

CoSlide 협업시스템을 위한 Jabber 메신저 클라이언트

이홍창⁰, 이태호*, 김성훈*, 이명준**

울산대학교 컴퓨터·정보통신공학부

⁰myhyunii, *{soulfree, heinz}@mail.ulsan.ac.kr

**mjlee@ulsan.ac.kr

A Jabber Messenger Client for the CoSlide Collaborative System.

Hong-Chang Lee⁰, Tae-Ho Lee, Seong-Hune Kim, Myungjoon Lee

School of Computer Engineering & Information Technology, University of Ulsan

요 약

*Jakarta Slide*는 아파치 프로젝트 중의 하나로 개발된 *WebDAV* 서버로서 인터넷 통하여 다양한 콘텐츠의 비동기적인 협업 활동을 지원한다. 하지만 *WebDAV* 프로토콜은 사용자와 그룹을 위한 작업공간을 명시적으로 지원하지 않기 때문에, *Jakarta Slide*를 통하여 복잡한 협업을 수행하기는 매우 어렵다. *CoSlide*는 이러한 문제점을 개선하기 위하여 *Jakarta Slide*를 확장한 협업시스템으로서 다수의 사용자와 그룹을 위한 다양한 작업공간을 제공하여 효과적인 협업 환경을 제공한다. 협업시스템을 통하여 복잡한 협업을 보다 효과적으로 수행하기 위해서는 협업 참여자들 간에 실시간적인 통신이 원활이 이루어져야 할 필요성이 있다.

본 논문은 *CoSlide* 협업시스템에서 실시간 메시지를 지원하기 위한 *CoJBother* 메신저 클라이언트의 개발에 대하여 기술한다. 개발된 *CoJBother*는 협업시스템의 사용자와 그룹을 정의하기 위하여 개발된 *Jabber* 프로토콜을 사용하며, 또한 이를 지원하기 위하여 개발된 메신저 서버와 함께 동작한다. 사용자는 *CoJBother*를 통하여 협업시스템의 작업그룹에 참여하는 사용자들을 확인하고 자동적인 그룹채팅 기능을 사용함으로써 그룹 참여자들과 협업 수행을 위한 원활한 의사소통을 할 수 있다.

1. 서 론

Jakarta Slide[1]는 아파치 웹서버에서 동작하는 모듈로서 *WebDAV* 프로토콜을 통하여 다양한 콘텐츠의 비동기적인 협업 환경을 제공한다. *Jakarta Slide*에서 지원하는 *WebDAV*[2,3,4]는 *HTTP/1.1*을 확장한 웹 통신 프로토콜로서 다양한 콘텐츠의 비동기적인 협업을 인터넷을 통하여 지원하기 위한 표준 하부구조를 제공한다. 사용자는 *Jakarta Slide* 서버를 이용하여 여러 사용자와 원격지 자원을 공유하면서 비동기적으로 협업을 수행할 수 있다.

CoSlide 협업시스템[5]은 원격지의 자원과 가상공간을 활용하여 효과적인 협업을 지원하는 시스템으로서 *Jakarta Slide*를 확장하여 구축되었다. 기존의 *Jakarta Slide*는 개별 사용자와 사용자 그룹에 대한 가상공간에 대한 체계적인 지원이 부족하여 다양한 그룹을 통하여 다수의 사용자가 협업을 수행하기 위한 효과적인 환경을 형성하는 것이 매우 어려운 작업이었다. *CoSlide*는 *Jakarta Slide*의 기능을 바탕으로 개별 사용자와 사용자 그룹에 대하여 체계적으로 가상공간을 제공함으로써 보다 효과적인 협업 환경을 제공한다. 이러한 *CoSlide*를 통하여 복잡한 협업을 보다 효과적으로 수행하기 위해서는

협업 참여자들 간에 실시간적인 통신이 원활이 이루어져야 할 필요성이 있다.

