

분산환경에서 CORBA를 기반으로 한 신뢰성 있는 그룹통신기법

The Mechanism for Reliable Group Communication Based on CORBA in Distributed Environment

안 계 호*
Kye-Ho Ahn

이 재 완**
Jae-Wan Lee

요 약

개방형 통신망 구조에서 어플리케이션 시스템은 다수의 컴퓨팅 노드에 분산되어 있는 객체들의 집합으로 구성된다. 이러한 분산된 환경에서 복잡한 분산 소프트웨어의 개발 및 관리의 복잡성을 줄이고 분산된 객체들을 효율적으로 관리하면서 객체들의 실시간 서비스를 제공하기 위해서는 객체 그룹 관리 및 통신 기법이 필요하다. 본 연구에서는 CORBA를 기반으로 하여 기존의 시스템에 큰 변환 없이 새로운 서비스를 적용시킬 수 있도록 객체그룹을 구성하고 신뢰성 있는 그룹통신 기법을 제공한다. 그룹관리의 효율향상을 위해 그룹을 서브그룹으로 나누고 서브그룹 관리자를 두어 서브그룹을 관리하도록 하였으며, 시퀀서를 두어 메시지의 순서성을 유지시킨다. 그룹간의 통신방법은 IP 멀티캐스트와 일대일 통신을 기반으로 한 멀티캐스트 기법을 혼용하여 사용하며 메시지 저장소를 두어 오류 발생시 선택적으로 메시지를 재 전송할 수 있도록 함으로서 성능을 향상시켰다. 제안된 기법을 분산환경에서 시뮬레이션을 통하여 성능을 평가하였다.

Abstract

In open communication architecture, application systems consist of objects distributed in lots of computing nodes. To reduce complexity of development and management of distributed software, to manage efficiently distributed objects and to provide realtime service, the mechanisms for object group management and communication are needed in distributed environment. In this paper we compose object groups and provide reliable group communication mechanism based on CORBA which can adopt a new service without lots of changes on existing systems. Group consist of some subgroups that subgroup manager manages for improving the efficiency of group management and message ordering is kept by using sequencer. Communication method among groups uses multicast based on point to point communication as well as IP multicast and we provide high performance by using selective retransmission of message through message history. We analyze the performance of the proposed mechanism through simulation in distributed environment.

1. 서 론

현재의 컴퓨팅 환경은 네트워크 기술의 급속한 발전과 더불어 기존의 대형 서버 중심의 클라이언트/서버 컴퓨팅에서 벗어나 개방형 통신망에 접속된 다양한 플랫폼을 이용한 분산 컴퓨팅 환경으로 바뀌어 가고 있다. 개방형 통신망 구조에서

어플리케이션 시스템은 다수의 컴퓨팅 노드에 분산되어 있는 객체들의 집합으로 구성된다. 이러한 분산된 환경에서 복잡한 분산 소프트웨어의 개발 및 관리의 복잡성을 줄이고 분산된 객체들을 효율적으로 관리하면서 신뢰성 있는 서비스를 제공하기 위해서는 객체 그룹 관리 및 통신 기법이 필요하다[1,2].

따라서 본 연구에서는 CORBA를 기반으로 하여 분산 객체들을 객체그룹으로 통합 관리하며, 그룹간의 통신방법은 IP 멀티캐스트와 일대일 통

* 군산대학교 대학원 정보통신공학과 졸업(석사)
khahn@kunsan.ac.kr

** 군산대학교 전자정보공학부 부교수
jwlee@kunsan.ac.kr

신을 기반으로 한 멀티캐스트 기법을 혼용하여 사용하고 그룹을 서브그룹으로 나누어 그룹의 멤버십 관리를 용이하게 한다. 또 그룹 내 메시지의 순서성을 보장하기 위하여 하나의 시퀀서를 두어 메시지에 일관된 시퀀스를 부여하고 이 시퀀스를 통하여 메시지 순서성을 유지함으로써 신뢰성 있는 그룹통신 기법을 제공한다. 시퀀서는 메시지 저장소를 가지고 있어 일정 기간동안의 그룹 내 모든 메시지를 저장하며 이 저장된 메시지는 메시지 수신자로부터 재전송요청이 발생하였을 때 재전송 하는데 이용된다. 메시지 재전송은 선택적인 메시지 재전송방법으로 해당되는 재전송 메시지만을 재전송 할 수 있도록 하여 성능을 향상시켰다. 본 연구결과는 분산환경에서의 그룹통신을 기반으로 자원공유, 다중 의사결정 시스템, 시스템통합 등에 활용될 수 있다.

