

광역 객체 컴퓨팅 환경에서 이름/속성기반의 통합 바인딩 서비스 방안

정 창 원[†] · 오 성 권^{††} · 주 수 종^{†††}

요 약

광역 컴퓨팅 시스템의 구조가 처음으로 네델란드 Vrije 대학의 분산 컴퓨팅 연구팀에서 명세화를 시키면서, 많은 연구개발자들은 분산객체의 광역 위치와 접속 서비스에 대한 연구들을 추진하고 있다. 이들 대부분의 연구들은 광역 컴퓨팅 사이트들 상에서 존재하는 비 중복된 연산 객체들간에 바인딩 서비스에 대해서만 고려하고 있다. 그러나 지구상에 존재하는 수많은 객체들은 이름이나 속성에 의해 중복된 특성을 지닌다. 이러한 같은 특성을 갖는 객체들은 중복된 연산객체로 정의할 수 있다. 기존의 네이밍이나 트레이딩 기법으로는 독립적인 위치투명성의 결여로 중복된 연산객체들의 바인딩 서비스 지원이 불가능하다. 따라서 본 논문에서는 광역 컴퓨팅 환경에서 중복된 연산객체들의 위치관리 뿐 아니라 시스템들간의 부하균형화를 유지시키면서 최적객체의 선정을 통한 바인딩 서비스의 시간을 최소화 할 수 있는 새로운 모델을 제시한다. 이 모델은 네이밍 및 트레이딩 기능들을 통합한 서비스에 의해 중복된 연산객체들에 대한 단일 객체들을 얻는 부분과, 이 객체들을 사용하여 노드관리자에 의해 중복객체들의 복수개의 컨택주소들을 제공하는 위치 서비스 부분으로 구성하였다. 위치 투명성을 제공하기 위해, 이 두 서비스는 서로 독립적으로 수행된다. 이러한 모델을 기반으로 분산객체의 광역 통합트리의 구조, 컨택주소들의 탐색 및 갱신 알고리즘을 기술하였다. 마지막으로 클라이언트 객체로부터 서로 다른 영역에 위치하는 분산객체들의 광역 바인딩을 제공할 수 있는 연합구조를 보였다.

A Study on Integrated Binding Service Strategy Based on Name/Property in Wide-Area Object Computing Environments

Chang-Won Jeong[†] · Sung-Kwun Oh^{††} · Su-Chong Joo^{†††}

ABSTRACT

With the structure of wide-area computing system which is specified by a researching team in Vrije University, Netherlands, lots of researchers and developers have been progressing the studies of global location and interconnection services of distributed objects existing in global sites. Most of them have focused on binding services of only non-duplicated computational objects existing wide-area computing sites without any consideration of duplication problems. But all of objects existing on the earth have the duplicated characteristics according to how to categorize their own names or properties. These objects with the same property can define as duplicated computational objects. Up to now, the existing naming or trading mechanism has not supported the binding services of duplicated objects, because of deficiency of independent location service. For this reason, we suggest a new model that can not only manages locations of duplicated objects in wide-area computing environments, but also provide minimum binding time by considering both the optimal selection of one of duplicated objects and load balance among distributed systems. Our model is functionally divided into 2 parts ; one part to obtain an unique object handle of duplicated objects with same property as a naming and trading service, and the other to search one or more contact addresses by a node manager using a given object handle, as a location service. For location transparency, these services are *independently executing each other*. Based on our model, we described structure of wide-area integrated tree and algorithms for searching and updating contact address of distributed object on this tree. Finally, we showed a federation structure that can globally bind distributed objects located on different regions from an arbitrary client object.

키워드 : 통합바인딩서비스(Integrated Binding), 광역컴퓨팅(Wide-Area Computing), 중복객체(Duplicated object), 연합모델(Federation Model), 최적객체선정(Optimal Object Selection), 부하균형화(Load Balancing)

1. 서 론

인터넷 기술의 발전에 따라 시스템 규모가 점차 커지고,

중앙 집중식 시스템에서 분산시스템들의 협력 형태인 광역 연합환경의 분산시스템으로 변화되고 있다. 현재 인터넷 기반의 웹서비스와 전자메일과 같은 광역 분산 서비스들이 사용자에게 제공되고 있지만, 국부적인 분산 투명성만을 제공하고 있다[1, 2, 18]. 이러한 분산투명성을 미들웨어 관점에서 해결하기 위한 연구들을 보면, 대표적인 사양으로 OMG의 CORBA[3]와 MS의 DCOM[4]을 들 수 있다. 최근

※ 본 연구는 한국과학재단 목적기초연구(2000-1-30300-010-2)지원으로 수행되었음.

† 준 회 원 : 원광대학교 대학원 컴퓨터공학과

†† 정 회 원 : 원광대학교 전기전자 및 정보공학부 교수

††† 정 회 원 : 원광대학교 컴퓨터 공학과 교수

논문접수 : 2001년 6월 9일, 심사완료 : 2002년 5월 17일

에 들어 이들간의 상호운용성을 지원하는 명세서가 제시되고 있으며, TINA[5]와 같은 컨소시엄에서도 기존의 통신망을 사용하면서 객체지향 기술을 접목시킨 프레임워크를 제시하고 있다. 이에 맞추어 광역 분산시스템의 등장과 함께, 이들 환경에 분산객체 컴퓨팅 방안을 광역화할 수 있는 광역 객체 컴퓨팅 모델의 제시가 요구되고 있다. 또한, 현재 광역 애플리케이션의 특성은 중복된 연산객체(이후, 중복객체라 부름)나 이동객체들의 바인딩 서비스가 요구되므로 위치 투명성이 요구되고 있다. 그러나 기존의 인터넷 기반에서 분산객체의 레퍼런스를 얻는데는 네이밍 서비스로만 가능하며 객체의 레퍼런스와 위치는 비독립적으로 유지되고 있다. 그러나 광역 객체 컴퓨팅에서 객체들의 분산 바인딩을 위해 네이밍 서비스와 위치서비스가 별도로 제공되도록 해야한다. 이 광역 객체 컴퓨팅 분야의 대표적인 연구는 네델란드 Vrije 대학의 A. S. Tanenbaum 연구팀에서 분산객체 컴퓨팅의 구조를 재정립한 Globe 시스템의 설계에서 네이밍 서비스와 위치서비스를 독립적으로 수행해야하는 근거를 제시하였다[6, 7]. 이러한 기반에서 많은 연구자들은 분산객체의 광역 위치와 접속 서비스에 대한 연구들을 추진하고 있다[8-12].