본 논문에서는 여러 사용자가 *CoSlide* 협업시스템을 이용하여 협업을 수행할 때 그룹 구성원들 간의 원활한 통신을 지원하는 *Jabber*[6] 메신저 클라이언트인 *CoJBother*의 개발에 대하여 기술한다. 개발된 *CoJBother*는 *CoSlide* 협업 시스템의 사용자와 그룹을 정의하기 위하여 개발된 *Jabber* 프로토콜[7]을 사용하며, 또한 이를 지원하기 위하여 확장, 구현된 *Jabberd* 서버[7]와 함께 동작한다. 협업 참여자는 *CoJBother*를 통하여 *CoSlide* 협업시스템의 작업그룹의 참여자들을 확인하고 자동적인 그룹채팅 기능을 사용함으로써 그룹 참여자들과 협업 수행을 위한 원활한 의사소통을 할 수 있다.

본 논문의 구성은 다음과 같다. 서론에 이어 2장에서는 관련 연구에 대하여 살펴보고 3장에서는 *CoJBother* 메신저 클라이언트의 구현에 대하여 살펴본다. 그리고 4장에서는 구현된 *CoJBother*의 사용 예를 살펴보고 5장에서는 메신저와 연동한 협업시스템에 대한 평가를 살펴본다. 마지막 6장에서는 결론 및 향후 연구 과제에 대하여 살펴본다.

* 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 육성지원사업의 연구 결과로 수행되었음.

2. 관련 연구

본 장에서는 본 논문에서 다루고 있는 확장된 Jabber 프로토콜과 Jabberd, JBother 그리고 CoSlide 협업시스템에 대하여 기술한다. 확장된 Jabber 프로토콜은 표준 프로토콜에 협업시스템의 정보를 표현할 수 있도록 확장하여 협업시스템을 위한 메시징 시스템에 사용될 수 있다. Jabberd는 표준 Jabber 프로토콜을 지원하여 메시징 서비스를 제공하는 서버 프로그램이다. JBother는 Jabber 프로토콜을 사용한 메신저 클라이언트로서 자바로 구현된 오픈소스 프로그램이다. CoSlide 협업시스템은 WebDAV 프로토콜을 사용한 Jakarta Slide를 확장한 시스템으로 개인과 그룹에 대한 가상공간을 지원하여 보다 효과적인 협업 환경을 제공할 수 있다.

2.1. Jabber 프로토콜과 확장된 Jabber 프로토콜

Jabber는 “실시간 메신저의 리눅스”라고 잘 알려진 오픈 소스 실시간 메신저 서비스이다. Jabber는 XML 프로토콜로서 인터넷 상의 두 요소들 간에 메시지나 프레젠스(presence), 또 다른 구조의 정보를 실시간으로 교환할 수 있도록 한다.

Jabber는 XML로 표현되며 간단한 구조로 정의되어 있어 응용이 용이하고 다양한 플랫폼에서 사용이 가능하다. [그림 1]은 Jabber 프로토콜에서 사용되는 실시간 메시지의 간단한 형태이다.

```
<message from='juliet@capulet.com'
          to='romeo@montague.net'>
  <body>Wherefore art thou, Romeo?</body>
</message>
```

[그림 1] Jabber 실시간 메시지의 간단한 구조

기존의 Jabber 프로토콜은 한 사용자가 한 그룹에만 소속될 수 있기 때문에 다수의 사용자가 여러 그룹에 소속되어 작업을 하게 되는 협업시스템의 정보를 표현하기는 어렵다. 이러한 문제점을 개선하기 위하여 확장된 Jabber 프로토콜은 협업시스템에 맞는 메신저 정보를 표현하기 위하여 표준 프로토콜에 새로운 XML 스키마가 확장되었다.

2.2. Jabberd 서버와 확장된 Jabberd 서버

Jabberd[8] 서버는 실시간 메시징 프로토콜인 표준 Jabber 프로토콜과 XMPP 프로토콜을 지원하는 서버이다. Jabberd는 C/C++로 구현되어있고 컴포넌트의 추가, 삭제가 쉬운 모듈 구조로 구성되어있으며 공개 소프트웨어

어로서 누구나 쉽게 사용할 수 있다. Jabberd 서버는 표준 Jabber 프로토콜을 통하여 메시징 서비스를 지원하기 때문에 본 연구에서는 협업시스템을 위하여 확장된 Jabber 프로토콜을 사용하기 위하여 Jabberd 서버를 확장, 구현하였다. 확장된 Jabberd 서버는 협업시스템의 저장소에 접근하여 사용자와 그룹의 정보를 확인하고 이 정보를 클라이언트에게 제공하여 준다.

