

## CORBA기반 지능형 에이전트에 관한 연구

장 일 동\*, 이 희 용\*\*

### A study of intelligent Agent Using CORBA

Il-Dong Jang\*, Hee-Young Lee\*\*

#### 요 약

본 논문에서는 에이전트 통신 언어(ACL)를 이용하여 분산된 데이터베이스에서의 에이전트 기반 모형을 제시한다. 즉 분산된 서로 다른 운영체제, 분산된 이종의 데이터베이스에서의 정보 공유와 교환이 가능한 CORBA를 이용한 지능형 상품검색 에이전트의 모형을 구현하였다.

최근 급속한 네트워크 기술의 발달로 데이터베이스 관리의 필요성이 제기되었고 이를 위해 CORBA와 같은 표준이 등장하여 분산환경에 사용되어졌다.

에이전트 응용은 여러 많은 분야 즉 전자상거래, 인터페이스, 정보 검색 등 다양하게 개발되고 있다. 사용자가 상품을 검색하고자 할 때 접속한 서버의 데이터베이스 내에서 먼저 검색하고 검색 상품이 없으면 다른 분산된 다수의 데이터베이스에 재 접속하여 검색하는 과정을 찾는 상품이 있을 때까지 반복한다. 이 과정을 사용자가 아닌 에이전트가 분산된 데이터베이스상의 지능형 에이전트와의 통신을 이용하여 수행한다.

#### Abstract

In this thesis, we propose an Intelligent Agent based on CORBA technology, which can exchange and share data under distributed computing environment using Agent Communication Language(ACL). Recently, due to the rapid development of network technology, the need for the database management appeared. So representative standard, CORBA have been applied and used under distributed environment.

Agent applications are being developed for various fields such as electronic commerce, user interface, information retrieval. When a user searches goods, he navigates several servers on different platform until he finds desired goods. The Intelligent Agent presented in this thesis, does these tedious navigations on behalf of the user through communication between different platform.

\* 부산여자대학 인터넷비즈니스 겸임교수  
\*\* 공학박사. (주)모바일넷 CEO

## I. 서론

인터넷의 대중화와 함께 네트워크 트래픽은 계속적으로 증가됨에 따라 과도한 트래픽의 영향으로 속도 저하로 호스트 중심의 중앙 집중식 방식이 한계를 가지게 되었다. 이러한 문제를 해결하기 위해 이종 시스템들 간에 프로그램을 분산시켜 부하를 줄임으로써 시스템의 성능 저하와 네트워크 병목현상을 해결하려는(1) 노력으로 OMG (Object Management Group)(2)에서 제안한 분산 객체 컴퓨팅(3)(4) 표준 구조인 CORBA(Common Object Request Broker Architecture)(5)(6)(7)에 초점을 맞추어 사용자 환경을 설계하고 구축하였다.

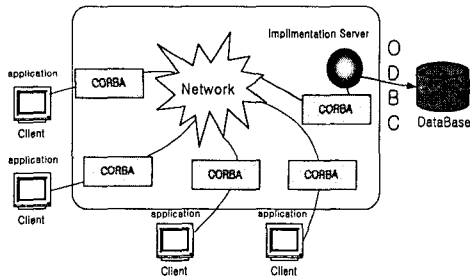


그림 2. CORBA기반 데이터베이스 접근 모델

네트워크 연결 및 관리에 대한 투명성, 다양한 시스템 호환성, 언어 독립성, 객체지향기술 등의 장점을 가진 미들웨어 CORBA는 본 연구에서 요구하는 속성을 거의 만족시키고 있으며, 다양한 플랫폼에 존재하는 데이터베이스로의 접근을 가능하게 한다(8).

그림 1은 3단계 방식으로 데이터베이스에 접근하는 방법으로(9)(10) 분산 객체 컴퓨터 환경에서 이기종 데이터베이스의 통합을 설계하고 구현하였다.

에이전트 기본 아키텍처는 ACL처리기와 분산 객체 환경인 CORBA를 이용하여 메시지를 전달하는 CORBA 객체로 구현하였다.

## II. 에이전트 시스템의 구성

### 2.1 구현된 에이전트 시스템의 설계

클라이언트/서버 구조는 한 쪽, 즉 프론트 엔드(front end)에는 최종사용자, 즉 클라이언트가 있고, 클라이언트가 다른 한 쪽, 즉 백 엔드(back end)에 있는 원격 컴퓨터를 말하며 본 논문에서는 클라이언트와 서버 사이에 다른 하나의 객체를 두어 3-tier 구조를 두었다.