이들 대부분의 연구들은 원격 사이트에 존재하는 비 중복객체와의 바인딩을 위한 네이밍 및 위치서비스의 관점만을 고려하고 있다. 그러나 지구상에 존재하는 수많은 객체들은 이름이나 속성에 의해 다양한 중복된 성질을 갖는다. 이러한 같은 성질을 갖는 객체들은 중복객체로 분류할 수 있다. 기존의 네이밍이나 트레이딩 서비스로는 중복된 객체들의 바인딩 서비스가 불가능하다.

따라서 본 논문에서는 광역 객체 컴퓨팅 환경에서 중복객체들의 관리 및 최적객체의 선정을 통한 바인딩 서비스의 시간을 최소화하고, 시스템들간의 부하균형화를 유지할 목적으로 광역 객체 컴퓨팅 환경에서 중복객체들간의 바인딩을 위한 네이밍/트레이딩 서비스 지원 모델을 제시한다. 이 모델은 중복객체들의 객체핸들(object handle)을 제공하는 네이밍/트레이딩 서비스부분(이름 또는 속성 -> 객체핸들)과 객체핸들을 사용하여 노드관리자에 의해 컨택주소(contact address)를 제공하는 위치 서비스(객체핸들-> 객체의 현 위치인 컨택주소로 매핑) 부분으로 구성된다. 이러한 서비스의 제공을 위한 기반구조로 광역 분산객체의 위치탐색 트리의 구조, 컨택주소의 갱신 및 탐색 알고리즘의 제안하였다. 트리의 구조에서 각 노드의 키는 객체핸들이며 잎노드의 포인터는 객체의 위치정보인 컨택주소를 나타내고 있다. 마지막으로 광역 통합트리 기반에서 서로 다른 영역(region)간에 광역 분산객체들의 연합 바인딩 과정을 기술하였다.

본 논문의 구성은 다음과 같다. 2장에서는 광역 객체 컴퓨팅의 기본이 되는 분산객체에 대해 설명하고, 3장에서는

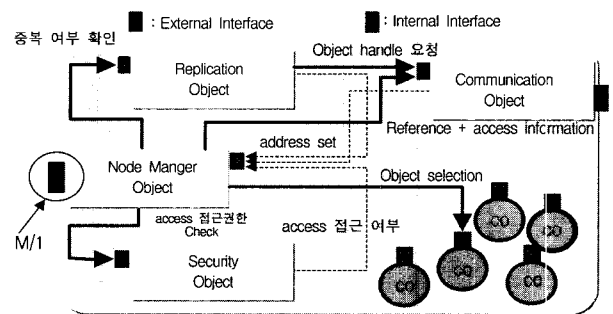
기존 서비스의 제한점과 이를 해결하기 위한 분산객체 바인딩 서비스에 대해 기술하고, 4장에서는 광역 통합 트리구조와 이를 기반으로 한 위치서비스의 알고리즘에 대해 기술한다. 5장에서는 서로 다른 영역간 혼합 서비스의 연합구조를 보인다. 그리고 끝으로 6장에서는 결론과 향후 연구내용에 대해 기술한다.

2. 광역 분산객체의 정의

광역 분산객체는 다중 주소공간(영역)에 즉 여러 사이트에 분산 및 중복객체로 존재한다. 객체그룹은 논리적 분산 영역내 객체들의 집합으로서, 하나 이상의 내·외부 객체와의 인터페이스와 수행상태 정보를 가지고 있으며, 객체그룹내 또는 다른 객체그룹내의 객체들과 상호작용을 가진다[13, 14]. 이들 그룹내 구성요소들은 노드관리자 객체를 중심으로 중복 및 분산객체들의 바인딩을 지원하는 복사, 통신객체들과 연산객체들, 그리고 연산 객체의 접근보안을 관리를 위한 보안객체로 이루어진다. 각 객체그룹 내에는 하나의 노드 관리자 객체를 가지며, 이 객체는 다른 영역인 그룹내의 통신객체(communication object)에 의해 호출된 관리 인터페이스 객체와 상호작용 하게된다. 또한, 객체그룹(이후, 분산객체라 부름)은 단일 객체와 같이 데이터와 메소드, 이를 조작하기 위한 오퍼레이션들로 캡슐화 되어있다. 네이밍 서비스와 위치서비스의 독립을 유지하기 위해 모든 객체들은 자신을 식별할 수 있는 유일한 객체 핸들(object handle)을 갖는다.

2.1 분산객체의 구조

본 연구의 기반이 되는 분산객체는 관리 인터페이스, 노드 관리자 객체, 복사객체, 보안 객체, 통신 객체 그리고 실질적인 서비스를 수행하는 연산객체로 구성된다. 또한 광역 분산객체로부터 지역 분산객체를 호출하는 사용자 정의(user-defined) 인터페이스인 외부 인터페이스와 지역 내 분산객체들간의 상호작용을 위한 내부 인터페이스로 구성된다. 아래 (그림 1)은 분산객체의 구조의 구성요소와 그들간의 인터페이스를 나타내고 있다.



