

XML 메시지 교환을 이용한 이기종 P2P 데이터 공유시스템 설계

정현^{*}, 강병욱^{**}, 서영수^{***}

Design of XML message-based the heterogeneous P2P Data sharing System

Hun Jung*, Byung-Wook Kang**, Yeung-Su Seo***

요 약

JXTA는 중앙 서버를 통한 관리가 필요 없고 네트워크에 연결되어 있다면 어떤 장치든 다른 장치들과의 통신이 가능하며, 이기종 P2P시스템들과의 통신이 가능하다는 특징들이 가지고 있다. 이러한 특징을 최대한 활용하면서 서로 상이한 프로토콜과 API를 사용하여 만들어진 P2P시스템간의 상호연동을 가능하도록 하기 위한 데이터 교환 시스템을 설계하였으며, 설계된 시스템의 중요한 모듈은 생성된 요청과 요청에 대한 응답을 파싱하는데 이용하는 XML Parser Component(XPC), 적절한 요청과 응답을 생성해 내는데 사용하는 XML Making Component(XMC)의 구조를 설명한다.

Abstract

If the management through the center server is unnecessary and JXTA is connected to a network, a communication with any kind of devices or some device is possible. And features that a communication with heterogeneous P2P systems is possible have. The important module of the designed Data exchange system an interworking between the P2P system which it is made we used API and the each other different protocol the possible data for replacement Data exchange system were designed this kind of a feature was to the utmost utilized may account for the XML Parser Component (XPC) which we use it parses the or about a request and the generated request, and the structure of the XML Making Component (XMC) which we use in generating the inappropriate a request and response.

*Keywords : XML 메시지 기반(XML Message-based), 이기종(heterogeneous), 데이터 공유 시스템(Data Sharing System), 적스타(JXTA)

1. 서론

인터넷은 한 컴퓨터에서 다른 컴퓨터 혹은 컴퓨터들이 네트워크를 통해서 정보를 접근하고자 하는 필요에 따라 발전되었다. 초기의 인터넷은 단말기를 통해 데이터를 접근할 수 있도록 해 주는 소수의 컴퓨터들로 구성된 규모가 크지 않은 네트워크였다. 하지만 점차적으로 이 개념은 무한한 영역으로 발전하여 글로벌 네트워크 개념으로 발전했다.

기존 인터넷 기술과 서비스들은 클라이언트/서버 모델에서 기초하여 다수의 클라이언트가 서버에 집중하는 구조이다. 이러한 집중 구조는 네트워크 트래픽 증가로 인해 서버 부하 증대의 문제를 발생 시키게 되었다. 이에 P2P기술은 클라이언트/서버 모델의 집중 구조의 문제를 보완 및 대체할 수 있는 기술로 새롭게 조명 받고 있다[1].

초고속 네트워크에 연동된 고성능 컴퓨터 자원들을 효율적으로 활용하려는 P2P기술은 네트워크를 통한 기존의 단순한 정보공유뿐만 아니라, 자원의 공유를 통해 분산된 자원들의 통합을 가능하게 한다.

현재 P2P분야는 지속적으로 발전하고 있으면, 이를 이용한 응용분야들도 발표되고 있다.

대부분의 P2P시스템은 서로 다른 체제를 가지고 있고, 바탕이 되는 P2P 인프라의 부재로 인해 P2P시스템은 서로 호환되지 않는다. 이것은 각 P2P시스템이 자신만의 P2P 사용자 커뮤니티를 만들고 있으며, 이에 따라 모든 P2P시스템에서 공통으로 사용할 수도 있는 시스템을 만드는 데 있어 중복된 일을 하고 있다는 것을 의미한다.

즉 P2P시스템들이 각자의 프로토콜과 API를 사용함에 있어 하나의 네트워크 환경에서 이용하기가 곤란하기에 SUN의 JXTA 플랫폼이 대두되었다.

본 논문에서는 상이한 P2P시스템들이 상존하는 상황에서 이들 간 상호운용성을 제공하기 위해 JXTA를 이용하고, 시스템 상호간의 차이점을 극복하고 메시지 교환을 하기 위한 수단으로 XML 프로토콜을 이용한 SOAP을 사용한다.

