

# 실시간 CORBA의 우선순위 모델 구현 및 성능분석

## An Implementation and Performance Analysis of Priority Model of Real-Time CORBA

“박 순례”, 정 선태\*\*

\* 영보 시스템(Tel : 82-2-780-1728 FAX 82-2-780-1850 ; E-mail : psunr@hanmail.net)

\*\* 숭실대학교 정보통신전자공학부(Tel : 82-2-820-0638; Fax : 82-2-817-5987 ; E-mail : cst@syccon.soongsil.ac.kr)

**Abstract :** The Current CORBA has many drawbacks to be deployed successfully in real-time system applications. Recently, OMG adopted Real-Time CORBA specification. In this paper, we report our efforts on an implementation of Priority Model of Real-Time CORBA spec., which is one of the most important components in Real-Time CORBA spec. The improvement of real-time performances of our implementation is verified by experiments.

**Keywords :** Real-Time CORBA, CORBA, Real-Time Systems, Priority Model, Real-Time ORB

### 1. 서론

CORBA(Common Object Request Broker Architecture)는 OMG(Object Management Group)에서 규정한 미들웨어 표준안으로써, 객체 지향 방식(OOD/OOP)이 갖는 장점을 지원하면서, 플랫폼과 사용 프로그래밍 언어에 독립적인 미들웨어 환경을 지원하므로, 현재 분산 컴퓨팅 환경의 미들웨어 표준으로 인기를 모으고 있다[1]. 더군다나, CORBA 가 제공하는 표준 통신 방식인 동기 호출은 클라이언트/서버의 ‘request/response’ 문맥에 잘 부합되어 최선(best efforts)의 성능을 기대하는 분산 환경 용용에 성공적으로 이용되어 왔다. 산업용 네트워크 시스템에서도 이제, 컴퓨팅 능력의 증가 및 용용 개발의 증가로 편리한 미들웨어의 필요성이 증대되고 있다. 그러나, 현재의 CORBA (버전 2.3)은 처음에 비즈니스환경의 클라이언트/서버 분산 컴퓨팅 환경 구축을 염두에 두고 주로 개발되어 왔기 때문에, 실시간성 지원이 부족하다. 현 CORBA에서 클라이언트의 호출과 서버의 요청 처리가 FIFO로 이루어지므로, 긴급한 태스크/통신 이 우선적으로 처리되지 못한다. 따라서, 실시간성 지원이 필요한 분산 시스템의 구축에서 태스크는 제시간에 처리하지 않으면 성능의 열화를 가져오는 경우가 많다(예를 들어, 센서로부터 데이터를 획득하고 이를 이용하여 단말을 제어하는 경우). 이에 최근 OMG에서는 실시간 CORBA 규약을 발표하였다[2]. 실시간 CORBA 규약은 (비실시간) CORBA 의 확장으로 명세 되어 있으며, 그 목표는 ‘end-to-end predictability’ 지원하는 CORBA ORB 구현을 위한 표준을 제공하는 것이다. 실시간 CORBA 규약에서 말하는 ‘end-to-end predictability’ 는 다음을 의미한다[2].

- 1) CORBA 호출 처리 도중, 클라이언트와 서버간에 자원을 경합하는 쓰레드들의 우선순위를 계속 존중하는 것이다. 예를 들어, 우선순위가 높은 클라이언트 쓰레드의 호출을 서버에서도 존중하여 높은 우선순위로 처리하여 준다.
- 2) ‘end-to-end processing’ 도중 쓰레드들의 우선순위가 역전되는 기간이 제한 되도록 한다.
- 3) 서버 메소드 호출 처리에 걸리는 시간이 제한되도록 한다. 이 ‘end-to-end predictability’ 지원을 위해 실시간 CORBA는

우선순위 모델, 통신 프로토콜 구성, 쓰레드 관리 등을 지원하는 규약을 표준화하고 있는 데, 이 중 가장 중요한 아이디어는 ‘우선순위 모델’ 지원이다. 본 논문은 실시간 CORBA 의 ‘우선순위 모델’을 설계/구현하고 성능 분석을 한 연구결과를 보고한다. 우선순위 모델 구현은 오픈 소스인 비실시간 ORB omniORB2 (v3.0.0)[3] 을 기반으로 확장한 형태로 수행하였다. 구현된 우선순위 모델은 성능 분석을 위해 서버 객체 메소드의 반복적 호출을 통해 비실시간 ORB 의 경우와 1) 평균 latency (평균 메소드 호출 출처 시간) 2) 저터( 메소드 호출 처리시간의 표준편차로 측정) 등의 실시간 QoS 측정을 비교 실험하였으며, 분석 결과 본 논문에서 구현된 우선순위 모델 실시간 ORB 가 높은 우선순위의 경우, 우수한 실시간 QoS (latency, jitter 등) 성능을 보여 줍니다.

현재 실시간 CORBA 명세는 미국 워싱턴 대학의 고성능 ORB 인 TAO[4]에서 일부 구현하고 있으나, 우선순위 모델을 포함한 포괄적 구현은 보고되어 있지 않으며. 또한 미국 OIS 사가 실시간 ORB 인 ORB express를 출시하고 있으나[5], 자세한 내용 설명이 공개되어 있지 않아 실시간 CORBA 규약에 의한 실시간 ORB 구현인지는 아직 명확하지 않다.

### 2. 실시간 CORBA 의 우선순위 모델

#### 2.1 CORBA 동작 구조

CORBA 구조에 대한 자세한 내용은 [1]을 참조하기로 하며, 여기서는 본 논문의 이해에 필요한 CORBA 의 동작 내용만을 간단히 기술한다. CORBA의 동작은 ORB(Object Request Broker), 객체 어댑터(POA; Portable Object Adapter), 객체 레퍼런스(IOR : Interoperable Object Reference) 등을 중심으로 이루어진다. ORB 는 클라이언트의 서버 객체메소드 호출 요청을 해당 서버를 찾아 전달하며, POA 는 서버 측에서 클라이언트의 메소드 호출을 해당 구현 객체 메소드로 매핑 한다. CORBA 공간에서 서버 객체는 IOR(Interoperable Object Reference) 로 유일하게 식별된다. 따라서, 서버 객체 메소드를 호출할 수있기 위해서는 클라이언트는 서버 객체의 IOR를 알아야 하며, 이를 통