웹 기반 그룹 협동 시스템에서 동기화된 협동을 지원하기 위한 능동형 웹 서버 설계

허순영* · 배경일**

A Design of the Active Web Server Supporting Synchronous Collaboration in the Web-Based Group Collaboration Systems

Soon-Young Huh · Kyoung-Il Bae

Abstract

The web-based group collaborative systems are emerging as enterprise-wide information systems. Since data in group collaborative systems are apt to be shared among multiple concurrent users and modified simutaneously by them, the web-based group collaborative systems must support synchronous collaboration in order to provide users with synchronized and consistent views of shared data. This paper proposes an active web server which can facilitate synchronous collaboration in web-based group collaborative systems. To accomplish such a goal, the active web server manages dependency relationships beween shared data and web browsers referencing them and actively propagates changing details of the shared data to all web browsers referencing them. And, this paper examines usefullness and effectiveness of the active web server to apply it to the ball-bearing design example of concurrent engineering design systems. The prototype system of the active web server is developed on a commercial Object-oriented Database Management System (ODBMS) called OBJECTSTORE using the C++ programming language.

* 한국과학기술원 테크노경영대학원
** 한국과학기술원 테크노경영대학원

1. 서론

최근 정보 시스템을 설계하고 구축하는 기반 구조로써 World-Wide Web (WWW)이 두각을 나타내고 있다 [10, 22, 23, 29]. 그 이유는 표준화된 네트웍 프로토콜을 이용하는 웹 (Web) 기반의 시스템이 운영 체제에 독립적이기 때문이다. 또 다른 이유는 개발자가 HTML (Hypertext Markup Language) 문서 [1] 를 이용해서 쉽고 빠르게 사용자 화면을 제작함과 동시에 기존의 프로그래밍 언어를 이용해서 기존 시스템과 수월하게 통합할 수 있기 때문이다. 이러한 장점에 의해 웹을 기반으로 하는 그룹 협동 시스템 (group collaborative systems)을 어떻게 구축해야 할 지에 관한 연구가 동시 엔지니어링 디자인, 전자 결재, 그리고 전자 화상 회의 시스템 등의 분야에서 활발하게 진행되고 있다 [3, 6, 27]. 이러한 웹 기반의 그룹 협동 시스템은 그룹 구성원들이 웹 환경에서 자신의 작업을 독립적으로 수행할 수 있게 함과 동시에 자원의 공동 사용을 활성화하여 전체적으로 작업 능률을 촉진시키는 것을 목표로 한다. 이를 위해서 웹 기반의 그룹 협동 시스템은 공유 자원이 수정되었을 때 공유 자원을 사용하는 그룹 구성원들에게 수정 내역을 즉시 제공하도록 하는 동기화 된 협동 (synchronized collaboration)을 지원해야 하며 [11, 15, 18, 19], 이를 통해서 그룹 협동 시스템은 공유 자원을 이용하는 모든 그룹 구성원들에게 동기화되고 일관성 있는 사용자 뷰를 제공할 수 있다 [5, 7, 12, 13, 15]. 그러나, 웹 기술의 기본 통신 프로토콜인 HTTP (Hypertext Transfer Protocol) 는 단방향 통신 (uni-directional communications)만을 지원하기 때문에 웹 서버는 웹 브라우저의 요구를 받지 않으면 특정 웹 브라우저에게 능동적으로 메시지를 보낼 수 없다. 이러한 수동적인 웹 기술의 한계는 동기화 된 협동을 지원하는 웹 기반의 그룹 협동 시스템의 개발을 매우 어렵게 만든다. 이 논문에서는 이러한 문제점을 지적하고 웹 기반의 그룹 협동 시스템에서 동기화된 협동을 지원하는 능동형 웹 서버를 제안한다. 능동형 웹 서버는 수동적인 웹 기술에 대한 대안을 제시할 뿐만 아니라 기존의 웹 서버 및 웹 브라우저 프로그램을 수정하지 않고 구현되어질 수 있도록 설계되었다. 본 논문은 능동형 웹 서버의 기본 구성 요소들을 쉽게 개념화하기 위해 객체지향 데이타베이스 모델 [14]과 에이전트 접근 방식 [8, 19, 28]을 채택한다. 또한, 능동형 웹 서버의 원형 시스템은 윈도우즈 NT [9] 위에 C++ 프로그래밍 언어 [25]와 객체지향 데이타베이스 관리 시스템인 OBJECTSTORE [14]를 이용하여 구현되었다. 마지막으로 이 논문은 볼 베어링 (ball-bearing) 설계의 예를 통해서 능동형 웹 서버가 동시 엔지니어링 디자인 시스템에서 어떻게 이용될 수 있는지를 설명한다.