(그림 1) 분산객체의 구조

2.2 분산객체의 구성요소

하나의 분산객체는 다른 주소공간들에 위치한 지역객체들로 이루어진다. 이들 구성요소들인 각각의 객체들은 클라이언트인 사용자로부터 객체의 상태인 분산, 중복 및 이주의 투명성을 제공하고 있다. 이러한 분산 투명성을 제공하기 위해 각 구성요소인 객체들은 다음과 같은 기능을 수행한다.

● 관리 인터페이스 객체(Management Interface Object : M/I Object)

외부의 분산객체로부터 다른 분산객체 내의 연산객체에 접근 할 때, 먼저 관리 인터페이스의 컨택주소를 통하여 노드 관리자 객체에게 서비스 요청한다. 이 때, 관리 인터페이스는 외부 객체의 서비스 요청을 노드 관리자 객체에게 보내기 전에, 서비스를 요청한 객체가 서비스에 대한 이용 가능여부를 보안 객체를 통하여 확인하고, 그 결과에 따라 서비스 요청을 노드 관리자 객체에게 전달한다. 그 밖의 기능으로는 분산객체의 생명주기 서비스 요청 및 객체에 대한 상태 정보의 유지 기능을 제공한다.

● 노드 관리자 객체(Node Manager Object)

관리 인터페이스 객체로부터 전달된 요구에 따라 요청된 기능을 수행한다. 요청된 기능들은 분산객체의 생명주기 서비스와 객체에 대한 정보 변경과 지정한 객체에 대한 보안 조건의 변경을 보안 객체를 통해서 수행한다. 또한 복사객체와 연산객체와의 인터페이스를 제공하고 있다. 특히 연산객체가 여러 영역내에 둘 이상의 동일한 형태인 중복객체들로 존재한다면 통신 객체와 상호작용을 가진다.

● 보안 객체(Security Object)

분산객체 내 각 연산 객체들을 사용하기 위한 보안 규칙을 관리한다.

● 복사 객체(Replication Object)

분산객체가 포함하고 있는 복사된 객체(replica), 즉 다른 주소공간들 내에 존재하는 둘 이상의 동일한 중복된 연산 객체들을 관리하고, 복제 일관성을 유지한다.

● 통신객체(Communication Object)

다른 주소공간에 존재하는 분산객체나 중복된 객체들과의 통신을 제공한다. 통신객체는 점대점 통신, 멀티캐스팅 또는 혼합된 통신기능을 제공한다. 이는 네트워크 프로토콜에 따라 통신기능이 의존할 수밖에 없다.

● 연산 객체(Computational Object)

실질적인 서비스를 제공하는 객체로서, 클라이언트로부터 서비스 요청을 받으면, 연산서비스를 제공하기 전에 보안 객체를 이용하여 요청 객체의 접근 가능 여부와 자신의 상태를 체크한 후, 연산서비스를 수행한다. 클라이언트 객체의 요청에 따라 최종 바인딩되는 목표객체로서 이 연산객체는

분산 또는 중복형태로 광역 주소공간에 존재하게 된다.

3. 분산객체간의 바인딩 서비스

3.1 기존의 바인딩 서비스의 한계성

현재 제공되는 바인딩 서비스(예, 도메인 네이밍 서비스)는 이름 기반의 비 중복 연산객체(이후 객체라 부름)의 식별자인 UUID(Unique User Identifier)를 제공하고 있고, 이 식별자는 객체의 위치정보와 독립적으로 유지될 수 없다. 다시 말해서 객체의 식별자가 곧 위치정보인 컨택주소를 의미하고 있어서 이동 객체나 중복된 객체들에 대한 이동위치나 중복위치정보를 나타낼 수 없다. 또한 현 네이밍 서비스 체계에서는 광역 주소공간에 존재하는 무수한 객체들에 대한 유일한 식별자를 부여한다는 것은 많은 어려움이 따르게 될 것이다. 결국 새로운 바인딩 서비스의 도입으로서 네이밍과 속성을 이용한 혼합서비스가 요구된다. 이러한 서비스는 중복된 동일한 객체들의 바인딩 전략을 필요로 하고 있다. 전자에서 언급한바와 같이, 이동객체나 중복객체는 하나의 객체 식별자인 UUID와 위치정보인 컨택주소가 여러 개가 될 수 있으므로 이들을 독립적으로 유지해야 한다. 결과적으로 <표 1>에서 나타낸바와 같이 기존의 네이밍[15]이나 트레이딩 서비스[16]로는 중복된 객체들이나 이동객체의 바인딩 서비스가 불가능하다. 현재 기존의 네이밍 서비스나 트레이딩 서비스에서는 비 중복객체들에 대한 바인딩 서비스만이 이루어지고 있다(○부분). 그러나 광역 객체 컴퓨팅 환경에서는 이동 또는 중복객체들에 대한 바인딩이 절대적으로 필요하다(×부분). 또한 현재 네이밍 서비스나 트레이딩 서비스의 특성상 중복객체 개념을 고려할 수 없는(-부분)도 바인딩 서비스를 지원할 수 있어야 할 것이다.

