

웹기반의 3 계층 구조 분산 객체 모니터링 시스템 설계에 관한 연구

이인수*,고상주*,박윤용*,정부금**,임동선**

*선문대학교 컴퓨터정보학부

**한국전자통신 연구소(ETRI)

e-mail:insul@nic.sunmoon.ac.kr

A Study on the Design of the 3-tier Distributed Object Monitoring System on the Web

In Soo Lee*, Sang ju Ko*, Yoon-Young Park*, Boo-Gaem Jung**, Dong-Sun Lim**

*Dept of Computer Science, Sunmoon University

**Electronics and Telecommunications Research Institute(ETRI)

요약

본 논문에서, 웹브라우저상에서 병행 객체들의 상태 정보 및 정보흐름을 보여주는 분산 병행 객체 모니터링 시스템의 설계에 관하여 기술하였다. 병행 객체는 UNIX의 쓰레드와같이 CPU에의해서 스케줄링 되어진다. 또한 모니터링 시스템을 설계해서, 분산환경에서의 병행 객체의 정적 및 동적정보를 수집하여, 웹브라우저상에 이 정보를 표시하고자 한다. 그리고, 설치한 모니터링 시스템을 동적라이브러리로 구현하고, 기존의 UNIX 또는 SROS상에서 실행되도록 구현할 것이다.

1. 서론

모니터링은 디버깅, 테스트 그리고 컴퓨터 프로그램들의 성능 평가를 지원해야 한다. 분산환경 시스템의 모니터링은 더욱 어려운 문제점들을 가지고 있다. 분산 환경의 모니터링 시스템들은 프로세스들간의 정보를 동적으로 수집해서 사용자가 원하는 유용한 형태로 표시한다. [1]

모니터링, 즉 객체들 또는 소프트웨어 프로세스들의 정보를 동적으로 수집, 해석, 표시하는 것은 분산 시스템을 관리 및 테스트 하기 위해서 필요하다. [1][2][3]

분산 모니터링 시스템의 수많은 문제점들 중 전송 지연으로인한 모니터링 정보의 일관성 문제 및 이벤트들의 리포트 전송 과정중의 다양한 지연은 부정확한 순서의 정보 전달을 발생시킬 수 있으므로, 클럭 동기화가 정확한 정보 전송을 위해 필요하다.

대형 시스템에서 모니터링 정보를 발생시키는 수많은 객체들은 관리자들에 어렵게 할 수 있으므로, 정보의 필터링과 처리기능이 필요하다.

또 다른 문제점은 모니터링 시스템 그 자체가 관찰되는 시스템과 동일한 자원을 사용하기 위해, 경쟁을 한다는 것이다. 이러한 문제들을 해결하기 위해, 모니터링 시스템은 모니터링 정보 생성, 처리,

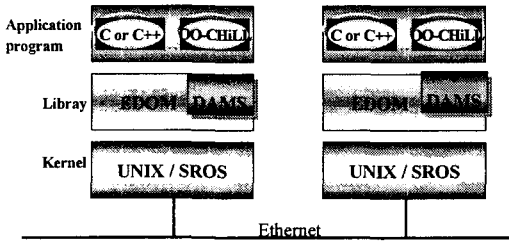
전송과 표시에 관한 수많은 기능을 제공해야 한다. 이 논문에서, 우리는 위에서 서술한 몇가지 문제점들을 해결할 수 있는 분산 모니터링 시스템 모델을 제안한다.

이 분산 모니터링 시스템은 OO-CHILL 언어로 생성된 병행 객체를 모니터링하는 시스템으로 채택한다. OO-CHILL은 전자교환 시스템을 위한 응용 프로그램의 개발 언어이다.

UNIX의 쓰레드와 같은 병행 객체는 CPU 에 의해 스케줄링 되어지는 데, 준비-실행-블록상태들을 갖는다. 또한 이것은 OOP 언어처럼 멤버 데이터와 함수를 갖는다.

OO-CHILL은 병행 객체를 선언하고, 생성할 수 있다. OO-Chill 로 쓰여진 응용 소프트웨어는 (그림1)과 같은 분산환경의 SROS상에서 실행되어 진다.

