

JXTA 플랫폼 기반 분산 정보 검색 시스템 설계 및 구현에 관한 연구

이승하*, 방세중*, 이필우**, 김양우*

*동국대학교 정보통신공학과

**한국과학기술정보연구원(KISTI) CI 미들웨어연구팀

e-mail: lesh915@dongguk.edu, neovega@dongguk.edu, pwlee@kisti.re.kr, ywkim@dongguk.edu

A Study on Design and Implementation Distributed IR(Information-Retrieval) System Based on JXTA Platform

Seungha Lee*, Sechung Pang*, Pilwoo Lee**, Yangwoo Kim*

*Dept of Information and Communication, Dongguk University

**CI Middlewqre Research Team, KISTI

요 약

일반적인 정보검색 시스템은 중앙 집중식의 서버/클라이언트 방식을 사용한다. 이 방식은 서버 집중 방식으로 시스템의 부하가 가중될 경우 추가적인 자원 확보에 어려움을 가진다. P2P(Peer-to-Peer) 기술은 이러한 중앙 서버의 문제점을 해결하기 위해 제안된 것이다. JXTA 플랫폼은 P2P 서비스를 제공하기 위한 오픈 소스 프로젝트로서 본 논문은 정보검색 시스템의 부하가 늘어날 경우 유연한 자원 확보를 위해 JXTA 플랫폼 기반의 JXIR(Jxta Information Retrieval) 시스템을 설계하고 구현하였다.

1. 서론

기존 클라이언트/서버 모델은 중앙 서버에서 효율적인 관리를 위해 시스템의 모든 기능을 포함하고 있다. 대표적인 클라이언트/서버 구조는 인터넷에서 살펴 볼 수 있다. 웹 서버와 브라우저로 대표되는 웹 구조에서 사용자들은 브라우저 만을 통해 여러 가지 서비스를 제공받을 수 있어 보편화 되었다. 그 중 대표적인 서비스가 웹 검색이다. 웹 검색의 경우 검색 사이트에 접속하면 사용자들은 간단히 자신이 알고자 하는 정보를 살펴 볼 수 있다. 하지만 대부분의 웹 검색 시스템들은 서버 집중형의 검색 엔진을 사용하고 있다. 이와 같은 시스템은 계속적으로 증가하는 대규모의 데이터들을 실시간으로 처리할 수 없다. 그렇기에 일정한 시간동안 주기적으로 업데이트하는 데 많은 컴퓨터 자원과 시간을 필요로 한다. 따라서 특정 시간 동안 보다 많은 데이터를 빠르게 처리하기 위해서 기존에 제안되었던 방법으로 병렬 시스템이나 클러스터링 구조의 시스템이 제시되었다. 하지만 위와 같은 시스템은 고정된 컴퓨터 자원을 사용한다. 따라서 시스템을 확장하는데 컴퓨터 자원이 필요한 경우 자원 확보와 시스템 연동에 있어 여러 문제점을 가진다. 이것은 시스템의 유연한 확장을 어렵게 한다.

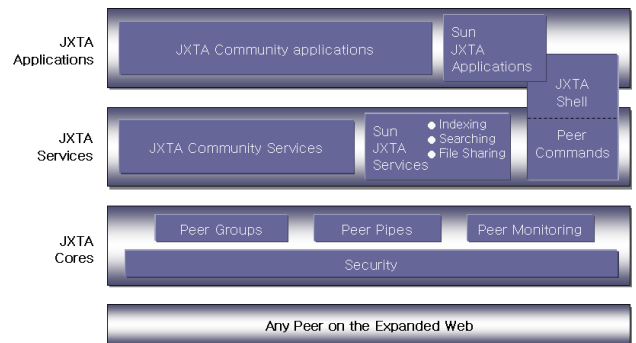
서버 집중형 시스템의 문제를 해결하기 위해 제안된 것이 P2P(Peer-to-Peer) 기술이다. 본 논문은 정보검색 시스템의 시스템 부하를 줄이고 정보검색 서비스의 실시간 처리를 위한 방안으로 P2P 기술을 이용하여 분산 정보 검색

시스템을 개발하였다. 이를 위해 SUN사에서 오픈 소스 프로젝트로 개발하고 있는 P2P 개발 플랫폼 JXTA[1][2]를 이용하였다. 또한 검색 처리를 위해서 오픈 검색 라이브러리인 루센(Lucene)[4]을 사용했다.

