

# MOM 기반 미들웨어 XGM의 개발에 관한 연구

최준기\*

## 요약

대용량의 자료 처리를 위해서는 확장성이 용이한 MPP(Massively Parallel Processor) 컴퓨터가 유리하다. 그러나, 이 시스템에 대한 전문지식이 없이도 쉽게 필요한 업무를 구현하기 위해서는 미들웨어가 필수적이다. 본 논문에서는 개발하고 있는 미들웨어인 MOM(Message Oriented Middleware) 기반의 XGM의 설계 과정과 구현 과정을 소개하고자 한다. 그리고, 실험을 통하여 그 우수성을 입증하였다.

## A Study on Developing of MOM based Middleware XGM

Joon-Kee Choi\*

## ABSTRACT

Scalability for handling large amounts of data, MPP (Massively Parallel Processor) computer is more advantageous. However, this system easy to use without specialized knowledge about the work required to implement the middleware is essential. In this paper, It is developing of a MOM (Message Oriented Middleware) based middleware XGM. Design and implementation of the XGM process is presented. And, prove that proposed system is the superiority of this work.

Key Words : Middleware, MOM, XGM, MPP, Credit Card

---

\* 인덕대학 비서과(✉jkchoi@induk.ac.kr)

· 제1저자(First Author) : 최준기 · 교신저자(Correspondent Author) : 최준기

· 접수일(2010년 7월 29일), 수정일(1차 : 2010년 9월 6일), 게재확정일(2010년 9월 10일)

## I. 서 론

21세기에 들어와 인터넷이 보편화되는 등 정보의 양이 급증함으로써, 이를 처리해야할 시스템의 성능 개선에 대한 요구도 끊임없이 증가되어 왔다. 또한 정보의 활용 형태도 다양해짐으로써 과중한 부하가 걸리거나 빠른 응답시간을 요하는 곳에서 단일프로세서로는 아무리 성능이 개선되어도 한계에 부딪치고 있다. 그래서, 끊임없이 연구되어온 분야중의 하나가 다수의 프로세서를 탑재하여 성능 개선을 꾀하는 병렬 처리 컴퓨터이다[1][2].

병렬처리 컴퓨터 즉, MPP 컴퓨터는 따라서 메모리의 공유를 효율적으로 지원해야 한다. 또한 선형확장, 상호간 유연성 그리고 관리에 이르기까지 설계가 잘 되어있다면 많은 장점을 가지고 있기 때문에, 앞으로의 발전 가능성도 크다고 할 수 있다. 이러한 하드웨어 구성의 추이에 따라 더욱 필요해지고 있는 소프트웨어가 미들웨어이다. 미들웨어는 응용프로그램과 운영체제 사이에 존재하여 사용자가 전문적인 시스템 지식 없이도 이질적인 통신 프로토콜에 독립적이고 컴퓨터의 구조, 운영체제 및 데이터베이스와 연동하여 응용 프로그램을 시스템 환경에 적합하게 구축할 수 있도록 해주는 소프트웨어이다[3][4][5][6].

본 논문에서는 개발중인 MOM 기반 미들웨어인 XGM(neXt Generation Middleware)의 개발 및 구현 과정을 소개하고자 한다. 우수한 기술력을 바탕으로 중소기업에서 개발중인 미들웨어지만 중소기업이 갖고 있는 경제적인 문제로 어려움을 겪고 있는 실정이다. 2장에서는 MOM 기반의 미들웨어에 대해 살펴보고 3장에서는 XGM 설계 및 구현에서 성능평가까지 논하고 4장에 결론을 맺는다. 이 과정을 통하여 토종 미들웨어로 개발중인 XGM이 좀 더 알려져서 중소기업의 애로사항들이 해결되어졌으면 하는 바램이다.

## II. MOM 기반의 미들웨어

미들웨어의 역할은 개발된 응용프로그램이 어떤 시스템 환경하에서도 동작할 수 있도록 도와주는 일이다. 미들웨어는 기술적 요소에 따라 여러 가지로 분류할 수 있다.