2. 능동형 웹 서버 (Active Web Server)

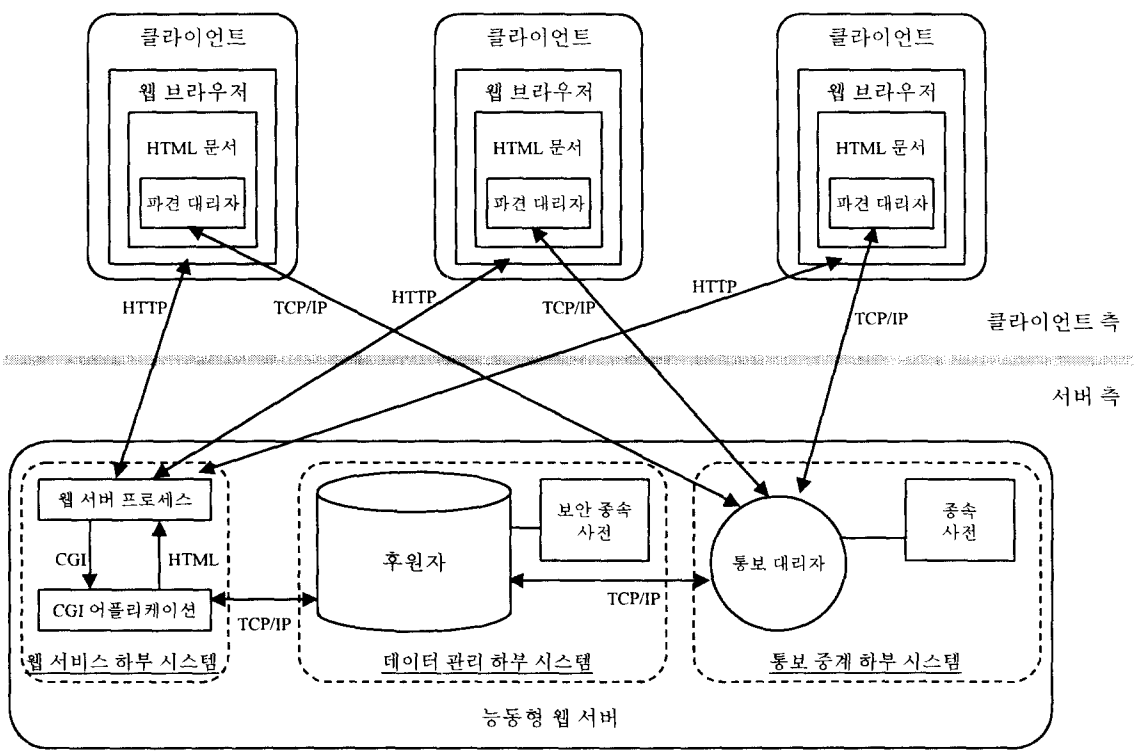
그룹 협동 시스템이 동기화된 협동을 지원한다는 것을 바꾸어 말하면, 공유 자원과 사용자 뷰 사이의 동기성을 유지하는 것이라 말할 수 있다. 그러므로, 본 논문은 공유 객체와 이를 동시에 참조하는 참조 객체 간의 동기화에 관한 모형을 도입하고자 한다. 이러한 목적에서 개발된 모형인 변화 관리 모형[12]은 다중 사용자 환경의 그룹 협동 시스템에 적용하기에 적합한 구조로 개발되었으므로, 본 논문은 변화 관리 모형에서 사용하는 후원자 (supporter), 종속자 (dependent), 그리고 종속 관계 (dependency relationship) 등의 개념과 관련 기법을 이용하고자 한다. 그룹 협동 시스템에서 후원자는 다중 사용자가 참조하는 공유 자원을, 종속자는 후원자를 참조하는 사용자 뷰를, 그리고 종속 관계는 후원자와 이를 참조하는 종속자 간의 동적인 관계를 의미한다. 특히 변화 관리 모형은 다중 사용자 환경에서 동적으로 변화하는 종속 관계를 종속 사전으로 관리한다.