본 논문은 다음과 같이 구성된다. 2절에서 본 논문의 시스템의 기반 기술인 JXTA 플랫폼에 관해서 설명하고 3절에서 루센에 대한 간단한 소개와 4절에서 본 논문에서 실제로 구현한 시스템에 대한 설계 및 구현 결과에 대해서 살펴보고 마지막으로 5절에서 결론을 맺는다.

2. JXTA™ Platform

2.1 JXTA™ Platform 구성

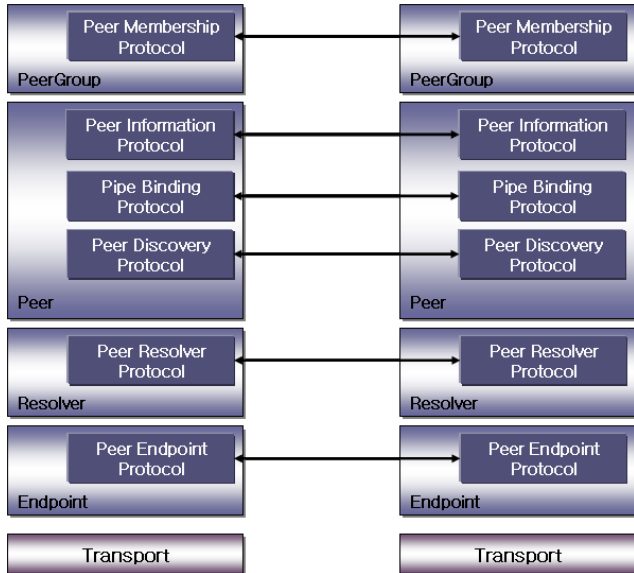


(그림 1) JXTA 플랫폼 구조

JXTA는 SUN Microsystems사에서 오픈소스 프로젝트

의 일환으로 개발을 시작한 P2P(Peer-to-Peer) 플랫폼(그림 1)으로 P2P를 비롯한 분산 컴퓨팅을 위한 기반 서비스를 제공하며 P2P 컴퓨팅 환경을 위한 공개 네트워크 컴퓨팅 플랫폼[3]이다.

2.2 JXTA™ 네트워크



(그림 2) JXTA 네트워크 프로토콜

가. Peer Discovery Protocol

피어로 하여금 네트워크상의 피어 서비스를 탐색할 수 있도록 한다. 다른 피어에게 광고(Advertisement)를 요청하고, 자신의 광고에 대한 다른 피어의 요청에 응답하기 위한 프로토콜을 정의한다.

나. Peer Resolver Protocol

피어로 하여금 일반형의 요청을 송신하고 처리하도록 한다. 단순 피어가 자신을 대신하여 다른 피어에게 메시지를 전파하도록 사용한다. 즉, JXTA 네트워크상에서 말단의 피어가 메시지 전파할 수 있도록 하는 프로토콜이다.

다. Rendezvous Protocol

피어 간의 메시지 전파에 대한 상세 내역을 처리하도록 한다. 다른 피어에 위치한 핸들러에게 일반형의 질의를 보내고, 질의에 대한 일반형의 응답을 처리하는 프로토콜을 정의한다.

라. Peer Information Protocol

피어에게 네트워크상의 다른 피어의 상태 정보를 입수하는 방법을 제공한다. 세부적인 모니터링 기능이 개발 필요에 따라 모니터링을 구현해야 한다. 모니터링 기능이 필요한 이유는 서비스를 효율적으로 만들 수 있고 상업적인 용도에서 서비스의 이용을 결정하고 사용자에게 사용료를 받을 때 사용할 수 있다. 실제 상

태 정보는 피어의 업타임, 트래픽 양 등을 제한적으로 제공하고 있다.

마. Pipe Binding Protocol

피어 엔드포인트(Peer Endpoint)에 대하여 가상의 통신 채널을 연동하는 메커니즘을 제공한다. 가상의 통신채널을 파이프(Pipe)라고 한다. 파이프를 엔드포인트에 연동하는 프로세스라고 볼 수도 있다. 파이프란 원격 피어와 데이터를 송신하거나 수신하는데 사용되는 JXTA의 전송 개념이다. 서비스는 다른 피어와 통신하기 위해 리졸버 핸들러를 사용하거나 파이프를 이용한다. 파이프는 실제 사용되기 전에 피어 엔드포인트에 연동되어야 한다.