<표 1> 이름/속성 분류에 따른 네이밍/트레이딩 서비스 지원

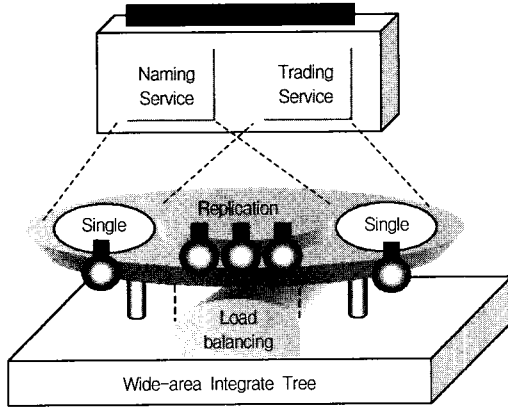
중복객체의 이름 및 속성 분류		바인딩 서비스 유형			
이름기반 객체1 : 객체2	속성기반 객체1 : 객체2	네이밍		트레이딩	
		비중복	중복	비중복	중복
이름 = 이름	전체속성 = 전체속성	-	×	-	×
이름 = 이름	전체 ≠ 전체	-	×	○	-
이름 = 이름	부분속성 = 부분속성	-	×	-	×
이름 = 이름	부분속성 ≠ 부분속성	-	×	○	-
이름 ≠ 이름	전체속성 = 전체속성	○	-	-	×
이름 ≠ 이름	전체속성 ≠ 전체속성	○	-	○	-
이름 ≠ 이름	부분속성 = 부분속성	○	-	-	×
이름 ≠ 이름	부분속성 ≠ 부분속성	○	-	○	-

따라서 네이밍/트레이딩 서비스를 통합한 새로운 바인딩 서비스 방안의 제안으로서 위 표에서 제시한바와 같이 (-, ×)부분들까지 제공함으로써, 단일객체나 중복객체에 대한 이름과 속성을 고려한 바인딩을 지원한다. 이러한 구조는 기

존의 네이밍 또는 트레이딩 서비스에서 제공하는 단일 객체를 선정하는 과정보다 양질의 서비스를 제공하는 객체들을 선택하여 바인딩할 수 있는 장점을 가진다. 또한, 중복객체들에 대한 바인딩 서비스의 요청 시, 분산된 중복객체들 중 한 객체의 선정과정으로 부하분배 전략을 추가시킬 수가 있다.

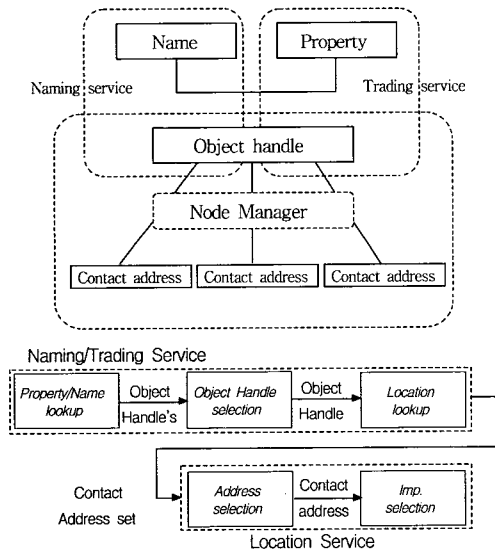
3.2 네이밍과 트레이딩 혼합서비스 구조 및 바인딩 과정

앞 절에서 제시한 문제점을 해결하고 <표 1>의 (-, X) 부분의 지원이 가능하도록, 우리는 광역 객체 컴퓨팅 환경기반에서 기존의 비 중복객체의 바인딩은 물론, 중복객체의 바인딩 서비스까지 지원할 수 있는 혼합서비스 구조를 (그림 2)와 같이 제시한다.



(그림 2) 네이밍과 트레이딩 혼합서비스 구조

위 구조에서 분산객체의 바인딩 서비스 과정을 기술하면, 분산객체간의 서비스 요청 및 결과 리턴을 위한 통신을 하는데 필수적인 처리과정은 객체들간의 바인딩 절차로 이루어진다. 기존의 네이밍이나 트레이딩 서비스는 객체의 위치



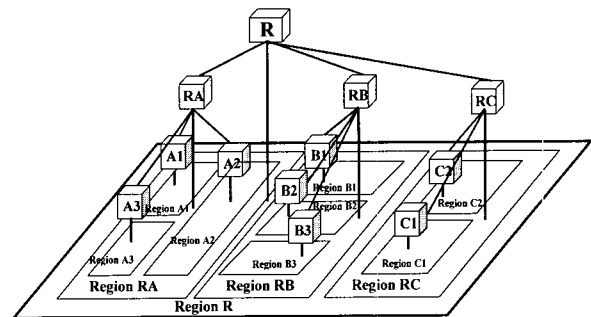
(그림 3) 이름 및 속성기반의 객체 바인딩 구조와 수행과정

가 변경이 되었을 경우, 변경할 이름이나 속성을 갱신해야 한다. 따라서, 기존의 서비스의 이름 대 객체의 UUID (= 컨택주소) 또는 속성 대 객체의 UUID (= 컨택주소)의 직접적인 매핑의 문제점을 해결하기 위해 다음 (그림 3)의 바인딩 구조로부터 바인딩 과정이 광역과 지역 두 단계로 이루어진다. 첫 번째 단계에서는 이름 또는 속성에 의한 단일객체 또는 중복객체를 탐색하고 이로부터 얻어진 하나 또는 그 이상의 객체들 중 요청 회수가 적은 바인딩 가능한 객체들을 선정한다. 둘째 단계에서는 선택한 객체들의 컨택주소들(중복객체인 경우, 여러 개의 컨택주소를 가질 수 있음)을 얻고, 노드 관리자를 통해 네트워크 통신비용이 가장 적고, 시스템 성능 대비 유휴처리능력이 큰 시스템 내의 연산객체에 대한 컨택주소를 선택한다.