공유 자원과 사용자 뷰 사이의 동기성을 제공하기 위해서는 종속성 관리 기법과 변화 통보 기법이 매우 중요하다 [12]. 먼저 종속성 관리 기법은 후원자와 종속자 간의 종속 관계가 생성되거나, 소멸될 때, 이를 종속 사전에 등록하는 시스템 절차이다. 그리고, 변화 통보 기법은 후원자가 변경되었을 때 변경된 후원자와 종속 관계를 가지는 모든 종속자들을 종속 사전에서 조회한 후, 이들에게 변경 내역을 통보하는 시스템 절차이다. 다음은 이 두 가지 기법을 지원하기 위해서 본 논문에서 제시하는 능동형 웹 서버의 기본 구성요소를 소개한다.

2.1. 능동형 웹 서버의 기본 구성 요소

그림 1은 본 논문에서 제시하는 능동형 웹 서버의 각 구성요소를 개념화하고 설명하기 위한 기본 구조도이다. 공유 자원를 참조하기 위한 HTTP 요청을 웹 서버에게 보내기 위해서 사용자는 웹 브라우저를 이용한다. HTTP 요청을 받은 웹 서버는 CGI 어플리케이션을 통해서 공유 자원 정보가 포함된 HTML 문서를 제작하여 이를 웹 브라우저로 전송한다. 이때 다중 사용자에 의해 동시에 참조되는 공유 자원은 후원자이며, 공유 자원을 참조하는 웹 브라우저는 종속자이다. 그리고, 공유 자원과 웹 브라우저 사이에는 종속 관계가 존재한다.

후원자의 형태는 그룹 협동 시스템의 목적에 따라 다양하게 정의 되어질 수 있다. 예를 들면, 동시 엔지니어링 디자인 시스템은 공동으로 작업하는 도면과 문서 등을 후원자로 정의한다. 후원자를 참조, 혹은 수정할 수 있는 권한을 가진 사용자들은 후원자를 동시에 참조하거나 수정함으로써 서로 긴밀하게 연계되어 작업한다. 또 다른 후원자의 예로는 워크플로우 시스템에서의 개인 정보 폴더와 개인 작업 리스트 등이 있다. 이 경우 동시 엔지니어링 디자인 시스템과는 달리 후원자는 다수의 사용자가 아닌 개인 사용자에 의해



[그림 1] 능동형 웹 서버의 기본 구조도.

소유되며, 소유자 만이 후원자에 저장된 전자 메시지, 혹은 작업 리스트를 읽을 수 있다. 따라서, 일반적인 그룹 협동 시스템에서는 후원자를 소유한 사용자 별로 필요에 따라 서로 다른 접근권한을 주고, 비소유자의 접근을 방지해야 한다. 이러한 요구를 충족하기 위해 본 논문은 보안 관계와 이를 관리하는 보안 종속 사전을 다음과 같이 정의한다.

- **보안 관계**는 후원자와 이를 참조, 혹은 수정할 수 있는 권한을 가진 사용자 간의 접근 권한을 의미한다. 따라서, 보안 관계는 [후원자, 사용자, 접근 권한]의 형태를 가지게 된다. 일반적으로 많이 쓰이는 접근 권한은 사용자가 후원자를 읽고 쓸 수 있는 Read Write (RW), 읽을 수만 있는 Read Only (RO), 그리고 쓸 수만 있는 Write Only (WO) 등이 있으며, 이러한 접근 권한은 시스템의 목적에 따라 더욱 다양하게 정의될 수 있다 [17, 24, 26].

- **보안 종속 사전**은 보안 관계들을 관리하기 위해 만들어진 정보 저장소이다.

