P2P 네트워크에서의 그룹 통신 매커니즘

손영성* · 김희정* · 김경석o
*한국전자통신연구원 · o부산대학교

Group Communication Mechanism in P2P Network

YoungSung Son* · HeeJeong Kim* · KyongSok Kimo
*ETRI · oPusan National University
Email : ysson@etri.re.kr, heejkim@etri.re.kr, gimgs@asadal.cs.pusan.ac.kr

요 약

본 논문은 인터넷 상의 PC 군을 네트워크로 연동하여 가상의 컴퓨팅 인프라를 구성하려는 연구에 관한 것이다. 자원의 복제 및 공유를 통해서 컴퓨팅 인프라 전반에 걸친 신뢰성과 결함감내 능력을 향상시키는 P2P 네트워크가 활발히 연구되고 있다. 그러나 현재까지의 P2P 네트워크는 복제된 데이터의 일관성 유지를 위해 노드들 간에 필요한 통신 매커니즘에 대한 고려가 부족하다. 본 논문은 이런 문제를 해결하기 위해서 그룹 통신 방식을 제안하고 IP 멀티캐스트 방식을 응용한 Spanning Tree 방식의 링 구성을 통하여 P2P 네트워크에서 그룹통신을 지원하기 위한 메시지 전송방식을 설명하고 자원의 복제를 통한 신뢰성 향상 기법에 대해서 소개한다.

키워드
P2P Network, Group Communication System, Routing Algorithm, Replication

I. 서 론

인터넷의 보급은 전통적인 서버 집중식 모델이 사용자 PC의 성능의 향상과 네트워크의 충분한 속도와 대역폭을 활용하지 못하는 문제점을 보여준다. 그러한 현실인식을 바탕으로 떠오르고 있는 Peer to Peer (P2P) 기술은 분산컴퓨팅 기술의 성공적인 상업화를 예견하고 있다. P2P는 불특정 다수가 참여하는 분산시스템으로서, 정보검색과 정보전송, 연산에서의 성능향상뿐만 아니라 컴퓨팅 시스템 전반적에 걸친 신뢰성과 결함 감내 능력을 향상시킬 수 있다. 이러한 결과는 분산 시스템에서 결함 감내, 고성능, 고가용성 등의 효과를 목적으로 자주 사용하는 기법이기도 한 자원의 복제 및 공유 특성 때문이다. 이러한 복제 개체들간의 일관성이 필수적인 응용을 개발함에 있어서의 어려움을 경감시키기 위해 그룹통신시스템이 개발되었다 [8].

그룹통신시스템이 제공하는 복제 개체의 일관성 보장은 신뢰성 있는 다중전송 프로토콜을 전제로 하지만, 기존의 그룹통신시스템에 적용되었던 신뢰성 있는 다중전송(Reliable Multicast) 기술들은 P2P 시스템에 적합하지 않다. IP 다중전송은 그 제어와 관리 관련 문제로 인하여 여전히 비현실적이며 [9], 응용수준의 다중전송은 이러한 이슈를 극복하지만 QoS 문제가 있다 [10]. 한편, P2P 망을 구성하고 있는 노드들이 상호 협력하여 파일이나 컴퓨팅 능력을 공유하는 Peer to Peer 네트워크가 소개되었으나, 그룹통신에 적합하지 않거나 확장성이 없다. [2]

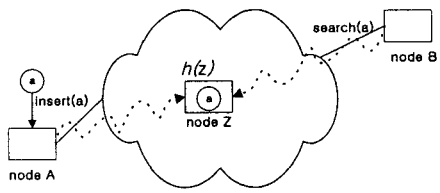
본 논문은 P2P 시스템에서의 그룹통신을 위하여 설계된 확장성 있는 메시지 전송방식과 신뢰성 향상을 위한 복제 기법에 대해서 설명한다.

