

# 분산 컴퓨팅 환경에서의 제어/감시 시스템 개발을 위한 기반 구조 설계

## A Design of Infrastructure for Control/Monitoring System in the Distributed Computing Environment

。이 원 구\*, 박 재 현\*\*

\* 인하대학교 자동화공학과 (Tel : 032-863-0442; Fax : 032-863-4386 ; E-mail : iwongu@rcsl.inha.ac.kr)

\*\* 인하대학교 자동화공학과 (Tel : 032-860-7713; Fax : 032-863-4386 ; E-mail : jhyun@inha.ac.kr)

**Abstract:** Recently, due to the advance of computer, network and Internet technology, control/monitoring systems are required to process the massive data. At the same time, the software development environment uses more and more component-based methodology. This paper proposes the services for the control/monitoring domain. Especially we define domain-specific interfaces and categories to acquire compatibility between products, and implement architecture for lightweight event service. As it is very important to support compatibility between heterogeneous systems, the proposed system provides modules for the web service and communication protocols based on the XML. And as proposed architecture consists of cluster of servers and Windows 2000's NLB service, it can guarantee more stable operation.

**Keywords:** distributed system, real-time, xml, dcom, control, monitoring

### I. 서론

오늘날 제어/감시 시스템은 컴퓨터와 네트워크 기술 그리고 인터넷의 발전에 따라 대량의 데이터를 대규모로 처리할 수 있는 능력을 요구 받고 있다. 이러한 시스템을 개발하기 위해서 새로운 개발 방법론뿐 아니라 다양한 서비스를 제공하는 안정된 기반 구조 (infrastructure)가 필요하게 되었다. 이러한 기반 구조를 바탕으로 시스템을 설계하고 개발함으로써 개발 비용 및 시간을 줄일 수 있을 뿐 아니라 같은 기반 구조를 사용하는 다른 시스템들, 즉 분석, 관리 및 보고서 생성 도구와 같은 시스템과의 통합이 비교적 용이해진다는 추가적인 장점이 있다[1].

한편 소프트웨어 개발 환경은 점점 컴포넌트화 되어 가고 있으며 이러한 컴포넌트들의 사용은 소프트웨어 모듈의 재사용성을 높여 주었고 이러한 미리 검증된 모듈의 재사용으로 전체 시스템의 완성도를 높일 수 있게 되었다[2]. 또한 이러한 컴포넌트를 위한 기반 구조 자체가 네트워크에 분산됨으로써, 즉 원격 객체 호출(RMI, Remote Method Invocation)을 지원함으로써 개발자들이 분산 시스템을 작성하는 것이 더욱 용이해졌다. 현재 이러한 컴포넌트를 지원하는 기반 구조로는 OMG(Object Management Group)의 CORBA(Common Object Request Broker Architecture), Microsoft사의 COM/DCOM(Component Object Model/Distributed COM) 그리고 Sun사의 EJB(Enterprise JavaBeans)등이 있다[3,4,5].

이러한 컴포넌트를 위한 기반 구조들이 개발자들에게 많은 서비스들을 제공해 주고 있기는 하지만 대부분의 서비스가 범용성을 추구하고 있어 어떤 도메인에 특정한 서비스를 기대하기는 어렵다.

제어/감시 시스템 역시 이러한 기반 구조들의 서비스만으로는 부족하거나 적합하지 않은 면들이 여럿 있다. 여러 벤더들간의 호환성 역시 제공되는 기본적인 기반 구조만을 바탕으로 하고 있어 같은 도메인 내의 다른 제품들간 호환은 거의 불가능했다. 본 논문에서는 이러한 결여된 서비스와 적합하지 않은 사항에 대해 알아 보고 이를 바탕으로 제어/감시 도메인에 필요한 서비스를 가지는 기반 구조를 설계한다.

앞으로 2장에서는 일반적인 기반 구조에서 제공하는 서비스에 대해 좀더 자세히 알아 보고 3장에서는 제어/감시 도메인에 필요한 서비스와 그러한 서비스를 이용하는 시스템의 구성에 대해 알아 본다. 그리고 4장에서는 이러한 서비스를 제공하는 기반 구조의 간단한 프로토타입의 구현에 대해 알아 보고 마지막으로 5장에서 결론과 앞으로의 연구 방향에 대해 논의한다.

### II. 기반 구조

#### 2.1 컴포넌트 기반 구조

컴포넌트 기반의 개발 방법들이 나오기 이전의 객체 지향 개발 방법들은 여러 가지 문제점을 가지고 있었다. 대표적인 것으로 투명한 객체 모델(white box model)의 사용으로 인한 모듈 재사용의 어려움과 코드의 공개 및 버전 관리 문제, 배포의 문제, 호환성의 부재 등이 있었다. 이러한 단점을 보완하기 위해 나타난 컴포넌트 기반의 개발 방법은 불투명한 객체 모델(black box model)을 사용하여 앞에 열거한 문제점들을 제거하거나 개선하였으며 현재 몇 가지의 컴포넌트 개발 모델들이 널리 사용되고 있다.

한편 기존에는 분산 시스템을 개발하기 위해서는 주로 소

본 논문은 선도 기술 개발 사업(G7-5-3-1-4)의 지원으로 수행된 연구 결과를 포함하고 있습니다.