RPC(Remote Procedure Call)은 클라이언트/서버 모델에서 원격지의 컴퓨터에 있는 프로세서를 실행시키고 그 결과를 받아볼 수 있도록하는 프로토콜이다. 따라서 이것의 특징은 동기(Synchronous)와 블로킹(Blocking)이라고 할 수 있다. ORB(Object Request Broker)는 객체들 간의 클라이언트/서버 관계를 맺어주는 미들웨어이다[7][8]. 클라이언트는 서버객체에 있는 메소드를 그것이 같은 컴퓨터에 있든, 또는 네트워크 상에 있든 상관없이 투명하게 호출할 수 있다. 그림 1에서 처럼 ORB는 독립적이고 유연성이 있으며, 이질적인 분산 환경에서, 서로 다른 컴퓨터 내에 있는 응용프로그램 간의 상호 운용성과 다중 객체시스템들에 대한 상호연결성을 제공한다.

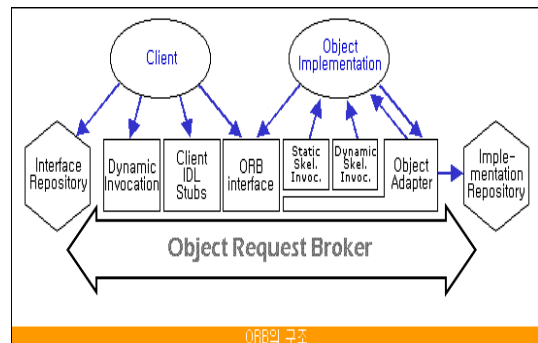


그림 1. ORB의 구조  
Fig. 1. Structure of ORB

MPP시스템에서 메시지 전달을 효과적으로 지원하는 메시지 지향 미들웨어(MOM)는 메시지를 큐(Queue)라고 불리는 전달 중계소에 넣어 처리하고 큐에 의한 메시지 관리 기능을 제공하며 메시징 기술을

기반으로 메시지를 송·수신 함으로써 신속하고 신뢰성 있는 메시지 전달, 비동기 통신을 쉽게 수행할 수 있다[9]. 그러나, 기존의 메시지 지향 미들웨어(MOM)는 시스템의 부하가 편중되어 병목 현상이 불가피하고, 특정 환경에서 운영되고 있어 사용자의 전문성이 필요하다는 단점이 있다.

TPM(Transaction Processing Monitor)은 데이터베이스에 접속된 많은 연결들로 인한 부하를 줄이기 위해 설계되었다[10]. 만일 에러가 발생하면 적절한 조치를 취하고 트랜잭션 처리를 보장한다. 아키텍처의 성격, 시스템의 API 등에 의해서 다양하지만 데이터베이스에 대하여 독립적인 트랜잭션 프로세서 구현에 중점을 두고 있다. TPM의 기능이 단순한 다중 송신 및 라우팅 기능 이상의 의미로 발전하고 있다.

본 논문에서 제시하는 MPP 환경하에서의 메시지 지향 미들웨어(MOM)는 시스템의 무정지성을 보장하는 모니터 레이어(Monitor Layer), 메시지 라우팅 및 큐를 관리하는 메시지 라우팅 레이어(Message Routing Layer), 메시지 변환과 프로토콜 및 타이머를 관리하는 네트워크 인터페이스 레이어(Network Interface Layer), 업무 로직 구현에 편리한 다이나믹 프레임(Dynamic Frame)과 다양한 API, 각종 라이브러리로 구성된 애플리케이션 레이어(Application Layer), 시스템 제어 운영, 상태 모니터링, 로그메시지 관리 등을 GUI 및 Web 기반 환경에서 관리하는 관리 레이어(Administration Layer)로 구성되어 있다. 즉, 기능별 독립성이 보장된 레이어드 구조(Layered Architecture)로 되어 있으며 시스템 구축 시 기간 및 비용 절감, 편리성, 안정성 등을 제공할 수 있다.

### III. MOM 기반의 미들웨어 XGM 설계 및 구현

#### 3.1 구조 및 기능

5개의 레이어로 구성된 XGM의 구조는 다음 그림 2와 같다.

