

이종 분산환경하에서의 정보관리시스템 구축에 관한 연구*

정석찬**, 조장혁**, 우훈식**, 주경준**

A Study on the Development of Information Management System On the Heterogeneous-Distributed Environment

Seok-Chan Jeong, Chang-Hyuk Cho, Hoon-Shik Woo, Kyung-Joon Ju

Abstract

현재의 기업간 비즈니스의 대상은 점차 대규모화되며, 범위도 지역적으로 넓은 지역에 분산되고 있다. 이러한 지리적 한계를 극복하기 위하여 현재 네트워크의 표준 사양이 되고 있는 인터넷/웹의 사용이 적극적으로 검토되고 있다. 그리고, 분산환경하에서의 응용 소프트웨어의 통합을 위하여 미들웨어로서 CORBA의 적용이 검토되고 있다. 본 연구에서는 컴퓨터 환경이 서로 상이하며 분산된 환경하에서의 정보관리 및 공유시스템에 대하여 CALS의 CITIS 개념을 사용하여 구현한다. 이 시스템은 다사용자간의 정보관리를 위한 DM(Document management) 엔진과 기업간 업무 프로세스의 진행에 따라 정보의 흐름과 사용자간의 연계 연계를 지원하며 정보의 유연한 전달을 지원하는 워크플로우 관리기(Workflow Management System), 사용자간의 정보 사용에 관한 이력을 관리하는 CM(Configuration Management) 관리기 및 분산환경하에서 다기업간에 수행되는 프로젝트를 관리하는 글로벌 프로젝트관리기(Global Project Management)로 구성된다.

1. 서론

최근 정보기술을 베이스로 한 전략적인 개념이 등장하고 있는 중에서, 1985년 미국방성을 중심으로 한 CALS(Continuous Acquisition and Life-cycle Support 또는 Commerce At Light Speed) 개념은 새로운 시대의 비즈니스 스타일로 전 세계적으로 주목으로 받고, 현재는 미국, 일본 및 EU를 중심으로 그 구현에 대하여 적극적으로 추진 중이다[2,4,16].

CALS 개념은 계획, 설계, 제조, 조달, 유지/보수, 유통, 판매, 폐기에 이르기까지의 생산활동의 모든 부문에 걸쳐서 정보의 표준화 및 공유화를 도모하여 생산성 향상 및 고객 서비스 향상을 도모하는 것으로, 초기의 CALS는 미국방성의 방위시스템을 대상으로 한 정보공유시스템 구축이

* 본 연구는 1997년도 정보통신부 "CITIS/CALS 통합 DB 기술 개발" 과제의 일부 연구 내용임

** 시스템공학연구소 시스템통합연구부 CALS 연구실

목적이었다.

그러나, 현재는 이러한 정보 공유 환경 구축은 기업의 비즈니스 대상이 점차 대규모화되며, 범위도 지역적으로 넓은 지역에 분산되어 다국적화 및 글로벌화됨에 따라 전 세계의 정보 시스템을 통합화하는 정보관리시스템, 즉 21세기의 제조 패러다임이 되는 가상기업의 구현이 요구되고 있으며, 가상기업의 구현에 CALS 개념이 유용하게 적용되고 있다.

이렇게 각 기업별로 분산되어 있는 정보 시스템을 통합하는 것은 각 정보관리시스템이 물리적으로 분산되어 있으며, 정보관리시스템이 사용되는 H/W, S/W, DBMS, Database 및 정보 교환에 사용되는 N/W 프로토콜의 이질성에 의하여 이질, 분산형 정보관리시스템의 형태가 된다.

그리고, 현재 급속하게 발전하고 있는 정보기술이 이러한 이질, 분산형 정보관리시스템의 구현을 가능하게 하고 있다.

이 중, 인터넷이 급속하게 보급되어 사실상의 네트워크의 세계 표준으로 된 것이다. 인터넷의 보급을 더욱 촉진시키게 된 웹(World-Wide-Web)의 보급은 누구나 쉽게 인터넷의 접속을 가능하게 하여, 인터넷의 대중화가 구현되게 되었다. 그리고, 기업 비즈니스의 영역이 점차 넓은 지역으로 확대됨에 따라 인터넷을 활용한 정보관리시스템의 구축을 추진하고 있다.

이러한 인터넷상에 구현되는 어플리케이션의 개발 언어로서 현재 JAVA[7,27,28,29]가 주목을 받고 있다. JAVA로 개발된 어플리케이션은 웹을 기본 환경으로 구현되기 때문에 플랫폼에 의존되지 않고 어느 플랫폼에도 실행 가능하여 분산 어플리케이션 개발에 적합하다. 그리고, 어플리케이션이 어디에 존재하고 있든, 개발자가 누구이든 관계없이 상호간 통신을 보장함으로써 다양한 H/W, S/W가 동시에 존재하는 이종 분산환경을 지원하는 미들웨어 기술로써 CORBA(Common Object Request Broker Architecture)[20,21,26,30,33]가 이종 분산형 정보관리시스템의 개발에 적극적으로 검토되고 있다.

즉, 이종 분산형의 정보관리시스템의 구현에는 네트워크 환경을 인터넷으로 하며, 어플리케이션 개발 언어를 JAVA로 사용하고, 개발된 어플리케이션의 통합을 CORBA로 사용하여 구현하는 것이 현재의 정보기술을 사용한 하나의 솔루션이라고 할 수 있다.

CALS 개념에서도 이종 분산형 정보관리시스템의 개념을 CITIS(Contractor Integrated Technical Information Service)로 제시하고 있다. 미국방성에서 사용된 CITIS는 미국방성의 MIL Spec인 MIL-STD-974[18]로 조달자가 되는 정부와 수주자가 되는 미 방위산업체간에서 정보의 자동 교환 및 공유의 개념이었지만, 현재는 기업간의 비즈니스에 사용되는 산업정보의 효율적인 교환 및 공유를 위한 개념으로, 그 구현이 추진되고 있다[3,5,6,9,10,11,14,15]. 즉, CALS 개념의 최종적인 구현 목표인 세계 수준에서의 정보공유체계를 구축에 CITIS 개념을 활용하고 있는 것이며, 현재의 기업 환경은 다양한 H/W, S/W가 혼재하는 이종 분산형 환경이라고 할 수 있다.