능동형 웹 서버는 후원자가 변경되었을 때 후원자를 참조하는 모든 종속자에게 변화 통보 메시지를 보낸다. 이를 위해서 능동형 웹 서버는 종속 관계와 이를 조직적으로 관리하는 종속 사전을 다음과 같이 정의한다.

- **종속 관계**는 후원자와 이를 참조하고 있는 종속자 간의 관계를 의미한다. 그림 1 에서 보듯이 능동형 웹 서버는 변화 통보 메시지를 보내기 위해서 후원자를 참조하고 있는 모든 종속자와의 통신 채널을 유지하므로, 종속 관계는 [후원자, 종속자, 통신 채널]의 형태를 가지게 된다.

- **종속 사전**은 종속 관계들을 관리하기 위해 만들어진 정보 저장소이다.

보안 관계와 종속 관계를 이용하여 능동형 웹 서버는 종속성 관리 기법과 변화 통보 기법을 제공한다. 본 논문에서는 이러한 기법의 절차를 수행하는 통보 대리자 (notification agent)와 파견 대리자 (delegate agent)를 다음과 같이 정의한다.

- **통보 대리자**는 클라이언트와 서버 사이에서 변화 통보의 중개 역할을 하고, 종속 사전 관리와 변화 통보 메시지 전달을 수행한다.

- **파견 대리자**는 통보 대리자로부터 변화 통보 메시지를 수령한다. 능동형 웹 서버는 종속자, 즉 웹 브라우저가 후원자를 참조하고자 할 경우 후원자의 정보를 포함한 HTML 문서와 함께 자바 애플릿 (JAVA applet) [4]으로 구현된 파견 대리자를 종속자에게 전달한다. 전달된 파견 대리자는 통보 대리자와의 통신 채널 생성과 변화 통보 메시지 수령을 수행한다.

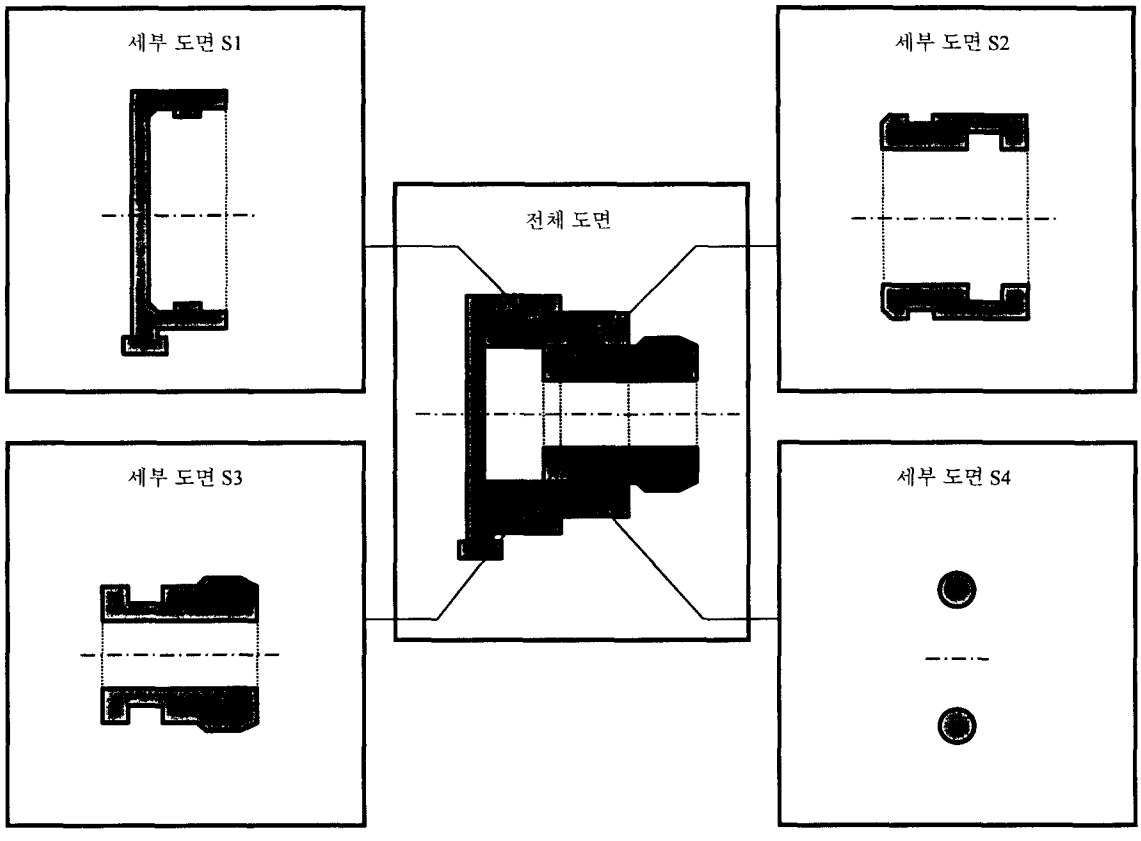
본 논문에서 제시하는 능동형 웹 서버는 그림 1 에서 보는 바와 같이 **웹 서비스 하부 시스템**, 데이터 관리 하부 시스템, 그리고 통보 중계 하부 시스템으로 이루어져 있으며, 이들은 한 컴퓨터에 위치할 수도 있고, 다수의 컴퓨터에 분산되어 있을 수도 있다. **웹 서비스 하부 시스템**은 웹 서버 프로세스와 CGI (Common Gateway Interfaces) 어플리케이션들로 이루어진다. 웹 서버 프로세스는 사용자가 웹 브라우저를 통해서 후원자를 참조하고자 할 경우 적합한 CGI 어플리케이션을 구동한다. CGI 어플리케이션은 후원자 정보와 파견 대리자를 담은 HTML 문서를 제작하여 웹 서버 프로세스에게 전달하고, 웹 서버 프로세스는 이를 사용자의 웹 브라우저로 전송한다. **데이터 관리 하부 시스템**은 후원자와 보안 종속 사전으로 이루어져 있으며, 이들은 데이터베이스 시스템으로 관리된다. 데이터 관리 하부 시스템은 웹 서비스 하부 시스템의 CGI 어플리케이션이 후원자 정보를 요청할 경우 이를 제공하며, 후원자가 변경되면 통보 대리자에게 변화 통보 메시지를 전달한다. 마지막으로 **통보 중계 하부 시스템**은 종속 사전과 통보 대리자로 이루어져 있다. 통보 대리자는 종속 사전을 관리하며 데이터 관리 하부 시스템에서