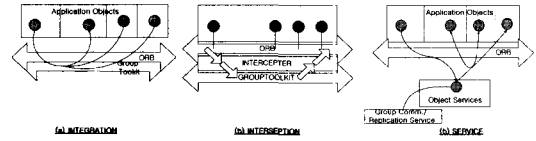
2. 관련연구

2.1 그룹서비스

분산된 객체들을 개별적으로 관리하려면 객체들을 관리하는데 매우 복잡하다. 따라서 분산된 객체들을 하나의 객체그룹으로 관리함으로써 객체들을 관리하는데 복잡함을 줄일 수 있다.

그룹서비스는 그룹멤버들과 통신하는 프리미티브를 제공하며, 각각의 그룹 멤버의 수, 식별자, 위치 정보를 알고 있을 필요 없이 한번의 요구로 모든 그룹 멤버들에게 동일한 요구를 발생할 수 있다.

기존의 CORBA를 기반으로 그룹통신을 추가하는 방법은 현재 존재하는 그룹통신 툴킷을 ORB에 통합하는 통합방법(integration approach)과 CORBA 서비스처럼 객체그룹 서비스를 구축하여 서비스하는 방법(service approach) 그리고 그룹 통신 시스템의 메시지에 해당하는 IIOP 메시지를 가로채서 복제 매니저에게 전송하는 인터셉터에 의한 방법(interception approach)이 있다[3,4,5].



(그림 1) CORBA를 기반으로 하는 그룹서비스방법

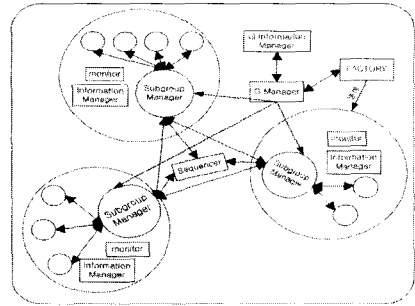
통합 방법[3]은 그룹통신 툴킷을 ORB내에 통합시킨 방법으로 Orbix+Isis와 Electra에 채택되어 왔으며 편리성과 투명성을 제공하지만 각 시스템 마다의 독자적인 시스템 툴킷이 존재하기 때문에 시스템에 의존하는 단점이 있고 인터셉터[5][6]에 의한 방법은 ORB를 수정할 필요가 없지만 인터셉터가 시스템 호출 수준에서 구현되어 있기 때문에 운영체제에 종속된다는 단점이 있다.

서비스 방법[4]은 CORBA의 표준을 따르는 서비스를 구축하고 이 서비스를 통하여 CORBA에 그룹 통신서비스를 추가함으로써 CORBA 명세의 변경을 요구하지 않는다는 장점이 있으나 서비스를 구축하는데 따른 비용과 사용하기가 불편하다는 단점이 있다.

