



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2009-0108872
(43) 공개일자 2009년10월19일

(51) Int. Cl.

G06Q 50/00 (2006.01)

(21) 출원번호 10-2008-0034197

(22) 출원일자 2008년04월14일

심사청구일자 2008년04월14일

(71) 출원인

울산대학교 산학협력단

울산 남구 무거2동 산29

(72) 발명자

이명준

울산 중구 다운동 702번지 다운동아름마을아파트 103동 1402호

이홍창

울산 중구 성안동 경동햇빛마을1차아파트 101동 907호

(뒷면에 계속)

(74) 대리인

장한특허법인

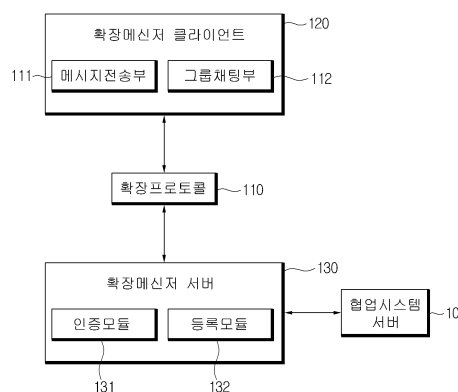
전체 청구항 수 : 총 11 항

(54) 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템

(57) 요약

본 발명에 따르면, CoSlide 협업시스템에서 지원되는 메신저 시스템의 구축에 관한 것으로서, 오픈 메신저 프로토콜인 Jabber프로토콜을 확장하여 구축된 확장프로토콜; 확장프로토콜을 지원하도록, Jabber프로토콜에 기반한 메신저서버인 Jabberd서버를 확장하여 구축되며, 협업시스템에 접속하여 협업 그룹에 참여한 사용자 및 사용자가 소속된 그룹 정보인 사용자/그룹 정보를 취출하여 제공하는 확장메신저서버; 및 확장프로토콜을 지원하도록, Jabber프로토콜에 기반한 메신저클라이언트인 JBother클라이언트를 확장하여 구축되며, 확장메신저서버에 접속하여 제공받은 협업시스템의 사용자/그룹 정보를 표시하여 사용자별 또는 그룹별 메신저 서비스를 제공하는 확장메신저클라이언트;를 포함하는, 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템이 제공된다. 개시된 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템에 따르면, 다수의 사용자가 CoSlide를 이용하여 협업 수행시 그룹 구성원들 간의 원활한 통신을 지원하며, 기존 Jabber 사용자와 그룹 간의 통신과 더불어 CoSlide 협업시스템의 사용자와 그룹 간의 통신을 지원하여 보다 효과적인 협업 환경을 제공할 수 있다.

대표도 - 도1



(72) 발명자

박진호

울산 남구 신정1동 636-12번지

김성훈

울산 남구 삼산동 평창3차아파트 602동 1507호

이 발명을 지원한 국가연구개발사업

과제고유번호 C1090-0701-0039

부처명 디지털제조정보기술연구센터(DMITRC)

연구사업명 대학IT연구센터육성지원사업

연구과제명 Augmented Reality 기반의 디지털 제조정보기술개발(4차년)

주관기관 울산대학교 산학협력단

연구기간 2007년 09월 01일 ~ 2007년 12월 31일

특허청구의 범위

청구항 1

CoSlide 협업시스템에서 지원되는 메신저 시스템의 구축에 관한 것으로서,

오픈 메신저 프로토콜인 Jabber프로토콜을 확장하여 구축된 확장프로토콜;

상기 확장프로토콜을 지원하도록, 상기 Jabber프로토콜에 기반한 메신저서버인 Jabberd서버를 확장하여 구축되며, 상기 협업시스템에 접속하여 협업 그룹에 참여한 사용자 및 사용자가 소속된 그룹 정보인 사용자/그룹 정보를 추출하여 제공하는 확장메신저서버; 및

상기 확장프로토콜을 지원하도록, 상기 Jabber프로토콜에 기반한 메신저클라이언트인 JBother클라이언트를 확장하여 구축되며, 상기 확장메신저서버에 접속하여 제공받은 상기 협업시스템의 사용자/그룹 정보를 표시하여 사용자별 또는 그룹별 메신저 서비스를 제공하는 확장메신저클라이언트;를 포함하는, 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템.

청구항 2

제 1항에 있어서, 상기 확장메신저서버 및 상기 확장메신저클라이언트는,

상기 확장프로토콜의 지원 이외에도, 확장되기 전의 프로토콜인 상기 Jabber프로토콜을 더 지원하는 것을 특징으로 하는, 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템.

청구항 3

제 2항에 있어서, 상기 확장메신저서버는,

상기 Jabber프로토콜에 기반하여 상기 Jabberd서버에 접속된 일반 Jabber의 사용자/그룹 정보를 더 추출하여 제공하고,

상기 확장메신저클라이언트는,

상기 협업시스템의 사용자/그룹 정보의 표시 이외에도, 상기 확장메신저서버로부터 제공받은 상기 일반 Jabber의 사용자/그룹 정보를 더 표시하는 것을 특징으로 하는, 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템.

청구항 4

제 3항에 있어서,

상기 협업시스템의 그룹 정보 및 상기 일반 Jabber의 그룹 정보에서 상기 그룹 정보는,

상기 그룹의 그룹명, 상기 그룹에 소속된 이웃 사용자의 정보를 포함하는 것을 특징으로 하는, 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템.

청구항 5

제 3항에 있어서, 상기 확장메신저클라이언트는,

상기 확장메신저서버로부터 전송받은 상기 일반 Jabber의 사용자/그룹 정보와, 상기 협업시스템의 사용자/그룹 정보를 이용하여, 상기 일반 Jabber의 사용자별, 상기 일반 Jabber의 그룹별, 상기 협업시스템의 사용자별, 상기 협업시스템의 그룹별로 각각의 가상공간을 제공하는 것을 특징으로 하는, 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템.

청구항 6

제 3항에 있어서, 상기 확장메신저클라이언트는,

상기 확장메신저서버로부터 전송받은 상기 일반 Jabber의 사용자/그룹 정보와, 상기 협업시스템의 사용자/그룹 정보를 이용하여, 해당 사용자가 소속된 Jabber 그룹별 또는 협업시스템 그룹별 채팅이 가능하도록 그룹채팅방을 제공하는 것을 특징으로 하는, 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템.

청구항 7

제 3항 내지 제 6항 중 어느 한 항에 있어서, 상기 확장프로토콜은,

상기 Jabber프로토콜에서 사용자 정보 표현을 위해 XML로 정의된 xdb파일의 XML스키마를 확장하여 구현하되,

상기 확장메신저서버에 접속한 사용자가 소속된 각 그룹별 타입 속성을 나타내는 타입태그(<type>)를 상기 xdb 파일 내의 아이템태그(<item>) 하부에 추가하여 하나의 사용자가 다수의 그룹에 소속되어 타 사용자와 협업하는 메신저 환경을 지원하며,

상기 타입태그를 이용하여, 상기 일반 Jabber의 그룹 내 사용자와 상기 협업시스템의 그룹 내 사용자 간의 구분이 가능한 것을 특징으로 하는, 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템.

청구항 8

제 3항 내지 제 6항 중 어느 한 항에 있어서, 상기 확장메신저서버는,

상기 일반 Jabber의 사용자와 상기 협업시스템의 사용자 간의 인증 구분을 위해,

상기 협업시스템의 사용자 인증을 위한 인증모듈; 및 상기 인증모듈을 통해 인증된 협업시스템의 사용자인 경우, 상기 협업시스템의 DB 내에 접속하여 그에 대응되는 협업시스템의 사용자/그룹 정보를 추출하여 상기 확장메신저클라이언트로 전송하는 등록모듈;을 각각 상기 Jabber서버에 추가하여 확장 구현되는 것을 특징으로 하는, 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템.

청구항 9

제 8항에 있어서, 상기 확장메신저서버는,

상기 일반 Jabber의 사용자로 접속 인증된 경우, 상기 xdb파일을 이용하여 사용자 인증에 관한 응답을 상기 확장메신저클라이언트 측으로 전송하는 것을 특징으로 하는, 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템.

청구항 10

제 3항 내지 제 6항 중 어느 한 항에 있어서, 상기 확장메신저클라이언트는,

상기 JBother클라이언트에서 Jabber프로토콜의 사용을 위해 이용되는 Smack라이브러리를 수정하여 구현하되,

상기 협업시스템의 사용자가 상기 확장메신저서버에 접속시, 상기 확장메신저서버 측을 향해 사용자 인증에 관한 XML전송을 수행하여 사용자 인증을 받는 인증용 네임스페이스, 및 상기 확장메신저서버로부터 전송받은 XML 문서로부터 상기 협업시스템의 사용자와 이웃한 이웃 사용자 정보를 구별하기 위한 이웃구별용 네임스페이스를 각각 상기 Smack라이브러리에 사용하는 것을 특징으로 하는, 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템.

청구항 11

제 10항에 있어서, 상기 확장메신저클라이언트는,

해당 사용자가 소속된 Jabber 그룹별 또는 협업시스템 그룹별로, 해당 그룹 내의 사용자에게 메시지 전송을 수행하는 메시지전송부, 및 해당 그룹 내의 사용자와 그룹채팅을 수행하는 그룹채팅부를 포함하는 것을 특징으로 하는, 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템.

명세서

발명의 상세한 설명

기술 분야

<1> 본 발명은 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템에 관한 것으로서, 보다 상세하게는 웹데브 기반의 협업에 참여한 사용자 또는 그룹 간의 메시징, 그룹채팅 등의 커뮤니케이션 기능을 제공하여 협업 효율을 향상시킬 수 있는, 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템에 관한 것이다.