즉, XML 메시지 교환을 이용한 이기종 P2P 시스템 간에 데이터 공유를 위한 시스템 설계에

목적을 둔다. 본 논문의 2장에서는 관련되는 연구들을 살펴보고 3장에서는 XML 메시지를 이용한 이기종 P2P 데이터공유 시스템을 설계한다. 마지막으로 4장에서는 결론 및 향후 연구방향을 제시한다.

II. 본론

2.1 P2P 서비스

기존의 네트워크 환경인 클라이언트/서버 모델에서는 서버와 클라이언트가 명확히 구분되어지고, 서버는 서비스를 제공하는 역할만 수행하며, 클라이언트는 서비스를 제공받는 역할을 수행한다. 즉, 클라이언트와 클라이언트 사이에는 직접적인 데이터 교환은 이루어지지 않으며 서버를 통해서만 이루어진다. 또한 데이터의 전달 역시 클라이언트의 요구와 서버의 응답방식으로 이루어진다.

이에 비해 P2P모델은 피어들 각각이 서로에게 서비스 제공자이면서 또한 서비스를 이용하는 사용자의 기능을 동시에 수행하는 구조이다. 즉 피어가 서버의 역할을 수행함과 동시에 클라이언트의 역할을 수행하는 것으로 데이터의 전달은 양방향이다. 데이터의 흐름을 비교해 볼 때 클라이언트/서버 모델은 주로 서버에서 클라이언트로의 데이터 흐름이 대부분을 차지하고 있으나, P2P모델의 경우 서로 간에 데이터를 교환하기 때문에 어느 한 피어로 트래픽이 집중되거나 방향성을 띄지 않고 피어들 사이에 자유로운 데이터 교환이 이루어지는 구조이다.

다음의 <그림1>은 클라이언트/서버 모델과 P2P모델에서의 데이터 흐름을 보여주는 그림이다.

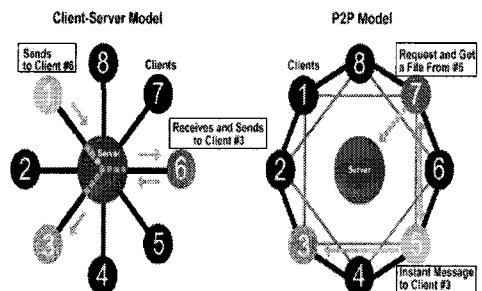


그림 1. C/S 모델과 P2P모델

현재 P2P시스템으로 사용되어지는 형태는 크

게 두 가지로 분류해 볼 수 있다. 하나는 메신저 형태의 시스템이며, 다른 하나는 파일 공유 형태의 시스템이다[2][5].

메신저 형태의 시스템은 피어들 간에 메시지를 주고받는 기본적인 기능부터 1:1 대화, 채팅방을 이용한 그룹대화 및 파일 전달의 기능에서부터 음성, 화상채팅 그리고 응용프로그램 공유에 이르기까지 다양한 기능을 제공한다.

파일 공유 형태의 시스템은 기본적으로 사용자가 원하는 파일을 검색하는 기능과 파일 다운로드 기능을 가진다.

각각의 P2P시스템들에서 사용되는 네트워크 구조 또한 크게 2가지 형태로 살펴볼 수 있다.

먼저 중앙 집중형 구조는 중앙의 서버가 피어들의 등록정보를 관리하고 있으면서, 어떤 피어가 P2P네트워크에 접속하게 되면 이 피어와 관련 있는 다른 피어들에게 그 피어의 접속을 알려줌으로써 피어들 간에 통신을 가능하게 하는 구조이다.

또 다른 구조로는 순수한 분산구조로서 그누텔라와 같은 파일공유 프로그램들에서 사용하는 방식으로 중앙의 관리서버 없이 모든 피어들이 서로 대등한 관계로 연결되는 구조이다.

이러한 분산구조에서는 중앙에 관리서버가 없기 때문에 트래픽이 한 곳에 집중하지 않고 P2P 네트워크가 견고하게 형성되는 장점은 있으나 서로 간에 데이터 전송이나 피어를 검색하는 기능에 있어서는 효율이 떨어진다.

대부분의 P2P시스템들은 이 형태를 그대로 따르거나 두 형태를 혼합한 하이브리드 형태를 취하고 있다[3].