2단계 방식을 적용하게 되면, 각 클라이언트 애플리케이션 마다 ODBC를 두어야 한다. 이는 서버의 DBMS가 변경되거나 서버 애플리케이션 변경 시 클라이언트마다 일일이 Driver를 변경시켜 주어야 하는 단점을 가지고 있다. 하지만 3단계 방식을 이용하게 되면 2계층에 ODBC를 두어 DBMS에 연결하는 Driver를 하나씩만 구현해 주는 된다.

즉, 기존의 분산 컴퓨터 방식이 상호 접속성(Interconnection)만을 보장하는 수준이었다면, 객체 지향 컴퓨팅 방식을 도입한 분산객체 컴퓨팅 기술은 나아가 상호 연동성(Interoperability)을 보장한다.

다시 말해 중계에이전트와 다른 원격 DataBase 서버와의 원활한 통신을 위하여 중계에이전트를 둬으로써, 각 서버들은 서로의 위치나 OS, DBMS의 종류에 관계없이 단지 중계에이전트와의 통신에만 신경을 쓰면 된다. CORBA 개발도구로 Visigenic사의 Visibroker(11)를 사용하였으며 구현 환경은 Client 에이전트 MS-SQL 데이터베이스 서버 에이전트, 중계 에이전트, InterBase 데이터베이스 서버 에이전트의 개발언어로는 델파이를 사용하였으며, 지역 데이터베이스는 MS-SQL, 리모트 데이터베이스 서버는 InterBase이고 운영체제는 Windows 2000 서버와 Windows 98에서 구현하였다.

### 2.2 전체적인 검색 에이전트의 구성

구현객체 측에서는 다음의 기능과 동작과정을 가지는 데, ID와 패스워드 입력 시 패스워드가 올바르게 입력되지 않는 경우 재 입력을 위한 메시지를 출력한다. 그리고 서버에 접근하기 위한 데이터베이스의 open과 close에

관한 메소드를 가진다. 서버는 두 개로 분리해 두었으므로 상품의 선택 시 해당 상품이 저장되어 있는 데이터베이스 서버로 접근하기 위한 적절한 드라이버를 선택하도록 하는 루틴이 정의되어 있다.

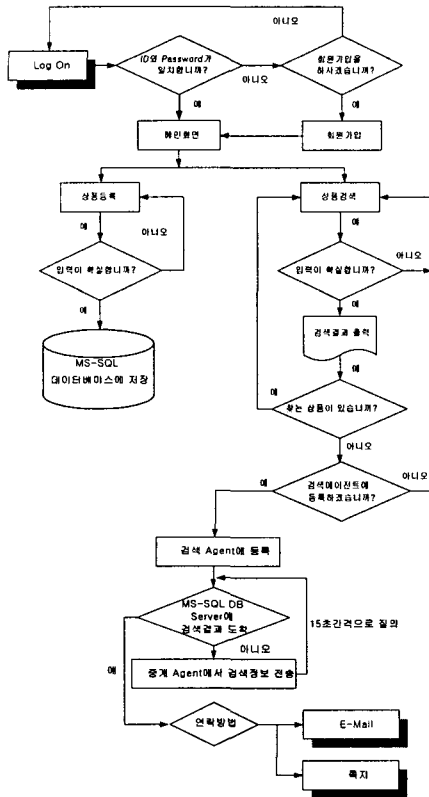


그림 2. 지능형 에이전트 흐름도

### 2.3 상품검색 클라이언트 에이전트의 구성

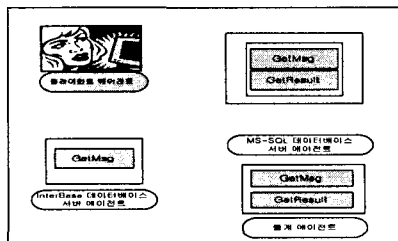


그림 4. CORBA를 이용한 상품검색 에이전트의 전체 구성도

사용자가 인터페이스를 통해 명령한 서비스를 서버쪽에 요청하는 것이 클라이언트의 첫 번째 임무이다. 클라이언트는 사용자의 지시에 따라 요청을 만들어서 서버에게 보내고 서버가 제공한 서비스를 적절히 사용자에게 보여준다.

상품중개 및 검색은 지리적으로 분산된 많은 DataBase 서버들이 존재하고 있기 때문에 서버들 간에 데이터의 교환이나 정보의 공유가 없기 때문에 상품 검색 입장에서는 많은 서버들을 찾아다니며 원하는 상품을 검색해야하는 시간적 낭비와 서버검색에 있어서 어려움을 겪게된다.