배경 기술

- <2> 협업시스템인 Jakarta Slide는 아파치 웹서버에서 동작하는 모듈로서 WebDAV 프로토콜을 통하여 다양한 콘텐츠의 비동기적인 협업 환경을 제공한다. Jakarta Slide에서 지원하는 웹데브(WebDAV) 프로토콜은 HTTP/1.1을 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.
- <3> 그런데, 웹데브 프로토콜은 사용자와 그룹을 위한 가상 작업 공간을 명시적으로 지원하지 않으므로, Jakarta Slide를 통하여 복잡한 협업을 지원하는 것은 매우 어려운 작업이다.
- <4> 여기서, CoSlide 협업시스템은 이러한 Jakarta Slide의 문제점을 개선하기 위하여 확장된 시스템으로서, 그룹 작업을 위한 다양한 가상 작업공간을 지원함으로써 보다 효과적인 협업 환경을 제공한다. 즉, 이러한 CoSlide 협업시스템은 원격지의 자원과 가상공간을 활용하여 효과적인 협업을 지원하도록 Jakarta Slide 협업시스템을 사용하여 구축되었다.
- <5> 기존의 Jakarta Slide는 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.
- <6> 그런데, CoSlide는 Jakarta Slide의 기능을 바탕으로 개별 사용자와 사용자 그룹 각각에 대하여 체계적으로 가상공간을 제공함으로써 보다 효과적인 협업 환경을 제공한다.
- <7> 이러한 CoSlide에서 제공하는 협업 환경을 사용하기 위해서는 CoSlide의 사용자로 등록되어야 하며, 개별 사용자는 여러 그룹에 소속되어 다양한 종류의 협업을 수행할 수 있다. 사용자는 CoDAView와 같은 CoSlide 전용 클라이언트를 통하여 자신의 가상공간과 자신이 속한 그룹의 가상공간에 접근하여 다양한 협업 활동을 수행할 수 있다.
- <8> 이렇게 여러 사용자가 다수의 그룹에 속하여 다른 사용자들과 협업을 수행할 때 사용자들의 원활한 통신을 위해서는 주어진 협업 환경에 부합하는 동기적인 협업을 위한 도구가 필요하다.
- <9> 이를 위해서는 기존에 Jabber 프로토콜을 기반으로 한 JBother 메신저를 확장할 필요가 있으며, 또한 기존의 Jabber 사용자 및 그룹과 더불어 상기 CoSlide 협업시스템의 사용자 및 그룹 간의 메시징 및 통신 기능을 지원할 필요가 있다.

발명의 내용

해결 하고자하는 과제

- <10> 본 발명은 상술한 문제점을 해결하기 위하여 창출된 것으로서, 다수의 사용자가 CoSlide를 이용하여 협업 수행 시 그룹 구성원들 간의 원활한 통신을 지원하며, 기존 Jabber 사용자와 그룹 간의 통신과 더불어 CoSlide 협업시스템의 사용자와 그룹 간의 통신을 지원하여 보다 효과적인 협업 환경을 제공하는, 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템을 제공하는 데 그 목적이 있다.

과제 해결수단

- <11> 상기와 같은 목적을 달성하기 위한 본 발명의 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템은, CoSlide 협업시스템에서 지원되는 메신저 시스템의 구축에 관한 것으로서, 오픈 메신저 프로토콜인 Jabber프로토콜을 확장하여 구축된 확장프로토콜; 상기 확장프로토콜을 지원하도록, 상기 Jabber프로토콜에 기반한 메신저서버인 Jabberd서버를 확장하여 구축되며, 상기 협업시스템에 접속하여 협업 그룹에 참여한 사용자 및 사용자가 소속된 그룹 정보인 사용자/그룹 정보를 추출하여 제공하는 확장메신저서버; 및 상기 확장프로토콜을 지원하도록, 상기 Jabber프로토콜에 기반한 메신저클라이언트인 JBother클라이언트를 확장하여 구축되며, 상기 확장메신저서버에 접속하여 제공받은 상기 협업시스템의 사용자/그룹 정보를 표시하여 사용자별 또는 그룹별 메신저 서비스를 제공하는 확장메신저클라이언트;를 포함한다.
- <12> 그리고, 상기 확장메신저서버 및 상기 확장메신저클라이언트는, 상기 확장프로토콜의 지원 이외에도, 확장되기 전의 프로토콜인 상기 Jabber프로토콜을 함께 지원하는 것이 바람직하다.
- <13> 한편, 상기 확장메신저서버는, 상기 Jabber프로토콜에 기반하여 상기 Jabberd서버에 접속된 일반 Jabber의 사

용자/그룹 정보를 더 추출하여 제공하고, 상기 확장메신저클라이언트는, 상기 협업시스템의 사용자/그룹 정보의 표시 이외에도, 상기 확장메신저서버로부터 제공받은 상기 일반 Jabber의 사용자/그룹 정보를 더 표시할 수 있다.

- <14> 또한, 상기 협업시스템의 그룹 정보 및 상기 일반 Jabber의 그룹 정보에서 상기 그룹 정보는, 상기 그룹의 그룹명, 상기 그룹에 소속된 이웃 사용자의 정보를 포함할 수 있다.
- <15> 그리고, 상기 확장메신저클라이언트는, 상기 확장메신저서버로부터 전송받은 상기 일반 Jabber의 사용자/그룹 정보와, 상기 협업시스템의 사용자/그룹 정보를 이용하여, 상기 일반 Jabber의 사용자별, 상기 일반 Jabber의 그룹별, 상기 협업시스템의 사용자별, 상기 협업시스템의 그룹별로 각각의 가상공간을 제공할 수 있다.
- <16> 또한, 상기 확장메신저클라이언트는, 상기 확장메신저서버로부터 전송받은 상기 일반 Jabber의 사용자/그룹 정보와, 상기 협업시스템의 사용자/그룹 정보를 이용하여, 해당 사용자가 소속된 Jabber 그룹별 또는 협업시스템 그룹별 채팅이 가능하도록 그룹채팅방을 제공할 수 있다.
- <17> 한편, 상기 확장프로토콜은, 상기 Jabber프로토콜에서 사용자 정보 표현을 위해 XML로 정의된 xdb파일의 XML스키마를 확장하여 구현하되, 상기 확장메신저서버에 접속한 사용자가 소속된 각 그룹별 타입 속성을 나타내는 타입태그(<type>)를 상기 xdb파일 내의 아이템태그(<item>) 하부에 추가하여 하나의 사용자가 다수의 그룹에 소속되어 타 사용자와 협업하는 메신저 환경을 지원하며, 상기 타입태그를 이용하여, 상기 일반 Jabber의 그룹 내 사용자와 상기 협업시스템의 그룹 내 사용자 간의 구분이 가능한 것을 특징으로 한다.
- <18> 또한, 상기 확장메신저서버는, 상기 일반 Jabber의 사용자와 상기 협업시스템의 사용자 간의 인증 구분을 위해, 상기 협업시스템의 사용자 인증을 위한 인증모듈; 및 상기 인증모듈을 통해 인증된 협업시스템의 사용자인 경우, 상기 협업시스템의 DB 내에 접속하여 그에 대응되는 협업시스템의 사용자/그룹 정보를 추출하여 상기 확장메신저클라이언트로 전송하는 등록모듈;을 각각 상기 Jabber서버에 추가하여 확장 구현될 수 있다.
- <19> 여기서, 상기 확장메신저서버는, 상기 일반 Jabber의 사용자로 접속 인증된 경우, 상기 xdb파일을 이용하여 사용자 인증에 관한 응답을 상기 확장메신저클라이언트 측으로 전송할 수 있다.
- <20> 한편, 상기 확장메신저클라이언트는, 상기 JBother클라이언트에서 Jabber프로토콜의 사용을 위해 이용되는 Smack라이브러리를 수정하여 구현하되, 상기 협업시스템의 사용자가 상기 확장메신저서버에 접속시, 상기 확장메신저서버 측을 향해 사용자 인증에 관한 XML전송을 수행하여 사용자 인증을 받는 인증용 네임스페이스, 및 상기 확장메신저서버로부터 전송받은 XML문서로부터 상기 협업시스템의 사용자와 이웃한 이웃 사용자 정보를 구별하기 위한 이웃구별용 네임스페이스를 각각 상기 Smack라이브러리에 사용할 수 있다.
- <21> 그리고, 상기 확장메신저클라이언트는, 해당 사용자가 소속된 Jabber 그룹별 또는 협업시스템 그룹별로, 해당 그룹 내의 사용자에게 메시지 전송을 수행하는 메시지전송부, 및 해당 그룹 내의 사용자와 그룹채팅을 수행하는 그룹채팅부를 포함할 수 있다.

효 과

- <22> 본 발명에 따른 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템인 CoJBother 메신저에 따르면, 표준 Jabber 프로토콜을 지원하여 일반 Jabber 사용자들 간에 통신을 지원할 뿐만 아니라 협업시스템에 맞게 확장된 jabber 프로토콜을 지원하여 협업시스템에 참여하는 사용자들 간의 통신도 지원한다.
- <23> 또한, 본 발명의 CoJBother 메신저는 협업시스템과 긴밀히 연동되어 협업에 참여하는 사람들과 자동적으로 연결이 되어 편리하게 의사소통할 수 있도록 하므로, 협업 참여자들로 하여금 협업에 필요한 의사소통을 위해 별도의 E-메일이나 전화 등의 도구를 사용하는 불편함 없이, 협업시스템과 연동됨에 따라 효과적인 협업을 수행할 수 있다.

발명의 실시를 위한 구체적인 내용

- <24> 이하 첨부된 도면을 참조하면서 본 발명에 따른 바람직한 실시예를 상세히 설명하기로 한다. 이에 앞서, 본 명세서 및 청구범위에 사용된 용어나 단어는 통상적이거나 사전적인 의미로 한정해서 해석되어서는 아니 되며, 발명자는 그 자신의 발명을 가장 최선의 방법으로 설명하기 위해 용어의 개념을 적절하게 정의할 수 있다는 원칙에 입각하여, 본 발명의 기술적 사상에 부합하는 의미와 개념으로 해석되어야만 한다.
- <25> 따라서, 본 명세서에 기재된 실시예와 도면에 도시된 구성은 본 발명의 가장 바람직한 일 실시예에 불과할 뿐

이고 본 발명의 기술적 사상을 모두 대변하는 것은 아니므로, 본 출원시점에 있어서 이들을 대체할 수 있는 다양한 균등물과 변형예들이 있을 수 있음을 이해하여야 한다.

