

자바 임베디드 시스템 기반 산업용 계측 및 진단을 위한 네트워크 미들웨어 실행환경의 설계 및 구현

정준영[†] · 김광수^{††} · 전진홍^{††}
[†]경남대학교 컴퓨터공학과 ^{††}한국전기연구원

Design and Implementation of Network Middleware Execution Environment for Industrial Measurement and Diagnosis based on Java Embedded System

Jun-Young Jung[†] · Kwang-Soo Kim^{††} · Jin-Hong Jeon^{††}
[†]Dept. of Computer Engineering, Kyungnam University ^{††}KERI

Abstract - 반도체와 네트워크 기술의 발전은 산업분야에서의 각종 컴퓨팅 환경을 독립형 시스템에서 네트워크 시스템으로 변화시키고 있다. 본 논문에서는 임베디드 시스템 기반 산업용 계측 및 진단 시스템을 네트워크 환경하에서 보다 능동적으로 상호 작용하기 위한 미들웨어 수준의 임베디드 시스템 실행환경을 구성하고, 시스템 플랫폼으로는 자바를 사용한다. 자바 임베디드 기반 계측 및 진단 네트워크 미들웨어 시스템은 각 대상기기의 고유작업들을 처리할 뿐만 아니라 이들 대상기기의 계측 및 진단 정보를 서로 공유함으로써 보다 다양한 상황에 유연하게 대처하고, 효율적인 시스템 관리를 위한 능동 계측 진단 미들웨어 시스템 실행환경을 구성하고자 한다.

반 미들웨어 관련 기술에 대해 살펴보고, 마지막으로 4장에서는 결론으로써 본 논문을 끝맺는다.

2. 본 론

2.1 산업용 계측 진단 네트워크 미들웨어 시스템

본 논문에서 제안된 산업용 계측 진단 시스템은 자바 런타임 환경을 포함하고 있는 네트워크 미들웨어 임베디드 시스템이다. 네트워크 미들웨어 임베디드 시스템 보드는 각 계측대상기기(Target System)에 연결된다. 자바 런타임 환경 기반 네트워크 미들웨어는 데이터 처리와 네트워크 기능을 수행할 수 있다.

2.1.1 네트워크 미들웨어 시스템 구성

그림1에서와 같이, 각 대상 시스템이 각각 운영되는 동안 계측 진단 네트워크 미들웨어는 전체 시스템을 구성한다. 네트워크 미들웨어는 산업용 계측 보드(Industrial Middleware Board: IMB)에 탑재되어 있다. IMB는 데이터 수집 기능, 데이터 처리 기능, 네트워크 기능, 그리고 네트워크 미들웨어를 이용한 시스템 구성 기능이 있다.

1. 서 론

최근의 계측기기는 정보통신기술의 접목을 통해 계측기의 네트워크화라는 새로운 패러다임을 모색 중이다. 이와 같은 흐름은 네트워크 기술의 발전과 더불어 현재 생활 전반에 걸쳐 광범위하게 변화되고 있는 디지털기의 정보화 추세와 동일하다. 이는 결국 계측기 분야도 네트워크화 방향으로 진전되어 계측이 시간적 공간적 제약 없이 이루어질 수 있도록 새로운 개념의 계측시스템, 즉 네트워크형 계측기의 출현이 불가피하게 되었음을 의미한다.

차세대 네트워크형 계측기기를 개발하기 위해서는 현재 다양한 운영체제 환경 하에서 운용되고 있는 컴퓨팅 환경의 통합된 네트워크 연결기술이 요구되어진다. 현재 임베디드 콘트롤러 시장에서 C/C++와 함께 많이 사용되고 있는 언어는 자바(Java)이다. 자바는 동적 다운로드, 다른 플랫폼과의 호환성, 상호간의 통신 능력, 많은 사용자, 보안문제 등의 이유로 주목할 필요가 있다 [1,2,3,4,10,12]. 자바는 정보통신기술을 산업용 계측기에 접목시켜 줄 수 있는 훌륭한 수단으로, 차세대 네트워크 형 계측기의 기반기술 확보를 위해 산업계측기용 자바기반 플랫폼의 개발이 가장 우선되어야 한다. 이러한 디지털 계측기 시스템을 구성하기 위해서는 각 계측기의 안정성을 보장하고 각 계측기간의 원활한 통신을 가능하게 하여 데이터의 실시간 공유를 가능하게 함으로써, 보다 진보된 계측기기 시스템으로 통합할 수 있는 표준 디지털 계측기용 미들웨어를 필요로 한다. 이러한 표준화된 디지털 계측기용 운영체제를 위해서는 개방구조를 가진 CPU와 디지털 계측기용 운영체제를 통해 모든 계측장비를 일반화 될 수 있는 플랫폼 구조로 통일하여 이들 기기 사이에서 정보를 교환할 수 있도록 해야 한다. 자바 플랫폼은 계측기기를 네트워크의 한 구성원으로서 인지하여 가상 네트워크 계측기기의 유연성을 가능하게 한다.