따라서, 이종 분산형의 정보관리시스템의 구현을 위한 개념은 CITIS의 개념을 활용하며, 개발환경을 인터넷/웹, JAVA, CORBA의 정보기술을 이용함으로써 구현 가능하다.

본 연구에서는 CITIS의 개념과 인터넷/웹, JAVA, CORBA를 사용한 이종 분산형 정보관리시스템을 지원하는 CITIS 서버의 프로토타입을 개발한다.

2. CITIS 개념에 의한 이질, 분산 정보관리시스템 개발

CALS는 종래의 CIM(Computer Integrated Manufacturing)과 같은 기업 내부 중심의 정보관리에서 외부 기업과의 정보 교환 및 공유가 가능한 이종 분산형 정보관리시스템을 구현함으로써 생산성을 향상시키며, 궁극적으로는 H/W, S/W 등의 물리적인 제약 조건을 한정되지 않고 정보로 결합하여 구성되는 가상기업의 구현을 목표로 한다. 그림 1은 CIM과 같은 기업 내부 정보관리시스템에서 가상기업으로의 구현절차를 나타낸 것이다.

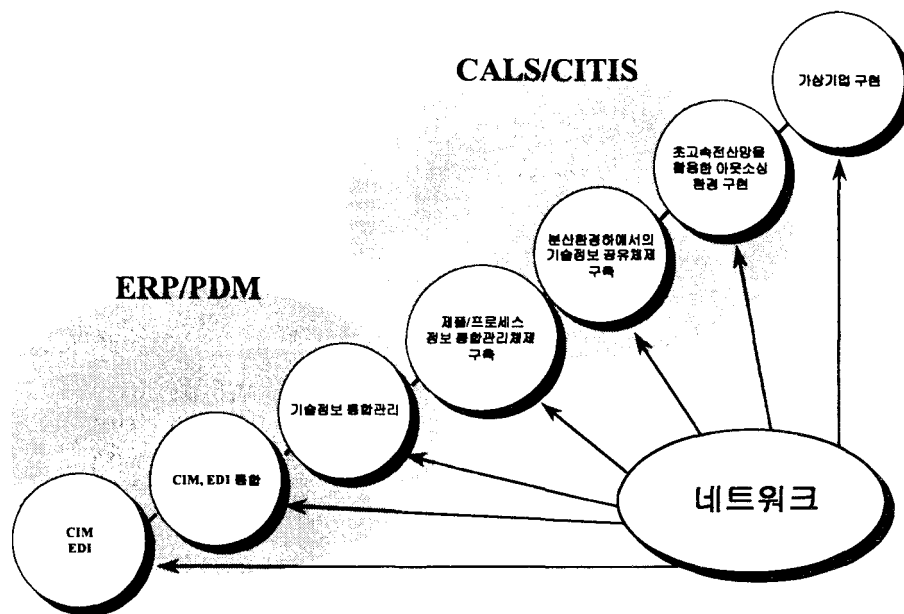


그림 1. CIM에서 가상기업으로의 구현 절차

CIM은 각자의 기업환경에 적합한 최적의 정보관리환경을 구축하여 생산성을 향상시키는 개념으로, 타 기업과는 차별화되고 독립적인 형태로 구축되었다. 그러나, CIM이 기업 내부만에 특화된 정보관리시스템으로 구축되었지만, 기업간의 기업활동에서는 일부적이지만 EDI(Electronic Data Interchange)와 같은 규정된 형식에 따라 상거래에 필요한 비즈니스 정보의 교환이 수행되고 있다. 이렇게 독립적으로 운영되어 온 CIM과 EDI를 통합화함으로써, 비즈니스 정보를 중심으로 한 기업 내부 정보관리시스템과 외부시스템과의 연계가 가능하게 된다. 그리고, 이러한 비즈니스 정보의 교환뿐만 아니라, 기업활동이 점차 글로벌화됨에 따라 제품정보의 교환도 급속하게 증가하고 있으며, 이러한 제품정보의 효율적인 관리를 위하여 PDM(Product Data Management) 시스템이 구축되고 있다[1]. CIM과 EDI의 통합 및 PDM 시스템의 구축은 기업간 정보교환체계 구축을 위한 효율적인 내부 정보관리시스템의 구축이라고 할 수 있으며, 현재는 통합정보관리시스템 구축의 개념으로 ERP(Enterprise Resource Planning) 시스템으로 구축되고 있다.

CALS는 기업간 정보공유체제를 구축하는 것으로, 여기에는 제품정보와 프로세스정보를 통합관리체제를 구축하는 것이 필요하다. 종래에는 설계에 활용되는 제품정보와 제조에 활용되는 프로세스정보가 분리되어 관리되었지만, 제품정보와 프로세스정보를 통합 관리함으로써 모든 생산활동의 통합이 가능하게 된다. 이러한 제품/프로세스정보의 통합관리체제 구축은 CALS의 표준에 의한 데이터의 표준화 작업으로 STEP(Standard for the Exchange of Product Model Data)[9] 활동이 대표적이라고 할 수 있다. 이렇게 기술정보가 제품정보 및 프로세스정보가 통합 관리가 가능하게 되면, 이러한 기술정보를 분산환경하에서 공유할 수 있는 체제를 구축하는 것이 필요하다. CITIS는 분산환경하에서의 정보공유체제를 구축하기 위한 CALS에서의 개념으로, 초기의 CITIS 개념이 미국의 군수시스템과 방산업체간의 단순한 조달정보의 공유체제 구축에서 현재는 기업간의 생산활동을 지원하며, 가상기업의 구현을 지원하는 개념으로 발전하고 있다. 이렇게 CITIS에 의하여 정보공유체제가 구축되며, 현재 인터넷과 같은 글로벌 네트워크환경하에서 전 세계적 아웃소싱(Outsourcing) 환경이 구현되며, 이것이 곧 가상기업의 구현이라고 할 수 있다.