2.2 JXTA 기술

JXTA는 JuXTApose(병행, 병치)의 약자로 전통적인 클라이언트/서버 방식과도 어울리면서 PC와 PC, PC와 가전제품, PC와 PDA등이 서로 교신할 수 있는 애플리케이션 개발환경을 제공하는 것이 목표이며 오픈소스 개발프로젝트 형식으로 진행되고 있다.

JXTA는 P2P시스템을 만들기 위해 정보처리 상호운용이 가능한 하나의 회로기판과 같은 것이다. JXTA는

피어 클라이언트들이 로그온, 로그오프, 재시동, 자료사제, 자료대체 등의 활동을 할 수 있으며 고정IP 및 유동 IP, 유선 및 무선환경, PC, PDA, 핸드폰, 기타 가전제품을 모두 소화할 수 있는 이상적인 개발 플랫폼이다[6].

이렇게 네트워크의 토폴로지가 지속적으로 변화할 수 있는 상황에서 피어들과 실제 네트워크 종점간에 비확정적인 맵핑을 하게 해주는 API환경을 제공하고 확장해 줌으로써 JXTA는 지속적인 발전을 하고 있다.

JXTA기술은 XML, 자바기술, UNIX 운영체계가 상당히 강력하고 유연하게 해주는 핵심개념과 같은 공개표준을 이용한다. 이와 같이 검증된 기술과 개념을 이용함으로써 개발자들에게 친숙하고 개발이 용이한 P2P시스템이 될 수 있을 것이다. JXTA는 P2P애플리케이션을 지원하는데 필요한 일련의 기본 기능들을 식별함으로써 P2P 시스템을 만들고 있고 상위계층 기능들과의 블록을 형성하여 기본기능들을 제공하고 있다.[7]

JXTA 플랫폼은 다음의 <그림2>와 같이 시스템 스택을 3계층으로 나눈다. 기본적인 수행에 필요한 피어설정과 통신기능의 코어 layer, 특정한 기능이 정의되어 있는 서비스 layer, 메일링이나 경매 스토리지 시스템과 같은 애플리케이션 layer의 계층으로 나뉘어져 있다[8].

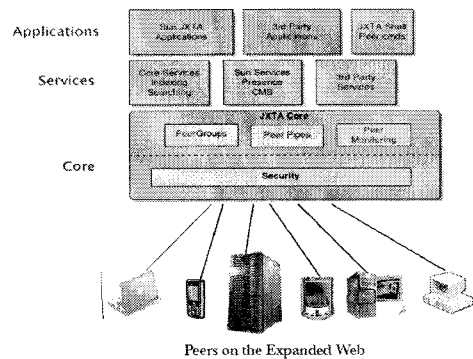


그림 2. JXTA 플랫폼

2.3 SOAP

최근 들어 IT업계에서 화두가 되고 있는 웹 서비스에 대해 W3C에서는 인터페이스와 바인딩이 XML에 의해 정의되고 기술되며 탐색이 가능하고 URI에 의해 식별이 되며, 인터넷 기반의 프로토콜을 통해 XML 기반의 메시지들을 이용하여 다른 소프트웨어 애플리케이션들과 직접적인 상호작용이 가능하도록 지원을 해주는 하나의 소

소프트웨어 애플리케이션으로 정의를 하고 있다.

이러한 웹 서비스의 기본적인 요소로는 인터넷을 통해 웹 서비스가 통신할 수 있도록 해주는 SOAP, 특정 조직이 자신의 웹 서비스를 온라인 디렉토리에 등록, 광고하거나 외부에서 웹 서비스를 검색하는데 사용되는 UDDI(Universal Description, Discovery and Integration), 웹 서비스를 정의하는 언어로서 특정 조직이 웹 서비스를 기술할 때 사용되는 WSDL(Web Service Description Language) 등이 있다.[2]

SOAP은 한 컴퓨터에서 다른 컴퓨터로 정보를 전달할 때 그 정보를 XML로 표현하는 표준적인 방법을 제공해 준다. 즉, 정보를 전달할 때 XML로 마크업 한 다음, 보통 HTTP로 전송되는 프로토콜을 정의하고 있다.