이러한 전체 환경의 통합과 협동적인 업무의 수행을 위해서는 각 전산 시스템간의 통신기능과 협의기능, 그리고 정보를 교환 공유할 수 있는 기능이 있어야 한다.

## III. 시스템의 구현을 위한 동작과정

### 3.1 MS-SQL DB서버

클라이언트에서 MS-SQL 데이터베이스 서버로의 구매등록, 판매등록, 물품검색은 MS-SQL Server 에 이전트의 CORBA객체의 함수에 생성되어있는 Stored Procedure를 사용하였다. 클라이언트가 MS-SQL 데이터베이스 서버에 물품검색 질의를 했을 때 원하는 물품이 서버에 있을 경우에는 질의에 해당하는 레코드를 Dataset으로 돌려주며 원격 인터페이스에 질의를 위한 필드별 데이터 생성 후 DivideMsg 프로토콜에 string에 맞게 자르며 tblsearch에 저장한다.

```
function TMSServerStub.GetMsg(const msg: WideString): WideString;
var
    OutBuf: IMarshalOutBuffer;
    InBuf: IMarshalInBuffer;
begin
    FStub.CreateRequest('GetMsg', True, OutBuf);
    OutBuf.PutWideText(PWideChar(Pointer(msg)));
    FStub.Invoke(OutBuf, InBuf);
    Result := UnmarshalWideText(InBuf);
end;
```

```
function TMSServerStub.getresult(const msg: WideString): WideString;
```

그림 4. 구현객체들을 실행시키기 위한 MS서버 측 모듈들 - Stub

구현객체를 호출하는 부분을 다시 세부적으로 살펴보면, 연결하고자 하는 데이터베이스 서버에서 접속하기 위한 객체 호출, 상품을 선택하여 데이터베이스 서버에 저장시키기 위한 객체 호출, 그리고 구매결과를 display하기 위해 화면상에 나타나야할 개인 ID와 구매물품을 되돌려 받기 위한 객체 호출, 마지막으로 연결된 데이터베이스 서버와의 접속을 중단하기 위한 객체 호출 등으로 구성된다.

```

procedure TMServerSkeleton.GetMsg(const
InBuf: IMarshalInBuffer; Cookie: Pointer);
.....
begin
msg := UnmarshalWideText(InBuf);
RetVal := FIntf.GetMsg(msg);
FSkeleton.GetReplyBuffer(Cookie, OutBuf);

OutBuf.PutWideText(PWideChar(Pointer(Retval))
);
end;
procedure TMServerSkeleton.getresult(const
InBuf: IMarshalInBuffer; Cookie: Pointer);

```

그림 5. 구현객체들을 실행시키기 위한 MS서버 측 모듈들 - Skeleton

일단 데이터베이스에 연결된 후 고객이 주문한 상품 데이터가 서버의 데이터베이스에 저장되기 위한 메소드가 사용되면, SQL문에 의해 각 고객별 테이블에 데이터를 저장한다. 또한 클라이언트 주문한 상품정보를 display하기 위한 메소드가 사용되어지며, SQL문에 의해 상품 테이블로부터 데이터를 가져온다.

### 3.2 중계에이전트 시스템 실행

본 논문에서는 분산된 데이터를 효율적으로 대처하기 위한 방안으로 지식 표현 능력을 가지는 에이전트 통신 언어(Agent Communication Language: ACL)와 분산 객체인 CORBA를 이용하여 분산된 시스템에서의 정보의 전달과 공유를 지원할 수 있는 에이전트 기반 시스템을 제시하고, 상품중계에 적용하여 분산된 환경에서 서로간의 정보를 공유하고 교환하는 중계에이전트 시스템을 구현하였다.

구현에 앞서 데이터베이스 연결성에 사실상 데이터베이스 서버의 부하를 줄인다는 측면을 고려할 때, 이의 해결을 위해 적어도 두 개 이상의 데이터베이스 서버를 운용함이 불가피하다는 것이다[12].

즉, 중계에이전트와의 통신에서 CORBA객체의 메소드를 이용함으로써, 실제로는 원격지에서 메소드를 호출하

지만 중계에이전트의 입장에서는 로컬의 메소드를 호출하는 것처럼 사용할 수 있다.

중계에이전트는 해당질의어를 각 데이터베이스 서버에 보내고, 각 데이터 베이스 서버는 string을 다시 질의어로 변환하여 지역 DBMS에게 질의를 한 후 결과 데이터셋을 protocol에 맞게 string으로 변환하여 중계에이전트에게 CORBA 객체의 메소드를 호출하여 데이터를 전달한다.