3. CITIS Server

CITIS는 분산환경하에서의 정보공유체제를 구축하기 위한 CALS에서의 개념으로, 초기의 CITIS 개념이 미국의 군수시스템과 방산업체간의 단순한 조달정보의 공유체제 구축에서 현재는 기업간의 생산활동을 지원하며, 가상기업의 구현을 지원하는 개념으로 발전하고 있다

이를 위해서는 이질적인 분산환경하에서 데이터를 공유하려는 기술들이 필요하며, 이러한 기술의 기반에는 WWW의 안정성과 아울러 JAVA와 같은 인터넷 엔지니어링 언어의 확산이 주목받고 있다. 이러한 기술적인 발전은 목표로 하는 시스템을 구축하기 위해서 다양한 측면에서 개발되는 기술들을 통합하는 것이 주요 목적이다. 이러한 시스템 통합 기술로서 현재 WWW와 CORBA 그리고 JAVA를 기반으로 하여 비즈니스 솔루션을 개발하는 것으로 모아지고 있다.

3.1 CITIS Server Architecture

CITIS는 CALS 구현을 위한 정보공유환경을 제공하는 것으로, 미국방성 조달정보의 공유환경체제에서 기업간에서 기술정보를 공유하기 위해서는 MIL-STD-974에 규정된 CITIS 기능을 보다 일반화할 필요가 있다. 본 연구에서는 CITIS를 이종 분산환경하에서의 기업간 기술정보 공유 및 기업활동을 지원하는 시스템으로 규정하여[5,6], 그림 2과 같은 CITIS 서버의 구현 모형을 제시한다. CITIS 서버는 기업간의 정보활동을 크게 2가지로 분류하여 데이터 모델링과 프로세스 모델링으로 구분하였다. 이것은 Object Oriented Technology에서 Object를 어트리뷰트와 매소드로 표현하는 것과 마찬가지로 기업간의 정보활동을 데이터적인 측면에서 접근방법과 프로세스 중심의 접근 방법으로 구분한 것이다. 데이터 모델링 측면의 기업활동으로는 Product Structure를 기반으로 하는 Configuration Management와 비정형정보공유를 위한 Document Management로 분리하여 접근하기로 한다. 기업간의 프로세스 모델링으로는

Workflow Management 와 Project Management로 구분하여 기업간에 발생할 수 있는 업무시스템을 모델링한다.

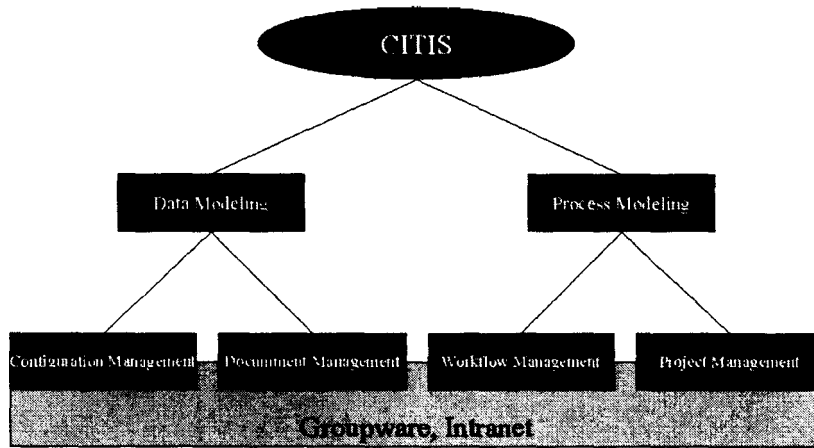


그림 2. CITIS Server Architecture

3.2 Configuration Management

CITIS에서는 제품 및 구성품의 정보를 설계, 제조, 조달 및 운용, 폐기의 Life-cycle에 걸쳐 일관성 있게 관리하여야 한다. 이를 위해서는 형상관리가 절대적으로 필요하며 형상관리를 위해서는 제품의 Product Structure 와 Configuration Management 기능이 필요하다. 이 형상관리 정보는 정부/주계약업체와 협력업체, 고객간에 유기적으로 저장, 관리, 교환되어야 한다.

형상관리는 다음과 같은 2가지 개념을 가지고 있다[13,17,19,22,25].

- (1) 형상항목(Configuration Item)과 형상항목을 정의하는 기술적 자료인 형상문서(Configuration Documentation)에 대한 형상관리
- (2) 일반적인 디지털 자료에 대한 형상관리원리의 응용

그림 3은 전체적인 형상관리 프로세스간의 관계를 IDEF-0 방법으로 표현한 그림이다. 형상관리의 프로세스는 크게 형상관리 운영(CM Management), 형상인식(Configuration Identification), 형상변경통제(Configuration Control), 형상상태회계(Configuration Status Accounting), 형상감사(Configuration Verification and Audit)의 5가지로 구성되어 있다.

3.3 Document Management

CITIS에서 사용하는 정보는 종래의 문서 중심의 정보이외에도 그 내용면에서 볼 때 일반문서, 도면, 기술시방서, 기술보고서, 연구보고서, 매뉴얼, 규정 등이며, 그 형태면에서 볼 때 스캐닝된 이미지, CAD 시스템으로 작성한 벡터 도형, 문자, 멀티미디어 등으로 다양하게 분류될 수 있는 비정형정보이다. 그리고, CITIS에서는 이러한 비정형정보가 물리적으로 분산되어 있지만, 논리적으로는 결합된 정보이다. 따라서, CITIS가 추구하는 산업계 차원에서의 정보 공유를 위해서는 이러한 비정형 정보의 효율적인 관리 운영을 위한 시스템이 필요하며, MIL-STD-974로 규정된 CITIS 정보관리 및 서비스를 제공한다.