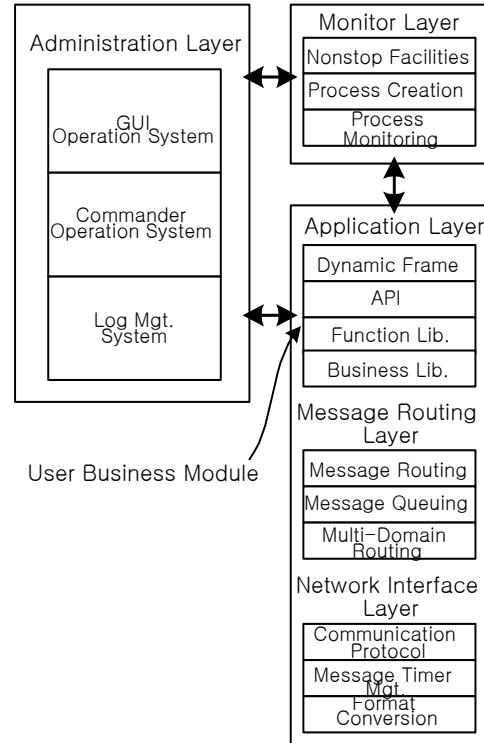


그림 2 XGM의 구조  
Fig. 2 Structure of XGM

그림에서 사용자 비즈니스 모듈(User Business Module)은 애플리케이션 레이어 바로 아래에 위치하고 있다. 시스템 전체의 무정지성을 보장하는 모니터 레이어는 논스톱 기능(Nonstop Facilities), 프로세스 생성(Process Creation), 프로세스 모니터링(Process Monitoring)의 세 기능으로 구성되는데 논스톱 기능은 HP와 같은 논스톱 아키텍처를 지원하면서 백업을 위한 옵션들을 제공한다.

그리고 그림 3의 상태 전이도에 따라 응용 프로세스 생성 및 시작(Startup) 메시지 전송, 옵션을 제공한다.

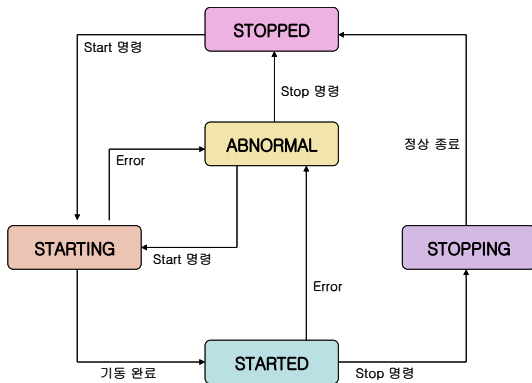


그림 3. 프로세스 상태 전이도  
Fig. 3. Process State Diagram

또한, 프로세스 상태 모니터링에 따라 응용 프로세스의 감시 기능을 제공하고 감시대상 프로세스들을 생성한다.

애플리케이션 레이어는 다이내믹 프레임, API, 기능 라이브러리(Function Library), 비즈니스 라이브러리(Business Library)의 네 가지로 이루어져 있다. 다이내믹 프레임은 사용자가 업무처리 코딩을 쉽게 할 수 있도록 다양한 프로그램 언어별 프레임웍을 지원한다. API는 업무 개발 시 시스템과의 인터페이스가 용이하도록 그림 4와 같이 모듈별로 함수를 제공한다.

