DMTF 표준 기반 경량 CIM 서버 기술 분석

박종근*, 안창원*, 우영춘*, 최훈**
*한국전자통신연구원 인터넷서버그룹
**충남대학교 컴퓨터공학과
e-mail : queue@etri.re.kr

Analysis of Lightweight CIM Servers based on DMTF Standards

Jong-Geun Park*, Chang-Won Ahn*, Young-Choon Woo*, Hoon Choi**
*Internet Server Technology Group, ETRI
**Dept. of Computer Engineering, Chungnam National University

요 약

이기종의 다양한 컴퓨터 시스템과 네트워크 장비 그리고 소프트웨어가 서로 혼재되어 있는 오늘날의 IT 환경을 효과적으로 관리하기 어려워짐에 따라 기업은 막대한 총 소유비용을 지출하고 있다. 이러한 상황을 효과적으로 대처하기 위하여, 분산 이기종 환경에서 플랫폼 독립적인 동시에 기술 중립적인 방법으로 시스템 자원을 감시, 통제, 조정하기 위한 표준 기술이 DMTF 를 중심으로 활발하게 개발되고 있다. 본 논문에서는 소형 단말기를 포함한 수많은 정보기기가 모두 네트워크로 연결되는 유비쿼터스 환경에서의 표준화된 시스템 관리를 가능케 하는 경량 CIM 서버가 가져야 할 요구사항을 도출하기 위해 현재 가장 대표적인 CIM 서버 솔루션을 중심으로 그 특징을 분석한다.

1. 서론

지난 수십년간 컴퓨터 및 통신기술의 급속한 발전에 힘입어 기업의 IT 환경은 이기종(heterogeneous)의 다양한 컴퓨터 시스템과 네트워크 장비 그리고 소프트웨어가 서로 혼재되어 운용되어 오고 있다. 이에 따라 1990 년대부터 네트워크 및 시스템 관리를 위한 통합 프레임워크의 필요성이 대두되었으나 기업별로 독자적인 시스템 관리 기술 및 솔루션을 개발하고 사용해옴으로써, 빠르고 다양하게 변화해가는 기업의 IT 환경을 효과적으로 관리하기가 어려워지고, 이는 곧 막대한 IT 총 소유비용(TCO; Total Cost of Ownership)의 지출을 초래하고 있다.

이러한 상황을 효과적으로 대처하기 위해 산업체가 주축이 되어 결성된 DMTF(Distributed Management Task Force, Inc.)는 기업 환경 및 인터넷 환경에서 요구되는 관리 표준을 개발하고 상호운용성을 보장하기 위한 통합 기술을 선도해 나가고 있다[1].

본 논문에서는 소형 단말기를 포함한 수많은 정보기기가 모두 네트워크로 연결되는 유비쿼터스 환경에서의 표준화된 시스템 관리를 가능케 하는 DMTF 표준 기반의 시스템관리 에이전트인 경량 CIM 서버가 가져야 할 요구사항을 도출하기 위해 가장 대표적인 CIM 서버 솔루션을 중심으로 그 특징을 분석한다.

2. DMTF 표준 기반 시스템관리

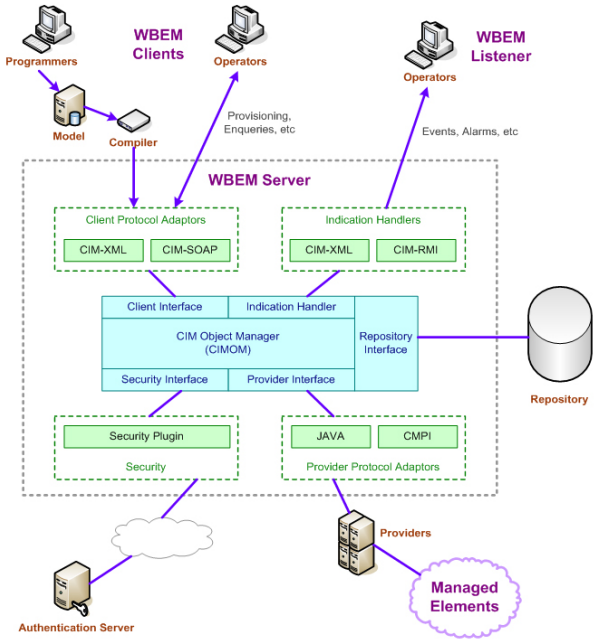
DMTF 표준 기술은 분산 환경에서 다양한 이기종 기기를 대상으로 플랫폼 독립적인 동시에 기술 중립적으로 시스템 자원을 감시, 통제, 조정하기 위한 기술로서, 그 중 가장 대표적인 표준 기술로는 CIM(Common Information Model)과 WBEM(Web-based Enterprise Management) 표준을 꼽을 수 있다.