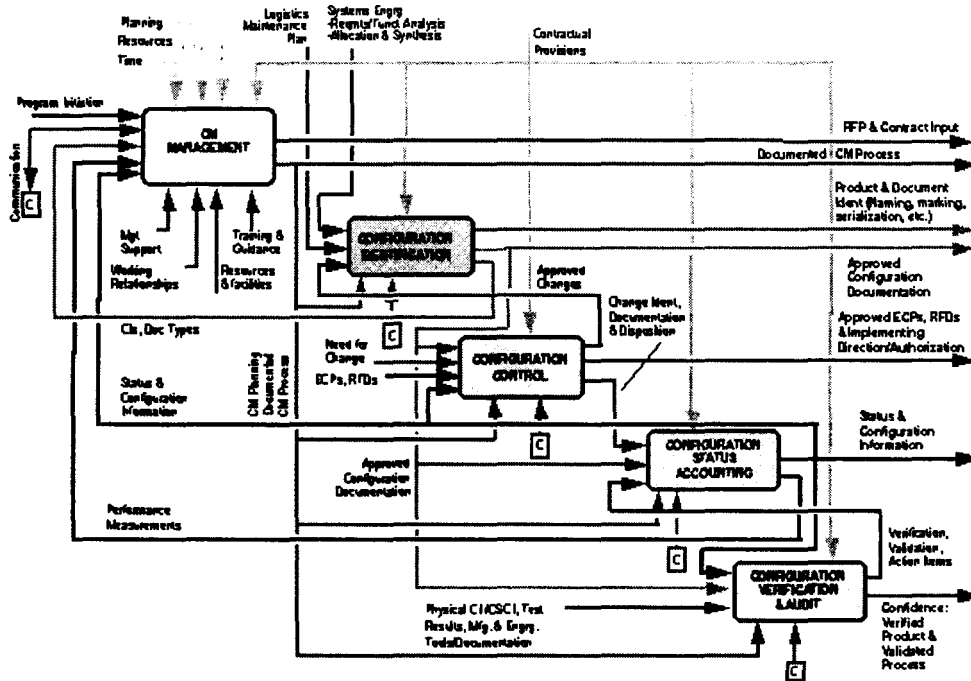


그림 3. Configuration Management Process의 관계

3.4 Workflow Management

워크플로우는 모든 업무 분야에서 문서, 정보, 업무 등과 관련된 업무 담당자들 사이에 수행되는 제반 업무 활동의 자동화된 과정이라 말할 수 있다. 워크플로우 시스템은 조직의 중요한 비즈니스 과정을 사정, 분석, 모델링, 정의하는 등의 BPR(Business Process Reengineering) 활동과 연계하여 생겨나기 시작하였다. 실제로 BPR 활동 중 워크플로우 시스템을 구현하는 경우도 많으며, 워크플로우 시스템은 업무 절차를 정의하고, 정보 활동을 지원하며, 업무 프로세스를 규정하고 있는 여러 절차적 룰에 따라 시스템을 적합화 시키는 도구로 유효하다.

워크플로우를 관리한다는 것은 정보의 흐름과 관련된 부서 및 관련자들 간의 업무 연계를 지원하고, 정보의 유연한 전달을 위한 기능들을 제공하는 것이다. 워크플로우 관리 기술이 업무 프로세스를 위한 모델을 제시해주고, 업무의 실행과 관리를 지원하는 해결책을 세우는 기반을 제공하기 때문에 지난 몇 년 동안 많은 주목을 받아왔다. 조직에서의 프로세스는 비슷한 여러 종류의 경우들을 다루는 활동들의 집합을 포함하고 있는데, 이는 대개 조직과 부서의 경계를 넘나드는 경우가 많다. 워크플로우 프로세스는 정보 기술을 사용하여 프로세스 내에서 정보의 통합, 통신, 제어가 이루어지는 자동화된 조직의 프로세스이지만, 프로세스 내의 활동들은 인간이나 자동화, 혹은 양자 모두가 될 수 있다. 워크플로우 시스템은 소프트웨어가 실행되는 동안 워크플로우를 정의하고, 이를 수행하게 하고, 관리하며, 업무 수행 과정을 일목요연하게 보여주는 시스템이다.

워크플로우 관리연합(WFMC)[12,24]에서 제시하는 워크플로우 참조모델에 따르면 전체적으로 워크플로우 시스템은 다음의 그림 4에서 보는 바와 같이 Build Time시의 프로세스를 디자인하고

정의하는 영역과, Run Time시에 프로세스를 실행시키고 통제하는 영역, 그리고 업무에 참여하는 작업자들과 애플리케이션들과의 인터페이스를 담당하는 영역의 세 가지 기능영역을 가지고 있다. 또한 워크플로우 관리연합은 워크플로우 시스템간의 상호운용성(interoperability)확보를 위한 다섯 가지의 인터페이스를 제안하고 있다.

- 인터페이스 1: 업무정의 도구와 워크플로우 엔진 간의 인터페이스
- 인터페이스 2: 워크플로우 클라이언트 프로그램과 워크플로우 엔진간의 인터페이스
- 인터페이스 3: 워크플로우 시스템이 업무를 수행하기 위해 사용하는 외부 애플리케이션과 워크플로우 엔진 간의 인터페이스
- 인터페이스 4: 다른 워크플로우 엔진 간의 인터페이스
- 인터페이스 5: 워크플로우 시스템을 제어하고 감사하는 관리 도구와 워크