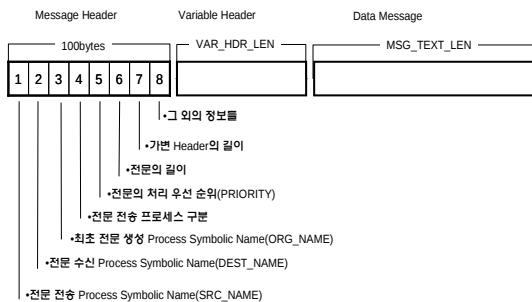


그림 4. XGM 메시지 구조  
Fig. 4. Structure of XGM Message

여기서, 메시지 헤더(Message Header)는 정보 검출 및 셋팅(Setting) 기능을 제공한다. 기능 라이브러리

는 업무구현시 유용한 기능들을 라이브러리 형태로 제공하는데 처리시간 관리시 필요한 타이머 모듈(Timer Module), 년,월,일별 로그 파일을 생성하고 오래된 로그 파일은 자동으로 삭제관리하는 로그 파일 모듈(Log File Module) 그리고 한글변환, 에디팅 모듈, 암호/복호 모듈 등을 제공하는 유틸리티 모듈(Utility Module)로 구성된다. 비즈니스 라이브러리는 구축하고자 하는 업무를 패키지 형태의 라이브러리로 제공한다. 이것은 개발기간을 단축시키는데 도움을 준다.

메시지 라우팅 레이어는 메시지 라우팅(Message Routing), 메시지 큐잉(Message Queuing) 그리고 멀티 도메인 라우팅(Multi-Domain Routing) 기능으로 구성된다. 메시지 라우팅은 같은 도메인 내의 메시지 전달을 담당하고 단일 메시지 라우팅(Single Message Routing), 블록 메시지 라우팅(Block-Message Routing), 브로드캐스팅 메시지 라우팅(Broadcasting Message Routing), 추적 메시지 라우팅(Trace Message Routing)의 4가지 전달 방식을 제공한다. 그림 5는 이들 중 추적 메시지 라우팅의 구성도를 보여준다.

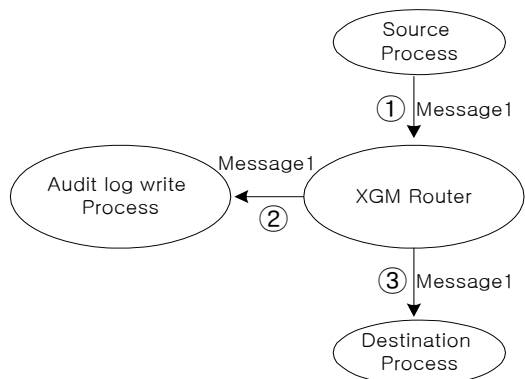


그림 5. 추적 메시지 라우팅 구성도  
Fig. 5. Diagram of Trace Message Routing

메시지 큐잉은 개별업무 프로세스 큐(Queue) 관리와 동일업무 프로세스 그룹을 위한 공통 큐 관리 및 로

드 밸런싱(Load Balancing)을 지원한다. 멀티 도메인 라우팅은 여러 도메인간의 메시지 전달을 담당한다. 이것은 단일 메시지를 타 도메인에 위치한 목적 프로세스로 전송하는 단일 메시지 라우팅, 메시지를 여러 도메인으로 전송하는 브로드캐스팅 메시지 라우팅, 해당 도메인에 목적 프로세스가 없을 때 실패 메시지를 원시 프로세스(Source Process)로 되돌리는 미확인 목적 라우팅(Unknown Destination Routing) 그리고 해당 도메인이 가용상태가 아닐 때를 처리하는 비가용 목적 라우팅(Unavailable Destination Routing)의 4 가지 전달방식을 제공한다.

네트워크 인터페이스 레이어는 XGM 시스템과 네트워크간의 연결을 담당한다. 이 레이어는 응용프로그램과의 커뮤니케이션 링크를 제공하고 제어 및 장애를 감시하는 커뮤니케이션 프로토콜(Communication Protocol)과 시간을 관리해주는 메시지 타이머(Message Timer) 그리고 포맷 검증 및 포맷 변환 기능을 제공하는 포맷 변환(Format Conversion)의 세 가지로 구성된다.

관리 레이어는 XGM 시스템의 운영 및 각종 편의 기능을 제공하는데, 시스템 운영을 위한 GUI 환경의 운영체제와 명령어를 관리하는 커맨더 운영체제(Commander Operation System) 그리고 로그 기록을 관리하는 로그 시스템(Log System)으로 구성된다. 여기서, GUI 운영체제는 다음의 다섯 가지 영역으로 나누어져 있다.