SROS(*Scalable Real-time Operating System*)는 ETRI에서 개발한 운영체제이다. 그리고 EDOM(*Etri Distributed Object Manager*) 또한 ETRI 자체에서 라이브러리로 병행 객체의 생성 및 종료기능을 가지며, 병행 객체들 사이에 멤버 함수를 호출 할 수 있고, 병행객체에 다양한 기능들을 제공한다. DAMS(*Distributed Active Object Monitoring System*)는 분산환경의 UNIX 또는 SROS 상에서 EDOM을 모니터링하기 위해 동적 라이



(그림 1) 병행객체실행 및 모니터링환경

브러리로 구현한다.

본 논문에서, 우리는 병행객체의 생명 주기 및 상태를 모니터링하고, 모니터링 정보를 수집해서 Web 브라우저상에 표시하기 위해 분산 병행객체 모니터링 시스템을 설계할 것이다.

2장에서, 모니터링 모델과 모니터링 시스템들의 구현 문제를 보고, 3장에서는 DAMS의 기능과 구조를 설명하겠다. 또한, 4장에서는 DAMS의 상세한 구현 측면을 설명하고자 한다.

2. 모니터링 모델과 구현.

모니터링은 컴퓨터시스템 또는 컴퓨터 프로그램들의 성능 평가에서 디버깅 그리고 테스트까지를 지원할 수 있는 기능들을 포함하고 있는 개념이다. 협의의 모니터링 개념은 단순히 실행중인 시스템의 상태 정보들을 수집하여 사용자들에게 보여주는 기능만을 의미한다. 그러나 광의의 모니터링 개념은 단순한 시스템 상태 정보 뿐만아니라 시스템의 상태를 제어할 수 있는 기능을 포함하고 있다. 시스템의 상태를 제어하기 위해서는 동적으로 모니터링 작업을 제어할 수 있는 기능이 필요하고, 동적으로 모니터링 작업을 제어하게 되면 실행중인 프로그램들의 디버깅과 테스트가 가능하게 된다.

프로그램이 분산 환경에서 실행하게 되면, 모니터링 작업은 더욱 어려워진다. 분산 시스템의 모니터링 작업은 동적으로 발생하는 프로세스 사이의 상호작용에 관한 정보들의 수집 그리고 이러한 정보들의 저장과 수집된 정보들을 사용자들에게 유용한 형식으로 보여주는 일련의 작업들을 포함한다. 따라서 모니터링 시스템을 구현하기 위해 중점적으로 고려해야 할 내용은 아래와 같이 요약할 수 있다.

첫째, 요구되는 모니터링 정보를 어느 곳에서 수집하는가 하는 문제이다. 모니터링 정보를 수집하기 위해서는 모니터 되는 시스템 내부에서 정보를 수집해야 하고, 이를 위해서는 모니터 되는 시스템 내부에 정보를 수집할 수 있는 코드를 삽입하여야 한다. 따라서 어느 부분에 필요한 코드를 삽입하여 어떤

내용의 정보를 추출할 것인가를 결정해야 한다. 또한, 정보 추출을 위해 삽입된 코드가 목표 시스템의 실행 성능에 영향을 최소화하도록 설계되어야 한다.

두 번째는 수집된 정보들을 분석해서 유용한 정보들을 만들어내는 문제이다. 많은 양의 정보가 수집되었지만, 이 정보들을 분석해서 시스템 모니터링에 필요한 정보들로 필터링하는 과정이 요구된다. 따라서 유용한 정보의 내용이 무엇인가를 정리하는 방법과 정보들의 분석 방법들에 관한 연구가 필요하다.

세 번째는 분석된 정보들을 사용자에게 편리한 형태로 보여주는 문제이다. 대부분의 모니터링 도구들은 분석하여 정제된 정보들을 보여주는 형태로서 GUI(Graphic User Interface) 형태를 일반적으로 사용하고 있다. 특히, 인터넷 인구가 확산되면서 본 연구에서는 모니터링 도구를 인터넷 상에 구현하고자 한다

이상의 내용을 요약하면 모니터링에 사용될 정보들 어느 곳에서(Where) 수집하여, 무엇을 보여줄 것인가(What)를 결정하기 위해 수집된 정보를 분석하고 필터링하는 과정이 필요하다. 마지막으로 분석되고 정제된 정보들을 어떻게(How) 보여 줄 것인가를 결정해야 한다. 이러한 과정을 근간으로 본 연구에서는 (그림 2)와 같은 3층 구조(3-tier)의 모니터링 시스템을 제안한다. 3장에서는 제안한 모니터링 시스템인 DAMS (Distributed Active Object Monitoring System)에 관하여 자세히 설명할 것이다.