2.2 그룹통신

그룹 서비스에서는 그룹 메시지를 모든 그룹의 멤버들에게 전송하기 위해서는 멀티캐스트 전송 방식이 필수적이다. IP 멀티캐스트는 인터넷 프로토콜의 멀티캐스트 주소로 예약되어 있는 D클래스의 멀티캐스트 주소를 이용하는 방법으로서 한번의 송신으로 그룹내의 모든 멤버들에게 메시지를 전송할 수 있다. 그러나 그룹 멤버십이 동적으로 변화하는 그룹에는 적합하지 못하고 메시지 전달에 대한 보장이 없기 때문에 멤버십이 동적으로 변화하는 그룹서비스에 적용하기 위해서는 그룹의 멤버십관리와 그룹 메시지 관리가 필요하다[8]. CORBA는 TCP를 기반으로 하고 멀티캐스트 기술을 제공하지 않기 때문에 CORBA를 이용한 그룹서비스에서는 멀티캐스트 전송을 위해 멀티캐스트 전송을 할 수 있는 도구를 이용하거나

각각의 멤버들에게 유니캐스트(Unicast)하는 방법이 사용되고 있다. 다중으로 유니캐스트 하는 방법은 메시지 전달의 신뢰성에 대한 장점이 있지만 메시지를 수신해야하는 노드의 수에 따라서 증가하는 오버헤드가 가장 큰 문제점이다. 본 논문에서는 IP 멀티캐스트 기법과 다중 유니캐스트 기법을 동시에 사용하며 IP 멀티캐스트의 단점인 멤버십 문제를 해결하기 위해 모니터링 서비스를 통해 그룹의 멤버십을 관리한다.



(그림 2) 그룹 통신 시스템의 구성

3. 그룹통신 시스템

3.1 시스템 구조

다수의 서로 다른 프로세스들이 서로 다른 컴퓨터 상에 분산되어 있는 분산환경에서는 분산된 객체들을 개별적으로 관리하려면 매우 복잡하고 여러 가지 어려움이 따를 뿐만 아니라, 프로세스들간 통신에서 결합허용과 높은 가용성이 요구된다. 따라서 분산된 객체들을 하나의 서브그룹으로 관리함으로써 객체들을 관리하는데 복잡함을 줄일 수 있으며, 또한 멀티캐스트 오퍼레이션은 한 프로세스에서 단일 메시지를 전송함으로써 다른 프로세스들에게 메시지를 전송할 수 있다. 그룹서비스에서 메시지를 송신하는 프로세스는 그룹서비스의 멀티캐스트 오퍼레이션을 호출함으로써 그룹의 모든 프로세스들에게 메시지를 송신할 수 있다. 그러므로 그룹서비스는 신뢰성 있는 메시지 전달을 보장하고 또한 메시지 전달의 투명성을 제공한다. 객체그룹은 여러 개의 서브그룹으로 구성되며, 또한 서브그룹은 여러 개의 객체들로 구성된 그룹을 의미한다. 객체그룹의 구성은 그림 2와 같다.

3.2 구성요소

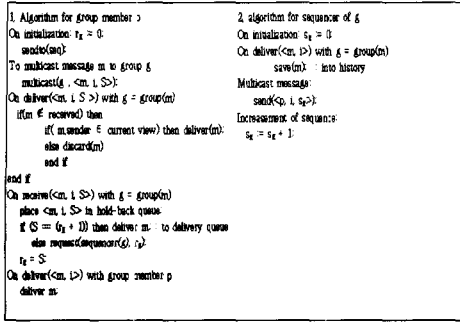
객체 그룹은 그룹 관리자 객체(GM : Group Manager Object), 그룹정보 관리자 객체(GIM : Group

Information Manger Object), 팩토리 객체(GFO : Group Factory Object), 그룹 모니터(GMO : Group Monitor Object), 서브그룹 관리자(SGM : SubGroup Manager Object), 서브그룹 정보관리자 객체(SGIM : SubGroup Information Manager Object), 시퀀서 객체(Sequencer Object)로 구성된다. 그룹관리자는 그룹을 초기화하고 객체 그룹을 관리하며 그룹관리를 효율적으로 관리하기 위해 뷰를 생성하고 관리한다. 또, 팩토리 객체와 상호작용 하여 그룹의 요소들을 생성하고 삭제를 수행한다. 그룹 정보관리자는 그룹내의 뷰에 관한 정보를 관리하며 그룹 및 서브그룹에서 사용하는 그룹통신 주소를 관리한다. 서브그룹 관리자는 서브그룹의 멤버십 관리 및 서브그룹으로 전송되는 메시지를 관리한다. 모니터객체는 그룹내 객체들의 상태를 감시하고 오류발생시 이를 그룹관리자에게 통보하여 그룹관리자가 뷰를 일관성 있게 유지할 수 있도록 한다. 시퀀서객체는 메시지의 순서화를 위한 시퀀스를 생성하고 오류제어 역할을 하며 그룹 메시지를 저장한다.