4. 광역 통합트리 기반의 위치 서비스 알고리즘

4.1 광역 통합트리의 논리적 구조

인터넷 기반의 광역 객체 컴퓨팅 환경은 객체지향 기술을 접목하면서 많은 어플리케이션이 객체의 형태로서 서비스된다. 특히, 중복된 객체의 바인딩 서비스는 기존의 이름이나 트레이더를 활용한 속성기반의 객체탐색 및 관리 서비스에서는 객체들의 중복성 유지에 대한 어려움을 가지고 있다. 따라서 광역 객체 컴퓨팅 환경에서의 이름과 속성을 혼합한 서비스 메커니즘이 요구된다. 이를 해결하기 위해, 우리는 광역 객체 컴퓨팅 환경에서 중복객체들에 대한 바인딩의 필요성을 제시했고, 네이밍과 트레이딩을 혼합한 서비스와 위치서비스는 3.2절에서 언급한 바와 같이 독립적으로 수행하도록 해야한다. 이러한 서비스의 제공을 위해, 우리는 중복객체들의 독립적인 위치서비스를 제공하도록 광역 통합 트리 구조와 트리를 통한 위치 서비스 알고리즘을 본 장에서는 제안한다. 제안한 논리적 광역 통합 트리의 구조는 (그림 4)와 같으며, 구조 내의 디렉토리 노드들은 분산 탐색 트리를 사용하여 구조화된다. 각각의 노드는 광역 분산 시스템의 지리적, 관리적 영역(region)으로 세분화되며, 중간노드는 서브트리에 대한 모든 지역의 결함을 보이며, 본 트리의 루트 노드는 광역 서비스의 영역을 나타낸다.

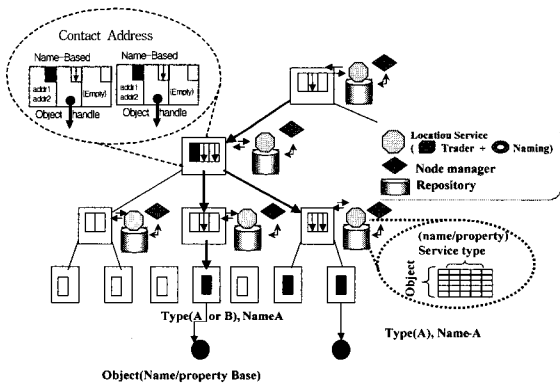


(그림 4) 광역 분산객체의 위치를 나타낸 논리적 광역 통합 트리

4.2 광역 통합트리의 물리적 구조

광역 통합 트리 기반에서 위치서비스는 분산객체의 유지 및 관리 시, 확장성, 위치성, 안정성, 허용결합성을 고려해야 한다. 특히 광역 객체 컴퓨팅 환경에서 분산객체의 유지 관리에 중복성과 객체들의 변환을 허용하는 이동성은 중요한 의미를 갖는다. 이름과 속성기반의 혼합 서비스는 클라이언트에게 단일객체에 위치투명성이 지원하지만 중복객체 및 중복된 객체의 이동성에 대한 투명성은 고려하지 않고 있다. 이를 해결하는 방법으로 객체핸들 개념을 이용하여 객체의 위치투명성을 제공한다. 그러나 객체의 이름과 속성의 혼합기반으로는 중복객체에 대한 위치투명성 및 이동성을 지원하지 못하고 있는 실정이다.

본 논문에서는 네이밍과 트레이더의 혼합 서비스를 이용하여 광역 객체 컴퓨팅 환경에서 이름과 속성기반의 중복객체에 대한 유지관리 및 투명성을 제공하기 위해서 위치 서비스의 기반이 되는 광역 통합트리의 물리적 구조는 다음 (그림 5)와 같이 제시한다. 제안한 모델 구조는 이름과 속성기반의 각각의 객체에 대한 유일한 객체핸들을 갖는다. 또한, 이러한 객체들의 중복여부에 따라 하나 이상의 컨택주소들을 가질 수 있다. 객체핸들은 객체에 대한 이름과 속성의 혼합서비스와 위치독립성을 제공하게 된다. 각 지역노드에는 컨택레코드(객체 객체의 위치정보인 Address와 forwarding pointer)를 포함한 광역환경의 분산객체의 추적을 위한 Look_up(Primitive Operation)을 제공하고, 노드매니저는 사용자편의의 이름과 속성 기반의 중복객체에 대한 상태의 변화 및 속성 변화에 대한 유지관리를 위한 Update(Primitive Operation)가 제공된다. 또한 각 지역노드의 레포지토리는 이름과 속성 기반의 객체상태정보를 저장하고 사용자가 원하는 최적의 객체선택에 필요한 정보를 포함한다.



(그림 5) 광역 통합 트리의 물리적 구조

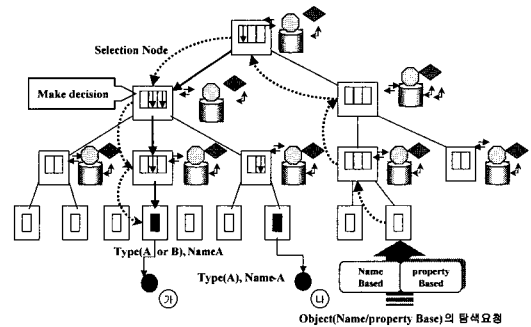
4.3 위치서비스 알고리즘

앞서 언급한 바와 같이 광역 객체 컴퓨팅 환경에서 이름과 속성 기반의 사용자가 요구하는 객체들에 대한 효과적인 컨택주소들의 탐색과 갱신은 DNS의 구성과 유사한 계층적인 지리적적, 위상학적 또는 관리상의 도메인과 같은

중요 광역 네트워크로 구성하였다.