중계에이전트는 받은 메시지를 MS-SQL서버 에이전트에게 CORBA 객체의 메소드를 호출하여 데이터를 전달한다. 서버 에이전트는 클라이언트의 고객이 연락방법으로 선택한 문자메시지, 클라이언트 화면의 쪽지, E-Mail의 선택 사항에 따라서 검색 결과를 전송하게 되고 만약 검색 결과가 없을 시에는 일정시간 간격으로 중계에이전트와 통신을 통해서 검색 물품을 검색한다.

### 3.3. 원격(InterBase) Database server의 통신 과정

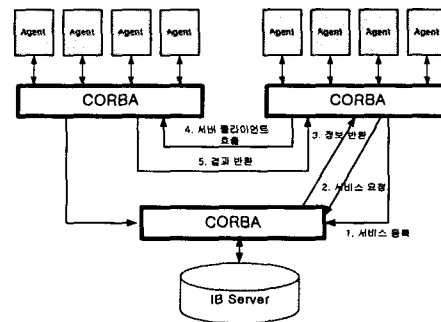


그림 6. 제안된 시스템의 통신과정

그림 6은 제안된 시스템의 통신 과정을 보여주는 것으로 원격 데이터베이스 서버가 많지만 본 논문에서는 InterBase 데이터베이스 서버만을 시범적으로 선택하여 구현하였다.

중계 에이전트에서 Remote DataBase Server로 메시지를 보내는 protocol은 다음에 대한 전체 구성도는 위 그림과 같으며 원격 데이터베이스 서버에이전트에 사용되고 있는 CORBA 객체는 IBase이고, method는 GetMsg가 있다.

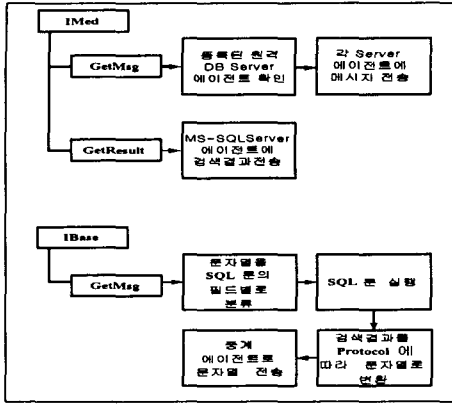


그림 7. 중계에이전트와 InterBase 데이터베이스 서버 에이전트

IV. 에이전트의 구현

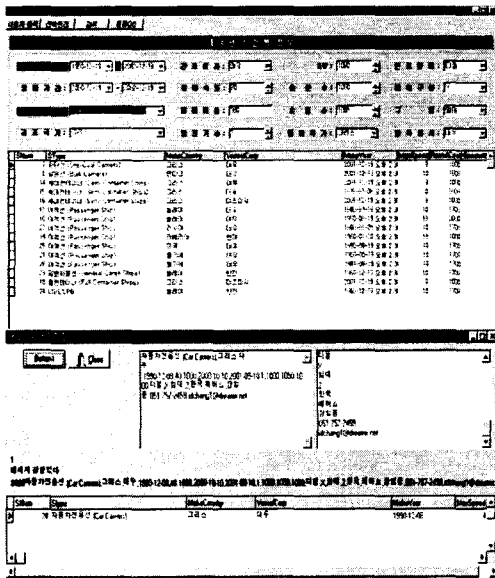


그림 8. 구현된 에이전트 통신과정 결과

그림 8은 IB Server에서 검색된 결과를 서버 에이전트로 보낸 결과이다. 서버에이전트에 등록된 상품이 없는 경우는 등록을 마치면 서버에이전트는 검색하며 요청한 상품을 없는 경우 다시 중계에이전트에 등록된 상품이 있는지 검색을 요청하고 다시 IB Sever에 검색을 요청한

다. 원하는 상품을 찾으면 다시 중계에이전트에서 서버 에이전트로 보내 선택한 쪽이나 E-mail로 검색된 내용을 보낸다.

즉 서버 에이전트에서 중계에이전트로의 데이터 전송이란 서버 에이전트 자체 DBMS에 질의 후 검색 물품이 있을 경우에는 검색결과를 클라이언트 에이전트에게로 전달해준다.

중계에이전트에서 서버 에이전트로의 데이터 전송은 중계에이전트는 Remote DataBase Server로 부터 전송된 메시지를 protocol로 CORBA객체 호출을 통하여 검색된 record들을 반환을 의미한다.

V. 결론

이제까지 분산 이기종 환경의 상품중계 검색 정보를 제공하기 위한 방법으로 CORBA기반으로 하는 시스템 설계 및 구현 결과를 살펴보았다.