2.3 JBother 메신저

Jabber 메신저는 Jabber 프로토콜을 이용하여 다양한 메신저의 기능을 확장하고 제공하는 메신저로서 JBother[9]는 자바언어로 구현된 대표적인 표준 Jabber 메신저이다. JBother는 윈도우 스타일 형식으로 그룹 채팅을 지원하며 다양한 리소스를 제공하고 트랜스포트 기능으로 다른 메신저와의 연동도 가능하게 한다. 그리고 여러 스킨을 제공해 각 사용자의 메신저 환경을 다양하게 변경할 수 있으며 JRE 환경이 갖춰진 플랫폼이라면 어디서든 쉽게 사용할 수 있다. 또한 다양한 플러그인을 통해서 여러 플랫폼에 맞게 최적화할 수 있다.

2.4 CoSlide 협업 시스템

CoSlide 협업시스템은 원격지의 자원과 가상공간을 활용하여 보다 효과적인 협업 환경을 제공하는 시스템으로서 Jakarta Slide를 확장하여 구현되었다. 기존의 Jakarta Slide는 협업을 하는 사용자와 그룹별 가상공간을 지원하지 않아 다수의 사용자와 그룹이 참여하는 협업 시 효과적인 협업을 제공하기 어려웠다. 이러한 문제점을 개선한 CoSlide는 협업에 참여하는 사용자와 그룹별로 가상공간을 제공하여 보다 효과적인 협업 환경을 제공한다. CoSlide 협업 시스템을 사용하는 사용자는 자신만의 가상공간을 제공받게 되며 참여하는 그룹의 가상공간에 접근하여 그룹의 구성원들과 용이하게 협업을 수행할 수 있다.

3. CoJBother 메신저 클라이언트의 구현

3.1 협업시스템과 연동되는 메시징 시스템

협업시스템을 통하여 다수의 사용자가 여러 그룹에 속하여 협업을 수행할 때는 작업의 규모나 일정에 따라 그룹 참여자들이 자주 바뀌거나 그룹 자체가 수정되는 일이 빈번하다. 이렇게 협업을 수행함에 있어 잦은 변동이 있을 시에 수시로 함께 작업하는 사람들과 의사소통을 하기 위하여 별도의 도구를 사용하는 것은 효과적인 협업 활동을 저해한다. 따라서 실시간적으로 협업시스템의

사용자와 그룹의 정보를 제공하여 별도의 도구 없이 그룹 참여자들과 의사소통을 전달할 수 있는 메신저가 있다면 보다 효과적인 협업을 수행할 수 있다.

‘Jabber 메신저’는 Jabber 프로토콜을 이용하는 메시징 시스템을 가리키는 것으로 본 연구에서는 협업시스템을 지원하도록 확장된 Jabber 프로토콜을 지원할 수 있는 Jabber 메신저를 개발한다. 이 메신저를 개발하기 위하여 표준 Jabber 프로토콜을 지원하는 JBother 메신저를 확장하도록 한다. JBother 메신저는 표준 Jabber 프로토콜을 지원할 뿐만 아니라 화려한 인터페이스와 그룹채팅 등의 다양한 기능을 지원한다.

하지만 JBother는 협업시스템의 작업그룹정보를 자동적으로 표현하지 못하기 때문에 협업시스템과 능동적으로 연동되려면 확장을 하여야 한다. 개발된 CoJBother는 확장된 Jabber 프로토콜을 지원하여 협업시스템에서 작업 중인 그룹 참여자정도를 자동적으로 표현하고 효과적인 통신을 위한 작업그룹별 그룹채팅기능을 제공한다.

CoJBother가 협업시스템을 지원하기 위하여 기존의 JBother 클라이언트에서 수정된 부분은 크게 두 가지이다. 한 부분은 확장된 Jabber 서버와의 통신을 위하여 확장된 Jabber 프로토콜을 사용할 수 있도록 Smack[10] 라이브러리를 수정하는 것이고 또 다른 한 부분은 서버로부터 받은 협업시스템의 정보를 사용자에게 맞도록 표현하는 인터페이스를 구현하는 것이다.