4.3.1 광역 통합트리의 탐색

광역 객체 컴퓨팅 환경에서 객체에 대한 탐색은 이름과 속성기반으로 잎 노드에서 수행이 시작된다. 클라이언트가 원하는 객체가 소속된 노드에 존재하면 바로 객체에 대한 컨택주소를 얻어 탐색하게 된다. (그림 6)은 광역 객체 컴퓨팅 환경에서 시나리오 중 최악의 탐색방법으로 근 노드까지 클라이언트가 요청한 객체에 대한 특히, 중복객체(이름/속성)에 대한 최적 객체 선정까지의 탐색 과정을 나타낸 그림이다. 즉, 잎 노드에서 원하는 객체에 대한 위치정보를 얻지 못했을 경우, 근 노드에 이르기까지 look up이 수행된다. 그리고 근 노드로부터 다시 forwarding pointer를 따라 하위레벨로 탐색하고, 해당 객체가 중복되어 있을 경우, 이들 중 최적 객체를 선정하고, 해당 객체의 컨택주소를 얻는다. 이때 탐색할 객체가 중복되어 여러 위치에 있는 경우, 객체의 요청회수, 네트워크 비용 그리고 시스템의 유희 처리 능력 등의 파라미터들을 이용하여 부하분배 전략을 세워서 하나의 최적의 객체를 선정한다.

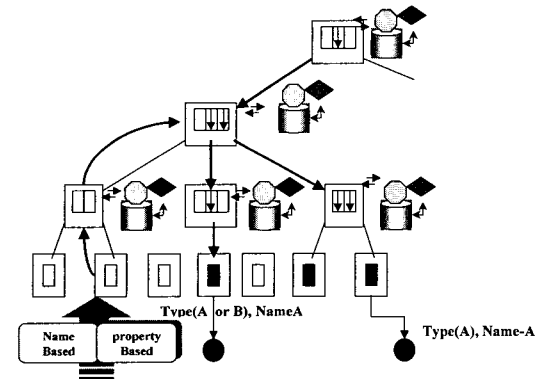


(그림 6) 광역 통합트리로부터 객체의 탐색과정

4.3.2 광역 통합트리의 갱신

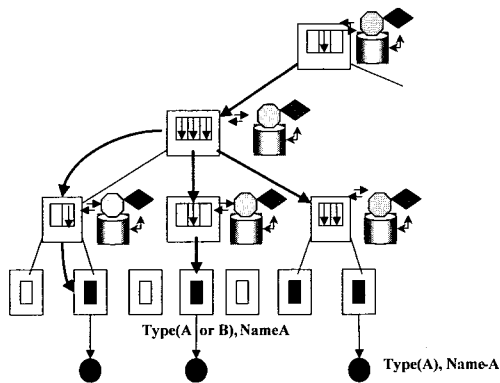
광역 통합트리의 갱신은 삽입과 삭제로 구분이 되며, 하위레벨(잎 노드) 상에서 이름과 속성 기반의 객체들을 추가하게 된다. 하위레벨의 객체 삽입 요청이 발생하면 잎 노드의 상위 노드에 객체 위치 정보를 표시하는 forwarding pointer를 등록하는 과정을 상위 노드로 반복 수행하며, 근 노드까지 위치 정보가 레포지토리(repository)에 등록이 된다. 다음 (그림 7)은 갱신의 삽입 알고리즘으로, 근 노드부터 잎 노드까지의 각 노드상에 추가를 원하는 객체에 대한 이름 및 속성의 등록정보를 위치서비스가 갖는 레포지토리에 저장하는 과정을 보인다.

또한, 삭제는 삽입과 마찬가지로, 하위레벨(잎노드)에서 수행되며, 잎노드의 이름/속성 기반의 컨택주소를 제거함으로써 수행된다. 그리고 상위 레벨의 이름/속성 트리의 일관성을 유지하기 위해, 컨택레코드가 이미 비어 있다면, 부모 노드는 해당 노드의 forwarding pointer를 레포지토리로부터 제거한다.



① Object(Name/property Base)의 Contact address 삽입요청

(A) 삽입요청



② Object(Name/property Base)의 contact address & Forward pointing추가

(B) 객체 삽입

(그림 7) 객체의 삽입 과정

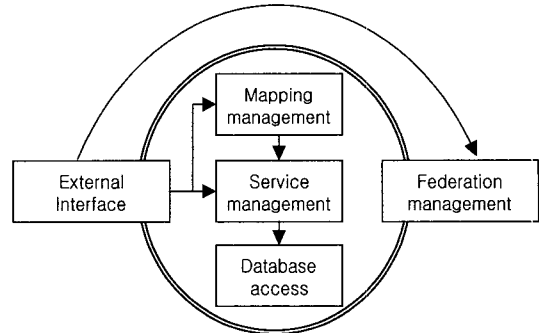
5. 광역 바인딩 서비스의 연합구조

광역 객체 컴퓨팅 환경은 이질적인 분산 시스템들이 모여 협력 형태로 변화되기 때문에 이들간의 연합은 필연적이다[17]. 기존의 연합방법으로 네이밍 서버들간의 연합 그리고 트레이더간의 연합에 대한 연구가 있으나 대부분이 네이밍 프로토콜 또는 트레이더 프로토콜을 기반으로 연합하는 형태이다. 그러나 광역 객체 컴퓨팅 환경에서 이루어지는 연합구조는 더 이상 기존의 단일 타입 모델을 공유하는 것이 아니라, 이질적인 모델을 지원해야만 한다. 따라서, 앞서 언급한 이름 및 속성 기반의 객체 바인딩 구조를 기반으로 본 연구에서 제시하는 연합 방법은 프록시의 개념을 추가한 이질적인 미들웨어 플랫폼에서 확장된 연합을 위해 Cooperator라는 중재자를 두어 분산객체와 혼합서비스 중간에서 상호작용을 돕는다.

5.1 Cooperator

Cooperator는 혼합 서비스의 proxy 기능으로 분산객체의 등록과 서비스 검색을 위한 기능을 제공하며, 로컬에 해당

객체가 없을 경우 다른 영역의 혼합 서비스의 정보 교환을 통하여 해당 객체의 레퍼런스를 얻도록 이들간의 중재자 역할을 하도록 한다. 또한, 교환되는 데이터의 변환과 바인딩 기능을 제공한다. 다음 (그림 8)은 Cooperator 모듈의 구조를 나타낸다.



