow

(그림 12)는 모델 영역과 뷰 영역을 구성하는 파일들을 연결하고 데이터의 흐름을 정의하는 Struts-Config.xml 파일의 일부를 보여준다. 포트릿에서 새로운 페이지를 호출하거나 데이터를 전달하는 행위를 Action이라고 표현하는데 Action은 Struts-Config.xml에 정의된다. Action을 정의하기 위해서는 <action-mappings> 하위에 <action> 태그를 한다. <action> 태그는 특정 페이지나 서블릿 클래스를 가리키고 있는데 해당 페이지나 서블릿 클래스에서 액션이 발생되어 새로운 연결이나 데이터 흐름이 발생할 때 이를 정의하기 위하여 하위에 <forward> 태그를 정의해야 한다. <그림 12>에서 "/ext/copportlet/server" 로 path 속성 값을 정의하고 있는 <action> 노드는 path 속성을 바탕으로 서블릿 클래스와 JSP 페이지 등에서 호출될 수 있으며, 이벤트가 발생하면 하위의 <forward> 노드의 name 속성 값에 따라 새로운 페이지를 호출한다.

IV. 협업지원 포털의 동작 및 성능

협업지원 포털은 웹 기반의 포털에서 사용자 중심의 인터페이스를 구성하는 콜스 nowrap으로 동작한다. 사용자는 자신의 작업 환경을 고려한 맞춤형 인터페이스를 통하여 다양한 서비스와 더불어 협업지원 포털을 사용함으로써 효율적으로 CoSlide 협업시스템을 이용할 수 있다.

그림 13. 포털에서 동작하는 협업지원 포털
Fig 13. Collaboration Portlet on a Portal
<일단 13>은 협업지원 포털을 비롯한 다양한 포털을 통한 포털에서 사용자를 고려한 다양한 인터페이스를 제공하는 모습을 보여준다.

4.1 협업지원 포털의 주요 기능

효과적인 협업 수행을 지원하기 위하여 협업지원 포털에서 제공하는 기능은 다음과 같다.

표 3. 협업지원 포털에서 제공하는 다양한 기능

<table>
<thead>
<tr>
<th>Table 3. Various Functions on Collaboration Portlet</th>
</tr>
</thead>
<tbody>
<tr>
<td>사용자별로 저장되는 자동로그인 기능</td>
</tr>
<tr>
<td>접근 기능 작업장의 정보 보기</td>
</tr>
<tr>
<td>작업을 위한 리소스 접근/검색/배치 기능</td>
</tr>
<tr>
<td>지정 할력선에 파일 업로드 기능</td>
</tr>
<tr>
<td>작업장 2기 파일 이동/복사 기능</td>
</tr>
<tr>
<td>작업장 하위 에이블선 생성 기능</td>
</tr>
<tr>
<td>지정 할력선/리소스 삭제 기능</td>
</tr>
<tr>
<td>다중 할력선/리소스 선택 기능</td>
</tr>
</tbody>
</table>

4.2 협업지원 포털의 사용자 인터페이스

사용자는 협업지원 포털의 초기 화면에서 서버의 정보와 사용자의 기본 정보를 입력함으로써 CoSide 협업시스템에서 접속할 수 있다. 자동로그인 기능을 통하여 반복적인 입력 작업 없이 쉽게 접속할 수 있도록 자동 접속할 수 있다. <그림 14>는 IP 주소와 사용자의 정보를 바탕으로 CoSide 협업시스템에서 접속하는 협업지원 포털의 인터페이스를 보여준다.

4.3 협업지원 포털의 성능

4.3.1 CoDAView 클라이언트와 비교

협업지원 포털은 협업 기반의 협업을 기반으로 한 포털에서 담장함으로써 기존의 협업 클라이언트인 CoDAView와 비교하여 품질에 독립적으로 서비스될 수 있다. 플랫폼에 속성하여 저장하는 기존의 CoDAView와는 달리 됨 화면을 보는 채 브라우저를 통하여 실행되므로 플랫폼에 구속받지 않고 웹을 수행할 수 있다. 또한, 다양한 포털들과 함께 사용자의 작업 환경을 고려한 인터페이스를 구성함으로써 보다 손쉽게 CoSide 협업시스템을 이용할 수 있다. 하지만, 협업지원 포털은 웹 기반으로 구현됨에 따라 상대적으로 협업 수행 기능과 협업 정보의 표현에 있어 약간의 제약조건을 가지고 있다. <그림 16>은 CoDAView와 함께 협업지원 포털의 비교를 보여준다.
4.3.2 CoSlide 협업시스템과 타 시스템 비교

CoSlide 협업시스템은 기존에 지원하던 여러 협업 클라이언트와 함께 협업지원 포털들을 지원함으로써 사용자들의 다양한 작업 환경을 고려한 협업 환경을 구현할 수 있으며, 다양한 플랫폼에 적합한 여러 클라이언트를 제공함으로써 보다 효과적으로 협업 수행을 지원할 수 있다. (표 4)는 기존의 협업시스템과 함께 개발된 협업지원 포털들을 지원하는 CoSlide 협업시스템의 차이점을 보여준다.