- XGM 오퍼레이션 관리
- 시스템 환경 관리
- 모니터링 관리
- EMS 로그 관리
- Audit 로그 관리

### 3.2 세부 설계

XGM의 업무 구성은 다음 그림 6과 같고 각각의 업무에 따른 모듈 구성은 그림 7에서 보아야 한다.

환경(Configuration) 업무에서는 크게 두 가지로 나누어지는데 윈도우 상의 GUI 환경과 파일서버들과 사용자간의 고리 역할을 하는 패스웨이(Pathway) 환경이다.

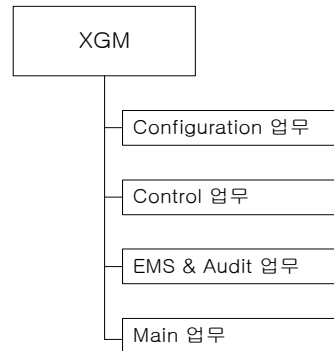


그림 6. 업무 구성

Fig. 6. Configure of XGM Work

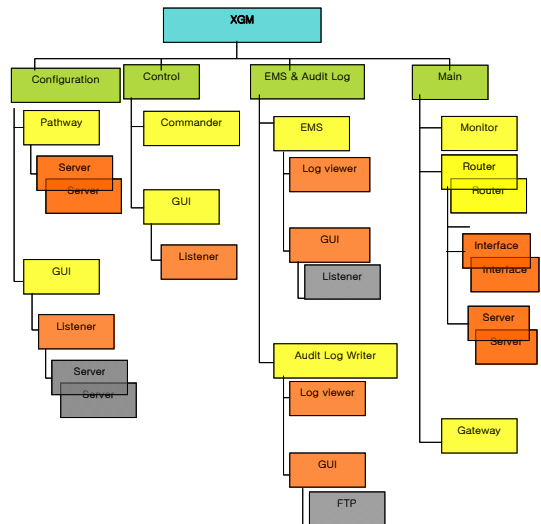


그림 7. 업무에 따른 모듈 구성

Fig. 7. Configure Module of XGM Work

이 두 업무에서 가장 큰 특징은 초기 환경정보, 각 프로토콜 프로세스에서 사용하는 정보, 메시지의 맵(Map) 정보, 사용자의 ID 및 비밀번호 그리고 사용되는 매개변수들에 대한 등록, 삭제, 수정 및 조회를 담

당한다는 것이다. 통제(Control) 업무는 각종 프로세스의 자동 제어 및 모니터링 그리고 각종 프로세스의 관리를 위하여 프로세스 상태를 조회, 기동, 정지 등의 기능을 수행하는 일이다. 각각의 세부 기능은 다음과 같다.

- 1) 프로세스 제어기능
  - START 명령 : 프로세스 기동
  - STOP 명령 : 프로세스 중지
  - Warmboot 명령 : 프로세스 초기 환경 정보 재 구성
- 2) 프로세스 조회기능
  - STATUS 명령 : 프로세스 상태 조회
  - INFO 명령 : 프로세스 등록 정보 조회
- 3) 프로세스 동적 관리기능
  - ADD 명령 : 프로세스 등록
  - Delete 명령 : 프로세스 삭제
  - ALTER 명령 : 프로세스 정보 변경
- 4) 보안 기능
  - SECURITY 명령 : 사용자의 ID 및 비밀번호를 확인하여 사용자 권한 변경
- 5) 큐 관리기능
  - PURGE 명령 : 라우터 프로세스 큐에 저장되어있는 데이터의 삭제
  - RESET 명령 : 라우터 프로세스 큐의 통계값 초기화
- 6) 데이터의 흐름을 Audit 기능
  - ALTER AUDIT 명령 : AUDIT 기능을 ON/OFF

EMS & Audit 로그 업무는 각종 프로세스에서 발생하는 로그 메시지를 출력한다. 사용자와 XGM 시스템 사이에 EMS 뷰어 프로세스와 Audit 뷰어 프로세스가 존재하고 이를 처리한다.