(그림 8) Cooperator 모듈 구조

• 외부인터페이스(External Interface)

분산객체가 Cooperator에 초기 접속하는 오퍼레이션으로, 혼합 서비스의 импорт, 익스포트 그리고 등록과 삭제 오퍼레이션을 호출하기 위한 모듈이다. 이는 매핑 관리 모듈, 서비스 관리 모듈과 상호작용한다.

• 매핑 관리(Mapping management)

각 영역에 위치한 Cooperator간의 교환되는 정보를 변환하기 위한 모듈로 데이터 타입이 상이한 경우, 관련 혼합 서비스에서 제공하는 импорт(import) 오퍼레이션에 맞추는 기능을 제공한다.

• 서비스 관리(Service management)

혼합 서비스의 연결 정보 리스트에 새로운 영역의 혼합 서비스의 서버를 추가하거나 삭제하기 위한 오퍼레이션을 제공한다.

• 연합 관리(Federation management)

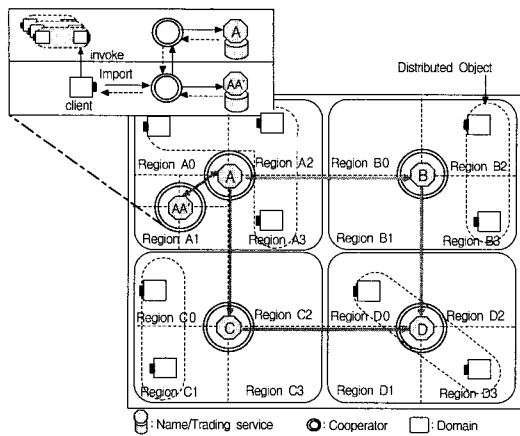
다른 영역의 Cooperator에게 импорт 요청을 전달을 처리하기 위한 오퍼레이션을 제공한다.

• 데이터베이스 관리(Database Access)

연결된 Cooperator 액세스 정보를 관리하기 위한 모듈로, Cooperator의 레퍼런스 리스트를 처리한다.

5.2 연합구조

광역 바인딩 서비스를 위한 연합구조는 광역 통합 트리 기반으로 영역별 혼합 서비스와 Cooperator가 위치하며, Cooperator간의 정보 교환은 혼합 서비스가 관리하고 있는 분산객체들의 객체 핸들에 대한 정보를 전달함으로써 이루어진다. 다음 (그림 9)는 이에 대한 연합구조를 나타낸다.



(그림 9) Cooperator를 통한 혼합 서비스의 연합구조

위 (그림 8)에 나타난 바와 같이 광역 통합 트리 기반으로 한 도메인 A, B, C, D로 나뉘며, 또한 세부적으로 4부분의 서브 영역을 갖는다. 이러한 영역은 혼합 서비스가 분산 객체들을 관리하고 있는 논리적인 영역과 같다. 여기에 Cooperator들은 각각의 영역에 혼합 서비스와 같이 위치하며, 분산객체를 구성하는 서비스 객체들의 상호접속을 지원한다. 임의의 영역 A1의 클라이언트가 서비스 객체를 탐색하기 위해 Cooperator에게 임포트 요청을 하면, 해당영역의 혼합 서비스를 통해 임포트 명세에 맞는 서비스 객체를 찾게 된다. 없을 경우, 연결된 Cooperator에게 임포트 요청을 전달하여, 탐색 범위를 넓혀 관련 정보를 교환함으로써 클라이언트의 임포트 요청에 가장 적합한 분산객체에 대한 레퍼런스를 얻는다. 이러한 임포트 요청은 사용자가 Cooperator의 함수를 지정함으로써 제한하도록 한다.

우리가 제안하는 연합구조는 이질적인 모델을 제공하기 위해 Cooperator를 통하여 혼합서비스가 관리하는 분산객체에 대한 정보를 얻을 수 있어, 서비스 선택의 폭을 넓히고, 서비스의 질을 향상시킬 수 있다.

6. 결 론

인터넷 기술의 발전에 따라 시스템 규모가 점차 커지고, 중앙 집중식 시스템에서 분산시스템들의 협력 형태인 광역 연합환경의 분산시스템으로 변화되고 있다. 더욱이 이러한 컴퓨팅 환경은 객체 지향 기술의 접목으로 광역 객체 컴퓨팅 환경을 새로이 만들고 있다.

현재 인터넷 기반의 웹서비스와 전자메일과 같은 광역 분산 서비스들이 클라이언트에게 제공되고 있지만, 국부적인 분산 투명성을 제공하고 있다. 특히, 이러한 분산 투명성을 제공하는 솔루션으로 네이밍 서비스가 대부분을 차지하고 있다. 이러한 이름만으로는 광역 환경에서는 클라이언트의 요구에 맞는 서비스를 제공하기엔 한계가 있다. 이를 해결하기 위해 속성 기반의 트레이딩 서비스가 출현하게 되었다. 이러

한 서비스들이 기반이 된 광역 객체 컴퓨팅 환경에서는 수많은 객체들이 이름이나 속성에 의해 중복됨을 예측하게 한다.