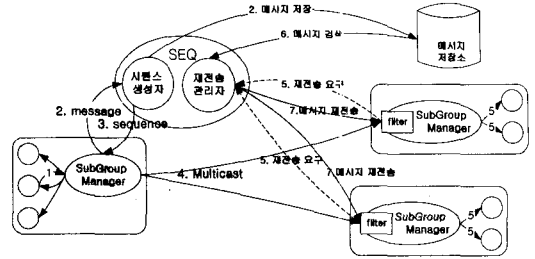
4. 그룹통신

4.1 그룹통신 알고리즘

그림 3은 그룹통신 알고리즘이다. 객체들간의 통신방법은 일대일 통신을 이용한 멀티캐스트와 IP 멀티캐스트를 사용한다. 메시지 송신자는 메시지를 전송하기 위해서 먼저 메시지 시퀀서를 부



(그림 3) 그룹통신 알고리즘



(그림 4) 그룹통신 수행과정

여받기 위해 메시지와 메시지식별자($\langle m, i \rangle$)를 시퀀서(seq)에 전송하고 대기한다. 메시지를 수신한 시퀀서는 메시지의 시퀀스(sg)를 부여하고 메시지 저장소(history)에 메시지를 저장한 후 메시지식별자와 시퀀스를 반환한다. 시퀀스를 부여받은 메시지는 그룹(g)으로 멀티캐스트 된다. 수신된 메시지는 일단 홀드백 큐로 이동되어 순서화된다. 홀드백 큐에서 수신된 메시지의 시퀀스(S)와 이전에 수신되었던 시퀀스(rg)를 비교함으로써 메시지를 순서화하며 순서화된 메시지는 수신큐로 이동된다. 일대일을 기반으로 하는 멀티캐스트는 단방향(oneway) 방식으로 전달되어지고 메시지를 전송한 쪽에서는 수신자의 메시지 수신 여부에 관계없이 다른 모든 멤버에게 메시지를 전송한다. 그림 4는 그림 3의 그룹통신 알고리즘을 그림으로 표현한 것으로 메시지의 송신에서 수신까지의 과정을 그림으로 표현한 것으로 송신될 메시지가 시퀀서에서 시퀀스를 부여받고 메시지 저장소에 저장된 후 전송되어 수신측에 수신되고 메시지 저장소에 저장되었던 메시지가 재전송되는 수행과정을 나타내었다.

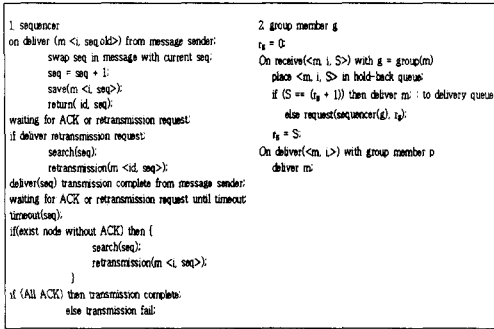
4.2 오류제어 알고리즘

시퀀서의 메시지 저장소에 저장된 메시지는 메시지 전달의 오류로 인한 재전송 요구 시 사용된다. 메시지 저장소에 저장되는 내용은 메시지와 메시지 식별자 그리고 메시지의 시퀀스가 저장되