공통정보모델인 CIM 은 비즈니스 컴퓨팅 및 네트워크 환경을 기술하기 위한 개념적인 모델로서, 플랫폼 독립적인 동시에 기술 중립적으로 관리 정보를 교환하기 위해 제정된 표준이다. CIM 은 특정 영역에만 국한되지 않고, 네트워크를 기반으로 클라이언트부터 서버에 이르는 종단간(end-to-end) 관리를 지향하고 있으며, 특히 객체 지향적인 모델 정의로 인해 모델 자체의 유연성과 확장성이 뛰어나 컴퓨터 시스템은 물론 통신, 전력 분배 시스템 등 다양한 산업 분야에서 채

택되어 사용되고 있다.

WBEM 표준은 분산 네트워크 환경에서 효율적인 시스템 관리 프레임워크를 제시하기 위해, 데이터 모델인 CIM 과 전송 프로토콜로서의 XML 및 HTTP 표준을 결합한 일련의 표준으로서 출발하여, WBEM 표준 기반의 관리 시스템에서 필요한 다양한 기술에 대한 표준화가 활발하게 진행되고 있다.

WBEM 표준 기반의 시스템 관리 시스템을 구성하기 위해서는 CIM(또는 WBEM) 클라이언트와 적어도 하나 이상의 CIM(또는 WBEM) 서버가 있어야 한다. CIM 서버는 실제 시스템 자원에 접근하여 정보를 추출하고 제어하는 다수의 CIM 프로바이더(provider)와 CIM 객체 관리자(CIM Object Manager), CIM 스키마(schema) 정보를 저장하는 CIM Repository, 그리고 CIM 클라이언트와의 통신을 위한 프로토콜 어댑터로 구성된다. 이에 반해 CIM 클라이언트는 시스템 관리를 위해 시스템 정보를 얻거나 제어하기 위한 사용자 인터페이스를 포함하여 CIM 서버와의 통신을 위한 프로토콜 어댑터로 구성된다. [그림 1]은 WBEM 표준 기반 시스템 관리 시스템의 구조를 도식화한 것이다[2].



(그림 1) WBEM 표준 기반 시스템 관리 시스템 구조

3. 경량 CIM 서버 기술 분석

3.1. 정의와 필요성

경량(lightweight) CIM 서버란, 일반적인 엔터프라이즈(enterprise) 서버 환경이 아닌, 데스크탑이나 임베디드 시스템 또는 시스템 자원이 한정된 소형 단말에 설치되어 운용될 수 있는 CIM 서버로 정의할 수 있다. 따라서, 시스템 자원이 풍부한 서버 시스템이 아닌 극히 제한된 시스템 또는 단말에서도 DMTF 표준에서 정의하는 다양한 기술을 지원하는 동시에 시스템 자원의 소비 또는 점유를 최소화해야 한다.