Main 업무는 모니터 프로세스, 라우터 프로세스, 인터페이스 프로세스 업무 서버 프로세스 그리고 게이트웨이(Gateway) 프로세스로 구성되어 있다. 각각의 역할은 다음과 같다.

- 1) 모니터 프로세스 : 프로세스의 제어 및 관리 기능을 수행 한다.
- 2) 라우터 프로세스 : 데이터의 버퍼링, 라우팅 등의 기능을 수행 한다.
- 3) 인터페이스 프로세스 : 프로토콜 및 데이터의 변환 기능을 수행한다.
- 4) 업무 서버 프로세스 : 사용자가 구현하는 업무 흐름 및 DB의 접근 기능을 수행 한다.
- 5) 게이트웨이 프로세스 : 라우터 프로세스간의 데이터 연결 기능을 수행한다.

### 3.3 구현 및 활용

#### 3.3.1 하드웨어 및 소프트웨어 구성

XGM의 구현을 위하여 구성된 하드웨어와 소프트웨어의 구성은 표 1과 표 2에서 보인다.

표 1. 하드웨어 구성  
Table 1. Hardware Configuration

| 구분   | 용량                      | 모델명                  | 수량   |
|------|-------------------------|----------------------|------|
| 시스템  | 1Set                    | HP 논스톱 서버 S7600      | 4CPU |
| 저장소  | 263GB                   | 4619(18.8G, 15K rpm) | 18대  |
| 통신포트 | SYN/ASYN C(X.25) 60 포트  | 3880 SWAN            | 10대  |
|      | Ethernet(TC P/IP) 10 포트 | 3861/10 Base         | 2대   |
|      |                         | 3863/100 Base        | 2대   |

하드웨어 중 MPP 기종으로 현재 무정지성을 요하는 업무 분야에서 가장 범용적으로 사용되고 있는 HP 논스톱 서버 플랫폼(platform)을 사용하였다. 소프트웨어에서 TigerSIM은 시뮬레이션과 테스트를 위한 도구로 프로세스간 메시지를 모니터링하거나, 캡처할 수 있다. 그리고, 개발도구는 윈도우 기반의 MS 비주얼 스튜디오에 닷넷이다.

표 2. 소프트웨어 구성  
Table 2. Software Configuration

| 구분    | 비고                         |
|-------|----------------------------|
| 운영체제  | 가디언90/OSS(G06.02)          |
| DB    | 논스톱 SQL/MP                 |
| 사용언어  | C, C++, Java, Tal, Cobol85 |
| 네트워크  | TCP/IP, X.25               |
| 시뮬레이션 | TigerSIM                   |
| 개발도구  | EIK-Nonstop 에디션            |
| 미들웨어  | XGM                        |

### 3.3.2 네트워크 구성과 업무 구성

도식화된 네트워크의 구성도는 그림 8과 같다. 네트워크 구성은 HP 논스톱 서버 플랫폼 기반으로 하였고, 시스템간의 신속하고 정확한 데이터 전송을 위하여 TCP/IP와 대외 접속을 위해 X.25를 각각 채택하였다.

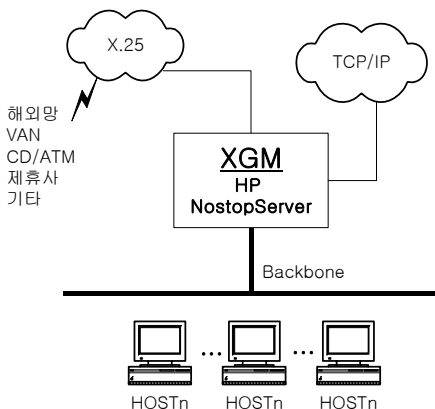


그림 8. 네트워크 구성도  
Fig. 8. Network Configuration

업무 구성은 은행의 신용카드 승인업무를 가지고 구축하였고 그림 9와 같다.