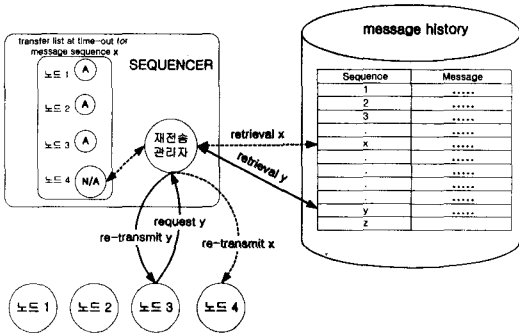
어 메시지 재전송 요구가 발생했을 때 시퀀서는 메시지의 시퀀스로 메시지를 메시지 저장소에서 검색하여 재전송 한다. 메시지 저장소에 저장된 메시지는 유효기간을 가지며 유효기간이 지난 메시지는 메시지 내용을 삭제하고 메시지의 식별자, 시퀀스 그리고 송신자의 정보만을 저장한다.

오류제어 알고리즘은 그림 5와 같고 그림 6은 알고리즘을 그림으로 나타내었다. 메시지 재전송 수신측에서 요구하는 경우(메시지 x)와 시퀀서에서 재 전송하는 경우(메시지 y)가 있다. 먼저 수신측에서 재전송을 요구하는 경우는 수신된 메시지의 시퀀스(S)가 이전에 수신된 시퀀스(rg) 보다 1 이상 클 때 발생한다. 이전에 수신된 메시지와 현재 수신된 메시지 사이의 메시지가 도착할 때까지 대기하지 않고 수신측에서는 시퀀스에 누락된 메시지의 재전송을 요구한다.

시퀀서에서 메시지마다 수신될 목적지들의 리스트(transfer list)를 작성하고 메시지의 수신응답을 확인한다. 일정시간 내에 확인응답 메시지가 수신되지 않을 때는 메시지를 재 전송한다. 메시지를 재전송하고 일정시간(timeout) 내에 재 전송된 메시지의 수신확인 메시지를 받지 못할 경우는 또 다시 메시지를 재 전송한다. 수신측에서는 재전송 메시지를 수신하면 먼저 전에 수신된 메시지인지를 검사하고 수신되지 않은 메시지는 메시지를 저장한 다음 수신응답메시지를 송신하고, 이미 수신된 메시지일 경우는 메시지를 폐기하고 시퀀서에게 수신응답 메시지를 송신한다.



(그림 5) 오류제어 알고리즘



(그림 6) 메시지 재전송

5. 성능평가

5.1. 시뮬레이션 환경

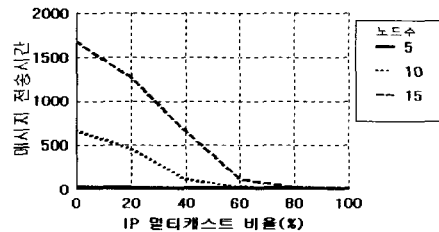
제안된 기법은 IONA 사의 orbix 2.02, visual C++ 6.0을 기반으로 하여 개발하였다. 또한, 시스템의 성능평가를 위해 그룹 통신에서 전체 멀티캐스트 메시지 중에서 1:1 통신을 기반으로 한 멀티캐스트의 비율에 따른 메시지의 전달 시간을 계산하였다. 시뮬레이션 환경은 window 2000, sun solaris 2.8 마이크로 소프트웨어 비주얼 C++ 6.0을 사용하여 실험하였다.

메시지의 발생은 임의의 노드에서 발생하고 노드에서 메시지를 전송하기 위해 전송을 준비하는 시간과, 노드에서 노드로 메시지가 전송되는 시간을 고려하였다. 메시지를 전송하기 위해 메시지

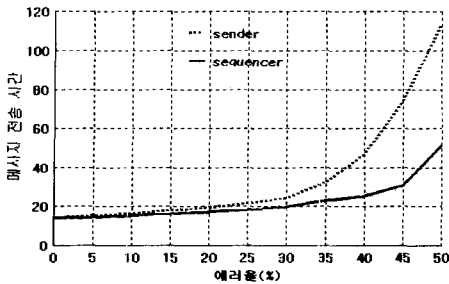
송신자는 먼저 시퀀서에 메시지를 전송하고 시퀀서는 시퀀스를 발생시키고 메시지를 저장한 후 메시지의 시퀀스를 반환한다. 송신자는 메시지에 대한 시퀀스를 수신하면 이 메시지를 그룹의 모든 멤버들에게 멀티캐스트 한다. 각 메시지의 발생시간부터 메시지가 모든 멤버들에게 전달될 때까지의 시간간격을 계산하였다. 본 연구에서는 1:1 전송과 IP 멀티캐스트의 비율에 따른 메시지 전송시간을 비교하고 메시지의 재전송을 송신측에서 처리할 경우와 시퀀서에서 처리할 경우의 메시지 전송시간을 비교하였다.