이러한 경량 CIM 서버는 앞서 언급한 바와 같이 향후 본격 도래할 유비쿼터스 서비스 환경에서는 더욱더 그 필요성이 대두될 것이다. 즉, 현재의 IT 환경도 분산된 네트워크를 기반으로 다양한 이기종의 시스템이 혼재되어 있지만, 유비쿼터스 서비스 환경에서는 그 관리 대상이 더욱 확대되고 다양해 질 것이기 때문이다. 결국 임베디드 시스템 또는 시스템 자원이 한정된 소형 단말까지도 DMTF 표준 기반의 일관된 시스템 관리 환경에 포함됨으로써, 시스템 관리자는 어떠한 시스템이든 상관없이 하나의 관리 기술과 인터페이스를 통해 모든 IT 자원을 효과적으로 관리할 수 있기 때문이다.

3.2. 공개 소프트웨어 기반 CIM 서버 분석

DMTF 의 WBEM 표준에 따라 다양한 공개 소프트웨어 프로젝트들이 활발하게 진행되고 있다. 먼저 대표적인 CIM 서버 개발 프로젝트로는 The Open Group 의 OpenPegasus[3], Quest Software 와 Novell 이 중심이 된 OpenWBEM[4], IBM SBLIM(Standards Based Linux Instrumentation for Manageability) 프로젝트의 sfcb(small footprint CIM broker)[5]가 있다. 이밖에도 SUN Microsystems 의 WBEMServices[6]가 있으나 2004 년 이후 개발이 지속되지 않고 있으며, Microsoft 의 WMI[7]는 MS Windows 운영체제에서 사용되고 있으나 표준을 완전히 준수하지도 않으며 소스를 공개하지도 않기 때문에 본 분석에서 제외한다. 본 분석에서 고려하는 OpenPegasus, OpenWBEM, sfcb 의 간략한 특징을 비교하면 <표 1>과 같다.

<표 1> 공개 소프트웨어 기반 주요 CIM 서버의 특징 비교

구분	OpenPegasus	OpenWBEM	SBLIM sfcb
라이센스	MIT Open source license	BSD Style license	Common Public license
개발자	The Open Group	Quest SW & Novell	IBM
사용언어	C++	C++	C
규모	Enterprise-grade CIM Server	Enterprise-grade CIM Server	Lightweight CIM Server
프로바이더 인터페이스	NPI CMPI JMPI Remote CMPI	C++ / Perl NPI CMPI Remote	CMPI
운영체제	Linux Solaris HP-UX HP-OpenVMS Mac OS MS Windows	Linux Solaris HP-UX AIX Mac OS X Novell Netware	Linux UNIX

<표 1>과 같이 OpenPegasus 는 기본적으로 서버급 환경을 대상으로 개발되어 온 CIM 서버이다. 그러나 경량 CIM 서버에 대한 필요성이 대두되면서 새로운 경량 CIM 서버의 개발이 아닌, 현재의 OpenPegasus 를 단순히 환경 설정을 통해 경량 CIM 서버로 사용하기

위한 OpenPegasus Lite 프로젝트를 진행하고 있다. 이 프로젝트에서는 현재 다음의 고려사항을 바탕으로 경량 CIM 서버의 요구사항을 도출하고 있으나, 아직까지는 뚜렷한 결과를 제시하지 못하고 있다[3].

- 메모리 및 디스크 사용량(footprint)에 대한 요구사항 분석
- OpenPegasus 의 기능 중 경량 CIM 서버에서 제공되어야 할 것과 제외해도 되는 기능의 검토
- 경량 CIM 서버가 운용될 운영체제와 기존 서버급 운영체제의 차이점을 결정하기 위한 경량 운영체제의 검토
- 경량 CIM 서버에서 대안 기술로 대체될 수 있는 기능 블록의 검토 (예, CIM repository 및 프로바이더 인터페이스 등)
- OpenPegasus 코드에 대한 컴파일러의 제한사항 결정

OpenPegasus 와 함께 현재 가장 널리 활용되고 있는 OpenWBEM 도 서버급 환경을 대상으로 개발되고 있는 CIM 서버이다. 그러나 OpenPegasus 와는 달리 OpenWBEM 은 주목할만한 경량 CIM 서버의 개발 움직임은 보이지 않고 있다.