3.2 확장된 Jabber 프로토콜을 위한 Smack 라이브러리 수정

Smack은 인스턴트 메시지와 상태 표시(Presence)기능을 제공하는 XMPP 기반의 오픈 소스 Jabber 클라이언트 라이브러리이다. 클라이언트는 Smack 라이브러리를 사용하여 Jabber 프로토콜을 사용하고 Jabber 메신저 서버와 통신할 수 있다.

클라이언트는 Jabber 서버로 접속할 때 협업시스템 사용자로서 접속을 하기 위해 확장된 Jabber 프로토콜의 ‘jabber:iq:CoSlide_auth’ 네임스페이스를 사용하여 인증에 관련된 XML 문서를 전송하고 서버로부터 인증을 받는다. 그리고 서버로부터 받은 XML 문서에서 협업시스템의 사용자 정보를 구별해내기 위해 추가한 ‘jabber:iq:CoSlide_roster’ 네임스페이스를 사용한다. [그림 2]는 Smack 라이브러리에 협업시스템 사용자로 접근할 수 있도록 코드를 수정한 부분과 서버로부터 받은 XML 문서로부터 협업시스템 사용자 정보를 분류하도록 코드를 수정한 부분이다.

```
// 추가된 네임스페이스를 이용하여 쿼리문을 전송
public String getChildElementXML(){
    StringBuffer buf = new StringBuffer();
    buf.append("<query xmlns=\"jabber:iq:coslide_auth\">");
    ....
}

// 서버로부터 받은 쿼리문에서 협업시스템 관련 정보를 분류
private IQ parseIQ(XmlPullParser parser) throws Exception{
    ...
    String name = parser.getName();
    String namespace = parser.getNamespace();
    ...
    else if(name.equals("query") &&
namespace.equals("jabber:iq:coslide_roster"))
        iqPacket = parseCoSlideRoster(parser);
    ...
}

// Jabber 프로토콜 패킷으로부터 사용자의 정보를 추출
private RosterPacket parserCoSlideRoster(XmlPullParser
parser) thows Exception{
    ...
    String jid = parser.getAttributeValue("", "jid");
    ...
}
```

[그림 2] Smack 라이브러리의 수정

3.3 확장된 Jabber 프로토콜을 위한 클라이언트 인터페이스 수정

수정된 Smack 라이브러리를 통해서 협업시스템의 정보를 가져오면 그 정보를 표현하기 위해 기존의 JBother의 소스코드를 수정할 필요가 있다. CoJBother는 JBother의 인터페이스 부분을 수정해서 협업시스템 정보를 나타낸다. 또한 편리한 그룹 채팅을 지원하기 위하여 새로운 팝업 메뉴를 추가하고 사용자가 그룹 채팅을 요청하면 그룹 정보와 사용자 정보를 이용하여 자동으로 그룹 채팅방을 생성한다. [그림 3]은 협업시스템의 정보를 나타내는 인터페이스 코드 부분과 그룹 채팅을 지원하도록 수정한 코드 부분이다.

```
// 협업시스템 사용자를 메신저 이웃 목록에 보여주는 코드
// Smack 라이브러리를 통하여 협업정보를 처리
Set set = CoSlideGroup.getObject().getData();
```

```

Iterator i = set.iterator();
if(group.equals("CoSlide") || group.equals("Contacts")){
    while(i.hasNext()){
        Map.Entry e = (Map.Entry)(i.next());
        String str = (String)e.getKey();
        StringTokenizer st = new
            StringTokenizer((String)connection.getUser(),"/");
        String name = st.nextToken();
        Vector v = (Vector)e.getValue();
        String t = (String)buddy.getUser().trim();
        for(int j = 0;j<v.size();j++){
            String s = (String)v.elementAt(j);
            if(s.equals(t))
                CoSlideAddGroup(str,buddy);
        }
    }
}

// 자동적으로 그룹 채팅을 지원하기 위한 기능의 소스 코드
// 그룹 채팅 기능을 위한 메뉴를 추가하고 이벤트 등록
private JMenuItem groupchat =
    new JMenuItem("GroupChat");
add(groupchat);
groupchat.addActionListener(listener);
...
// 기능을 활성화 하면 자동으로 그룹 채팅을 하기 위한 정보
// 를 담은 북마크 실행
GroupChatBookmarks gc = new GroupChatBookmarks
    (BuddyList.getInstance().getTabFrame());
gc.setCoSlideGroupChat(groupname,
    "conference.203.250.77.124");
    
```