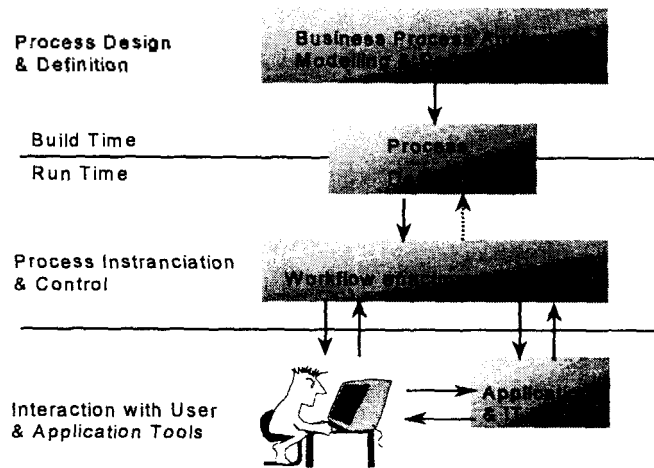


그림 4. WfMC의 Workflow 참조 모델

3.5 Project Management

프로젝트관리기술은 과거 PERT/CPM 기법 이후로 꾸준히 발전하여 다양한 형태의 프로젝트의 계획 및 관리에 응용되어 왔다. 흔히 프로젝트관리기술이라고 하면 일정계획 기법인 PERT/CPM 만을 생각하는 경우가 많지만 이는 단지 프로젝트관리기술 중의 극히 일부분일 뿐이다. 프로젝트관리기술의 교육, 표준, 기술 발전, 국제 협력을 위한 국제기구인 PMI(Project Management Institute)[31]는 프로젝트관리기술을 대략 다음과 같이 구분하고 있다.

- 프로젝트 통합관리(Project Integration Management)
- 프로젝트 범위관리(Project Scope Management)
- 프로젝트 일정관리(Project Time Management)
- 프로젝트 비용관리(Project Cost Management)
- 프로젝트 품질관리(Project Quality Management)

- 프로젝트 인력관리(Project Human Resource Management)
- 프로젝트 통신관리(Project Communications Management)
- 프로젝트 위험관리(Project Risk Management)
- 프로젝트 조달관리(Project Procurement Management)

프로젝트관리 프로세스는 크게 다섯 가지의 그룹으로 나누어 볼 수 있다.

- Initiating Process: 프로젝트나 프로젝트의 각 수행단계가 시작되어야 하는지를 인지하고 실제로 시작하는 단계
- Planning Process: 운용할 수 있는 계획을 개발하고 유지하는 단계
- Executing Process: 계획을 수행할 수 있도록 인력 및 다른 자원을 조정
- Controlling Process: 프로젝트가 목표에 맞게 수행되고 있는지 모니터링하고 필요한 경우 적절한 조치를 취한다.
- Closing Process: 프로젝트나 프로젝트의 각 수행단계의 끝을 정식으로 수용하는 단계

프로젝트는 각 개인의 업무 및 과업으로부터 건설 및 플랜트 건설, 신제품 개발, 소프트웨어 개발, 무기체계 개발, 우주개발 등의 대형 복합 프로젝트 엔지니어링에 이르기까지 다양한 형태를 가질 수 있다. 따라서, 이들 프로젝트에 공통으로 적용될 수 있는 프로젝트관리기술 이외에 응용분야별로 특화된 프로젝트관리기술이 있을 수 있다. 그림 5는 프로젝트 관리시스템의 개념도이다.

4. 이종 분산형 정보관리시스템 구현을 위한 정보기술

기업간의 정보활동을 다루고 있는 CITIS는 그 특성상 이질 분산환경에서의 공동작업이 중요하다. 현재까지 이러한 이질 분산환경에서의 정보의 공유를 위하여 다양한 정보기술의 발전이 추진되어 왔고, CALS의 실제 구현에 이러한 정보기술의 발전이 크게 기여하고 있다. 3.에서 기술한 CITIS 시스템을 구축하기 위한 기반 기술은 다음과 같다.

4.1 WWW/Internet

정보관리시스템의 근간 기술로서 자리 매김하고 있는 WWW (World Wide Web)은 안정된 프로토콜을 기반으로 다양한 요구사항 들을 수용하고 기존의 시스템들을 통합하며 발전해오고 있다.

1994년 초가 WWW은 HTTP(Hyper Text Transfer Protocol)과 하이퍼텍스트 인터페이스를 기반으로 텍스트나 이미지, 오디오 등을 포함하는 멀티미디어 HTML(Hyper Text Markup Language) 문서를 제공하였다. 이후 1995년에 들어 WWW는 기존의 데이터나 응용 프로그램을 연동하고 보다 풍부한 서비스를 제공하기 위해 HTML문서 내에 초보적인 사용자 인터페이스를 제공해주는 FORM 태그와 이를 통해 제출된 사용자 요구사항을 외부 응용 프로그램과 연동해주는 CGI(Common Gateway Interface) 방식을 통해 기존의 응용 시스템과 연동되기 시작했다. 아울러 테이블을

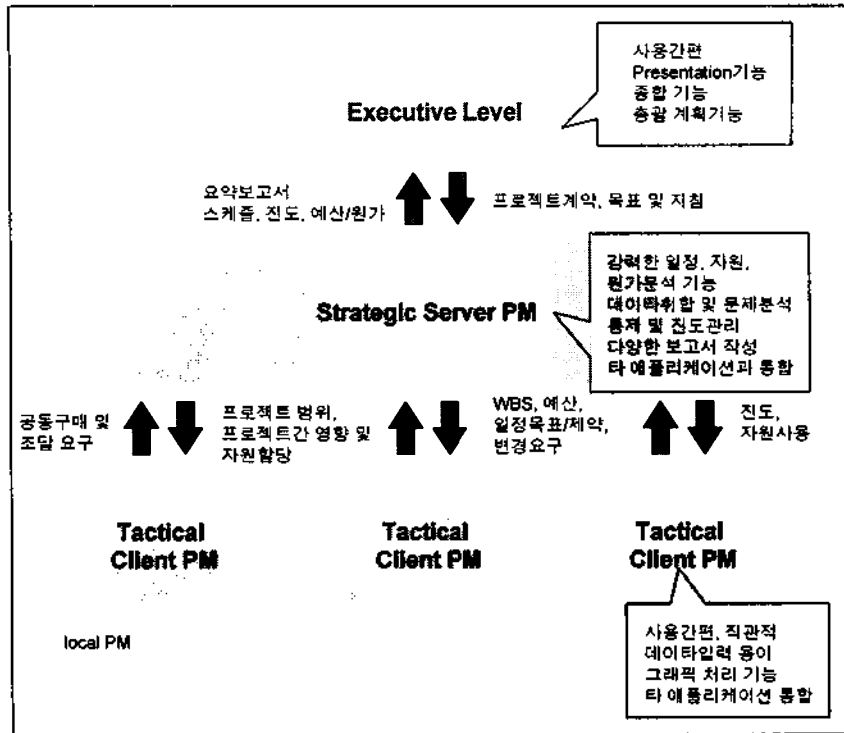


그림 5. 프로젝트 관리시스템의 개념도

비롯하여 강화된 HTML태그들과 보안 기능이 추가되었다.