IBM 의 SBLIM 프로젝트의 일환으로 개발되고 있는 sfcb 는 애초부터 경량 CIM 서버를 목표로 개발된 유일한 CIM 서버이다. Sfcb 의 경량 CIM 서버와 관련된 주요 특징을 열거하면 다음과 같다[5].

- 모듈 기반 방식으로, 운용 환경에 따라 기능을 쉽게 추가, 삭제, 조정 가능
- GetProperty, SetProperty 를 제외한 모든 표준 기반 CIM operations 지원
- CIM Indication 지원
- 열거형 CIM operation 응답 메시지 전송을 위한 HTTP chunking 지원
- 동작중 시스템 자원 소비를 최소화하기 위해, 여러 개의 CIM 클래스를 하나로 묶어 단일 프로세스로 실행될 수 있도록 하며, 사용자 환경 설정으로 유휴한 프로바이더의 자동 로딩/언로딩 기능 제공
- 프로바이더 인터페이스로서 CMPI 인터페이스만 지원
- CIM 클라이언트 인증 기능을 CIM 서버가 아닌 별도의 CIM 프로바이더로 개발하여 추가 가능 (예제 프로바이더 제공)

이와 같은 서버급 환경을 대상으로 한 CIM 서버에 비해서도 손색없이 갖춘 특징으로 인해 sfcb 는 가장 대표적인 동시에 이목의 집중을 받고 있는 대표적인 경량 CIM 서버로서 자리매김하고 있다.

그러나 한가지 주목할 것은 sfcb 가 대표적인 경량 CIM 서버이고 또한 OpenPegasus Lite 프로젝트가 진행중이지만, 경량 CIM 서버가 갖추어야 할 구체적인 시스템 요구사항이 아직 정립되어 있지 않다는 것이다. 따라서 이러한 시스템 요구사항의 분석 및 정의가 선행되어야만 경량 CIM 서버의 개발이 본격화될 수 있으며, 앞으로 도래할 유비쿼터스 서비스 환경에서의 표준 기반 시스템관리 환경이 확산 적용될 수 있을 것이다. 또한 sfcb 의 경우와 마찬가지로 경량 CIM 서버가 Linux 나 UNIX 이외에 아직도 많은 사용자층을 확보하고 있는 MS Windows 등 다양한 플랫폼에서도 동작할 수 있는 유연한 경량 CIM 서버의 개발이 필요하다.

4. 결론

지금까지 DMTF 표준 기반의 시스템관리 기술을 소개하고 현재의 경량 CIM 서버 기술에 대해 분석하였다. 앞서 살펴본 바와 같이, 세계 유수의 IT 기업과 표준화 단체가 앞장서서 DMTF 표준 기반의 기술을 개발하고 공개 소프트웨어 프로젝트를 중심으로 솔루션을 개발하고 있지만, 지금까지는 서버급을 대상으로 한 기업 IT 환경에 그 초점이 맞추어져 왔다.

그러나 급속한 IT 기술의 발전에 힘입어 머지 않은 미래에 도래할 유비쿼터스 서비스 환경에서는 현재보다 더욱 다양하고 복잡한 IT 환경이 조성될 것으로 예상된다. 따라서, 머지않은 미래의 IT 환경을 효과적으로 관리하기 위해서는 다양한 시스템 플랫폼 및 운용 환경에 따른 시스템관리 요구사항을 도출하고 관련 기술을 개발해 나가야 할 것이다.

참고문헌

[1] DMTF, http://www.dmtf.org
[2] Chris Hobbs, A Practical Approach to WBEM/CIM Management, Auerbach Publications, 2004
[3] OpenPegasus Project, http://www.openpegasus.org
[4] OpenWBEM Project, http://www.openwbem.org
[5] SBLIM Project, http://www.sblim.org
[6] WBEMServices Project, http://wbemservices.sourceforge.net
[7] WMI, http://www.microsoft.com/whdc/system/pnppwr/wmi/default.mspx