SOAP은 CORBA와 같이 ORB(Object Request Broker)의 통신 프로토콜로 XML을 사용 하는 것으로 객체에 대해 원격지에서 정해진 XML 규약에 맞추어 보내면 그 컴포넌트가 지시에 맞추어 동작하게 된다. 이와 같은 일은 원래 분산객체 미들웨어인 CORBA나 DCOM에서도 가능했던 것이지만 SOAP은 XML을 기반으로 하여 분산환경에서 분산객체를 위하여 정보를 교환하기 위한 텍스트 기반 프로토콜이다. 또한 SOAP은 메시지에 있는 사항과 그 사항을 처리하는 방법을 설명하기 위해 프레임워크를 정의한 봉투, 응용 프로그램에서 정의한 데이터 형식의 인스턴스를 나타내는 일련의 인코딩 규칙, 원격 프로시저 호출 및 응답을 나타내는 규칙으로 구성된 XML 기반의 프로토콜이다[11].

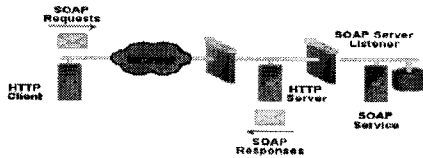


그림 3. SOAP 서비스 프로세스

SOAP 메시지는 XML을 이용하여 인코딩 되어 있고 SOAP 규격은 추상화된 다른 계층으로 나타낸다. 클라이언트로부터 파생된 SOAP 메시지는 SOAP Requests로 서버에 요구하고, 서버가 접수를 받고 요청을 처리한 후에, SOAP Response는 클라이언트로 반환된다. 이 프로세스는 <그림3>과 같다.

III. XML 메시지 교환을 이용한 이기종 P2P 데이터공유 시스템 설계

3.1 시스템 개요

본 논문에서 설계하게 될 P2P시스템은 조회/록업서버를 가진 P2P 데이터 공유 시스템을 설계한다. 서버의 역할은 접속된 피어들의 목록을 각각의 이용 가능한 자원들과 함께 제공하는 기능을 한다. 다음의 <표 1>은 본 논문에서 설계하고자 하는 시스템의 구성요소 및 역할을 나타낸다.

표 1. 시스템 구성요소 및 역할

구성요소	역할
리스너	피어의 접속 요청 처리
조회/록업서버	피어정보 관리
브라우저	UI를 통해 요청/응답에 대한 결과 표시

<표 1>에 정의한 구성요소들의 상호관계를 살펴보면 다음과 같다.

(1) 리스너-서버 관계

서버는 모든 리스너들에 대하여 이름, IP주소, 공유된 자원에 대한 정보를 데이터베이스에 저장한다. 리스너는 서버에 로그인해서 공유된 자원을 선언한 후, 파일과 폴더의 이름을 위치와 함께 전송한다. 서버가 제공하는 리스너들의 목록은 로그아웃상태의 리스너들은 제외하며 현재 온라인 상태인 리스너들만을 보여준다.

(2) 리스너-브라우저 관계

브라우저가 모든 요청을 리스너에게 보내고, 처리된 질이나 메시지의 형태로 응답을 수신하는 요청/응답의 관계이다. 리스너는 리스너 수준의 모든 요청을 처리하고 브라우저는 처리된 결과를 요청 피어에게 전달한다.

(3) 브라우저-서버 관계

브라우저는 리스너들의 목록을 서버로부터 가지고 와서, 어느 리스너가 자원을 공유하고 있는지 확인한다. 브라우저는 서버로부터 리스너들의 목록을

얻는 것과 함께 서버 수준의 검색요청을 전달할 수 있다

3.2 시스템 흐름도

시스템의 구성요소와 구성요소와의 관계를 토대로 본 논문에서 설계하고자 하는 시스템의 흐름을 살펴보면 아래의 <그림4>와 같다.

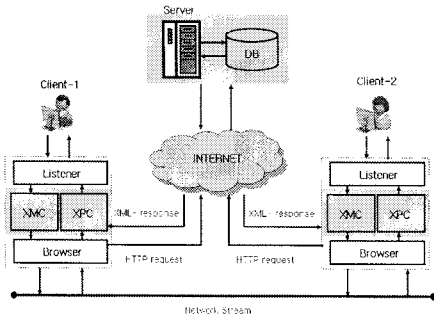


그림 4. 시스템 흐름도

본 논문에서 설계하고자 하는 P2P시스템의 구성요소는 리스너, 브라우저, 서버 3개의 모듈에 기반을 둔다.