- <26> 도 1은 본 발명의 실시예에 따른 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템의 구성도, 도 2는 도 1의 상세 구성도, 도 3은 도 1의 다른 구성도이다. 그리고, 도 4는 종래의 JBother 메신저의 기본인터페이스 구성도, 도 5는 CoSlide 협업시스템의 시스템 동작도, 도 6은 CoSlide 협업시스템의 DAView 클라이언트 구성도이다.
- <27> 또한, 도 7은 CoSlide 협업시스템의 그룹 정보를 표시하는 CoJBother의 기본 인터페이스 구성도, 도 8은 CoSlide 협업시스템 클라이언트와 연동된 CoJBother의 사용 예시도, 도 9는 CoJBother를 이용한 그룹채팅 구성도, 도 10은 그룹 구성원들이 CoJBother를 통하여 통신하는 예시도, 도 11은 본 발명의 메신저 시스템의 도입 전후 협업시스템 작업 비교도이다.
- <28> 본 발명은 다수의 사용자가 CoSlide 협업시스템(200)을 이용하여 협업을 수행할 때 그룹 구성원들 간의 원활한 통신(실시간 메시징, 그룹채팅 등)을 지원하는 메신저 시스템(100)인 'CoJBother 메신저'의 개발에 대하여 기술한다.
- <29> 이러한 CoJBother 메신저(100)는, 메신저를 위한 표준 프로토콜 중의 하나인 표준 Jabber 프로토콜을 기반으로 하는 JBother 메신저를 확장하여 구현된 것으로서, CoSlide 협업시스템(200)의 사용자와 그룹을 지원하기 위하여 Jabber 프로토콜을 확장하여 구성된다.
- <30> 이에 따라, CoJBother 메신저(100)는 '기존의 일반 Jabber 사용자와 그룹'의 정보와 더불어 'CoSlide 협업시스템(200)의 사용자와 그룹'의 정보를 표시하고 각 정보를 통한 상호 간의 통신을 지원하여 보다 효과적인 협업 환경을 제공할 수 있도록 한다.
- <31> CoSlide 협업시스템(200)을 통하여 협업을 수행하는 사용자는, CoJBother 메신저(100)를 통하여 별도의 관리 작업 없이 그룹 구성원들과 그룹 채팅을 수행할 수 있다.
- <32> CoJBother 메신저(100)에서 사용한 프로토콜인 Jabber 프로토콜은 XML로 구성된 XMPP 기반으로 플랫폼이나 서버, 클라이언트에 상관없이 사용될 수 있으며, Jabber를 사용하는 모든 사용자들과 연결이 될 수 있는 오픈 메신저 프로토콜이다.
- <33> 이러한 CoJBother 메신저(100)는 기존의 JBother 메신저에서 사용하는 Jabber 프로토콜을 확장하여 CoSlide 협업시스템(200)의 사용자 간 및 그룹 간의 통신을 지원한다.
- <34> 요약하면, 본 발명에서는, 메신저 시스템(100)이 CoSlide 협업시스템(200)과 연동되기 위한 Jabber 프로토콜의 확장, 그리고 확장된 Jabber 프로토콜을 사용하여 CoSlide 협업시스템(200)의 사용자와 그룹 간의 원활한 통신을 지원할 수 있도록 JBother 메신저의 확장이 요구된다.
- <35> 물론, CoSlide 협업시스템(200)의 사용자와 그룹을 지원하기 위하여, 상기 Jabber프로토콜을 확장함과 동시에, 확장된 Jabber 프로토콜(110)을 지원하는 도 2의 Jabberd 서버(137)의 확장 또한 요구된다.
- <36> 이하에서는, 본 발명의 구체적인 설명에 앞서, Jabber 프로토콜, Jabberd 서버(137), JBother 메신저, CoSlide 협업시스템(200), 웹데브프로토콜(20)에 관하여 간단히 살펴보기로 한다.
- <37> 1) Jabber 프로토콜
- <38> Jabber 프로토콜은 '실시간 메신저의 리눅스'라고 잘 알려진 오픈 소스 실시간 메신저 서비스이다, XML로 구성된 XMPP 기반으로 정형화된 형식의 메시지를 주고받음으로써 플랫폼이나 서버, 클라이언트와 무관하게 여러 사용자가 다양한 형식을 통하여 서로 통신을 할 수 있도록 한다.
- <39> 즉, Jabber 프로토콜은 XML 프로토콜로서 간단한 구조로 정의되어 있어 응용이 용이하고 플랫폼에서 사용 가능한데, 특히 인터넷상의 두 요소들 간에 메시지나 프레젠스(presence), 혹은 또 다른 구조의 정보를 실시간으로 교환할 수 있도록 한다. Jabber프로토콜의 특징은 표 1과 같다. 후술하겠지만 본 발명은 이러한 Jabber 프로토콜을 확장하여 도 1 또는 도 2와 같은 확장프로토콜(110)을 구축하게 된다.

[표 1] Jabber 기술의 특징

Open	Jabber 프로토콜은 무료이며 공개되어있고 이해하기 쉬운 구조로 되어있음
Standard	XMPP라는 인스턴트 메시징과 프레젠테이션 기술을 바탕으로 IETF에 의해 표준화됨
Proven	1988년 Jeremie Miller에 의해 개발된 이후 Jabber 기술은 현재 보다 안정화 되었고 수천 명 이상의 개발자들이 참여하고 있음
Decentralized	Jabber 네트워크의 구조는 이메일과 비슷하게 Jabber 사용자는 서버의 위치나 존재 유무를 알 필요 없이 이용할 수 있음
Secure	SASL과 TLS 등의 강력한 보안 시스템과 독립된 서버로 보안성 향상
Extensible	누구나 응용할 수 있는 XML 네임스페이스로 쉽게 확장할 수 있음
Flexible	Jabber 응용프로그램은 다양한 도구와 함께 연동될 수 있음
Diverse	다양한 기업과 오픈 소스 프로젝트에서 실시간 응용프로그램과 서비스 개발을 위해 Jabber 프로토콜을 사용함

아래의 표 2는 Jabber 프로토콜에서 간단한 인스턴트 메시지 전송의 구조다.

[표 2] Jabber 인스턴트 메시지의 간단한 구조

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

2) Jabberd 서버

Jabberd 서버(137)는 실시간 메시징 프로토콜인 Jabber 프로토콜과 XMPP 프로토콜을 지원하는 서버로서, C/C++로 구현되어 있고, 컴포넌트의 추가, 삭제가 쉬운 모듈 구조로 구성되어 있으며 공개 소프트웨어로서 누구나 쉽게 사용할 수 있다.

특히 본 발명에서는 본 연구에서는 표준 Jabber 프로토콜뿐만 아니라 CoSlide 협업시스템(200)의 정보를 제공할 수 있도록, 확장된 Jabber 프로토콜(110)을 지원하기 위해 도 2와 같이 Jabberd 서버(137)를 확장하여 구현하였다.

3) JBother 메신저

JBother 메신저는 Jabber 프로토콜을 사용한 메신저로서 자바로 구현된 오픈소스 프로그램이다. Jabber 메신저는 Jabber 프로토콜을 이용하여 다양한 메신저의 기능을 확장하고 제공하는데, JBother는 자바언어로 구현된 대표적인 표준 Jabber 메신저이다. JBother는 도 4와 같이 윈도우 스타일 형식으로 그룹 채팅을 지원하며 다양한 리소스를 제공하고 트랜스포트 기능으로 다른 메신저와의 연동도 가능케 한다. 그리고 여러 스킨을 제공하여 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.

후술하겠지만, 본 발명은 이러한 JBother 메신저를 확장하여 CoJBother 메신저를 구축한다.

4) CoSlide 협업시스템(200)

도 2와 같은 CoSlide 협업시스템(200)은 원격지의 자원과 가상공간을 활용하여 효과적인 협업을 지원하는 시스템으로서, 기존의 WebDAV 프로토콜을 사용한 Jakarta Slide를 확장하여 구축된 것이다. CoDAView는 CoSlide에서 제공하는 협업 환경을 효과적으로 사용할 수 있도록 도와주는 클라이언트 도구로서, GUI 인터페이스를 통하여 가상공간의 자원을 쉽게 관리할 수 있다.

도 5와 같이, 협업시스템 서버(10)와 협업시스템 클라이언트(30)로 구성된 CoSlide 협업시스템(200)은 WebDAV 프로토콜(20)을 이용하여 협업을 수행할 수 있는 효과적인 기능들을 제공한다. 도 5에는 이러한 CoSlide 협업시스템의 전체적인 동작 과정을 보여준다.

4-1) CoSlide 서버(10)

Jakarta Slide는 협업을 지원하기 위한 표준 하부구조를 제공하는 웹 통신 프로토콜인 WebDAV를 사용하여 다양한 콘텐츠의 비동기적인 협업을 지원하는 협업 모듈이다. 하지만 기존의 Jakarta Slide는 협업을 하는

사용자별, 그룹별 가상공간을 지원하지 않아 다수의 사용자와 그룹이 참여하는 협업시 효과적인 협업 환경을 제공하기 어려웠다. Jakarta Slide의 문제점을 개선한 CoSlide는 협업에 참여하는 사용자와 그룹별로 가상공간을 제공하여 보다 효과적인 협업 환경을 제공한다. 도 5와 같이 CoSlide 협업시스템(200)을 사용하는 사용자는 자신만의 가상공간(개인작업장)을 제공받게 되며 참여하는 그룹의 가상공간(그룹작업장)에 접근하여 그룹의 구성원들과 용이하게 협업을 수행할 수 있다.

<56> 4-2) CoDAView 클라이언트(30)

<57> 도 6을 참고하면, CoDAView 클라이언트(30)는 CoSlide 서버(10)에 접근하여 협업을 수행할 수 있는 인터페이스와 자원 관리를 위한 WebDAV 명령을 수행하는 기능을 제공하는 도구이다. CoDAView 클라이언트(30)의 인터페이스는 QT의 클래스들로 구성되며 시그널/슬롯의 이벤트 핸들 방식으로 이벤트를 처리한다. WebDAV 명령을 수행하기 위해서는 CoSlide 서버(10)에 요청을 보내는 클래스들과 WebDAV 클라이언트 라이브러리를 통하여 사용자의 요청에 따른 자원 관리를 수행한다.