본 논문에서는 다른 운영체제, 다른 데이터베이스에 있는 데이터의 효율을 극대화하기 위하여 서로의 데이터를 질의할 수 있는 에이전트의 모델을 설계하였다.

다른 데이터베이스, 운영체제, 다른 언어로 작성된 에이전트에서 중계에이전트에 등록을 하기 위해서는 중계 에이전트의 수정이 필요하다. 확장성을 위하여 중계 에이전트의 등록에 관련된 모듈의 개발도 필요할 것이다.

하지만 이러한 기대만큼 해결해야 할 과제들도 많다. 가장 우려되는 것이 보안과 사용자식별에 대한 문제이다. 이러한 문제점들이 보완된다면 전자상거래를 통한 에이전트의 역할이 더욱 증대될 것으로 기대하며 특히 이동전화와 PDA 등의 발달로 이동 컴퓨터를 지원해주는 이동 에이전트의 중요성은 더욱 커질 것이다. 이밖에도 계획이나 학습 등을 위한 지식의 보강으로 좀 더 지능을 갖춘 에이전트의 개발이 기대된다.

이러한 연구의 성과를 정리하여 보면 다음과 같다.

첫째, CORBA를 사용함으로써 데이터 레코드, 데이터 타입, 필드에 상관없이 실용적인 검색이 가능하도록 구현하였다.

둘째, 운영체제, 데이터베이스에 무관하게 등록과 검색이 가능하도록 CYBER 시장을 활용하였다.

셋째, 클라이언트/서버 기반으로 통합된 중계에이전트

시스템을 제안하고 구현하였다

앞으로 언어의 투명성 즉 중계 에이전트가 다국적 언어 번역기를 갖춘다면 언어의 장벽을 없앨 수 있을 것으로 기대하며 나아가 InterBase 서버에 웹서버를 접속시킨다면 웹상에서 검색과 등록이 가능할 것으로 기대한다.

## 참고문헌

- [1] David Perkins, Evan McGinnis, "SNMP, SNMPv2, SNMPv3, and RMON 1 and 2 : William Stalling's, 3ED, Addison Wesley, 1999
- [2] S.Vinoski, "CORBA: Intergration Diverse Applications Withn Distributed Hetergeous Environments", IEEE Communications Magazine, vol.14, Feb 1997
- [3] K. Mani Chandy Adam Rifkin, Systematic Composition of Objects : Processes and Sessions, Oxford University Press Computer Journal, Volume 40, Number 8, pp 465-478, October 1997
- [4] K. Mani Chandy, Joseph Kiniry, Adam Rifkin, and Daniel Zimmerman, A Framework for Structured Distributed Object Computing, Parallel Computing, Volume 24, Number 12-13, pp. 1901-1922, November 1998
- [5] George Couloursi, Jean Dollimore, Tim Kindberg, Distributed Systems: Concepts and Design. 2nd Edition, Addison Wesley, 1994.
- [6] OMG, "The Common Object Request Broker Architecture and Specification : Revision 2.0", OMG Document 1995.
- [7] Zhonghua Yang, Keith Duddy, "Distributed Object Computing with CORBA", DSTC Technical Report 23, 1995.
- [8] Evaggelia Pitoura, Omran Bukhres, Ahmed

Elmagarmid, "Object Orientation in Multidatabase System", ACM Computing Surveys, June 1995

- [9] 왕창중, 이세훈, CORBA 프로그래밍, 1th, 대림, 1998: pp8-22.
- [10] Robert Orfali, Dan Harkey, Client/Server Programming with JAVA and CORBA, 2thed . USA, John Wiley & Sons, 1998.
- [11] <http://www.inprise.com/visibroker>, Visibroker.
- [12] Silvano Maffeis, Douglas C. Schmidt, Construction Reliable Distributed Communication Systems with CORBA, IEEE Communications Magazine, Vol. 14, No. 2, 1997

## 저자 소개



### 장 일 동

1990 동의대학교 전산통계학과  
이학사

1993 경성대학교 전산통계학과  
이학석사

1998 한국해양대학교 산업공학  
박사수료

1997~현재 부산여자대학 인터  
넷비즈니스과 겸임교수

연구분야 : 데이터마이닝, 데이터  
베이스, 전자상거래, 지능  
형 에이전트

### 이 회 응

2002.9 한국해양대학교경영과  
학공학박사

2002.12~현재 (주)모바일넷  
대표

연구분야 : Ship scheduling,  
VRP, Marine GIS,  
ECDIS/ECS