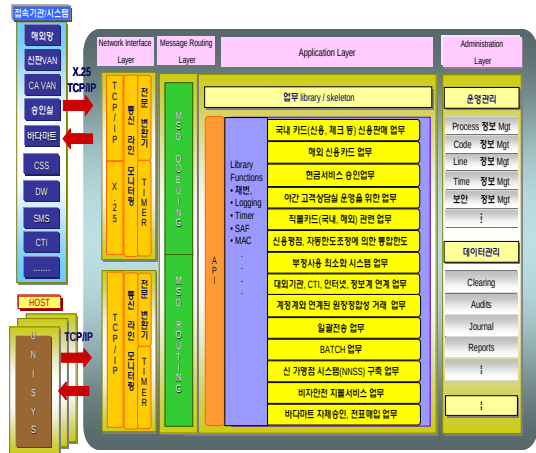


그림 9. 구축된 카드사 업무 구성  
Fig. 9. Established Credit Card Works

그리고, 각 레이어별 처리 기능은 다음과 같다.

#### 1) 네트워크 인터페이스 레이어

- 접속기관 및 시스템과의 통신 프로토콜 제어
- 메시지 검증 및 변환
- 메시지 별 처리 시간 관리
- 통신라인 장애 시 대응

#### 2) 메시지 라우팅 레이어

- 메시지를 목적 프로세스로 라우팅
- 목적 프로세스 Busy시 큐잉

#### 3) 응용 레이어

- 국내 카드(신용, 체크) 신용판매 업무
- 현금서비스 승인업무
- 해외 신용카드 승인 업무
- 승인실/당직실 조회 업무
- 부정사용 조기 경보 업무

- 한도 관리 업무
- 가맹점 관리 업무

#### 4) 관리 레이어

- 운영관리
- 데이터 관리

#### 5) 모니터 레이어

- 업무 시스템 전체의 무정지성을 담당

산된 레이어 구조로 설계된 반면 기존 제품은 주요 기능, 즉 XGM의 라우터, 인터페이스, 모니터 기능을 하나의 프로세스가 처리하는 구조적 차이에 기인한다.

## IV. 결 론

오래전부터 요구되어온 대용량의 자료 처리를 위해서는 확장성 측면이 탁월한 MPP 방식이 주류를 이루고 있으나, 기술적 어려움과 MPP 방식의 이점을 잘 살리기 위해서는 사용자의 시스템에 대한 전문지식이 필수적이다. 이를 위한 방안으로 사용자가 전문적 시스템 지식 없이도 이질적인 통신 프로토콜, 시스템 아키텍처, 운영체제, 데이터베이스 시스템과 다양한 응용 프로그램 서비스 등을 시스템 환경에 적합하게 구축할 수 있게 하는 미들웨어가 꼭 필요하다.

본 논문에서는 MPP 컴퓨터에서 효과적인 MOM 기반 미들웨어인 XGM의 설계 과정과 구현 과정을 소개하였다. 그리고, 실험을 통해 XGM의 우수성을 측정하였다. 특히, 실제 가동중인 금융업무시스템과 그 업무를 XGM을 적용한 결과 정형화된 업무 라이브러리와 프레임워크 프로그램을 지원함으로 모듈의 재사용성이 향상되었다. 기존 MOM 제품과 비교하여 시스템 부하시 병목현상 해소로 유연성 및 확장성이 증대되었고, GUI 및 웹 환경에서 운영 가능한 클라이언트 모듈 제공으로 사용자 편의성 측면에서 차별화 되는 결과를 얻을 수 있었다.

중소기업에서 개발중인 이 XGM은 우수한 미들웨어임에도 불구하고 다른 중소기업들처럼 경제적으로 어려움을 겪고 있는 실정이다. 정부차원의 적극적인 지원을 통해 우리의 우수한 기술이 지속적으로 발전할 수 있도록 제도적 장치가 마련되어야 하겠다. 향후, 다양한 실제 업무분야에서 다양한 검증 과정을 거쳐 완전한 미들웨어로 거듭나야하며, 유닉스 운영체제를 기반으로 하는 미들웨어로 진화를 해야한다. 이 또한

### 3.3.3 성능평가

신용카드의 승인관련업무들이 있는 은행의 카드승인 시스템을 모델로 XGM을 적용한 파일럿(Pilot) 시스템을 구축하여 현 은행 시스템과의 성능평가를 하였고, 또한 기존 MOM 제품과의 비교를 실시하였다.