<58> 5) 웹데브 프로토콜(WebDAV 프로토콜;20)

<59> WebDAV 프로토콜(20)은 WebDAV 인터넷을 통하여 다양한 콘텐츠의 비동기적인 협업 저장을 지원하기 위한 프로토콜이다. WebDAV는 HTTP/1.1을 확장한 프로토콜로서 사용자들에게 원거리 서버들의 파일을 수정하고 관리할 수 있도록 한다. WebDAV의 기능은 속성을 이용한 자원관리, 덮어쓰기 방지, 이름 공간 관리 등이 있다. WebDAV는 각각의 기능들을 위해 아래의 표 3에서 보는 것과 같이 HTTP1.1의 메서드도 사용하고 있으며 일부 메서드는 확장하였고 몇몇의 메서드들은 추가되었다.

<60> [표 3] WebDAV에서 제공하는 메서드들

<61>

메 서 드	기 능
HEAD, TRACE	네트워크 행동을 찾고 추적하는 기능
GET	문서를 서버에서 받음
PUT, POST	문서를 서버에 전달
DELETE	자원 삭제
MKCOL	컬렉션 생성
PROPFIND, PROPPATCH	자원의 속성을 검색하고 설정
COPY, MOVE	이름 공간 문맥 내에 있는 자원 관리
LOCK, UNLOCK	덮어 쓰기 방지 기능
OPTIONS	서버가 지원하는 메서드 출력

<62> 간단히 요약하면, 본 발명은 CoSlide 협업시스템(200)을 통하여 협업 수행시, 참여자들 간에 원활한 의사소통을 위한 메신저 시스템(100)을 개발하여 협업시스템(200)과 연동하는데 그 목적이 있다. 본 발명에서 메시징 프로토콜은 메시징 프로토콜로 널리 사용되는 Jabber 프로토콜을 확장하여 협업시스템(200)의 정보를 표현할 수 있도록 하였다. 메신저 서버(130)는 표준 Jabber 프로토콜을 지원하는 Jabberd 서버(137)를 확장하여, 확장된 Jabber 프로토콜(110)을 지원하도록 하였고, 메신저 클라이언트(120)는 JBother 메신저를 확장하여 구현하였다.

<63> 도 2는 협업시스템(200)의 구조와 함께 연동된 본 발명의 메신저 시스템(100)의 구조이며, 메신저 시스템(100)의 서버(130)는 협업시스템의 서버(10) 내에 저장된 DB에 접근하여, 협업시스템(200)에서 협업그룹에 참여한 사용자와 그룹에 대한 정보를 가져온다.

<64> 그리고, 이러한 정보를 확장된 Jabber 프로토콜(110)을 통하여, 클라이언트(120) 측에 제공하여 줌으로써, 본 발명의 메신저 시스템(100)에 접속한 사용자는 자신이 속한 협업 그룹의 사용자와 편리하게 의사 전달을 수행할 수 있다.

<65> 이하에서는, 앞서 1)~5)항목에서 설명한 내용을 바탕으로 하여, 본 발명의 구성에 관하여 도 1 내지 도 11을 참고로 하여 상세히 설명하고자 한다.

<66> 도 1 또는 도 2와 같이, 본 발명의 실시예에 따른 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템(100)은, 확장프로토콜(110), 확장메신저서버(130), 확장메신저클라이언트(120)로 구성된다.

<67> 먼저, 상기 확장프로토콜(110)은 상술한 오픈 메신저 프로토콜인 Jabber프로토콜을 확장하여 구축된 것이다.

<68> 상기 확장메신저서버(130)는, 상기 확장프로토콜(110)을 지원하도록, 상기 Jabber프로토콜에 기반한 메신저서버

인 Jabberd 서버(137)를 확장하여 구축되며, CoSlide 협업시스템(200)의 서버(10)에 접속하여 협업 그룹에 참여한 사용자 및 사용자가 소속된 그룹 정보인 사용자/그룹 정보를 추출하여 제공하는 부분이다.

- <69> 여기서, 확장메신저클라이언트(120)는 상기 확장프로토콜(110)을 지원하도록, Jabber프로토콜에 기반한 메신저 클라이언트인 JBother클라이언트를 확장하여 구축된다. 또한, 확장메신저클라이언트(120)는 확장메신저서버(130)에 접속하여 제공받은 CoSlide 협업시스템(200)의 사용자/그룹 정보를 표시하여 사용자별 또는 그룹별 메신저 서비스를 제공할 수 있다.
- <70> 물론, 이러한 확장메신저클라이언트(120)는 상기 확장메신저서버(130)에 사용자에게 의해 유무선 접속 가능한 PC, 핸드폰, 노트북, PDA 등의 수단일 수 있다.
- <71> 한편, 상기 확장메신저서버(130) 및 확장메신저클라이언트(120)는, 상술한 확장프로토콜(110)의 지원 이외에도, 확장되기 전의 프로토콜인 Jabber프로토콜을 더 지원한다. 이는 확장프로토콜(110) 자체가 기존의 프로토콜 체계를 유지한 상태에서 XML 스키마의 수정을 통해 프로토콜이 확장된 것이기에 가능하다.
- <72> 여기서, 확장메신저서버(130)는, Jabber프로토콜에 기반하여 Jabberd 서버(137)에 접속된 일반 Jabber의 사용자/그룹 정보를 더 추출하여 제공 가능하다.
- <73> 이에 따르면, 확장메신저클라이언트(120)가 협업시스템(200)의 사용자/그룹 정보의 표시 이외에도, 확장메신저서버(130)로부터 제공받은 일반 Jabber의 사용자/그룹 정보 또한 표시 가능함을 의미한다.
- <74> 즉, 본 발명은 기존 Jabber 사용자와 그룹 간의 통신과 더불어, CoSlide 협업시스템(200)의 사용자와 그룹 간의 통신을 모두 지원함으로써, 보다 효과적인 협업 환경을 제공할 수 있다.
- <75> 한편, 협업시스템(200)의 그룹 정보 및 일반 Jabber의 그룹 정보에서 지칭하는 상기 '그룹 정보'란, 도 7과 같이, 다수의 해당 그룹에 대한 그룹명, 해당 그룹에 소속된 이웃 사용자의 정보를 포함하여, 이를 바탕으로 추후 사용자별 또는 그룹별 메시지 전송, 그룹채팅 등의 환경을 제공할 수 있게 된다.
- <76> 이상과 같은 개략적인 구성을 기반으로, 본 발명의 보다 상세한 구성에 관하여 설명하면 다음과 같다.
- <77> 우선, 상기 확장프로토콜(110)이 확장되기 전인 상기 'Jabber프로토콜'을 살펴보면 다음과 같다.
- <78> 표준 Jabber 서버인 Jabberd 서버(137)에서는 메신저 사용자의 정보를 저장하고 불러오기 위해 XML로 정의된 xdb 파일(133)을 사용한다. xdb 파일(133)은 사용자의 기본 정보 및 로그인 정보를 표현한다. Jabberd 서버(137)는 이 정보를 이용하여 사용자에게 이웃 목록을 전송할 수 있다.
- <79> [표 4] 사용자의 정보를 표현하는 xdb 파일

```
<xdb>
<password xmlns='jabber:iq:auth' xdbns='jabber:iq:auth'>1</password>
<query xmlns='jabber:iq:register' xdbns='jabber:iq:register'>
<username>jinho</username>
<password xmlns='jabber:iq:auth'>1</password></query>
<query xmlns='jabber:iq:roster' xdbns='jabber:iq:roster'>
<item jid='john@203.250.77.124' name='john' subscription='both' /></query>
<foo xmlns='jabber:x:offline' xdbns='jabber:x:offline' />
<query xmlns='jabber:iq:last' last='1165563334' xdbns='jabber:iq:last' />
</xdb>
```

- <81> 표 4에서 <query> 태그는 Jabber 프로토콜에서 표현하는 Infor/Query 정보(iq) 중에서 특정 정보를 얻거나 전송하기 위한 쿼리 정보를 나타낸다. <xmlns> 태그는 Jabber에서 일어나는 수많은 통신 이벤트에 대한 유형을 정의한 네임스페이스(NameSpace)를 나타낸다.
- <82> 'jabber:iq:register' 네임스페이스는 접속한 사용자의 아이디와 패스워드 정보를 표현한다. 'jabber:iq:roster' 네임스페이스는 접속한 사용자의 이웃 목록을 표현하며 하위에 <item> 태그로 이웃들을 개별적으로 저장한다.
- <83> Jabberd 서버(137)는 xdb 파일(133)로부터 사용자의 이웃 목록 정보를 얻는다. 그리고 클라이언트에게 다음과 같은 표준 Jabber 프로토콜을 통하여 사용자의 이웃 목록과 그룹 정보를 전달한다.

<84> [표 5] 사용자의 이웃 목록 정보

<85>

```
<query xmlns='jabber:iq:roster' xdbns='jabber:iq:roster'>
  <item jid='jinop@203.250.77.124' name='jinop' subscription='to'>
    <group>my friends</group>
  </item>
  <item jid='heinz@203.250.77.124' name='heinz' subscription='to'>
    <group>my friends</group>
  </item>
</query>
```

<86> 표 5에서 각각의 <item> 태그는 등록된 이웃을 나타내는데 jid는 등록된 이웃의 Jabber ID를 나타낸다. <item> 태그의 하위 태그로 <group> 태그가 존재하는데 이 태그를 사용하는 경우, 클라이언트 인터페이스에서는 해당 사용자가 특정 그룹 안에 포함되어 있음을 보여준다. 위의 그림에서 사용자 'jinop'와 'heinz'는 my friends라는 그룹에 소속되어 있음을 나타낸다.

<87> 하지만 상술한 Jabber프로토콜은 사용자의 이웃을 정의하는 <item> 태그에 <group> 태그를 하나 밖에 기술할 수 없기 때문에 한 사용자가 여러 그룹에 소속되는 것을 표현할 수 없다.

<88> 따라서, 다수의 사용자가 여러 그룹에 속해 협업을 하는 CoSlide 협업시스템(200)의 환경을 지원하기 위해서는 표준 Jabber프로토콜의 확장이 필요하다.