바. Endpoint Routing Protocol

출발지 피어에서 목적지 피어에 대한 메시지 라우팅이 가능하도록 사용되는 메시지의 집합을 제공한다. 탐색된 경로 정보를 이용하여 원격 피어에게 메시지를 발송하는 메커니즘을 제공한다. 파이프는 편리한 프로그래밍 모델을 제공하기 위해 엔드포인트 상에서 구축된 추상적인 개념으로 엔드포인트가 네트워크상에서 실제적인 정보 교환을 수행하는 실체이다.

3. Lucene 검색 엔진

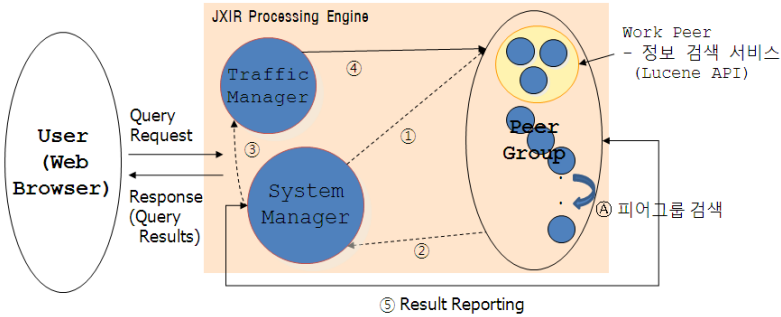
루씬(Lucene)은 확장 가능한 고성능 정보검색 (IR, Information Retrieval) 라이브러리이다. 루씬은 사용자의 소프트웨어 프로그램에 색인과 검색 기능을 간단하게 추가할 수 있도록 지원한다. 루씬은 오랜 기간 많은 사람들이 사용하여 그 능력과 성능을 인정받았고, 자바로 만들어진 오픈소스 소프트웨어이다.

루씬은 사용자의 프로그램에 텍스트 색인과 검색 기능을 추가할 수 있도록 지원한다. 그러나 루씬을 이용해 임의의 바이너리 파일을 직접 색인하고 검색할 수 없으며, 모두 문자열 형태로 변경된 이후에야 루씬으로 색인하고 검색할 수 있다[5]. 예를 들어 루씬의 입장에서, 마이크로소프트 워드의 파일인 DOC 파일은 바이너리 파일이다. 따라서 DOC 파일은 루씬에 직접 추가해서 색인하거나 검색할 수 없으며 여러 방법을 통해 일반 텍스트로 변경된 이후에 색인하고 검색할 수 있다. 즉 색인하고 검색하고자 하는 자료는 텍스트 형태로 루씬에게 제공되어야 한다. 이런 규칙이 단점일 수도 있지만, 텍스트로 변환할 수 있는 모든 파일은 루씬이 색인하고 검색할 수 있다는 장점을 갖는다. 다시 말해 웹 사이트의 웹 페이지, 로컬 디스크에 저장된 문서, 일반 텍스트 파일, 마이크로소프트 워드 문서, HTML파일, PDF 파일 등 일반 텍스트를 추출할 수 있는 모든 문서를 처리할 수 있다.

특히 텍스트라면 어떤 자료든 색인해서 검색할 수 있기 때문에 기업의 데이터베이스에 저장돼있는 대량의 자료를 빠르게 검색하도록 해주는 응용 프로그램에도 루씬을 많이 사용한다. 또한 루씬은 색인하고 검색하는데 있어 텍스

트를 다루기 때문에 이메일 메시지, 디스크에 쌓인 메일링 리스트 메시지, 메신저 대화 내용 등도 무난히 처리할 수 있다.

과들을 받게 된다.



(그림 3) JXIR 모델 구조도

4. JXIR(Jxta Information Retrieval) 시스템 설계 및 구현

4.1 JXIR 시스템 설계

JXIR(Jxta Information Retrieval)은 우리가 본 논문에서 설계하고 구현한 'JXTA 기반의 분산 정보 검색 시스템'의 이름으로 (그림 3)과 같은 형태로 살펴 볼 수 있다. 사용자가 검색하고자 하는 쿼리 트래픽을 시스템 관리자(System Manager)에 전달하면 트래픽 관리자(Traffic Manager)가 이 작업들을 피어 그룹(Peer Group)으로 분배, 처리하는 구조이다. 피어 그룹 내의 각 피어(Peer)의 정보는 시스템 관리자가 관리하며 이를 트래픽 관리자에게 통보하여 트래픽 작업의 분산을 용이하게 한다. 시스템 관리자는 항상 시스템에 속한 피어들의 상태를 감시하고 가용한 자원들을 이용할 수 있게 한다. JXIR 모델은 (그림 3)에 보이는 것과 같이 가운데 사각형 부분의 트래픽 관리자, 시스템 관리자, 그리고 피어 그룹 간 작업 처리 엔진 기반 하에 검색 작업을 처리하도록 한다.