II. 관련연구

2.1 P2P 기술과 일반적인 P2P 메시지 전송 방식

P2P 기술은 불특정 다수가 참여하는 분산시스템이며 검색의 효율성, 익명성, 저장 비용의 분산 등의 특징을 가진다. P2P 구조에서는 각 클라이언트의 네트워크 상황도 고정적이 아니며 PC 의 작업로드 상황도 안정적이지 않다. 특히 PC 의 도메인

도 없고 IP가 고정되어 있지 않는 것이 일반적이기 때문에 매 서비스시에 새로운 네트워크 상황을 초기화하는 작업 등이 필요하다. 또한, 사용자나 네트워크 상황으로 인한 빈번한 단절상황을 고려해야하는 어려움이 있다. P2P의 파일 공유방식은 Napster [1]와 같이 중앙에 서버에 공유 파일의 인덱스를 두는 방식과 Gnutella [2]와 같이 플러딩(flooding)을 이용해서 파일을 찾는 순수 P2P 방식이 있다. 순수 P2P 방식은 플러딩의 양이 어떤 시점에서부터 감소하기 때문에 실제로 시스템에서 파일을 못 찾을 수도 있다.



[그림 1] P2P 파일 저장 방법

각 노드는 전체 시스템의 유일한 해쉬함수(h(x))를 가진다. 이 해쉬함수는 자신의 대표 id 를 만들기 위해서 사용되고 객체 저장, 검색시에도 사용된다. 일반적인 P2P 네트워크에서 객체 저장 방법은 그림 1과 같다. 파일을 저장하기 위해서는 노드A는 저장할 객체(a)의 특성(객체 이름, 크기, 생성날짜, 그외 정보) 를 해쉬함수(h(x))를 이용해서 저장위치키 (destination key) 를 결정한다. 이 저장위치키를 이용해서 이 값을 담당하는 노드 Z 를 찾아낸다. 그리고 객체(a) 를 노드 Z 에 저장한다. 노드B 에서 해당 객체(a)를 찾기 위해서는 해쉬함수(h(x))를 이용해 알아낸 저장위치키를 이용해서 노드Z를 찾아서 객체(a)를 얻는다.

2.2 그룹통신시스템

CSCW, 그룹웨어, 네트워크 게임 등 많은 응용에서는 공동의 목적을 위한 공유데이터를 복제함으로써, 시스템의 신뢰성이나 결함감내, 가용성 향상을 도모하지만, 그 복제 데이터의 일관성을 보장하는 것은 상당히 어려운 작업이다. 그룹통신시스템은 현재 일관성을 유지하고 있는 멤버들의 리스트를 관리하면서, 그 멤버쉽이 동적으로 변경될 때마다 멤버들에게 통보하고, 공유 데이터에 대한 업데이트를 모든 멤버들에게 정확히 다중 전송함으로써, 복제 데이터의 일관성을 보장한다 [8].

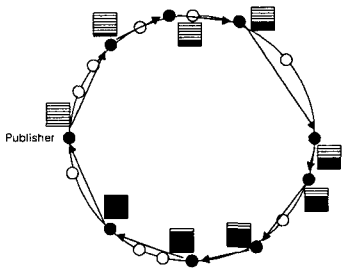
III. 그룹통신 메시지 전송 방식

3.1 메시지 전송 방법

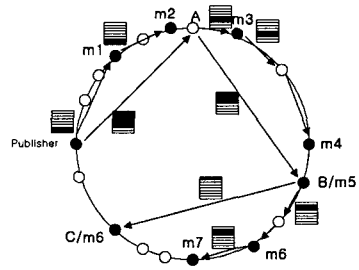
Magic Square 네트워크에서 메시지 전송은 시작지점(start)에서 목적지점(destination)까지 Peer를 건너뛰며 진행된다. 우선 그림 4와 같이 시작지점에서 라우팅 테이블을 검색하여 목적지점와 가장 가까운 Peer를 정해서 메시지를 전송한다. 메시지를 받은 Peer 는 자신의 아이디와 비교한 뒤 라우팅 테이블을 검색해서 위의 작업을 반복해서 수행한다.

3.2 순차 전송 방식 (Basic Scheme)

그룹 통신 서비스에서 일반적인 메시지 전송은 한 전송 멤버에서 다수의 수신 멤버에게 메시지를 전송하고 모든 멤버가 그 메시지를 받았을 경우에 끝난다(commit). 기본적인 IP multicast 를 이용할 경우는 순차적으로 1대1 통신을 수행하기 때문에 전송 멤버의 블록킹 현상이 발생한다. 이를 해결하기 위해서 Magic Square 네트워크의 특성을 이용한 그림 2의 순차 전송 방식을 설명한다. 메시지 송신 멤버는 메시지의 헤더에 수신 멤버를 모두 기록해서 메시지를 구성한다. 송신 멤버는 메시지 헤더에 있는 멤버를 순차적으로 찾아서 메시지를 시계방향(clock wise)으로 전송한다. 각 멤버는 메시지를 받아 수신여부를 메시지 헤더에 기록한 뒤 다음 멤버로 재전송한다.