<89> 다시 말해서, 기존의 Jabber프로토콜은 한 사용자가 한 그룹에만 소속될 수 있으므로 다수의 사용자가 여러 그룹에 소속되어 작업을 하게 되는 협업시스템(200)의 정보를 표현하기는 어렵다. 이러한 문제점을 해결하고 협업 시스템(200)에 맞는 메신저 정보를 표현하기 위해서는 Jabber 프로토콜의 XML 스키마를 확장해야 한다.

<90> 즉, 상기 확장프로토콜(110)은 기존의 Jabber프로토콜에서 사용자 정보 표현을 위해 XML로 정의된 xdb파일(133)의 XML스키마를 확장하여 구현되는 것이다.

<91> 더 상세하게는, 확장메신저클라이언트(120)를 통해 확장메신저서버(130)에 접속한 사용자가 소속된 각 그룹별 타입 속성을 나타내는 상기 타입태그(<type>)를, 상기 xdb파일(133) 내의 아이템태그(<item>) 하부에 추가함으로써, 하나의 사용자가 다수의 그룹에 소속되어 타 사용자와 협업하는 메신저 환경을 지원할 수 있게 된다.

<92> 이상과 같이, 확장된 XML 스키마는 다수의 그룹에 속할 수 있는 사용자를 표현하기 위해 각 그룹별로 사용자를 복수 개로 표현할 수 있다. 아래의 표 6을 참고하면, <item> 태그 하위에 <type> 태그를 두어 사용자 그룹과 협업 작업을 하는 그룹의 구분 표현도 가능하다.

<93> [표 6] 협업시스템의 정보를 표현하기 위한 XML 문서의 예

<94>

```
<query xmlns='jabber:iq:CoSlide_roster' xdbns='jabber:iq:CoSlide_roster'>
  <item jid='jinop@203.250.77.124' name='jinop' subscription='to'>
    <group>my friends</group>
    <type>usergroup</type>
  </item>
  <item jid='heinz@203.250.77.124' name='heinz' subscription='to'>
    <group>my friends</group>
    <type>usergroup</type>
  </item>
  <item jid='jinop@203.250.77.124' name='jinop' subscription='from'>
    <group>CoSlide</group>
    <type>workgroup</type>
  </item>
</query>
```

<95> 상기 표 6의 XML 문서에서는 협업시스템의 정보를 표현하기 위해 jabber:iq:CoSlide_roster라는 XML 네임스페이스를 추가적으로 사용하였다. 또한, 새로운 네임스페이스에서 사용자의 그룹을 분류하기 위한 <type> 태그를 추

가하여 일반 Jabber 사용자와 협업시스템 사용자를 구별한다. 예를 들어, <type> 태그의 값이 'usergroup'이면 '일반 Jabber의 사용자'를 나타내고, <type> 태그의 값이 'workgroup'이면 '협업시스템(200)에 참여하고 있는 사용자'를 나타낸다.

- <96> 즉, 타입태그를 이용한다면 Jabberd 서버(137)에 접속한 일반 Jabber의 그룹 내 사용자와, 협업시스템(200) 그룹 내 사용자 간의 구분이 가능하게 되고, 추후 메신저 시스템(100) 상에서 각각에 대한 개별 정보의 구분 표현도 가능하게 된다.
- <97> 이렇게 확장된 XML 스키마를 통하여, 확장메신저서버(130)는 한 사용자가 여러 그룹에 소속되는 협업시스템(200)의 정보를 표현할 수 있는 확장된 Jabber 프로토콜, 즉 확장프로토콜(110)을 제공할 수 있게 된다.
- <98> 따라서, 확장메신저클라이언트(120) 측의 사용자가 이러한 확장프로토콜(110)을 통하여 확장메신저서버(130)에 접속 가능하도록 하여, 협업시스템(200)에 소속된 사용자들 간에 원활한 통신 지원할 수 있다.
- <99> 한편, 상기 확장메신저서버(130)는, 표준 Jabber 서버인 Jabberd 서버(137)를 CoSlide 협업시스템(200)에 맞도록 확장하여 구현되는데, Jabberd 서버(137)는 모듈 구조로 되어 있어서 서비스의 확장 및 추가가 용이한 이점이 있다.
- <100> 본 발명에서는 확장메신저서버(130)의 구성은, 도 1 또는 도 3과 같이 인증모듈(CoSlide_auth module;131)과 등록모듈(CoSlide_roster module;132)이 Jabberd 서버(137)에 추가되어 확장된 것이다, CoSlide 협업시스템(200)과 연동된 확장메신저서버(130)는 일반 Jabber의 사용자와 협업시스템(200)의 사용자 간을 구별하여 인증 가능하게 한다.
- <101> 인증모듈(131)은 협업시스템(200)의 사용자 인증을 위한 컴포넌트이고, 등록모듈(132)은 CoSlide 협업시스템(200)의 사용자와 그룹의 정보를 확장메신저클라이언트(120) 측으로 전송하기 위한 컴포넌트이다.
- <102> 확장된 Jabberd 서버(137)인 확장메신저서버(130)는 위의 2가지 모듈을 통해 CoSlide 협업시스템(200)과 연동한다. 추가된 모듈(131,132)은 도 3과 같이 JSM(Jabber Session Manager;135)의 하부 컴포넌트로 구현되는데, JSM의 하부 컴포넌트들은 Jabberd의 설정 파일인 jabberd.xml(136) 내의 <load main="jsm"/> 엔트리에 등록된다.
- <103> 이러한 확장메신저서버(130)는, 사용자가 일반 Jabber의 사용자로 접속 인증된 경우, 앞서 상술한 xdb파일(133)을 이용하여 사용자 인증에 관한 응답을 확장메신저클라이언트(120) 측으로 전송하게 된다.
- <104> 그런데, 상기 인증모듈(131)을 통해, 일반 사용자가 아닌 협업시스템(200)의 사용자로 인증받는 경우, xdb파일(133)뿐만 아니라 CoSlide 협업시스템(200)의 사용자/그룹 정보를 가져오기 위한 등록모듈(132)을 통해 협업시스템(200)의 사용자 정보 저장소(DB)에 인터페이스(134)를 통해 접속하게 된다.
- <105> 즉, 등록모듈(132)은 사용자가 인증모듈(131)을 통해 협업시스템(200)의 사용자로 인증되면, 협업시스템(200)의 DB 내에 접속하여 그에 대응되는 협업시스템(200)의 사용자/그룹 정보를 취출하여 확장메신저클라이언트(120)로 전송한다.
- <106> 아래의 표 7은 users.c 파일의 협업시스템 사용자로 인증을 받는 부분의 코드를 보여준다.

[표 7] 협업시스템의 사용자로 인증받는 부분의 코드

```
// CoSlide 협업시스템의 사용자 패스워드를 가져오는 메서드
char *get_iplace_pass(char * userid){
    ...
    char *name = xmlnode_get_attrib(cur, "name");
    // 입력된 사용자의 패스워드가 존재하면 패스워드를 반환
    if(strcmp(name, "password") == 0){
        value = xmlnode_get_attrib(cur, "value");
        return conv_encoding("EUC-KR", "UTF-8", value);
    }
    ...
}
// Jabberd에서 사용자 인증을 처리하는 메소드
udata js_user(jsmi si, jid id, HASHTABLE ht){
    ....
    newu->pass = pstrdup(p, get_iplace_pass(newu->user));
    newu->id = jid_new(p, jid_full(uid));
    return newu;
}
```

사용자가 인증모듈(131)을 통해 CoSlide 협업시스템(200)의 사용자로 인증을 받고 난 후, 서버는 그 사용자가 속해 있는 그룹의 정보를 클라이언트 측으로 전송한다. 협업시스템의 사용자/그룹 정보는 각각 XML문서로 정의되어 저장된다.

등록모듈(132)은 기존 Jabberd 서버(137)의 Roster 컴포넌트를 확장한 컴포넌트로 확장메신저클라이언트(120)가 확장메신저서버(130)에 접속을 하면, 협업시스템(200) 내의 사용자/그룹 정보를 얻어서 확장프로토콜(110)을 통하여 확장메신저클라이언트(120)로 보내도록 한다.

C로 구현된 Jabberd 서버(137)에서 XML 문서로부터 정보를 얻기 위해 XMLParser 라이브러리를 사용하였다. 표 8은 등록모듈(132)인 CoSlide_Roster 컴포넌트가 협업시스템(200)의 저장소(DB)에 접근하여 정보를 가져오는 코드 부분이다.

[표 8] 협업시스템의 저장소로부터 사용자 정보를 가져오는 코드

```
// CoSlide 협업시스템의 정보를 가져오는 메소드
xmlnode coslide_roster_get(char *user)
{
    ...
    DIR *dp;
    xmlnode ret;
    if((dp = opendir("/usr/local/tomcat-src/store/metadata/users")) == NULL )
        return ret;
    ret = xmlnode_new_tag("query");
    xmlnode_put_attrib(ret,"xmlns",NS_COSLIDE_ROSTER);
    while((dir_info = readdir(dp)) != NULL){
        ...
        strcat(dirtemp, dir_info->d_name);
        xmlnode temp = xmlnode_file(dirtemp);
        char *name = xmlnode_get_attrib(temp,"name");
        xmlnode tmp_item = xmlnode_new_tag("item");
        xmlnode_put_attrib(tmp_item, "jid", strtemp);
        ...
        xmlnode_insert_tag_node(ret,tmp_item);
        ....
    }
    ...
}
```

이상과 같이, 본 발명의 CoJBother 메신저(100)는 Jabber 메신저인 JBother을 확장한 것으로서, 표준 Jabber 프로토콜을 지원하여 일반 Jabber 사용자들 간에 통신을 지원할 뿐만 아니라, 협업시스템에 맞게 확장된 jabber 프로토콜을 지원하여 협업시스템에 참여하는 사용자들 간의 통신도 지원한다.

다음으로, 확장메신저클라이언트(120)에 관하여 상세히 알아보고자 한다.