[그림 3] 협업시스템 사용자 목록과 그룹 채팅 기능을 구현하는 소스 코드

4. CoJBother 메신저 클라이언트의 사용 예

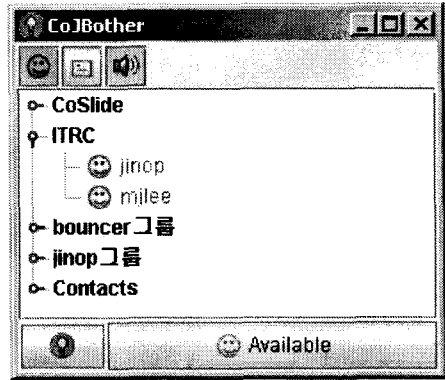
CoSlide 협업시스템은 XML 파일로 정의된 문서를 통하여 시스템 정보를 표현한다. 다음 [그림 4]는 CoSlide 협업시스템의 그룹 정보를 정의하는 XML 문서의 일부이다. 이 문서는 ITRC이라는 그룹의 속성을 정의하고 참여하는 사용자들의 정보를 표현한다.

```

// ITRC 그룹에 참여하고 있는 사용자들의 정의
<property
name="Groupmemberlist" namespace="DAV:"
value="jinop,mathpf,mjlee" type="" protected="false">
    
```

[그림 4] CoSlide 협업시스템의 그룹을 정의하는 XML 문서의 일부

다음 [그림 5]는 'mathpf' 사용자가 CoJBother를 통하여 CoSlide 협업시스템에 접근하였을 때 자신이 소속된 그룹과 그룹원의 정보를 보여주는 기본 인터페이스를 보여준다.



[그림 5] 협업시스템의 그룹 정보를 표시하는 CoJBother의 기본 인터페이스

CoJBother는 협업시스템을 위한 편리한 그룹 채팅 기능을 제공한다. 사용자가 팝업 메뉴를 통한 그룹 채팅방 생성을 요청하면 CoJBother는 그룹의 정보와 사용자의 정보를 사용하여 서버로 그룹 채팅방 개설을 요청한다. 그에 따라 Jabberd 서버는 conference 콤포넌트를 사용하여 그룹 채팅방을 개설하게 되어 사용자는 자동적으로 그룹 채팅방에 참여하게 된다.

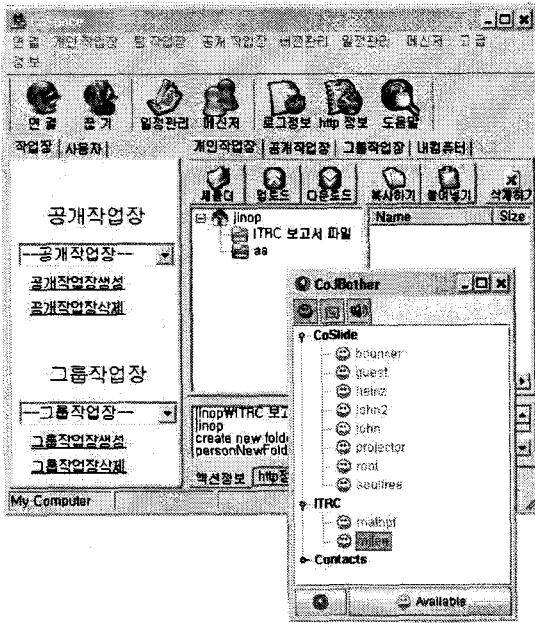
4.1 CoJBother 메신저 클라이언트의 활용

다음은 CoJBother를 활용하여 협업을 수행하는 과정을 보여준다. 사용자는 원격지에 떨어진 상태로 CoSlide 협업시스템을 통하여 협업에 참여할 수 있고 다른 참여자와 정보를 주고받기 위하여 CoJBother를 사용할 수 있다. 사용자는 CoJBother의 그룹채팅기능을 이용하여 손쉽게 협업 참여자들과 정보를 교환할 수 있다.