5.2 결과 및 분석

1:1 전송의 비율이 0%인 부분은 모든 메시지가 IP 멀티캐스트에 의해 전송되었음을 의미하고 100%인 부분은 모든 메시지가 1:1 전송에 의해 전송되었음을 의미한다. 그림 7은 어려움을 고려하지 않고 그룹내의 멤버수와 IP 멀티캐스트의 비율에 변화를 주었을 때의 결과이다. 결과에 나타난 실선과 점선들은 수신노드의 수를 나타낸다. 결과에서 보는바와 같이 메시지 전송에서 일대일 전송이 차지하는 비율이 커질수록 또 수신노드의 수가 많을 수록 메시지의 평균 전송시간이 증가한다. 이것은 IP 멀티캐스트 기법은 한번의 전송으로 모든 수신노드들이 메시지를 수신할 수 있지만 1:1 전송에 의한 멀티캐스트는 수신노드의 수만큼 송신을 해야하기 때문에 발생한다. 그러므로 수신노드의 수가 많을수록 1:1 전송 비율이



(그림 7) IP 멀티캐스트의 비율과 멤버수의 변화에 따른 전송시간 변화



(그림 8) 오류제어 위치에 따른 평균전송시간 비교

많아질 때 메시지 전송시간이 큰 폭으로 증가하는 것을 볼 수 있다.

그림 8은 오류제어를 송신측에서 했을 때와 본문에서 사용된 기법인 시퀀서에서 오류제어를 했을 때의 메시지 전송시간을 에러율에 따라 비교하였다. 오류제어를 송신측에서 했을 때 보다 시퀀서에서 했을 때 메시지 전송시간이 작게 측정되었으며 메시지의 재전송 횟수가 많아질수록 송신측에서 메시지 재전송을 처리하는 것보다 시퀀서에서 처리할 때 메시지 전송시간을 향상시킬 수 있음을 알 수 있다. 이는 메시지 재전송 처리를 시퀀서에서 함으로써 송신측에서 1:1 전송 시 다른 노드로의 전송이나 다음 메시지의 전송에 영향을 주지 않고 전송할 수 있기 때문이다.

6. 결 론

본 연구에서는 기존의 통신망에 큰 변화 없이 새로운 서비스를 적용시킬 수 있는 분산처리 기술, 객체지향형 모델링 및 설계 기법을 적용하여 분산된 객체들을 효율적으로 관리할 수 있고 신뢰성과 병렬성을 제공하여 소프트웨어 개발과 분산시스템의 성능을 향상시킬 수 있도록 분산된 객체들을 그룹화하고 서브그룹으로 분할하여 그룹관리를 효율적으로 할 수 있도록 하였다.

그룹에 등록되거나 탈퇴하는 객체에 대해 실시간에 뷰(View)를 관리하기 위해 그룹정보 관리자를 두어 그룹에 등록된 멤버들에 대한 일관된 뷰를 가질 수 있도록 하였고 시퀀서와 히스토리 기

능을 돕으로서 메시지의 순서화와 선택적인 메시지 재전송이 가능하고 그룹 모니터와 뷰 관리를 통해 멤버십이 동적으로 변화하는 환경에서도 멀티캐스트 기법을 사용하여 신뢰성 있고 투명하게 메시지가 전달될 수 있도록 하였다.