'Jabber 메신저'는 Jabber 프로토콜을 이용하는 메시징 시스템을 가리키는 것으로 본 발명에서는 협업시스템(200)을 지원하도록 확장된 Jabber 프로토콜을 지원하는 Jabber 메신저를 개발한다. 즉, 이 메신저를 개발하기 위하여 표준 Jabber 프로토콜을 지원하는 'JBother 메신저'를 확장하도록 한다. JBother 메신저는 표준 Jabber 프로토콜을 지원할 뿐만 아니라 화려한 인터페이스와 그룹채팅 등의 다양한 기능을 지원한다.

하지만 JBother는 협업시스템(200)의 작업그룹정보를 자동적으로 표현하지 못하기 때문에 협업시스템(200)과 능동적으로 연동되려면 확장을 하여야 한다. JBother 메신저를 확장하여 개발된 본 발명의 CoJBother 메신저(100)는 확장된 Jabber 프로토콜을 지원하여 협업시스템에서 작업 중인 그룹 참여자정보를 자동적으로 표현하고 효과적인 통신을 위한 작업그룹별 그룹채팅 기능을 제공한다.

따라서, CoJBother 메신저(100)는 확장된 Jabber 프로토콜을 지원하고 그에 맞는 인터페이스와 기능을 제공한다.

CoJBother 메신저(100)가 협업시스템(200)을 지원하기 위하여 기존의 JBother 클라이언트에서 수정된 부분은 크게 2가지이다. 하나는 확장된 Jabberd 서버(137)와의 통신을 위하여 확장된 Jabber 프로토콜(110)을 사용할 수 있도록 Smack 라이브러리를 수정하는 것이고, 또 다른 하나는 서버(10)로부터 받은 협업시스템(200)의 정보를 사용자에게 맞도록 제공하는 인터페이스를 구현하는 것이다.

즉, 확장메신저클라이언트(120)는 JBother클라이언트에서 Jabber프로토콜의 사용을 위해 이용되는 Smack라이브러리를 수정하여 구현된다. 그 배경에 대하여 설명하면 다음과 같다.

기존의 JBother는 표준화된 Jabber프로토콜의 사용을 위해 Smack 라이브러리를 사용한다. Smack은 인스턴트 메시지와 상태 표시(Presence)기능을 제공하는 XMPP 기반의 오픈 소스 Jabber 클라이언트 라이브러리이다.

Smack 라이브러리는 자바로 구현되어 있으며 XML 포맷으로 데이터를 주고받는다. Smack 라이브러리는 Jabberd 서버(137)에 접근하는 방법과 접근하고 나서 Jabber프로토콜에 맞는 형식으로 데이터를 주고받을 수 있는 방법

을 제공한다.

<123> 표 9는 Smack 라이브러리를 통하여 간단하게 Jabber 프로토콜을 사용하는 방법을 보여준다.

<124> [표 9] Smack 라이브러리의 사용

<125>

```
// Jabber 서버에 접속
XMPPConnection connection = new XMPPConnection("jabber.org");
// Jabber 서버에 사용자로서 인증
connection.login("mtucker", "password");
// 지정한 사용자에게 메시지 전송
connection.createChat("jsmith@jivesoftware.com").sendMessage("Howdy!");
```

<126> 그리고, 표 10은 표 9와 연관되어 구성된 Jabber 프로토콜 메시지의 내용을 보여준다.

<127> [표 10] Smack 라이브러리를 통한 Jabber 프로토콜 메시지의 내용

<128>

```
// Jabber 서버에 사용자로서 인증
<query xmlns="jabber:iq:auth">
  <username>mtucker</username>
  <password>password</password>
</query>
// 지정한 사용자에게 메시지 전송
<message to='jsmith@jivesoftware.com' id='message22'>
  <body>Howdy!</body>
</message>
```

<129> 그런데, 본 발명의 경우, 상술한 바와 같은 Smack 라이브러리를 수정하여 확장메신저클라이언트(120)를 구성해야 한다.

<130> 즉, 확장메신저클라이언트(120)는 협업시스템(200)의 사용자가 확장메신저서버(130)에 접속시, 확장메신저서버(130) 측으로 사용자 인증에 관한 XML전송을 수행하여 사용자 인증을 받는 인증용 네임스페이스(jabber:iq:CoSlide_auth)를 상술한 Smack라이브러리에 사용하며, 또한 확장메신저서버(130)로부터 전송받은 XML문서로부터 협업시스템(200)의 사용자와 이웃한 이웃 사용자 정보를 구별하기 위한 이웃구별용 네임스페이스(jabber:iq:CoSlide_roster)를 상기 Smack라이브러리에 사용하게 된다.

<131> 더욱 상세하게는, 확장메신저클라이언트(120)는 Jabberd 서버(137)로 접속을 할 때 협업시스템(200)의 사용자로서 접속을 하기 위해 Smack 라이브러리에서 'jabber:iq:CoSlide_auth' 네임스페이스를 사용하여, 인증에 관련된 XML 문서를 전송하고 또한 확장메신저서버(130)로부터 인증을 받는다. 그리고, 확장메신저서버(130)로부터 전송받은 XML 문서에서 협업시스템(200)의 사용자에 대한 이웃 사용자를 구별해내기 위해 'jabber:iq:CoSlide_roster' 네임스페이스를 사용한다.

<132> 아래의 표 11은 협업시스템(200)의 사용자로 접근할 수 있도록 Smack 라이브러리에서 코드를 수정한 부분과, 확장메신저서버(130)로부터 받은 XML 문서로부터 협업시스템(200)의 사용자 정보를 분류하도록 코드를 수정한 부분이다.

[표 11] Smack 라이브러리의 수정

```
// 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 이웃정보를 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);
    ....
}
// 서버로부터 받은 Jabber 프로토콜 패킷으로 부터 사용자의 정보를 추출
private RosterPacket parseCoSlideRoster(XmlPullParser parser) throws Exception{
    ...
    String jid = parser.getAttributeValue("", "jid");
    ...
}
```

한편, 확장메신저클라이언트(120)는 수정된 Smack 라이브러리를 통해 협업시스템(200)에서 가져온 정보를 표현하기 위해, 기존의 JBother의 소스코드를 수정할 필요가 있다.

CoJBother 메신저(100)는 JBother의 인터페이스 부분을 수정해서 협업시스템(100)의 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.

아래의 표 12는 협업시스템(200)의 정보를 나타내는 인터페이스 코드 부분과 그룹 채팅을 지원하도록 수정한 코드 부분이다.

[표 12] 협업시스템 사용자 목록과 그룹 채팅 기능을 구현하는 소스 코드

```
// 협업시스템 사용자를 메신저 이웃 목록에 보여주도록 하는 코드
// Smack 라이브러리를 통해 협업시스템 사용자에게 대한 정보를 가져온 뒤 처리함
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 정보를 담은 북마크 실행
GroupChatBookmarks gc =
    new GroupChatBookmarks(BuddyList.getInstance().getTabFrame());
gc.setCoSlideGroupChat(groupname, "conference.203.250.77.124");
```

이렇게 소스들이 수정되며, 본 발명의 확장메신저클라이언트(120)는 확장메신저서버(130)에 접속된 해당 사용자가 소속된 Jabber 그룹별 또는 협업시스템 그룹별로, 해당 그룹 내의 사용자에게 메시지 전송을 수행하는 메시지전송부(121), 및 해당 그룹 내의 사용자와 그룹채팅을 수행하는 그룹채팅부(122)를 포함한다.

이상과 같이 구성되는 본 발명의 메신저 시스템인 CoJBother 메신저(100)의 사용 예를 살펴보면 다음과 같다.

CoSlide 협업시스템(200)은 XML 파일로 정의된 문서를 통하여 시스템 정보를 표현한다. 다음 표 13은 CoSlide 협업시스템(200)의 그룹 정보를 정의하는 XML 문서의 일부이다. 이 문서는 ITRC이라는 그룹의 속성을 정의하고 참여하는 사용자들의 정보를 표현한다.

[표 13] CoSlide 협업시스템의 그룹을 정의하는 XML 문서의 일부

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

다음 표 14는 'mathpf' 사용자가 CoJBother 메신저 시스템(100)을 통하여 CoSlide 협업시스템(200)에 접근하였을 때 자신이 소속된 그룹과 그룹원의 정보를 보여주는 기본 인터페이스를 보여준다.

도 7은 협업시스템(200)의 그룹 정보를 표시하는 CoJBother의 기본 인터페이스를 나타낸다.

확장메신저클라이언트(120)는, 확장메신저서버(130)로부터 전송받은 일반 Jabber의 사용자/그룹 정보와, 협업시스템(200)의 사용자/그룹 정보를 이용하여, 일반 Jabber의 사용자별, 일반 Jabber의 그룹별, 협업시스템(200)의

사용자별, 협업시스템(200)의 그룹별로 각각의 가상공간을 제공한다.