전달된 변화 통보 메시지를 적합한 종속자들에게 중계하는 역할을 한다.

3. 동시 엔지니어링 디자인 시스템 사례

동시 엔지니어링 디자인 시스템은 전체 도면을 다수의 세부 도면으로 나누어 각 세부 도면을 작업자들에게 할당하고, 각 작업자는 다른 작업자들의 작업 상태를 참조하여 자신이 맡은 부분을 설계한다. 그러므로, 작업자 간의 동기화된 협동을 지원하기 위해서는 동시 엔지니어링 디자인 시스템이 모든 작업자에게 자신이 담당한 세부 도면은 물론, 담당하지 않은 세부 도면에 대해서도 동기화된 뷰를 제공하는 것이 필요하다 [2, 16, 20]. 그림 2에 나타난 볼 베어링 도면의 예는 능동형 웹 서버가 이러한 시스템 요구를 수용함을 보이기 위해 사용된다. 볼 베어링의 전체 도면은 네 개의 세부 도면, S1, S2, S3, 그리고 S4로 나뉘어져 있으며, 각 세부 도면은 전체 덮개 (housing block), 외부 링 (outer ring), 내부 링 (inner ring), 그리고 볼 (ball)을 나타낸다 [21]. 특히 각 세부 도면은 면의 길이와 형태에 대해 다른 세부 도면과 일관되게 설계되어야 한다. 예를 들면, S1와 S2는 서로 연결된 부분에 대해 같은 형태로 설계되어야 한다.

작업자 U1, U2, U3, 그리고 U4가 그림 2에 나타난 세부 도면, S1, S2, S3, 그리고 S4, 각각의 설계를 담당하고 있다고 가정하자. 모든 작업자는 자신이 담당한 세부 도면을 독립적으로 작성함과 동시에 다른 세부 도면도 참조해야 한다. 그러므로, 세부 도면은 담당자에 의해 수정되고 다른 작업자에 의해 참조되는 후원자이고, 표 1은 그림 2의 예를 표현한 능동형 웹 서버의 보안 종속 사전이다.