실험은 멤버수가 10개와 15개의 경우 전체 메시지 전송에서 IP 멀티캐스트의 비율이 약 35% 일 때 IP 멀티캐스트를 전혀 사용하지 않을 때 보다 50%의 성능이 향상되었고, IP 멀티캐스트의 비율을 증가시켰을 때 시스템의 성능도 향상되었다. 또한, 오류제어를 시퀀서에서 할 경우에는 에러율이 20%의 경우 송신측에서 오류제어를 했을 때 보다 약 10%의 성능향상이 있었으며 에러율이 증가할수록 성능은 향상되었다.

실시간 객체그룹 서비스는 분산된 객체들을 효율적으로 관리할 수 있으며, 이로 인해 분산시스템의 성능을 향상시킬 수 있다. 또한 실시간 멀티미디어 서비스를 제공할 수 있는 어플리케이션 개발 및 시스템간의 분산성 및 이질성으로부터 투명한 분산시스템을 개발할 수 있고, 그룹간의 관리를 통한 의사결정 시스템 지원, 객체그룹간의 보안 기법 개발 그리고 다중 질의 검색 서비스에 활용될 수 있고 분산된 객체들의 실시간 관리뿐만 아니라 CORBA를 기반으로 하여 특정 플랫폼이나 기술에 독립적인 어플리케이션 개발, 이기종 분산환경에서의 자원공유, 분산 의사결정 시스템 등에 활용될 수 있다.

Acknowledgement

이 논문은 정보통신부에서 지원하는 대학기초 연구지원사업으로 수행되었음

참 고 문 헌

[1] N.Natarajan, F.Dupuy N.Singer, H.Christensen, "Computational Modeling Concept", TB-A2.HC. 012-12-94, TINA-C, Feb. 1995.

- [2] Christensen, E.Colba, "Information Modeling Concepts", DRAFT, TBEAC.001-1.1-93, TINA-C, Nov. 1994.
- [3] Silvano Maffei, "The Object Group Design Pattern", Dept. of Computer Science, Cornell Univ., Dec. 1996.
- [4] P. Felber, R. Guerraoui, and A. Schiper. "The implementation of a CORBA group communication service", In Theory and Practice of Object Systems, Volume 4, Number 2, 1998.
- [5] P. Narasimhan, L. E. Moser, P. M. Melliar-Smith, "Exploiting the Internet Inter-ORB Protocol Interface to Provide CORBA with Fault Tolerance", Proceedings of the Third USENIX Conference on Object-Oriented Technologies and Systems, June 1997.
- [6] L. E. Moser, P. M. Melliar-Smith, D. A. Agarwal, R. k. Budhia, C. A. Lingley-Papadopoulos, "Totem : A fault tolerant multicast group communication system", Communications of the ACM, Vol.39, No.4, pp. 54~63, April 1996.
- [7] OMG, "The Common Object Request Broker : Architecture and Specification v2.4.1", October 2000, <ftp://ftp.omg.org/pub/docs/formal/00-11-03.pdf>
- [8] George Coulouris, Jean Dollimore, Tim Kindberg, Distributed System: Concepts and Design, 3rd edition, pp.153-158, Addison Wesley, 2001.

● 저 자 소 개 ●



안 계 호

2000년 군산대학교 정보통신공학과 졸업(학사)

2002년 군산대학교 대학원 정보통신공학과 졸업(석사)

관심분야 : 분산운영체제, 컴퓨터 네트워크, 멀티미디어, 인터넷 응용 등

E-mail : khahn@kunsan.ac.kr



이 재 완

1984년 중앙대학교 전자계산학과 졸업(학사)

1987년 중앙대학교 대학원 전자계산학과 졸업(석사)

1992년 중앙대학교 대학원 전자계산학과 졸업(박사)

1996년 3월~1998년 1월 한국학술진흥재단 전문위원

1992년~현재 : 군산대학교 전자정보공학부 부교수

관심분야 : 분산시스템, 운영체제, 실시간 시스템, 컴퓨터 네트워크, 멀티미디어 등

E-mail : jwlee@kunsan.ac.kr