4.2 JXIR 시스템 구현

가. 시스템 관리자 및 트래픽 관리자 구성

시스템 관리자, 트래픽 관리자는 피어 그룹을 생성하고 사용자가 요청한 쿼리를 작업 피어에게 분산 처리한다. ① 피어의 그룹을 생성하는 것은 시스템 관리자의 역할이다. ② 시스템 관리자는 처음 피어 그룹을 생성하고 그룹에 참여한 상태에서 작업 피어들이 자신들의 정보를 보내줄 것을 대기하고 있다가 피어가 자신의 정보를 보내면 ③ 입력받은 피어의 정보를 우선순위 테이블(Priority Table)로 작성하여 트래픽 관리자에게 전송한다.

④ 트래픽 관리자는 시스템 관리자로부터 받은 우선순위 테이블 정보를 바탕으로 작업 피어에게 요청 받은 작업 쿼리들을 전송한다. ⑤ 피어들 각각은 요청 받은 쿼리들을 처리하여 이 결과를 시스템 관리자에게 전송한다. 시스템 관리자는 이 정보를 사용자에게 보내기 위해 결과 데이터들을 정리한다. 최종적으로 사용자는 이 처리된 결

나. 작업 피어(Work Peer)의 구성

작업 피어는 처음 시스템 관리자가 동작을 시작했을 경우 ① 그룹 정보를 검색해서 그룹 정보를 획득한다. 획득한 정보는 JXTA에서 사용하는 광고(Advertisement) 형태를 취하고 있다. 이 정보를 이용하여 작업 피어는 그룹에 참여할 수 있게 된다. 획득한 정보를 바탕으로 ② 시스템 관리자에게 자신의 피어 정보를 전송하면, ④ 트래픽 관리자로부터 처리할 쿼리를 전송 받고 이를 처리하여 ⑤ 그 결과를 시스템 관리자에게 전송한다.

다. JXIR 시스템 구현 클래스

(1) SystemManger Class

시스템 관리자의 역할은 피어 그룹 생성, 참여, 그룹 정보 광고 발행, 작업 피어의 정보 획득 대기 및 획득한 정보를 우선순위 테이블로 작성, 트래픽 관리자에 우선순위 테이블 정보 전송 등이다.

<표 1> SystemManger Methods

메소드	내용
startJXTA()	JXTA 플랫폼 시작
startServer()	작업 피어로부터 정보수집을 위한 서버(서비스) 동작
receivePeerInfo()	작업 피어로부터 정보 수신 메서드
buildPriorityTable()	우선순위 테이블 구성
run()	작업 피어로부터 정보 수신을 위한 스레드 메서드

(2) TrafficManger Class

트래픽 관리자는 시스템 관리자로부터 받은 우선순위 테이블 정보를 바탕으로 작업 피어에게 들어오는 작업 요청 쿼리들을 전송한다.

<표 2> TrafficManger Methods

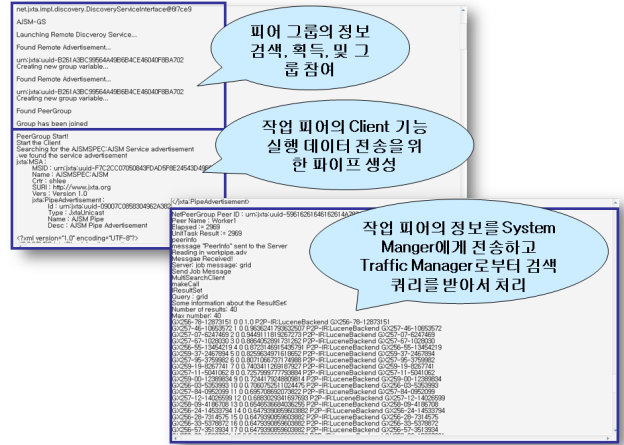
메소드	내용
TrafficManager()	트래픽 관리자 생성자: 사용할 스레드(최대개수) 초기화
sendQueryEvent(Vector logEvent)	피어로 부터 작업 요청 쿼리의 정보를 받는다.
run()	트래픽 관리자를 스레드로 동작하기 위한 실행 메서드
sendQueryEvent()	데이터 전송 메서드
setPeerGroupConfigure(SystemManager mySysManager)	피어 그룹 설정 메서드

(3) WorkPeer Class

작업 피어는 피어에 대한 정보를 시스템 관리자에게 전송하고 검색 쿼리 작업을 위한 대기 상태에 들어간다. 트래픽 관리자에게 검색 쿼리 작업을 요청 받으면 처리를 마치고 시스템 관리자에 결과를 전송한다.