XGM을 적용한 시스템과 현 은행 신용카드 승인 시스템의 성능 비교를 위한 조건은 승인 처리 건수 50만 건, 1 트랜잭션(transaction) 처리시 평균 디스크 I/O가 Select 15회, Update 10회, 평균 레코드의 크기는 500Bytes, 메시지 크기는 200Bytes를 기준으로 검증하였다.

검증 결과 XGM을 적용한 시스템 처리능력이 현 은행 신용카드 승인시스템 대비 10배 이상의 처리능력(XGM: 173 건/1초당, 현 S 은행 신용카드 승인시스템: 16 건/1초당)을 확인하였다. 또한 XGM은 GUI 및 웹 환경에서 운영 가능한 클라이언트 모듈을 제공하여 HP 논스톱 서버 환경에서만 운영이 가능한 현 시스템의 문제점을 개선할 수 있었고, 정형화된 업무 라이브러리와 프레임워크 프로그램을 지원함으로 모듈 재사용성이 측면에서 유리함이 입증되었다.

기존 MOM 제품과의 성능비교는 CPU 사용량이 70% 이내까지는 유사하였으나, CPU 부하가 커짐에 따라 기존 제품은 오히려 처리능력이 감소하는 현상이 발생하여, 시스템 부하 시에는 XGM이 우수한 처리능력을 나타냈다. 주된 원인은 XGM은 주요 기능이 분

막대한 자금과 시간을 필요로 하기 때문에 아낌없는 지원을 통해 세계적인 우수한 제품으로 발전해 나가기를 기대해 본다.

Monitor (TPM) Real-time Benchmark  
Testbed," *European Computing Conference, WSEAS*, pp. 150 ~ , 2008.

## 참고문헌

- [1] 박규호, 정봉준, 최종혁, 홍준성, "병렬처리 컴퓨터의 구조 및 응용," *한국정보과학회 논문지*, vol. 14 no. 6, pp. 35 ~ pp. 51, 1996.
- [2] 박승권, 유승화, "병렬처리 컴퓨터의 추세," *한국정보과학회지*, vol. 13 no. 7, pp. 5 ~ pp. 15, 1995.
- [3] 김학선, "MPP 기반 메시지 지향 미들웨어," 석사학위 논문집, 2003.
- [4] 양해술, 이하용, 박주석, "미들웨어 소프트웨어 시험사례 연구," *한국콘텐츠학회논문지*, Vol. 9 No. 7, pp. 110 ~ pp. 120, 2009.
- [5] 윤여봉, 윤희용, 정윤철, "유비쿼터스 미들웨어 기술," *한국인터넷정보학회*, 제5권 제 3호, pp. 49 ~ pp. 57, 2004.
- [6] Object Management Group(OMG), <http://www.omg.org>
- [7] 김수중, 윤용익, "임베디드 시스템 미들웨어의 기술 동향," *대한전자공학회지*, v.31 no. 11, pp. 99 ~ pp. 108, 2004.
- [8] Schmidt D. C. and Cleeland C., "Applying patterns to develop extensible ORB middleware," *IEEE Communications Magazine*, vol. 37 no. 4, pp. 54 ~ pp. 63, 1999.
- [9] 백광진, 김태운, "M-MOM;이동 클라이언트를 위한 메시지 지향 미들웨어 서비스," *한국정보과학회 논문지*, vol. 5 no. 2, pp. 247 ~ pp. 262, 1999.
- [10] Catalan, M.L., Ludena R., D.A., Umeno, H. and Aritsugi, M., "Transaction Processing



최준기(Joon-Kee Choi)

1995년 순천향대학교 전산학과  
(공학석사)

1999년 순천향대학교 전산학과  
(공학박사)

1999년~현재 인덕대학 비서과 부교수

※ 관심분야 : 디지털 워터마킹, 임베디드 시스템, RFID