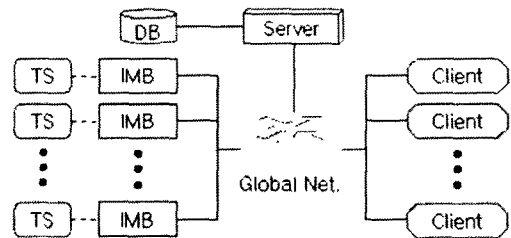


그림1. 산업용 계측 진단을 위한 네트워크 미들웨어 시스템 구성

IMB는 시스템의 상태를 분석하기 위한 진단 모듈을 실행하는 임베디드 시스템이다. 이것은 자바 런타임 환경에 기반을 두고 있는 계측 모듈과 진단 모듈을 포함하고 있다. 산업용 미들웨어 시스템은 시스템의 유연성을 부여하여 산업분야에서 적용하기 위해 자바 런타임 환경을 재구성한다.

2.2 자바 임베디드 시스템 기반 소프트웨어 아키텍처

IMB를 위한 자바 플랫폼은 자바가상기계, 자바 API 그리고 자바 애플리케이션으로 구성된다. 자바 API는 Core API, Standard API, Signal API 그리고 Measurement API를 포함한다. Signal API와 Measurement API는 산업분야에서 계측 대상기기를 계측하는 기기에 사용될 수 있는 산업용 API이다. 그림은 IMB를 위한 자바 플랫폼 아키텍처를 나타내고 있다 [1,2,3,10,13].

본 논문의 구성은 다음과 같다. 2장에서는 네트워크 미들웨어 시스템의 구성과 자바 플랫폼 기반 소프트웨어 아키텍처를 기술하고 구현 및 테스트에 대해서 묘사한다. 3장에서는 산업분야에서 적용 가능한 임베디드 시스템 기

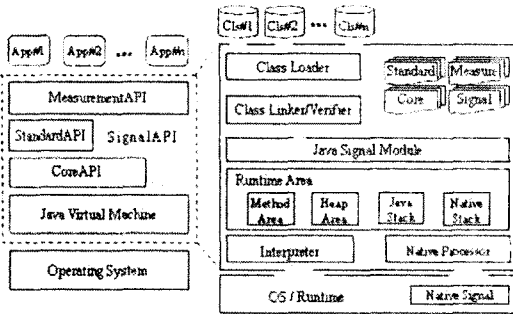


그림2. IMB를 위한 자바 플랫폼의 아키텍처와 실행

그림 2에서 보듯이, IMB를 위한 자바의 실행과정은 표준자바의 실행과정과 유사하다. IBM를 위한 자바와 표준자바와의 차이점은 네이티브코드의 사용이다. 표준자바는 네이티브코드를 포함하지 않는다. IMB를 위한 자바는 네이티브 코드를 포함하고 있는데, 이것은 자바 실행환경의 유연성을 떨어뜨린다.[4,5,8]

2.2.1 산업용 계측 및 진단 시스템 컴포넌트

네트워크 미들웨어는 그림3과 같이 다섯 개의 컴포넌트로 구성된다. Data Analysis Module 은 시스템으로부터 수신된 계측 데이터를 분석한다. 분석된 데이터는 시스템의 진단하기 위한 기본 정보를 생성한다.

System Diagnosis Module은 분석된 데이터를 사용하여 시스템의 상태를 진단하고, 시스템의 이상유무를 결정한다. Pre-Emergency Status Scan Module은 사전에 정의된 이상상태를 가지고 비교하여 대상 시스템의 상태를 체크하는 모듈이다.