한편 1996년 선마이크로시스템에서 개발한 JAVA라는 객체지향 인터프리터 언어는 기존의 WWW에 대화형 응용 프로그램을 추가하고 상태정보를 유지할 수 있는 가능성을 제시하였다. 이후, WWW는 JAVA의 이러한 장점을 받아들여 기존의 정적WWW에서 동적WWW으로 변환하기 시작한다. 이 시기의 동적WWW의 특징은 WWW 서버에 위치시킬 수 있는 애플릿이라는 코드와 이 코드를 JAVA가 가능한 웹 브라우저를 통해 다운로드 받은 후 이를 실행시킬 수 있는 모빌코드 환경이다.

1997년에 이르러 WWW는 기존의 JAVA 기술외에 하드웨어나 소프트웨어에 상관없이 응용 프로그램을 개발하고 다양한 개발 서비스를 이용하게 해주는 객체지향 미들웨어인 CORBA를 수용하게 되었다. 좀더 자세히 말하자면 CORBA의 표준 프로토콜인 IOP (Internet Inter-ORB Protocol)와 WWW의 표준 프로토콜인 HTTP의 통합에 의해 WWW상에서 component 서비스를 개발하기 위한 토대를 갖추게 된 것이다. CORBA외에도 JAVA의 component 기술인 Java Beans나 마이크로소프트의 component 기술인 ActiveX가 기존 WWW의 확장을 위해 적극 채택되고 있다. 이들을 묶어 동적 WWW라 한다.

특히 이러한 기술 중에서 WWW과 CORBA, 그리고 CORBA IDL의 JAVA 매핑은 JAVA에서 제공하는 모빌 코드 기능과 CORBA에서 제공하는 객체 component 기술을 적용하여 대규모의 정보 관리시스템을 기존의 WWW상에서 구축할 수 있게 해준다. 이렇게 구축된 정보 시스템을 CORBA 비즈니스 객체라 하며 이러한 개발환경을 ObjectWeb이라 하며, 이종 분산환경하의 정보관리시스

템 구축을 위한 기본 구조로 되고 있다.

4.2 Java

JAVA 언어를 사용하면 인터넷 같은 이질적인 네트워크에서 아키텍처 중립적인 코드를 동적으로 불러오고 실행할 수 있다. JAVA의 이러한 특징 때문에 인터넷과 WWW은 기존의 수동적 성격을 벗어나게 되었다. JAVA가 이러한 기능들을 제공하는 것은 그 아키텍처 내에 플랫폼 독립성, 견고성, 안정성, 객체지향성, 동적 실행과 같은 특징들을 담고 있기 때문이며 이 특징들 덕분에 JAVA는 인터넷의 중심언어로 자리잡았다. JAVA 가상머신(Virtual Machine)은 일종의 소프트웨어 CPU이다. JAVA로 작성된 소프트웨어가 어느 플랫폼에서나 수정 없이 동작할 수 있는 것은 잘 정의된 표준 API와 JAVA 가상머신 때문이다. 그림 6은 JAVA의 가상머신을 설명한 그림이다[7].

하지만 현재 JAVA는 처리 속도가 늦다는 점이 문제가 되지만, 플랫폼에 독립적이며, 이식성이 용이하다는 점에서 이질 분산형 정보관리시스템 개발에 유용한 개발 언어가 된다.

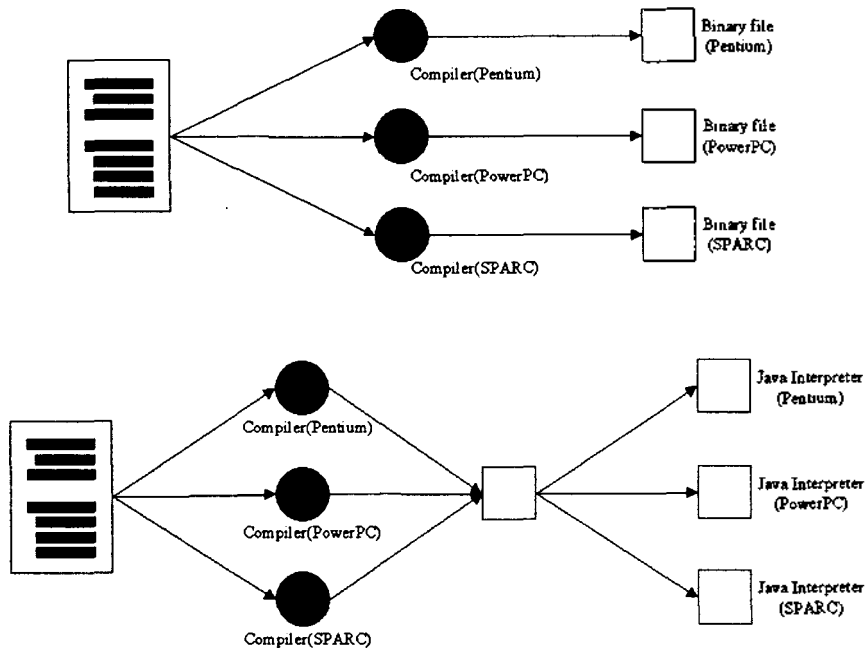


그림 6. JAVA 가상머신(Virtual Machine)