- <148> 이러한 확장메신저클라이언트(120)는 상기 일반 Jabber의 사용자/그룹 정보, 협업시스템(200)의 사용자/그룹 정보를 각각 이용하여, 해당 사용자가 소속된 Jabber 그룹별 또는 협업시스템 그룹별 채팅이 가능하도록 그룹채팅방을 제공하는데, 물론 그룹채팅방은 그룹채팅부(122)에 의해 가능하다.
- <149> 이상과 같이, 본 발명의 CoJBother 메신저는 협업시스템(200)을 위한 편리한 그룹 채팅 기능을 제공한다.
- <150> 확장메신저클라이언트(120) 측의 사용자가 팝업메뉴를 통한 그룹 채팅방 생성을 요청하면, 그룹의 정보와 사용자의 정보를 사용하여 확장메신저서버(130)로 그룹 채팅방 개설을 요청한다. 그에 따라 내부의 Jabberd 서버(137)는 conference 컴퍼넌트를 사용하여 그룹 채팅방을 개설하게 되어 사용자는 자동적으로 그룹 채팅방에 참여하게 된다.
- <151> 다음은 CoJBother 메신저(100)를 활용하여 협업을 수행하는 과정을 보여준다.
- <152> 도 8 내지 도 10과 같이, 사용자는 원격지에 떨어진 상태로 CoSlide 협업시스템(200)을 통하여 협업에 참여할 수 있고 다른 참여자와 정보를 주고받기 위하여 CoJBother 메신저, 즉 본 발명의 메신저 시스템(100)을 사용할 수 있다.
- <153> 사용자는 CoJBother 메신저(100)의 그룹채팅기능을 이용하여 손쉽게 협업 참여자들과 정보를 교환할 수 있는데, 그 간단한 예는 다음과 같다.
- <154> 1) 연구소 프로젝트의 참여자인 'jinop'와 'mathpf'는 프로젝트를 수행하기 위하여 각각 미국과 프랑스로 출장을 가게 되었음.
- <155> 2) 도 8과 같이 원격지에 떨어진 상태에서도 CoSlide 협업시스템(200)의 클라이언트(30)를 사용하여 출장 중 연구보고서 작성을 위한 자료를 웹데브프로토콜(20)을 이용하여 그룹작업장에 올리는 등의 협업을 수행함
- <156> 3) 원격지에 떨어져 있는 협업그룹 참여자들 간에 연구보고서 작성 문제로 긴급한 의사소통이 필요하게 되었음.
- <157> 4) E메일이나 전화와 같은 별도의 도구를 사용하지 않고, 도 8 내지 도 10에 도시된 협업시스템 클라이언트에 연동된 CoJBother 메신저(100)를 사용하여 협업그룹의 참여자들을 확인하고 원클릭으로 그룹채팅을 시작함.
- <158> 5) 프로젝트를 진행하기 위한 회의를 시작하고 성공적으로 의사를 교환함.
- <159> 다음은, 본 연구에서 개발된 CoJBother 메신저(100)의 성능을 평가하기 위하여 본 발명에 의해 연동된 협업시스템(200)과 다른 협업시스템을 비교하고 메신저의 연동에 따른 작업과정의 차이를 비교하였다.
- <160> 표 14는 자바 기술 기반으로 다양한 작업공간을 지원하는 iplace, 대표적인 웹기반 협업시스템인 BSCW와 함께 CoSlide 협업시스템의 차이점을 보여준다.

[표 14] 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
자원처리프로그램 자동 구동	×	×	○
실시간 메시징	×	×	○

- <163> CoSlide 협업시스템(200)은 기존의 협업시스템과는 달리 그룹작업장과 공개작업장을 완벽히 지원하고, 상술한 본 발명의 CoJBother 메신저(100)와 연동되어 협업을 수행하는 동시에 실시간 메시징 까지 가능하게 된다.
- <164> 다음으로, 도 11은 메신저가 없는 협업시스템(200)과, CoJBother 메신저(100)를 도입한 CoSlide 협업시스템

(200)에서, 협업 참여자들이 의사소통을 할 때 각각의 작업과정을 보여준다.

- <165> 협업시스템(200)을 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 정작 협업의 효율성을 떨어뜨린다.
- <166> 본 발명에서 개발된 CoJBother 메신저(100)를 도입하면 참여자들 간에 의사소통을 위해 별도의 도구를 사용하는 불필요한 작업이 제거된다. CoJBother 메신저(100)는 협업시스템(200)과 긴밀히 연동되어 협업에 참여하는 사람들과 자동적으로 연결이 되어 편리하게 의사소통할 수 있도록 한다. 따라서 협업 참여자들은 협업에 필요한 의사소통을 위하여 별도의 도구를 사용하지 않고 협업시스템(200)과 연동되는 메신저(100)를 사용하여 보다 효과적인 협업을 수행할 수 있다.
- <167> 이상과 같은 본 발명을 요약하면 다음과 같다.
- <168> 본 발명은 여러 사용자가 CoSlide 협업시스템(200)을 이용하여 협업을 수행할 때 그룹 구성원들 간의 원활한 통신을 지원하는 메신저인 CoJBother의 개발에 대하여 기술한 것이다. 이를 구현한 CoJBother 메신저(100)는 메신저를 위한 표준 프로토콜 중의 하나인 Jabber 프로토콜을 사용하는 JBother 메신저를 확장하여 구현되었으며 기존의 Jabber 사용자와 그룹 간의 통신과 더불어 CoSlide 협업시스템(200)의 사용자와 그룹 간의 통신을 또한 지원하도록 구현되었다.
- <169> CoSlide 협업시스템(200) 환경의 정보를 제공하기 위하여 Jabber 프로토콜을 확장하고 메시징 시스템을 위한 Jabber 서버인 Jabberd 서버(137)를 기반으로, 메시징 시스템 서버(130)를 개발하였다. CoJBother는 확장된 Jabberd 서버와의 효과적인 통신을 위하여 협업시스템 정보를 표현할 수 있도록 수정된 Smack 라이브러리를 사용하였으며 협업시스템 환경에 맞는 인터페이스를 제공한다. CoJBother는 CoSlide 협업시스템에 부합하는 기능을 제공함으로써 협업시스템의 사용자와 그룹 간의 효과적인 통신을 지원한다.
- <170> 현재의 CoSlide 협업시스템(200)의 사용자는 협업을 수행하기 위하여 협업 클라이언트와 더불어 다른 사용자이나 그룹과 통신을 하기 위하여 CoJBother 메신저를 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. 여기서, 이러한 융합된 메신저 환경 또한 본 발명의 범주에 속함은 자명하다.
- <171> 이상과 같이, 본 발명은 비록 한정된 실시예와 도면에 의해 설명되었으나, 본 발명은 이것에 의해 한정되지 않으며 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자에 의해 본 발명의 기술 사상과 아래에 기재될 청구 범위의 균등 범위 내에서 다양한 수정 및 변형이 가능함은 물론이다.

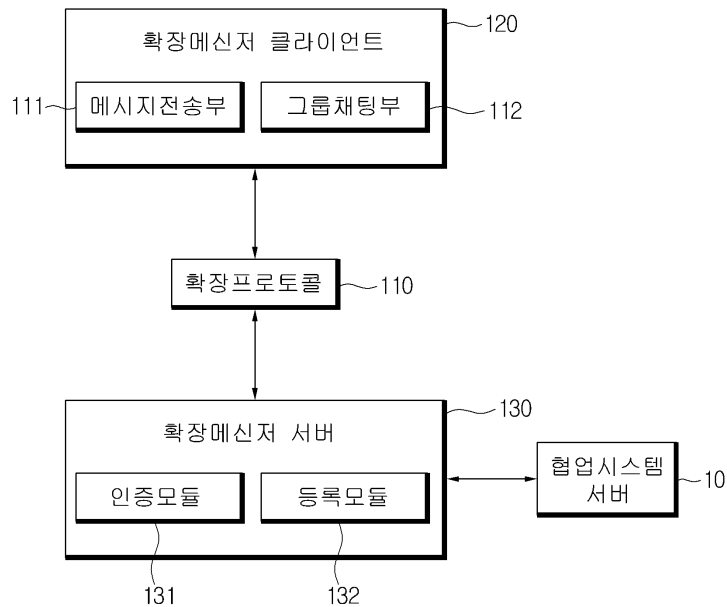
도면의 간단한 설명

- <172> 도 1은 본 발명의 실시예에 따른 웹데브 기반 협업시스템의 지원을 위한 메신저 시스템의 구성도,
- <173> 도 2는 도 1의 상세 구성도,
- <174> 도 3은 도 1의 다른 구성도,
- <175> 도 4는 종래의 JBother 메신저의 기본인터페이스 구성도,
- <176> 도 5는 CoSlide 협업시스템의 시스템 동작도,
- <177> 도 6은 CoSlide 협업시스템의 DAView 클라이언트 구성도,
- <178> 도 7은 CoSlide 협업시스템의 그룹 정보를 표시하는 CoJBother의 기본 인터페이스 구성도,
- <179> 도 8은 CoSlide 협업시스템 클라이언트와 연동된 CoJBother의 사용 예시도,
- <180> 도 9는 CoJBother를 이용한 그룹채팅 구성도,
- <181> 도 10은 그룹 구성원들이 CoJBother를 통하여 통신하는 예시도,
- <182> 도 11은 본 발명의 메신저 시스템의 도입 전후 협업시스템 작업 비교도이다.
- <183> <도면의 주요 부분에 대한 부호의 설명>

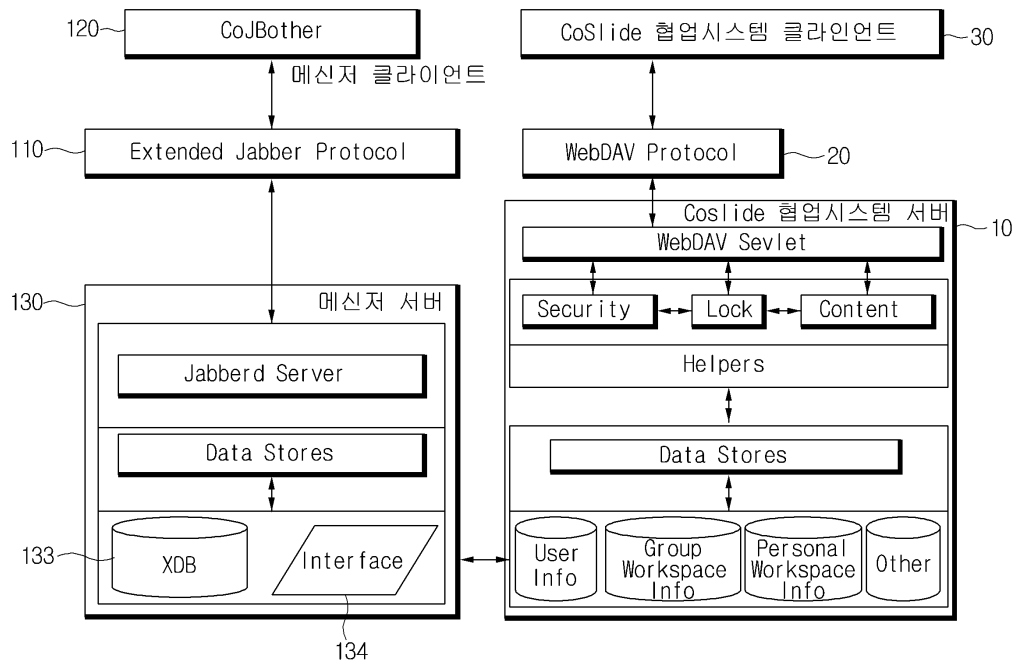
<184>	10...협업시스템 서버	20...웹데브프로토콜
<185>	30...협업시스템 클라이언트	
<186>	100...웹데브 기반 협업시스템의 지원을 위한 메신저 시스템	
<187>	110...확장프로토콜	120...확장메신저클라이언트
<188>	121...메시지전송부	122...그룹채팅부
<189>	130...확장메신저서버	131...인증모듈
<190>	132...등록모듈	133...XDB
<191>	134...인터페이스	135...JSM
<192>	136...Jabberd.xml	