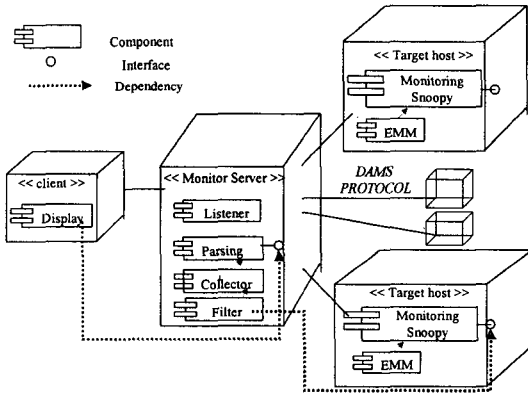
3. DAMS 의 구조와 기능.

3.1 . DAMS 의 구조

DAMS의 구조는 (그림2)와 같이 모니터 되는 시스템들, 모니터 서버 그리고 모니터 클라이언트들로 구성되어 있다. 모니터 되는 시스템은 실제로 모니터 해야 되는 시스템들로서 분산환경으로 구성되어 있다. 이곳에서는 주로 필요한 정보들을 수집하는 작업이 이루어진다. 모니터서버는 모니터 되는 시스템들로부터 모니터링 정보를 수집, 저장하고 분석하여 필요한 정보들로 정제하는 역할을 한다. 또한, 분석되고 정제된 정보들을 별도로 저장시키게 된다. 모니터 클라이언트에서는 모니터 서버에서 정제된 정보들을 인터넷 상에 표시하는 역할을 한다.

3.2 DAMS 의 기능

DAMS 의 기본 기능은 병행 객체의 생성, 소멸들의 생명 주기 및 병행객체들간의 멤버함수 호출을 모니터링하는 도구이다. 특히 DAMS는 실행시간 라이브러리 형식으로 설계하였으며, 이미 개발 완료된



(그림 2) 3-tier 모니터링 구조

EDOM과 함께 사용하여 분산 병행객체들을 모니터링 하게 된다.

부가적인 기능은 다음과 같다.

1) 모니터링 정보의 통합

분산시스템 상에서 모니터링 시작시 모든 관찰되는 호스트들에 대한 모니터링 정보를 생성을 하는 것은 어렵다. DAMS는 여러 모니터링 정보 파일을 합칠 수 있다. 또한 추적 이벤트들은 자동적으로 저장되어 질 수 있다.

2) 모니터 정보의 저장, 추적.

모니터 정보는 저장되고, 디버깅 단계에서 전체 병행 객체들의 상호 관계를 추적하기 위해 사용되어질 수 있다.

3) 모니터 정보의 통계적 처리.

통계 정보, 즉 생성된 병행 객체의 수와 병행 객체들의 메시지 호출 각각의 병행 객체의 상태들은 수집되어 지고, 처리되어 질 수 있다.

4) 병행 객체 성능 측정.

실행시간, 즉 병행 객체의 생성 시간 및 메시지호출시간은 측정되어지고, 수집되어 진다. 이것은 시스템 병목현상을 발견하는 데 도움이 된다.

5) 모니터링 정보 요약 및 표시

가공되지 않은 추적 정보는 매우 복잡하고, 많은 시스템 정보들이 교류하므로 이해하기 어렵다. DAMS에서는 이 저수준 정보를 분류하고 요약해서 순차적인 그래프 형태로 보여준다. 또한, 필터링기능을 통해 불필요한 정보를 제외한 중요한 핵심 정보를 요약해서 보여준다.

6) 웹상에 모니터링 정보 표시.

수집된 정보는 웹상에 표시되어 진다. 이것은 JAVA 언어로 구현된다.

각각의 DAMS 구성요소는 몇 개의 소프트웨어 모듈로 구성된다.