위의 <그림4>와 같이 원격지에서 실행되는 두 개의 피어(리스너와 브라우저)가 서로 자원을 공유하며, 자원 공유를 시도하기 위하여 두 개의 피어는 다음의 <표 2>와 같이 두 가지 동작을 수행한다.

표 2. 피어 동작

동작	기능
요청	-피어사이에 계속적으로 수행할 상호작용을 위해 통신을 수립하는 단계
응답	-요청에 대한 적절한 응답을 해주는 단계. -응답의 경우 미리 정의된 메시지에 대한 반환 또는 요청에 대한 단순 응답.

리스너로 동작하는 피어는 다른 피어들에 의해 만들어진 요청에 대하여 응답하는 역할을 하는 반면, 브라우저는 모든 요청을 생성하는 피어의 역할을 한다.

서버는 현재 로그인한 모든 피어들의 정보(IP주소, 로그인명, 공유자원에 관한 정보 등)를 보유하고 있는 데이터베이스를 관리 한다.

P2P시스템으로부터 전달되어진 신호를 XML을 이용하여 처리하게 되며, XML로 구성된 각종 메시지를 처리하는 다음의 <표 3>과 같이 두 개의 모듈이 필요하다.

표 3. XML 처리모듈

구성	기능
XML 파서모듈	생성된 요청과 요청에 대한 응답을 파싱.
XML 작성모듈	적절한 요청과 응답을 생성.

시스템의 시작은, 리스너가 자신의 로그인 이름, IP주소, 공유된 자원들에 대한 정보로 구성된 요청을 서버에 전송함으로써 시작된다. 리스너는 서버에 접속한 리스너들의 목록에 자신의 항목을 추가하며, 서버는 리스너로부터 넘겨받은 사용자 정보의 인증작업을 시작한다. 사용자 인증의 과정은 XML 형태로 적절한 응답을 반환하며, 이 응답은 XML 파서모듈에 넘겨져서 파싱되어 최종적으로 적절한 메시지가 리스너에게 전달된다.

이제 리스너 역할을 하는 피어가 서버에 성공적으로 로그인하였다면 이 리스너는 다른 피어의 요청에 응답할 준비가 되었다고 판단한다.

또한 콘텐츠를 검색하고 있는 브라우저는 서버에 요청신호를 보냄으로서 서버에 접근한다.

서버는 브라우저의 요청에 대하여 서버가 자신에게 접속한 모든 피어들의 목록을 XML형태로 전달한다. 서버는 접속된 리스너 이름, 공유된 자원, 리소스의 IP주소 등의 구성 목록을 브라우저에게 제공한다.

브라우저는 서버에 접근한 후 원하는 리스너와 통신을 수립하는 단계를 수행한다.

먼저 브라우저와 리스너 사이에 통신이 수립되면 두 피어들 사이에는 자료 교환이 가능하다. 브라우저는 XML 형식으로 데이터를 액세스하기 위하여 네트워크 스트림을 오픈한다, 피어들 사이에서 일어나는 프로세스는 네트워크 스트림을 통해 이루어진다.

브라우저는 특정 데이터를 원격 리스너로부터 다운로드하는 방법으로 원하는 콘텐츠를 획득할 수 있으며, 브라우저가 데이터를 다운로드 하려면 XML작성모듈

(XMC)을 사용하여 요청 XML을 생성하고, 리스너의 네트워크 스트림을 통하여 전달한다.

리스너는 자신에게 전달되어진 요청XML을 파싱하고 어떤 파일이 브라우저에게 전송할 것인지 파악한다. 이 정보는 브라우저의 요청에 대한 XML응답형태 또는 브라우저가 다운로드 요청한 파일일 수도 있다. 이 정보를 브라우저의 네트워크 스트림을 통하여 브라우저에게 전달한다.

최종적으로 브라우저는 리스너에 의해 업로드 된 파일을 읽어 들이고 이때 읽어 들인 파일이 브라우저의 요청에 대한 응답인지 다운로드 요청을 한 파일인지 확인하여 적절한 메시지를 화면에 표시한다.