[그림 2] 순차 전송 방식



[그림 3] Spanning Tree 방식

3.3 Spanning Tree 전송방식 (Spanning Tree Scheme)

순차 전송 방식은 모든 멤버에 순차적으로 메시지를 전송하기 때문에 Magic Square 네트워크의

효과적인 Peer 건너뛰기 진행방식으로 메시지 전송 방법 사용할 수 없다. 효과적인 전송을 위해서는 큰 라우팅 테이블을 가진 노드만을 건너뛰어야 하나 순차 전송 방식은 작은 라우팅 테이블을 가진 노드(멤버)를 많이 거치게 되어 메시지 재전송이 발생하게 된다. 이를 개선하기 위해서 spanning tree 전송방식을 고안하였다. Spanning tree 전송 방식은 메시지 전송시에 메시지 전송 효율을 높이기 위해서 큰 라우팅 테이블을 가진 노드(A,B,C)에서 그림 3 와 같이 메시지를 나누어서 병렬 전송을 한다. 전체 Magic Square 네트워크는 노드 A, B, C 를 기준으로 크게 영역이 나눠진다. Publisher 가 멤버들(m1~m6) 에게 메시지를 전송할 경우에 노드 A가 메시지를 받으면 멤버 m3,m4 를 향한 메시지와 노드 B 를 향한 메시지를 나눠서 동시에 전송한다. 노드 B와 노드 C 에서도 동일한 작업을 수행한다.

3.4 메시지 전송 성능 평가

이 절에서는 앞에서 소개한 메시지 전송 방식의 성능을 해석해본다. P2P 네트워크에서는 메시지 전송 시에 목적지점에 도착하는데 거친 노드의 개수를 홉(hop) 수로 정하고 이를 평가 기준으로 삼았다. 또한, 동시에 발생하는 메시지의 개수도 중요한 평가 척도가 된다.

N개의 노드로 구성된 Magic Square 네트워크에서 메시지의 전송시의 평균 홉수는 O(logN) 이다.

M개의 멤버로 구성된 그룹에서의 메시지 전송시의 평균 홉수는 다음과 같다. 순차 전송 방식은 항상 하나의 메시지가 전송되나 모든 멤버에게 메시지를 전송하기 위해서는 M 번의 재전송이 발생한다. 따라서 순차 전송 방식에서는 모든 멤버가 메시지를 받아 전송을 종료(commit)하려면 M* logN 의 홉수가 필요하다.

Spanning Tree 전송 방식에서는 전송중에 메시지가 나눠져서 최대 O(Height) 개의 메시지가 전송된다. 하지만, 모든 메시지가 최종 전달되는 시간은 logN 이 된다.

	Basic	Spanning tree
메시지 전송 경로	Clockwise skiplist sequence	Spanning tree sequence
메시지 재전송 횟수	O(M)	Avg: O(logN) Worst: O(N)
동시 메시지 개수	1	O(Height)
평균 홉수	O(logN)	O(logN/M)
최종전송홉수	O(M*logN)	O(logN)

[Table 1] The Comparison between two schemes

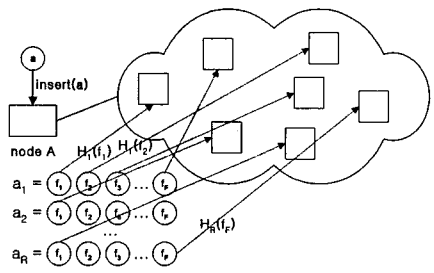
IV. 가용성 향상 기법

4.1 복제(Replication) 방법

복제 방법은 저장할 파일을 P2P 네트워크의 여러 노드에 복제해서 중복 저장한다. 복제하는 방법으로 다중 해쉬 방식과 재귀 해쉬 방식이 있다. 다중 해쉬 방식은 복제할 수(R개)만큼 해쉬 함수(h1(x), h2(x), h3(x), … , hR(x))를 만들어 복제를 저장할 때 사용한다. 재귀 해쉬 방식은 해쉬함수의 결과값을 다시 해쉬한다. 맨 처음 파일 저장시의 해쉬 함수는 h(x) 이고 두번째 복제 파일 저장시의 해쉬 함수는 h(h(x)) 가 된다. 두 방식은 성능면에서 차이는 없다. 본 논문에서 다중 해쉬 방식을 채택하였다. 다만, 복제된 파일이 같은 노드에 저장되는 것은 금지한다. 복제 방식은 복제하는 횟수만큼 가용성이 향상되는 것을 보장한다.