- EMS : EDOM Monitoring Server.
- EMV - EDOM Monitoring Viewer
- EMM - EDOM Monitoring Manager

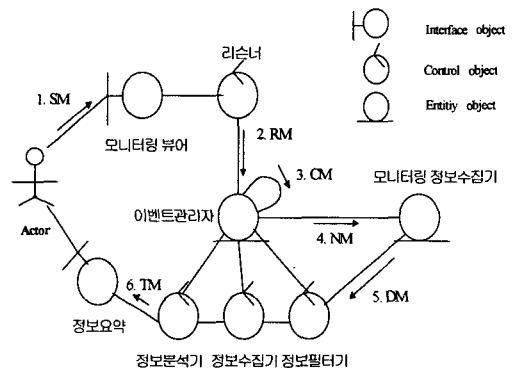
EMS 는 DAMS 의 핵심 부분이며, 모니터 서버에 있다. 모니터링 당하는 호스트들로부터 상태 정보를 모으고, EMV로 정보를 보낸다. 또한, EMS는 수집된 정보를 분석하고, 필터링하는 기능들을 가지고 있다. 특히, EMS는 "Listener"라는 쓰레드를 생성하고, 갱신한다. 이것은 모니터할 호스트의 이름에 대한 정보를 가지고 있다. 또한, 각각의 호스트의 모니터링 작업을 시작하고, 종료하며, 모니터할 시스템의 EMM과 연결 방법을 제공하거나, 모니터 클라이언트의 EMV와 연결한다.

EMV 는 모니터 클라이언트에 있고, 웹브라우저에 EMS 에 의해 수집된 다양한 정보를 표시한다.

EMM은 지역 저장 장치에 다양한 모니터링 이벤트들을 저장한다. 이 모듈은 모니터할 각각의 시스템에 위치해 있다. 저장 된 정보들은 모니터링 정보들을 추적하는 데 사용될 수 있거나, 서비스 객체들의 성능을 측정한다. 그리고, 이 정보는 모니터 서버의 EMS모듈에 전송된다.

4. DAMS 의 구현

이 장에서는 (그림 1)과 같은 개방분산 시스템을 위한 모니터링시스템, DAMS의 구현에 관하여 기술하였다. DAMS는 (그림 2) 와같이 모니터서버, 모니터 타겟 호스트, 그리고 모니터 클라이언트로 구성한다. DAMS구현을 위한 소프트웨어는 JDK1.2이후 버전 및 GNU C 컴파일러가 사용되어진다. 그래픽 사용자 인터페이스는 JAVA Applet으로 구현한다.



(그림 3) 이벤트 흐름도

3.3 DAMS 소프트웨어 구성요소

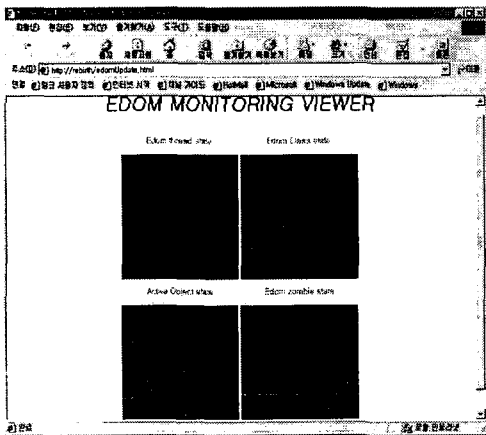
4-1 DAMS 의 이벤트

본 절에서는 DAMS에서 사용하고자 하는 이벤트를 정의하고, 모니터링이 실행되는 과정을 (그림 3)과 같이 표시하고, 설명하고자 한다.

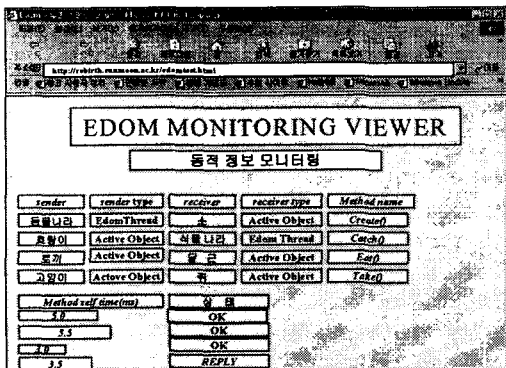
기본적으로 DAMS에서 정의하고 있는 이벤트는 SM(Start Monitor) 이벤트, RM(Request Monitor),CM(Create Manager) 이벤트, NM(Notify Message) 이벤트, DM(Delivery Message) 이벤트, TM(Transfer Message)이벤트등 6가지가 있다. 아래에서 위의 6가지 이벤트들을 설명하였다.