4.3 CORBA

1989년 4월, 현재 존재하는 객체 기술을 바탕으로 응용 프로그램들을 결합하기 위한 객체지향 표준을 제정하기 위하여 OMG라는 비영리단체가 탄생하였다. 이 단체에는 MS사를 비롯하여 600여 개의 컴퓨터 관련 단체들이 참가하여 객체지향 기술을 바탕으로 이종의 분산된 환경하에서 응용 프로그램을 서로 통합할 수 있는 OMA(Object management Architecture)라는 표준기술을 개발하게 되었다. OMA는 응용 프로그램의 결합뿐만 아니라 객체의 생성, 소멸, 저장, 트랜잭션 기능에 이르기까지 분산객체 환경에서 필요한 모든 서비스를 총칭하는 것이다. 이들 기능 중

표 1. JAVA/CORBA ORB와 다른 방법의 비교

Feature	CORBA/IIOP	DCOM	RMI	HTTP/CGI	Sockets
Abstraction level	★★★★	★★★★	★★★★	★★	★
Seamless Java integration	★★★★	★★★	★★★★	★★	★★
OS platform support	★★★★	★★	★★★★	★★★★	★★★★
All-Java implementation	★★★★	★	★★★★	★★★★	★★★★
Typed parameter support	★★★★	★★★★	★★★★	★	★
Ease of configuration	★★★	☆	★★★	★★★	★★★
Distributed method Invocation	★★★★	★★★	★★★	☆	☆
State across invocation	★★★★	★★★	★★★	☆	★★
Dynamic discovery and Metadata support	★★★★	★★★	☆	☆	☆
Dynamic invocation	★★★★	★★★★	★	☆	☆
Performance(remote Pings)	★★★★	★★★	★★★	☆	★★★★
Wire-level security	★★★★	★★★★	★★★	★★★	★★★
Wire-level transactions	★★★★	★★★	☆	☆	☆
Persistent objects	★★★★	★	☆	☆	☆
URL-based naming	★★★	★	★★	★★★★	★★★
Multilingual object Invocation	★★★★	★★★	☆	★★★	★★★★
Language-neutral wire Protocol	★★★★	★★★★	☆	★★★★	☆
Intergalactic scaling	★★★★	★	★	★★	★★★★
Open standard	★★★★	★★	★★	★★★★	★★★★

CORBA는 컴퓨터 내부의 버스처럼 서로 다른 프로그램들 사이의 버스 역할을 하는 모듈로 OMA 구조의 핵심이 된다. CORBA는 네트워크상에 분산된 클라이언트와 서버간의 표준화된 객체 인터페이스를 IDL(Interface Definition Language)를 이용하여 정의하고 클라이언트 및 서버의 객체간의 상호작용을 관리하기 위한 다양한 분산 서비스(Distributed Service)를 제공한다. 표 1은 Java/CORBA ORB(Object Request Broker)와 그 외의 다른 방법과의 비교 분석표이다[20].

4.4 Client/Server 환경

그림 7은 지금까지 앞에서 열거한 기술을 사용하여 CITIS 시스템의 C/S 환경을 정리하여 본 것이다. 크게 Client 와 Server로 분류하여 생각하면 중간에 사용되는 미들웨어로는 HTTP 또는 CORBA가 사용될 것이다. 이 경우 단순한 정보는 HTML을 사용하여 정보를 사용자에게 제공하고 복잡한 어플리케이션이 필요한 경우에는 JAVA 애플릿 또는 ORBlet을 사용하여 서버쪽 프로세스에 CORBA를 통하여 직접 접속할 수도 있다. 클라이언트 모듈로는 Netscape Navigator나 Internet Explorer와 같은 웹 브라우저가 기본모듈에 해당하고 이와 JAVA 애플릿, ActiveX, 또는 Plug-ins 등의 기술이 사용될 수 있을 것이다[23]. 서버쪽에는 일단 WWW 서버가 필요하고 이 서버에서 제공하는 API 함수 또는 CGI, Fast-CGI 등의 기술이 사용되게 될 것이다. 또한 CITIS 서버가 구현하고자 하는 기능적 요소들을 담고 있는 어플리케이션 로직이 서버쪽에 자리잡게 된다. 서버와 DB 사이에도 또 하나의 Layer를 두었는데 이는 CALS 환경에서 얘기하는 IDE(Integrated Data Environment) 환경을 고려한 것이다. 이질분산환경의 특성상 실제 데이터가 저장되는 DB는 여러 곳에 분산되어 있을 수 있다. 이러한 분산 DB는 또한 서로 다른 종류의 DBMS를 사용할 수도 있는데 이러한 문제를 해결하기 위하여 여러 DBMS 시스템에 일관된 인터페이스를 제공하는 라이브러리 함수들이 개발되어 있다[32].