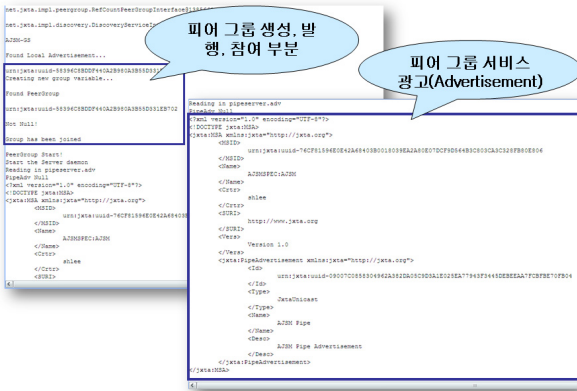
<표 3> WorkPeer Methods

메소드	내용
startJXTA()	JXTA 플랫폼 시작
run()	작업 피어 작업 처리를 위한 쓰레드 메서드
runClient()	Pipe 정보 설정 및 Client 기능 시작
sendAndReceiveData (PipeAdvertisement pipeAdv)	쿼리 작업(Job) 처리를 위한 메서드

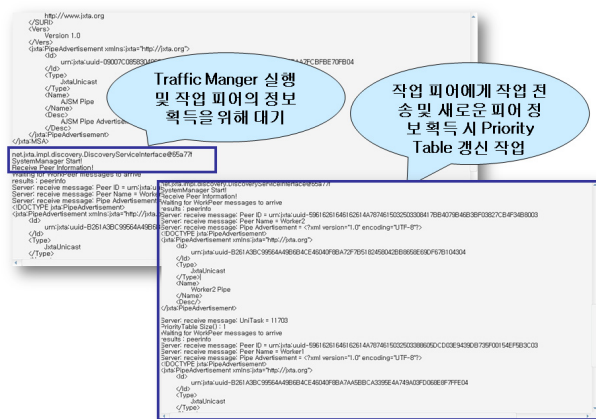


(그림 6) 작업 피어 실행 - 피어 그룹 참여 및 요청 작업 쿼리 처리

라. JXIR 시스템 동작 화면



(그림 4) 시스템 관리자 실행 - Jxta 플랫폼 실행 및 피어 그룹 생성, 발행



(그림 5) 트래픽 관리자 실행 및 작업 피어로 작업 전송

5. 결론

JXTA 플랫폼은 P2P 서비스를 위한 기반 기술이고 루씬은 확장 가능한 고성능 정보검색 라이브러리이다. 일반적인 정보검색 시스템은 중앙집중형으로 이는 중앙 서버의 처리량이 많아질 경우 시스템을 확장하는데 많은 어려움을 가지고 있다. 본 논문은 위 두 기술을 사용하여 일반적인 중앙 집중형 정보검색 시스템에 필요한 자원을 적합한 시간에 제공하여 시스템 확장을 유연하게 할 목적으로 JXIR(Jxta Information Retrieval) 시스템을 설계하고 구현하였다.

JXIR 시스템의 시스템 관리자와 트래픽 관리자는 피어의 확장을 관리하고 피어에 검색을 요청하는 역할을 수행한다. 이 기능을 통해 중앙 집중식 시스템이나 일반 개인 정보검색 시스템이 추가적인 자원을 필요로 하는 경우 자연스럽게 시스템을 확장하여 처리할 수 있게 된다. 하지만 본 시스템은 우선순위 테이블 작성에 사용된 정보를 피어 컴퓨터의 단순한 사양 정보를 기준으로 우선순위를 작성하여 처리하였다. 향후 연구를 통해 실질적으로 사용가능한 리소스에 대한 정보를 기반으로 시스템을 확장 할 필요가 있다.

참고문헌

- [1] Li Gong, "JXTA: a network programming environment", IEEE Internet Computing, Volume: 5 Issue: 3, May-June 2001 Page(s): 88-95
- [2] Project JXTA web site, <http://www.jxta.org>, Sun Microsystems. 2001.
- [3] Project JXTA, "JXTA v2.0 Protocols Specification", Sun Microsystems 2003.
- [4] Lucene web site, <http://lucene.apache.org>, The Apache Software Foundation.
- [5] Erik Hatcher, Otis Gospodnetic, "Lucene IN ACTION", Manning Publications