따라서, 단일 객체뿐만 아니라 이름 또는 속성 기반의 중복된 객체들의 효율적인 관리와 시스템간의 부하균형화를 유지하며, 최적객체 선정을 통한 바인딩 서비스를 제공하고 자 한다. 이를 위해, 먼저 기존의 TINA의 객체그룹 개념과 Globe 시스템의 분산 공유 객체에 대한 분석을 통해 광역 객체 컴퓨팅 환경에 기반이 되는 분산객체를 모델링 하였다. 또한, 이를 기반으로 이들의 상호접속 바인딩 서비스를 제공하기 위해 중복객체에 대한 분류 방안을 제시하고, 이름과 속성 모두를 지원하는 네이밍과 트레이더 기능이 통합된 혼합 서비스와 실제 분산객체의 컨택주소를 관리하는 위치 서비스를 각각 독립적으로 수행하여 이름 대 주소의 문제점을 해결하였다. 또한, 지역적인 영역 개념을 도입한 광역 통합 트리 기반의 위치 서비스의 탐색, 갱신알고리즘을 기술하였다. 그리고 서로 다른 영역에 위치하는 분산객체들의 광역 바인딩을 제공하는 연합구조를 Cooperator 모듈을 통해 이질적인 모델을 지원하도록 하였다.

향후 연구로는 논문에서 제시한 내용을 근거로 성능 평가에 초점을 맞추고, 네이밍과 트레이딩 기능을 통합한 혼합 서비스의 구현과 광역 통합 트리를 기반으로 한 위치 서비스의 구현을 통해 분산객체간의 접속 시험을 하고, 이에 대한 최적 부하 성능 평가에 대한 연구가 진행되어야 한다. 또한, 분산객체의 탐색 경로의 최적화를 위한 연구가 필요하다.

참 고 문 헌

- [1] Wide-area systems research, <http://ringer.cs.utsa.edu/faculty/jon/wa.html>.
- [2] The GLOBE Project : <http://www.few.vu.nl/~steen/globe/>.
- [3] Object Management Group, "The Common Object Request Broker : Architecture and Specification, version 1.2," Technical Report 93.12.43, OMG, December, 1993.
- [4] DCOM, Technical Overview, Microsoft Corp./ Redmond, Wash., 1996.
- [5] INA-C, 1995, "Overall Concepts and Principles of TINA," TB_MDC.018_1.0_94.
- [6] M. Van Steen, "Scalable Naming in Global Middleware," Internal report IR-464, October, 1999.
- [7] M. van Steen, "Algorithmic Design of the Globe Wide-Area Location Service," The Computer Journal, 41(5) : pp.297-310, 1998.
- [8] Infospheres, <http://www.cs.caltech.edu/~adam/CALTECH/infospheres.html>.
- [9] Globus, <http://www.globus.org/>.
- [10] Berkeley's WebOS, <http://www.wizzo.demon.co.uk/webcom.html>.
- [11] Web computer, <http://www.wizzo.demon.co.uk/webcom.html>.
- [12] A. S. Grimshaw, Wm, Wulf, and the Legion Team. "The

legion vision of a worldwide virtual computer,” Communications of the ACM, 40(1), January, 1997.

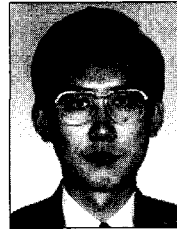
- [13] 김명희, 신경민, 이재완, 주수종, “분산처리환경에서 객체그룹간의 연산객체들의 상호접속”, 한국정보과학회논문지, 제25권 제8호, pp.815-824, 1998.
- [14] 이승용, 정창원, 신영석, 주수종, “개방형 분산 환경에서 객체그룹 모델의 설계”, 한국통신학회논문지, 제23권 제9호, pp.2258-2270, 1998.
- [15] DSTC : http://www.dstc.edu.au/Products/CORBA/Naming_Service/.
- [16] DSTC : Trader, <http://archive.dstc.edu.au/AU/projects/corba-trader/documents.html>.
- [17] 정창원, 주수종, “객체그룹간의 상호접속을 지원하는 연합 트레이더 모델”, 한국정보과학회논문지, 제26권 제9호, pp.1126-1134, 1999.
- [18] P. Homburg, M. van Steen, A. S. Tanenbaum, “An Architecture for A Scalble Wide Area Distributed System,” In Proc. Seventh SIGOPS European Workshop, Connemara, Ireland, ACM. To appear, Sept., 1996.



정 창 원

e-mail : mediblu@wongkwang.ac.kr
 1993년 원광대학교 컴퓨터공학과 졸업 (학사)
 1998년 원광대학교 컴퓨터공학과 졸업 (교육석사)
 1998년~현재 원광대학교 컴퓨터공학과 박사과정

관심분야 : 분산 컴퓨팅, 분산 객체 모델, 위치 서비스, 트레이딩 서비스 등



오 성 권

e-mail : ohsk@wongkwang.ac.kr
 1981년 연세대학교 전기공학과 졸업 (학사)
 1983년~1989년 금성산전연구소(선임 연구원)
 1993년 연세대 대학원 전기공학과 졸업 (박사)

1996년~1997년 캐나다 Manitoba대학 전기 및 컴퓨터공학과 Post-Doc.

1993년~현재 원광대 전기전자 및 정보공학부 부교수
 2002년~현재 대한전기학회(KIEE) 및 제어자동화시스템공학회(ICASE) 편집위원

관심분야 : 시스템자동화, 퍼지이론 및 신경회로망 응용, 계산지능 및 소프트웨어 등



주 수 종

e-mail : scjoo@wongkwang.ac.kr
 1986년 원광대학교 전자계산공학과(학사)
 1988년 중앙대학교 대학원 컴퓨터공학과 (석사)
 1992년 중앙대학교 대학원 컴퓨터 공학과 (박사)

1993년~1994년 미국 Univ. of Massachusetts at Amherst, 전기 및 컴퓨터공학과, Post-Doc.

1990년~현재 원광대학교 컴퓨터 공학과 교수
 관심분야 : 분산실시간 컴퓨팅, 분산 객체 모델, 시스템 최적화, 멀티미디어 데이터베이스