브라우저를 통하여 파일을 업로드 하는 경우는 브라우저는 다른 피어들이 공유한 메모리 영역을 선택하고 자신이 업로드 할 파일을 선택하게 된다. 그리고 리스너에게 보낼 요청XML을 생성한다. 리스너가 요청XML을 받으면 작성되어야 할 파일에 대한 권한이 있는지 확인하게 되며, 권한이 있는 경우 브라우저가 쓰고 있는 파일을 읽어 들이고 동시에 데이터를 파일에 쓴다. 만약 권한이 없다면 업로드 거부 응답을 생성하여 브라우저에게 반환한다.

3.3 구성요소 모듈사이의 통신

(1) 리스너와 서버 통신

리스너 애플리케이션을 시작하면 로그인 창이 표시되며, 사용자는 접속 아이디를 입력한다. 리스너는 서버에 자신의 정보를 등록하기 위하여 공유된 자원들의 목록과 로그인명, 자신의 IP주소를 전달하게 된다. 다음의 <그림 5>는 리스너와 서버사이의 통신을 나타내고 있다.

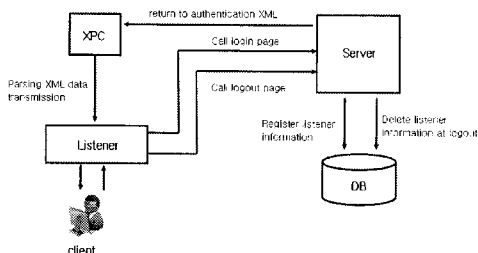


그림 5. 리스너와 서버 통신

서버는 넘겨받은 모든 인자를 확인한 후 해당 리스너의 항목을 작성한다. 그리고 요청에 대한 응답으로 리스너에게 인증 XML 메시지를 전달한다.

(2) 브라우저와 서버 통신

리스너가 성공적으로 서버에 연결된 후, 브라우저는 서버와 통신을 하게 된다. 다음의 <그림 6>은 브라우저와 서버사이의 통신을 나타낸다.

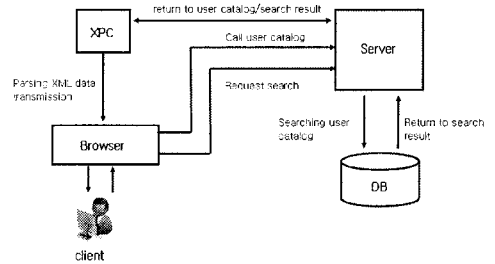


그림 6. 브라우저와 서버 통신

브라우저는 현재 실행중인 모든 리스너 목록을 얻어서 자신의 창에 표시한다.

서버가 브라우저로부터 요청을 받으면 현재 실행하고 있는 모든 리스너들의 목록을 데이터베이스에서 검색하여 브라우저에게 XML메시지 형태로 응답을 전달한다. 브라우저는 이 XML응답메시지를 파싱한 결과를 사용자에게 보여 준다.

브라우저가 반환된 사용자 목록에서 특정 사용자를 선택하며 브라우저는 선택된 리스너와 소켓연결을 수립한다. 즉 리스너와 브라우저 사이에 IP주소를 사용하여 직접 통신이 수립된다. 이후 수행될 브라우저와 리스너 사이의 모든 통신은 서버를 거치지 않고 소켓을 통해 직접 이루어진다.

(3) 리스너와 브라우저 통신

애플리케이션의 통신 기능은 소켓에 기반을 두고 있는데 소켓은 서로 통신을 하기 위해 시스템의 네트워크 스트림을 이용한다. 리스너와 브라우저는 각각의 네트워크 스트림에 데이터를 읽고 쓰는 작업을 수행한다.

브라우저는 리스너의 네트워크 스트림에 데이터를 쓰는 방식으로 요청을 XML의 형태로 전송하고, 리스너는

네트워크 스트림으로부터 데이터를 읽어 들이는 방법으로 요청을 처리한다. 다음의 <그림 7>은 리스너와 브라우저사이의 요청/응답을 나타내고 있다.