- 1) 연구소 프로젝트의 참여자인 'jinop'와 'mathpf'는 프로젝트를 수행하기 위하여 각각 미국과 프랑스로 출장을 가게 되었음.
- 2) 원격지에 떨어진 상태에서도 CoSlide 협업시스템의 클라이언트를 사용하여 출장 중 연구보고서 작성을 위한 자료를 그룹작업장에 올리는 등의 협업을 수행함.
- 3) 원격지에 떨어져있는 협업그룹 참여자들 간에 연구보고서 작성 문제로 긴급한 의사소통이 필요하게 되었

음.

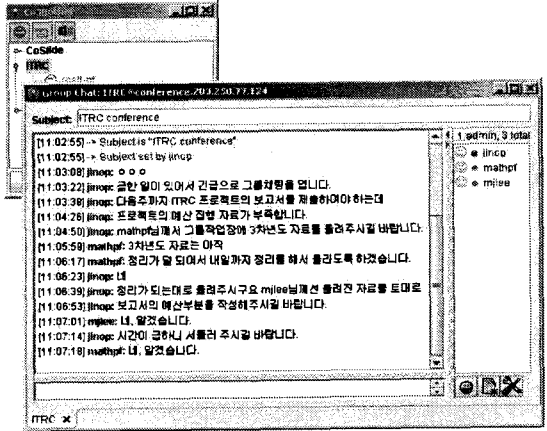
- 4) E메일이나 전화와 같은 별도의 도구를 사용하지 않고 협업시스템 클라이언트에 연동된 CoJBother를 사용하여 협업그룹의 참여자들을 확인하고 원클릭으로 그룹채팅을 시작함.



[그림 6] 협업시스템 클라이언트와 연동된 CoJBother를 사용하는 모습



[그림 7] CoJBother를 이용한 손쉬운 그룹채팅 기능



[그림 8] 그룹 구성원들이 CoJBother를 통하여 통신하는 예

5. 평가

본 연구에서 개발된 CoJBother 메신저의 성능을 평가하기 위하여 연동된 협업시스템과 다른 협업시스템을 비교하고 메신저의 연동에 따른 작업과정의 차이를 비교하였다.

5.1 협업시스템 비교

[표 1]은 자바 기술 기반으로 다양한 작업공간을 지원하는 iplace[11], 대표적인 웹기반 협업시스템인 BSCW[12]와 함께 CoSlide 협업시스템의 차이점을 보여준다.

[표 1] 협업시스템 비교

	BSCW	iPlace	CoSlide
기반 프로토콜	http	http	webdav
개인작업장	○	○	○
그룹작업장	△	○	○
공개작업장	×	×	○
구현기술	Python	EJB+JSP	JAVA
사용플랫폼	Unix, Windows	Unix, Linux, Windows	Unix, Linux, Windows
그룹참여	수동적	능동적/ 수동적	능동적/ 수동적
자원 처리프로그램	×	×	○
자동 구동	×	×	○
실시간 메시징	×	×	○

○:지원 △:부분지원 ×:미지원

- 5) 프로젝트를 진행하기 위한 회의를 시작하고 성공적으로 의사를 교환함.

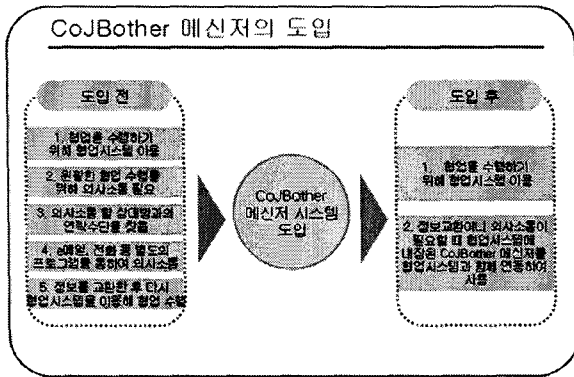
CoSlide 협업시스템은 기존의 협업시스템과는 달리 그룹작업장과 공개작업장을 완벽히 지원하고 CoJBother와 연동되어 협업을 수행하는 동시에 실시간 메시징이 가능

하다.