<table>
<thead>
<tr>
<th>표 4. 각 협업시스템들의 특징 Table 4. Features of the Collaborative Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>BSCW</td>
</tr>
<tr>
<td>기반 프로토콜</td>
</tr>
<tr>
<td>개인작업장</td>
</tr>
<tr>
<td>그룹작업장</td>
</tr>
<tr>
<td>공유작업장</td>
</tr>
<tr>
<td>서버 구현기술</td>
</tr>
<tr>
<td>서버 플랫폼</td>
</tr>
<tr>
<td>원도우 클라이언트</td>
</tr>
<tr>
<td>리눅스 클라이언트</td>
</tr>
<tr>
<td>웹 기반 클라이언트</td>
</tr>
</tbody>
</table>

BSCW(19)는 대표적인 웹 기반의 협업시스템으로서 사용자에 대한 개인작업장과 그룹 사용자에 대한 그룹작업장을 부피적으로 지원한다. 하지만 다양한 작업 환경을 고려한 클라이언트와 기능을 지원하지 않고 복잡적인 사용자간을 지원하지 않아 효과적인 협업서비스를 제공하지 못한다. iPlace(20)는 웹 기반의 협업시스템으로서 다양한 사용자간을 지원하여 효과적인 협업 수행을 지원한다. 하지만 http 프로토콜의 제약에 따라 사용자간 관리나 서버 제어를 하기 위한 새로운 기능을 추가하기 어려워 다양한 클라이언트를 지원하지 못한다. CoSlide 협업시스템은 기존의 협업시스템에서 제공한 사용자간을 사용자와 그룹을 위하여 세분화하였으며, 각 작업장에서 협업 수행을 위한 다양한 기능들을 제공한다. 또한, 사용자

의 작업 환경을 고려한 다양한 플랫폼의 클라이언트를 지원하여 보다 효과적으로 협업을 수행할 수 있도록 도와주며 사용자 중심의 맞춤형 인터페이스를 제공하는 포털 클라이언트를 통하여 그 활용 범위를 더욱 넓히고 있다.

V. 결론


본 논문에서는 Struts 프레임워크를 기반으로 하고 WebDAV 프로토콜을 통하여 CoSlide 협업시스템과 연동하여 협업 수행 기능을 제공하는 협업지원 포털들을 구현하였다. 그러나 이러한 포털들의 개발을 체계적으로 도와주는데 별도의 개발도구가 존재하지 않아 구현에 상당한 어려움이 있었다. 이에, 본 논문에서 기술한 협업지원 포털들의 개발 경점을 바탕으로 하여 이론 포털 전용의 개발도구의 가능성을 탐색하고 이를 구현하는 연구를 수행할 예정이다.
참고문헌

[17] JSR-168, the Java Portlet Specification version 1.0, p19, Figure 4-1
[18] JSR-168, the Java Portlet Specification version 1.0, p20, Figure 4-2
저자 소개

이 흥 찬
2006년 : 울산대학교 컴퓨터정보통신
공학부 졸업(공학사)
2008년 : 울산대학교 컴퓨터정보통신
공학부 졸업(석사)
2008년 ~ 현재 : 울산대학교 컴퓨터정보
통신공학부 박사 과정
관심분야 : 웹기반 정보시스템, 합업시스템, 웹기반 분산시스템.

이 명 준
1980년 : 서울대학교 수학과 졸업(학사)
1982년 : 한국과학기술원 전산학과 졸업(석사)
1991년 : 한국과학기술원 전산학과 졸업(박사)
1982년 ~ 현재 : 울산대학교 컴퓨터정보
통신공학부 교수
1993년 ~ 1994년 : 미국 버지니아대학
교환 교수
2006년 ~ 2006년 : 미국 캘리포니아주
립대학 교환 교수
관심분야 : 웹기반 정보시스템, 프로그램
래더, 분산 프로그래밍, 생물정보학, 선형대수학, 워크 프로그래밍 환경.