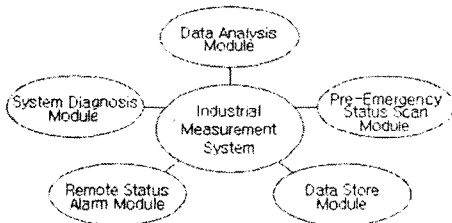


그림3. 네트워크 미들웨어 시스템 컴포넌트

Remote Status Alarm Module은 대상 시스템의 이상상태가 발생할 경우 시스템 관리자에게 알려주기 위한 모듈이다. Data Store Module은 Data Analysis Module과 System Diagnosis Module에서 생성된 데이터를 저장하기 위한 보드의 data store module과 data backup server module로 구성된다.

2.3 시스템 구현 및 테스트

본 논문에서 구현한 산업용 네트워크 미들웨어 보드는 자바 플랫폼을 내장하고 있으며, 네트워크 서비스를 위한 웹서비스 환경을 기본적으로 구성한다. 네트워크 미들웨어에서 사용되고 있는 자바 플랫폼은 임베디드 시스템 환경에 적합하도록 기능들을 추가 및 삭제를 통하여 성능의 최적화를 꾀한다.

그림4는 2.1절과 2.2절에서 제안된 산업용네트워크 미들웨어 시스템과 자바 임베디드 시스템 기반 소프트웨어 아키텍처를 기반으로 구현한다. 네트워크 미들웨어 보드에서 제공하고 있는 계측 및 진단을 위한 자바 환경과 웹 서비스를 이용하여 대상계측기기의 상태를 모니터링하고 있다. 본 시스템을 실행하는데 있어서 가장 중요한 요소 중의 하나는 계측 데이터의 무손실 보장해야 한다. 계측

데이터는 대상기기의 전원이 실시간적으로 이루어져야 하기 때문이다.

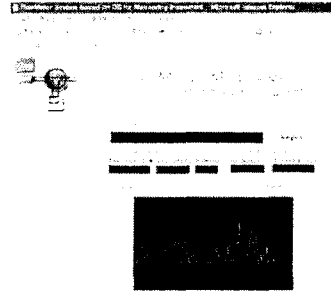


그림4. 네트워크 미들웨어기반 서비스의 구현

3. 관련연구

3.1 OpenPLANET

OpenPLANET이란, 전기를 이용하여 동작하는 모든 기기를 인터넷 경유로 전세계 어디에서나 컨트롤 할 수 있는 획기적인 '원격 감시 제어 시스템'이다. 그림5에서의 아키텍처에서 보여지는 리얼머신과 버추얼머신은 OpenPLANETdml 중요한 컴포넌트이다[14].

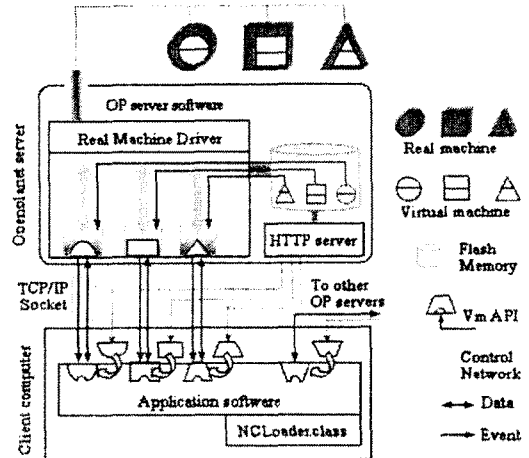


그림5. OpenPLANET 아키텍처

리얼머신은 감시 제어의 대상이 되는 여러 가지 전기 기기 제어에 네트워크상에서의 통신기능을 가진 원칩 마이크로컴퓨터를 내장함으로써 정보의 송수신 능력을 갖고 있다. 버추얼머신은 데이터 통신에 의해 실기계와 동일하게 동작하는 가상 기기. 자바 언어로 만들어진 소프트웨어 오브젝트로서, 인터넷 등의 정보제 네트워크상을 자유자재로 이동할 수 있다.

버추얼머신 서버는 OpenPLANET의 소프트웨어를 내장하고 가상기계 관리나 배송, 액세스 제어, 통신 시큐리티 관리 등을 하는 관리용 퍼스널컴퓨터, 일반적인 시판 퍼스널컴퓨터를 그대로 사용할 수 있는 것 외에 보다 콤팩트화, 저가격화를 지향한 전용 서버를 현재 개발 중이다.

2.2 Java Virtual Machine

자바가상머신은 자바의 런타임 실행환경중에서 실제 코

드를 소프트웨어적으로 해석하고 실행하는 가상화된 칩이다. 자바 가상머신은 실행되는 운영체제에 관계없이 독립적으로 수행해줄 수 있는 기능을 제공한다.