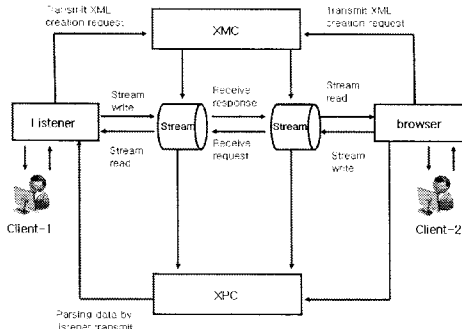


그림 7. 리스너와 브라우저 통신

3.4 이기종 P2P 플랫폼 구조

다음의 <그림 8>은 본 논문에서 제안한 이기종 P2P 시스템에 대한 소프트웨어 플랫폼을 나타낸다.

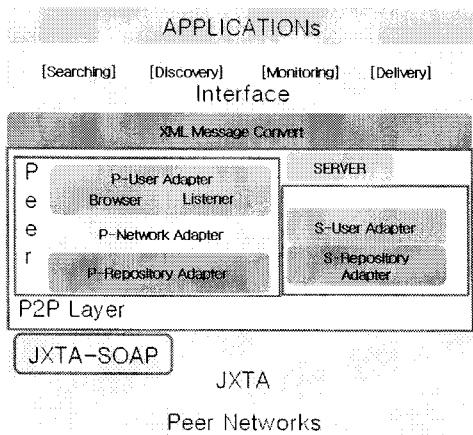


그림 8. 소프트웨어 플랫폼

본 논문에서 설계된 소프트웨어 플랫폼은 실제 다양한 네트워크 환경에서 통신을 담당하는 Peer 네트워크 계층과 Peer 네트워크 계층위에서 P2P 네트워크 환경을 꾸미기 위한 JXTA 프레임워크가 존재한다.

이러한 계층위에 P2P시스템에서 사용하게 될 P2P Layer가 위치하게 된다.

P2P Layer는 서로 다른 P2P 네트워크 환경 상에서

각 피어들간의 P2P 네트워크 환경설정을 통한 데이터 공유를 담당하게 될 계층으로 이 계층에는 서로 다른 P2P 네트워크 환경에 접속하여 실질적으로 다운로드 할 수 있는 네트워크 스트림을 형성하는 역할을 담당하는 Network Adapter와 각 피어들이 자신의 정보를 저장하기 위하여 조회 및 록업 서버에 접속하고 피어들의 상태 정보를 확인하고 원하는 자료 검색 등의 역할을 담당하는 User Adapter 그리고 각 피어들의 상태 정보를 XML 포맷으로 저장하여 접속한 각각의 피어들에게 제공하고 피어들의 상태정보가 변경되어질 경우 피어들의 정보의 저장 및 접속한 다른 피어들에게 알려주기 위하여 수정된 피어정보와 기존 피어정보를 비교하는 Repository Adapter로 구성되어진다.

P2P Layer 계층위에는 피어와 피어, 피어와 조회/록업 서버 사이의 요청과 응답 메시지를 XML 형식으로 변환시켜주는 XML Message Convert 계층이 위치한다.

(1) 피어

(가) P-User Adapter

각 피어가 서버로 접속할 수 있는 통로 역할을 수행하며, 리스너와 브라우저로 구성된다.

(나) 리스너

P-User Adapter안에 구성되어져 있으며, P2P 네트워크 환경에 접속하는 역할과 특정 소켓상에서 브라우저의 접속을 대기하는 역할을 수행한다. 즉, 브라우저의 요청을 리스너에게 보내면 처리된 질의나 메시지의 형태로 응답을 수신한다.

(다) 브라우저

브라우저는 “Share Manager”와 “Browser Manager”와 같이 사용자와 컴퓨터 사이에 인터페이스 역할을 할 수 있는 GUI 환경이 있다.

(라) P-Network Adapter

브라우저의 검색요청이 리스너에게 전달되어졌을 때 실제 네트워크 스트림을 형성하여 공유 데이터를 다운로드 할 수 있는 부분이다.

(마) P-Repository Adapter

다른 피어로부터 전송되어진 각 피어들의 정보와 자신의 정보가 저장되어지는 부분이며 피어들의 갱신정보가 도착하였을 때 기존 피어정보와 비교하고 저장하는 역할을 담당한다.