4.2 DAMS의 정보 표시.

클라이언트는 웹브라우저 상에서 실시간 또는 로그 방식으로 병행객체의 정적 및 동적 정보를 관찰할 수 있다. (그림4)는 DAMS VIEWER의 일부면으로, 실시간으로 현재 생성된 Edom thread의 수 또는 병행 객체의 수를 보여준다.



(그림4) DAMS Viewer



(그림5) 동적모니터링

모니터링 관리자는 더 자세한 정보관찰을 위해, (그림5)와 같이 실시간 모드를 로그 모드로 변경해서, 현재 병행 객체의 자세한 정보를 검사할 수 있으며, 병행 객체간의 메시지 통신의 결과를 웹브라우저 상에 시각적으로 보여 줄 수 있다.

모니터링 정보는 EDOM의 메시지 큐를 통해 정보를 생성할 수 있다. 특히, 병행 객체간의 멤버 함수 호출 테스트는 멤버 함수를 호출하는 송신자의 이름과 유형 그리고 멤버 함수를 수신하여 처리하는 수신자의 이름과 유형을 도시하였고, 멤버 함수 호출의 상태 결과를 표시한다

5. 결론

본 논문에서, 우리는 병행 객체의 개념을 소개했고, OO-Chill,SROS 그리고 EDOM 에 대해 소개했다.

그리고 우리는 병행 객체 모니터링 시스템 DAMS를 구현했다. DAMS는 UNIX와 SROS에서 테스트 중이다. 우리는 분산 모니터링 시스템의 주요 기능들을 설명했다. 또한, 우리는 DAMS의 자세한 설계 및 Diagram을 제시했다. 앞으로 계속해서 DAMS를 구현하여 EDOM 과 연동되어 사용할 수 있도록 하는 것이 큰 의미있는 연구과제가 될 것이다.

참고문헌

- [1] J.Joyce et al., "Monitoring distributed Systems,"ACM Trans, Comutut.Syst., vol. 5. no. 2 .May 1987, pp. 121 -50.
- [2]Masoud Mansouri-Samani and Morris Sloman, "Montiroing distributed Systems",IEEE Network,November 1993.
- [3] Young-Seok Shin , San-baek Lee and Dong-Sun Park, " Design of An Object Monitoring System for Open Distributed Services" ,IEEE 1998.
- [4] D.Marinescu et al., "Models for Monitoring and Debugging Tools for Parallel and Distributed Software," Journal of Parallel distributed Computing 9, 2, June 1990 , pp. 171 -84 (specialissue : software tools for parallel programming and visualisation)
- [5] M.Van Riek and B. Tourancheu, " A General Approach to the Monitoring of Distributed Memory Machines. A Survey, Laboratoire de Informatique du Parallelisme," Ecole Normale Supérieure de Lyon, Institut des Sciences de la Matière de Université Claude Bernard de Lyon, Institut IMAG, Unite de Recherche Associee au CNRS no. 1398, Research Report no. 91/28,September 1991.
- [6] R.Holmann et al., "Distributed Performance Monitoring : Methods,Tools, and Applications ,"University of Erlangen-Nurnberg,IMMD VII, Martensstrabe 3,D-8520 Erlangen,Germany,1992.
- [7]E.Lumpp, Jr et al., "Specification and Identification of Events for Debugging and Performance Monitoring of distributed Multiprocessor Systems, " Proc. 10th International Conference on Distributed Systems, June 1990, pp. 476 -83.
- [8] R.Klar, A . Quick, and F.Soeiz , "Tools for a Model-driven Instrumentation for Monitoring," G.Balbo,editor, Proc. of the 5thInt. Conf. on Modelling Techniques and Tools for Computer Performance Evaluation, Rorino, Italy, pages 165-180. (Elsevier Science Publisher B.V., 1992)