4.2 복제 분할(Replicated Fragmentation) 방법

이 방법은 위의 복제 방법과 분할 방법을 혼합한 것으로 파일을 다수(F개)의 조각(fi)으로 분할한 뒤 각 조각을 다수(R개)로 복제해서 각각을 저장하는 방식이다. 그림2에서 복제 분할 방법을 설명하고 있다. 파일(a)는 R개 복제되고 각 복제파일(ai)는 F개의 조각(f1, f2, f3, … , fF)으로 분할되었다. 이 각각의 조각을 해쉬함수 Hi(x)를 이용해서 저장할 노드를 정하도록 하였다.



[그림 4] 복제 분할 방법

4.3 가용성 분석

본 절에서는 앞에서 소개한 가용성 향상 기법의 효과를 분석해 본다.

분석에 필요한 파라메터는 다음과 같다.

네트워크에 참여하는 노드 수 : N
결함(Failure)있는 노드 수 : M
파일 조각의 수 : F
복제 파일의 수 : R

따라서, 파일 분할과 파일 복제가 함께 적용된 경우의 파일이 가용하지 않을 확률은 다음과 같다.

$$Failure(a) = 1 - \left(1 - \frac{M}{N} \times \frac{M-1}{N-1} \times \cdots \frac{M-R+1}{N-R+1}\right)^{F}$$

V. 결 론

본 논문에서는 그룹통신시스템의 기반 기술인 신뢰성 있는 다중전송을 P2P 네트워크에서 구성하는 방법에 대해서 소개하였다. P2P 네트워크에 참여하는 모든 노드를 일정한 방식으로 정렬하고 노드들간의 상호 연결성을 라우팅 테이블로 구성하고 메시지를 재전송하여 신뢰성있는 멀티캐스트 전송방식을 대체할 수 있는 순차 전송 방식과 Spanning Tree 전송 방식을 소개하였으며, 기존 P2P 메시지 전송방식과의 비교를 통하여 동시 메시지 개수는 늘어나지만 최종 전송 홉수에서 성능 향상을 보였다. 또한, 객체의 접근 가능성이 매우 중요한 요소이다. 이러한 객체에 대한 접근 가능성을 가용성 분석을 통해서 복제 방법, 분할 방법, 복제와 분할 방법을 동시에 사용한 경우에 대해서 살펴보았다.

참고문헌

[1] Napster. http://www.napster.com
[2] Gnutella. http://gnutella.wego.com/
[3] I. Clark, et al., "Freenet: A distributed anonymous information storage and retrieval system in designing privacy enhancing technologies," In Proc. International Workshop on Design Issues in Anonymity and Unobervability, LNCS 2009, 2001.
[4] A. Rowstron, et al., "Pastry: Scalable, decentralized object location and routing for large scale peer to peer systems," 18th IFIP/ACM International Conference on Distributed Systems Platforms, 2001.
[5] I. Stocia, et al., "Chord: A Scalable Peer to peer Lookup Service for Internet Applications," SIGCOMM, 2001.
[6] S. Ratnasamy, et al., "A Scalable Content Addressable Network," SIGCOMM, 2001.
[7] William Pugh, "Skip Lists: A Probabilistic Alternative to Balanced Trees," Communication of ACM, June 1990.
[8] Gregory V. Chockler, et al., "Group Communication Specifications: A Comprehensive Study," ACM Computing Surveys, Dec. 2001.
[9] S. Deering, "Host Extensions for IP multicasting," Internet RFC 1112, Available at http://www.ietf.org/rfc/rfc1112.txt, 1989
[10] J. Jannotti, et al., "Overcast: Reliable Multicasting with an Overlay Network," In Proceedings of OSDI, Oct. 2000.