룹 간의 통신을 또한 지원하도록 구현되었다.

4.4.2 메신저의 도입에 따른 작업 과정

다음 [그림 9]은 메신저가 없는 협업시스템인 iplace와 CoJBother 메신저를 도입한 CoSlide 협업시스템에서 협업 참여자들이 의사소통을 할 때 각각의 작업과정을 보여준다.



[그림 9] 메신저의 도입 전후의 협업시스템 작업 과정

협업시스템을 이용해 협업을 수행할 때 보다 효과적인 협업을 위하여 참여자들 간에 의사소통이 필요하다. 이러한 의사소통의 도구로서 E-메일이나 전화가 오랫동안 사용되어 왔다. 하지만 이러한 방법은 불필요한 작업을 수반하여 효율성을 떨어뜨린다. 본 연구에서 개발된 CoJBother 메신저를 도입하면 참여자들 간에 의사소통을 위해 별도의 도구를 사용하는 불필요한 작업이 제거된다. CoJBother는 협업시스템과 긴밀히 연동되어 협업에 참여하는 사람들과 자동적으로 연결이 되어 편리하게 의사소통할 수 있도록 한다. 특히 따라서 협업 참여자들은 협업에 필요한 의사소통을 위하여 별도의 도구를 사용하지 않고 협업시스템과 연동되는 메신저를 사용하여 보다 효과적인 협업을 수행할 수 있다.

5. 결론 및 향후 연구

본 논문에서는 여러 사용자가 CoSlide 협업시스템을 이용하여 협업을 수행할 때 그룹 구성원들 간의 원활한 통신을 지원하는 메신저인 CoJBother의 개발에 대하여 기술하였다. CoJBother는 메신저를 위한 표준 프로토콜 중의 하나인 Jabber 프로토콜을 사용하는 JBother 메신저를 확장하여 구현되었으며 기존의 Jabber 사용자와 그룹 간의 통신과 더불어 CoSlide 협업시스템의 사용자와 그

현재의 CoSlide 협업시스템 사용자는 협업을 수행하기 위하여 협업 클라이언트와 더불어 다른 사용자이나 그룹과 통신을 하기 위하여 CoJBother를 별도로 사용하여야 한다. 두 개의 프로그램을 따로 사용하여 통신을 하고 협업을 수행하는 것이 효율성을 저해하는 요인이 될 수 있으므로 향후 연구는 협업시스템 클라이언트와 CoJBother 메신저를 융합하여 보다 효과적인 협업 환경을 제공하고자 한다.

6. 참고 문헌

- [1] "http://jakarta.apache.org/slide/", Jakarta Slide
- [2] Y. Goland, E. Whitehead, A. Faizi, S. Carter, D. Jensen, "HTTP Extensions for Distributed Authoring - WEBDAV," RFC 2518, Standards Track, February, 1999
- [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] G. Clemm, E. Sedlar, J. Whitehead, "Web Distributed Authoring and Versioning
- [5] 김동호, 박진호, 신원준, 이명준, "웹데브 기반의 효과적인 협업 작업 지원" 한국정보과학회, 한국컴퓨터종합학술대회 2006
- [6] "http://www.jabber.org", Jabber Software Foundation
- [7] 이홍창, 박진호, 신원준, 이명준, "웹데브 기반의 협업 시스템을 위한 Jabber 프로토콜의 확장", 한국정보과학회, 한국컴퓨터종합학술대회 2007
- [8] "http://jabberd.org", Jabberd
- [9] "http://www.jbother.org", JBother
- [10] "http://jivesoftware.org/smack", Smack
- [11] 안건태, 정명희, 이근웅, 문남두, 이명준, "iPlace: EJB 기술을 이용한 웹 기반 협업시스템", 한국정보처리학회논문지, 제8-D권, pp. 735-746, 2001년 12월.
- [12] Appelt, W., Mambrey, P. "Experiences with the BSCW Shared Workspace System as the Backbone of a Virtual Learning Environment for Students," Proceedings of the World Conference on Educational Multimedia, Hypermedia and Telecommunications ED-MEDIA 99, Seattle, June 1999