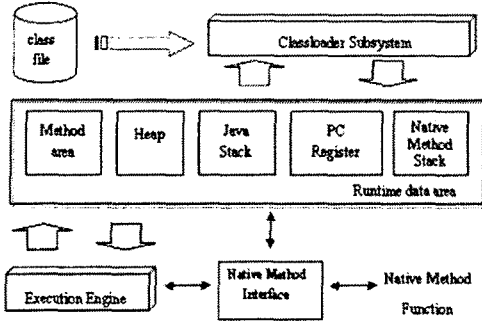


그림6. 자바 가상 머신의 내부구조

자바 가상머신은 내부적으로 그림6과 같이 구성되는데 모든 프로그램은 쓰레드 단위로 실행되며 메소드 영역과 힙은 모든 쓰레드가 공유한다. 자바 스택은 여러 개의 스택 프레임으로 구성되며, 스택 프레임은 쓰레드가 호출해 실행하고 있는 메소드의 상태를 저장하기 위한 단위이다. 또한 자바 가상머신에서 실행하는 애플리케이션 클래스에 해당하는 클래스 파일의 바이너리 데이터를 읽어 들여 메소드 영역에 저장하며, 프로그램 실행 중에 생성한 모든 객체는 힙에 둔다[1,2,3].

2.2 TRON

TRON(The Realtime Operating System Nucleus)은 사용자가 컴퓨터의 기능과 사용법을 정의하는 컴퓨터 운영체제를 말한다. TRON 은 개방구조인 CPU 와 운영체제를 포함하는 단일구조로 정보기기를 통합한다[11,12].

TRON은 다섯 가지 구성요소로 이루어져 있으며, ITRON과 JTRON이 핵심적인 역할을 담당하고 있다. ITRON은 임베디드 시스템을 위한 실시간 운영체제이고, JTRON은 자바와 ITRON의 특징들을 결합한 하이브리드 운영체제이다. 그림은 자바 프로그램과 실시간 프로그램 사이의 통신을 위한 인터페이스를 계층적으로 나타내고 있다.

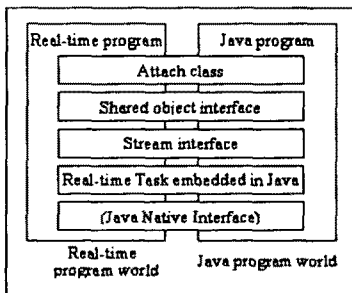


그림7. 자바 프로그램과 실시간 프로그램의 상호관계

4. 결 론

본 논문에서 산업용 계측 및 진단을 위한 임베디드 시스템을 구축하고, 임베디드 컨트롤러 시장에서 C/C++ 와 더불어 각광받고 있는 자바플랫폼을 이용하여 네트워크 미들웨어 실행환경을 구성한다. 이는 임베디드 자바를 기반으로 자바 실행환경을 구축하여 산업기기의 네트워크화를 실현하고 산업기기간의 커뮤니케이션과 관리자나 사용자가 요구하는 새로운 기능을 다운로드하여 추가, 변경할 수 있어, 산업기기 시스템의 유연성을 부여할 수 있어

서, 보다 유동적인 시스템을 구성하는 것이 가능하다.

[참 고 문 헌]

- [1] B. Venners, Inside Java Virtual Machine, McGraw-Hill, 1997.
- [2] Tim Lindholm and Frank Yellin, The Java Virtual Machine Specification, ADDISON-WESLEY, 1997.
- [3] A. Taivalsaari, Implementation a Java Virtual Machine in the Java programming Language, SUN Lab, 1997.
- [4] Peter W. Madany, JavaOS:A Standalone Java Environment, JavaSoft, 1996.
- [5] Dkramer, Java Native Interface Specification, SUN Microsystems, 1997.
- [6] Yukikazu Nakamoto, JTRON 2.0 Specification, TRON Association, 1999.
- [7] Ken Sakamura and Hiroaki Takada, ITRON4.0 Specification, TRON Association.
- [8] Y. J. Kwag, Java Virtual Machine Programming, Infobook, 1999.
- [9] <http://Java.sun.com/>, Sun Microsystems, Java Home Page.
- [10] <http://www.tron.org/>, TRON ASSOCIATION.
- [11] <http://www.t-engine.org/>, T ENGINE FORUM.
- [12] <http://www.javaworld.com/>, Java World COMPANY.
- [13] <http://www.openplanet.co.jp/>, OpenPlanet Project, Shiko-ku Electric Power COMPANY.