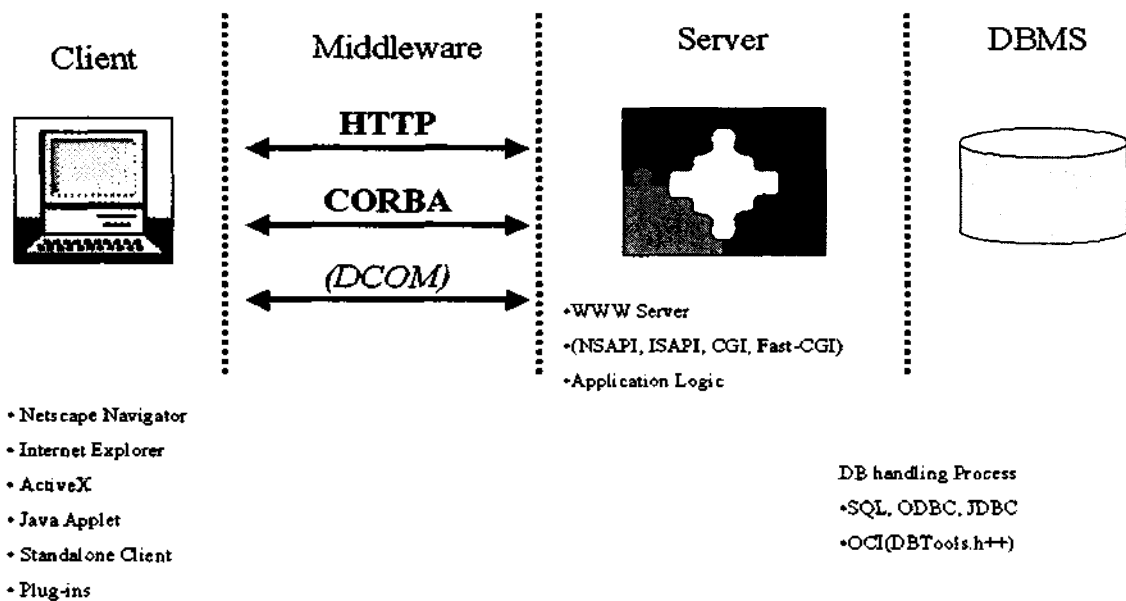


그림 7. Client/Server 환경에 의한 CITIS Server 구현 모형

5. 결론

CALS 는 기업이 정보로 결합하여 형성되는 가상기업의 구현을 목표로 한다. 이렇게 CALS 구현으로 구축되는 가상기업을 지원하는 정보관리시스템은 이종 분산형의 형태로 된다.

즉, 현재의 기업 비즈니스는 지역적으로 분산되어 있으며, 사용되는 시스템도 다양한 플랫폼,

응용 소프트웨어, DBMS 등에 의하여, 그 구현이 용이하지 않다.

따라서, 본 연구에서는 이중 분산형 정보관리시스템의 구현에 CITIS의 개념을 활용하며, 이를 구현하기 위한 기술로서 JAVA, CORBA, 웹을 기반으로 사용한다. 그리고, 이중 분산환경하에서 기업 활동을 지원하기 위한 정보관리시스템으로서 Document Management, Configuration Management, Workflow Management, Project Management로 구성되는 CITIS 서버의 구현 아키텍처를 제시한다.

[참고문헌]

- [1] 김선호, CALS를 위한 제조업의 통합 제품정보 관리, CALS/EC Journal June, pp.10-17, 1996
- [2] 정석찬 역, 後藤明也저, CALS 구상, 도서출판 문원, 1996(CALS構想, 生産性出版, 1995)
- [3] 정석찬, CALS의 정보인프라 - CITIS(Contractor Integrated Technical Information Service), Proceedings of CALS PACIFIC KOREA '96, 1996
- [4] 정석찬, 고도 정보화사회 구현을 위한 CALS 적용에 관한 연구, 한국경영과학회/대한산업공학회 '97 춘계공동학술대회 논문집, pp.633-636, 1997
- [5] 정석찬, 우훈식, 백종명, 주경준, CITIS(Contractor Integrated Technical Information Service) 구현에 관한 고찰, 한국경영과학회/대한산업공학회 '97 춘계공동학술대회 논문집, pp.637-640, 1997
- [6] 정석찬, 우훈식, 조장혁, 백종명, 주경준, CITIS(Contractor Integrated Technical Information Service) 구현에 관한 연구, IE Interface, 1997(계재 예정)
- [7] 최지환, 자바애플릿, 파워북, 1996
- [8] 木村文彦, 小島俊雄, 製品モデル表現とその利用技術 - STEP, 日本規格協會, 1995
- [9] Asahi Yoichi, Implementation of Information Sharing Environment with Bilateral CITIS, Proceeding of CALS Japan '96, pp.9-17, 1996
- [10] CITIS Solution Enables On-line Access to Contractually Required Information, FORMTEK Journal, Fall, Issue 26, 1996
- [11] Commercial Standard for Contractor Integrated Technical Information Services(CITIS), CALS Industry Steering Group Draft, prepare for CALS Industry Steering Group and A Joint
- [12] David Hollingsworth, Workflow Management Coalition The Workflow Reference Model, 1994
- [13] EIA/IS-649, National Consensus Standard for Configuration Management, Electronic Industries Association, 1995
- [14] Heisterberg, R. J., CITIS Implementations and Future Trends in the USA, Proceedings of CALS
- [15] Kitagawa Riichro, Strategy in NCALS, Proceeding of CALS Japan '96, pp.1-8, 1996
- [16] MIL-HDBK-59B, Continuous Acquisition and Life-cycle Support(CALS) Implementation Guide, Department of Defense, USA, 1994
- [17] MIL-HDBK-61, Configuration Management Guidance, Department of Defense, USA, 1997
- [18] MIL-STD-974, Contractor Integrated Technical Information Services(CITIS), Department of Defense, USA, 1993

- [19] MIL-STD-2549, Configuration Management Data Interface, Department of Defense, USA, 1997
- [20] Robert Orfali, Dan harkey, Client/Server Programming with JAVA and CORBA, Wiley Computer Publishing, 1997
- [21] Robert Orfali, Dan harkey, Jeri Edwards, The Essential Distributed Objects Survival Guide, John Wiley & Sons, 1996
- [22] <http://cmis.caci.com/>, CM IS Homepage
- [23] <http://developer.netscape.com/>, Netscape DevEdge Online
- [24] <http://www.aiai.ed.ac.uk:80/project/wfmc/>, Workflow Management Coalition
- [25] <http://www.icmhq.com/>, Institute of Configuration Management
- [26] <http://www.iona.com>, Welcome to IONA Technologies
- [27] <http://www.javasoft.com/>, Java Homepage
- [28] <http://www.javaworld.com/>, IDG's magazine for Java community
- [29] <http://www.oasis.leo.org/java/documentation/faqs/00-index.html>, The Java Oasis : FAQs and HOWTOs
- [30] <http://www.omg.org>, OMG Homepage
- [31] <http://www.pmi.org/>, PMI
- [32] <http://www.roguewave.com/>, RogueWave Homepage
- [33] <http://www.visigenic.com/>, Visigenic Software