도면

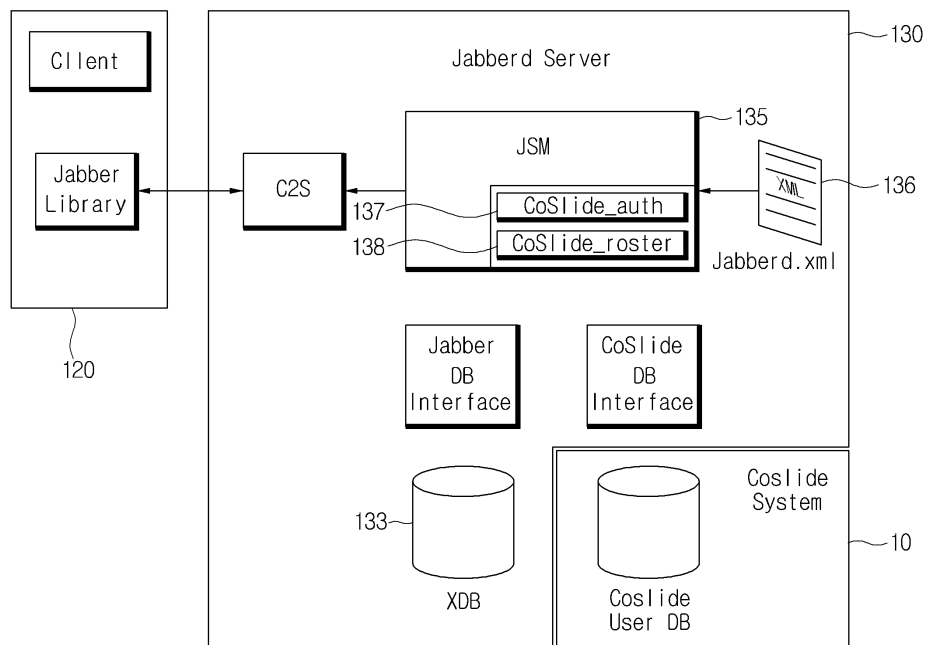
도면1



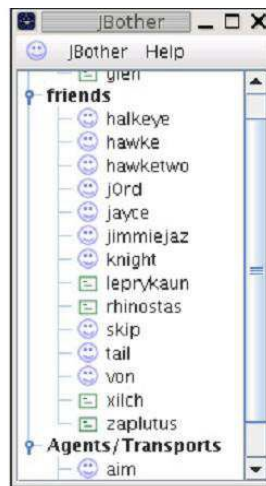
도면2



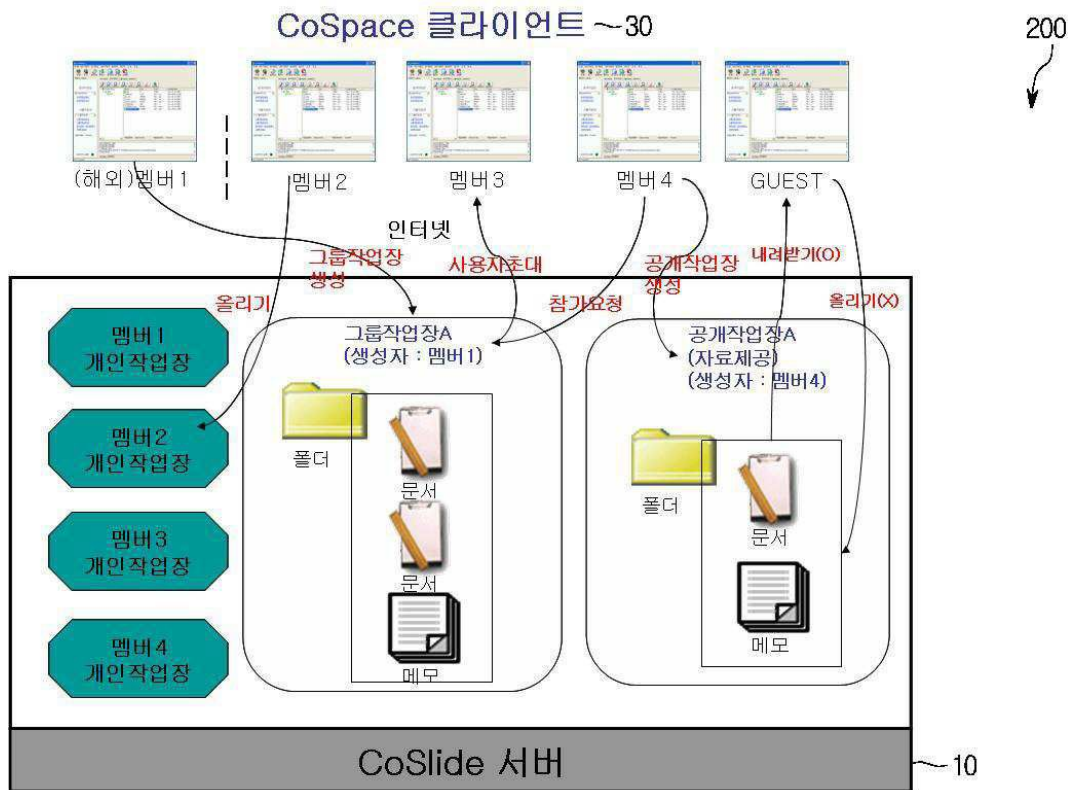
도면3



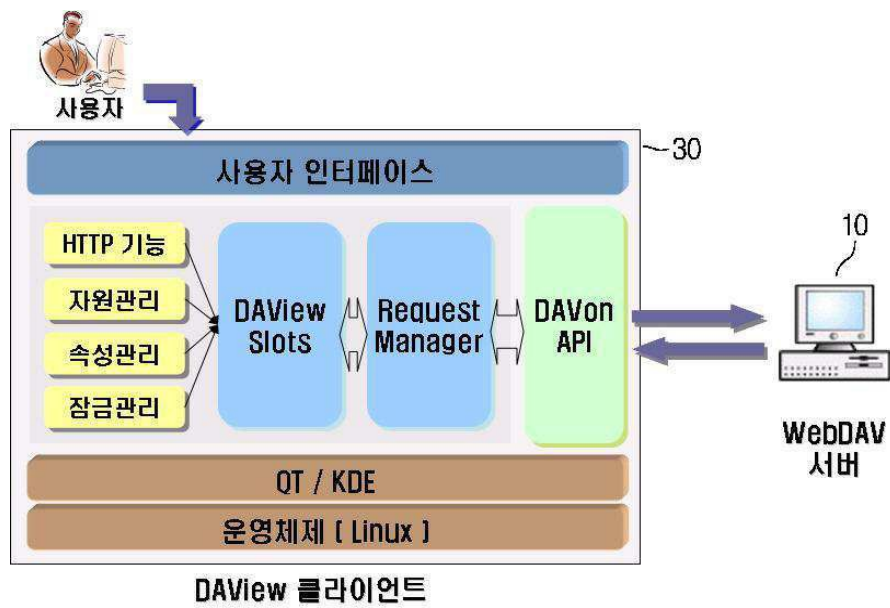
도면4



도면5



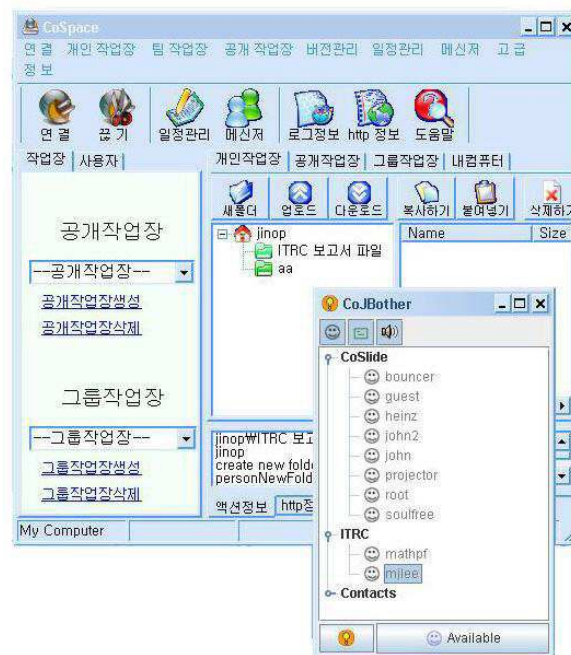
도면6



도면7



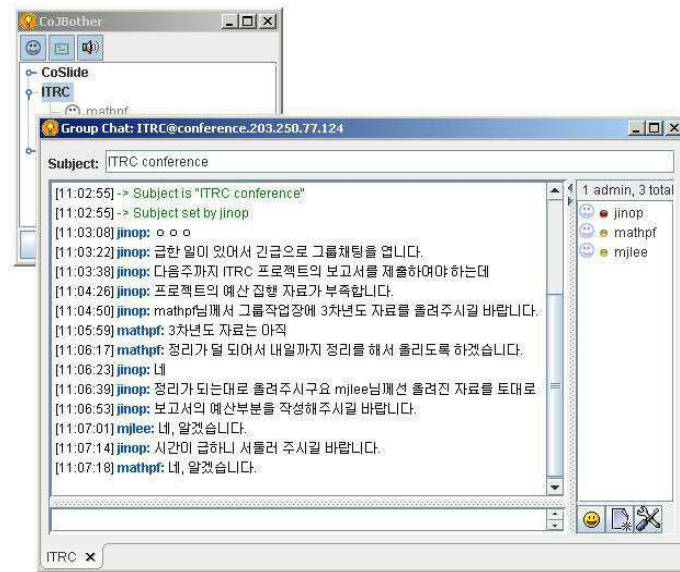
도면8



도면9



도면10



도면11