(2) 조회/록업 서버

(가) 서버

모든 등록된 피어(P-User Adapter)들을 자원들에 관한 상세한 정보와 함께 “S-Repository Adapter”를 통하여 DB에 저장하고 있다. 이 자원들은 “P-User Adapter”들이 다른 피어들과 공유하기 위해 제공하는 것이다. 서버에서는 “P-User Adapter”가 실행되는 컴퓨터상의 파일을 서버로 전송하는 것이 아니라, “P-User Adapter”를 통해 접속되어진 피어들의 목록을 제공한 후 접속한 피어들의 정보를 관리하여 주는 역할만 수행한다.

(나) S-User Adapter

피어가 공유파일을 검색하고자 하는 경우 서버로 접속하여 또 다른 피어들의 정보를 받아오기 위한 통로 역할을 담당한다.

(다) S-Repository Adapter

피어들의 정보를 데이터베이스에 저장 관리한다.

(3) XML Message Convert

피어와 피어사이, 피어와 조회/록업서버 사이의 요청과 응답 XML 메시지를 생성하고 파싱하는 역할을 수행한다.

IV. 결론 및 향후 과제

JXTA 프로젝트는 중앙 서버를 통한 관리가 필요 없고 네트워크에 연결되었다면 모든 장치에서 다른 장치들과의 통신이 가능하며 타 P2P시스템과의 통신이 가능한 장점을 가지고 있다. 이러한 장점에 조회기능을 가진 서버를 중심에 두어 피어들이 자신이 필요한 정보를 얻기 위해 각각의 피어를 일일이 방문할 필요성이 없게 하였으며, 실제적인 통신은 피어들 상호간에 이루어 질수 있도록 설계하였다. 또한 서로 상이한 프로토콜과 API를 사용하는 P2P시스템들 간의 통신을 위하여 데이터 교환 시스템을 설계하였다. 본 논문에서 제안한 데이터 교환 시스템을 통하여 상호 운용성이 유지되어진다면 P2P 통신망의 응용범위를 확장할 수 있을 것이다.

향후에는 설계된 데이터 공유 시스템을 구현 할 것이며, P2P 시스템간의 병목으로 성능 저하가 발생할 수 있으므로 병목형상방지에 대한 연구가 필요할 것이다.

참 고 문 헌

- [1] 송은지, “효율적인 P2P 시스템 개발에 관한 연구”, 남서울대학교 논문지 Vol.11 No.2, 2005.
- [2] 권태숙, 이일수, 이승룡, “XML기반 전역 Peer-to-Peer 엔진 설계 및 구현”, 한국통신학회 Vol.29 No.1B, 2004.
- [3] 양광민, 주형렬, “P2P 기반 자료공유시스템의 응용”, 경영학논문집, Vol.29 No.1, 2002.
- [4] 이정환, 조인준, “Peer-to-Peer에서 채팅 프로그램 설계 및 구현(Design and Implementations of Chatting program in Peer-to-Peer)”, 한국인터넷정보학회 추계학술발표대회 논문집 제5권 제2호, 2004.
- [5] 한진수, 김태형, “Jxta를 이용한 타 P2P 시스템과의 연결”, 한국정보과학회 학술발표논문집 제31권. 1호, 2004.
- [6] Wilson, Brendon J, “JXTA : 차세대 P2P 플랫폼”, 생능출판사, 2003.
- [7] Gradecki, Joe, Wiley Pub, “Mastering JXTA : building Java peer-to-peer applications”, 2002
- [8] Brookshier, Dan, Sams, “JXTA : Java P2P programming”, 2002.
- [9] Qinghai Miao, Zhixue Wang, Fei-Yue Wang, Shuming Tang, Feng He, “An Implementation of Artificial Transporation System based of JXTA”, ICVES 2006. IEEE International Conference on 13-15 Dec. 2006.
- [10] Nakamura, M., Ma, J., Chiba, K., Shizuka, M., Miyoshi, Y., “Design and implementation of a P2P shared Web browser using Jxta”, Advanced Information Networking and Applications, 17th International Conference on 27-29 March, 2003.
- [11] Alex Toussaint and Wrox Author Team. “Professional Java Web Services”, Wrox Press, 2002.
- [12] 이정환, 조인준, “Peer-to-Peer에서 채팅 프로그램 설계 및 구현”, 한국인터넷정보학회 추계학술발표대회 논문집 제5권 제2호 2004.