[그림 2] 볼 베어링 도면 예.

이 보안 종속 사전에서는 각 작업자가 자신이 담당한 세부 도면을 참조 및 수정할 수 있으며 (RW 접근 권한), U1 은 S2 를, U2 는 S1, S3, 그리고 S4 를, U3 는 S2 와 S4 를, 그리고 U4 는 S2 와 S3 를 참조한다 (RO 접근 권한).

<표 1> 그림 2 의 예에 대한 보안 종속 사전.

	U1	U2	U3	U4
S1	RW	RO		
S2	RO	RW	RO	RO
S3		RO	RW	RO
S4		RO	RO	RW

U2 가 웹 브라우저를 통해서 S1 을 참조할 때 능동형 웹 서버는 적합한 CGI 어플리케이션을 구동한다. CGI 어플리케이션은 먼저 보안 종속 사전으로부터 U2 의 접근 권한을 점검한 후 S1 의 도면 정보와 파견 대리자를 담고 있는 HTML 문서를 U2 의 웹 브라우저로 전송한다. U2 의 웹 브라우저로 전달된 파견 대리자는 통보 중계 하부 시스템의 통보 대리자와의 통신 채널을 생성하고, 통보 대리자는 S1 과 웹 브라우저 윈도우 사이의 종속 관계를 종속 사전에 등록한다. 한편, U1 이 S1 을 수정할 때 능동형 웹 서버는 S1 의 하부 도면을 수정하는 CGI 어플리케이션을 구동한다. CGI 어플리케이션은 보안 종속 사전으로부터 U1 의 접근 권한을 점검한 후 S1 의 하부 도면을 수정한다. 수정이 완료되면 변화 통보 기법의 절차를 수행한다. 즉, 데이터 관리 하부 시스템은 S1 의 변화를 나타내는 변화 통보 메시지를 통보 중계 하부 시스템의 통보 대리자에게 보낸다. 이때 통보 대리자는 종속 사전으로부터 현재 S1 을 참조하고 있는 종속자 (웹 브라우저)의 리스트를 얻은 후 리스트에 속한 모든 웹 브라우저에게 변화 통보 메시지를 보낸다. 종속자 측에서는 파견 대리자가 변화 통보 메시지를 수령하고 웹 브라우저를 사용하는 작업자에게 이를 알린다. 만약 U2 가 S1 을 참조하고 있다면, U2 는 파견 대리자를 통해서 S1 이 변경되었다는 것을 알 수 있을 것이다.

이러한 방법으로 능동형 웹 서버는 웹 환경에서 구현된 동시 엔지니어링 디자인 시스템에 동기화된 협동 기능을 제공하고, 동시 엔지니어링 디자인 시스템은 세부 도면별 설계 작업이 병렬적으로 수행될 수 있도록 함으로써 전체 설계의 생산성 향상을 도모한다.

4. 결론

능동형 웹 서버의 장점은 다음과 같이 요약될 수 있다. 첫째, 능동형 웹 서버는 대부분의 협동 시스템에 적용될 수 있는 일반적인 시스템 구조을 제시한다. 이를 증명하기 위해서 본 논문은 능동형 웹 서버가 동시 엔지니어링 디자인 시스템에 적용될 수 있음을 보였다. 둘째, 능동형 웹 서버의 개발 과정에서 소프트웨어 호환성이 고려되었다. 그러므로, 현재 사용되고 있는 웹 서버 프로그램과 웹 브라우저 프로그램이 수정 없이 그대로 이용될 수 있다. 이것은 능동형 웹 서버가 독립적인 세 하부 시스템으로 분리되어져 있고 지능 에이전트 접근 방식을 채택했기 때문이다. 셋째, 능동형 웹 서버는 종속 관계를 관리하고 변화 통보 메시지를 중계하는 데에 있어서 위치 투명성 (location transparency)을 제공한다. 후원자와 이를 참조하는 웹 브라우저 사이를 추상화된 종속 관계로 관리함으로써 능동형 웹 서버는 후원자와 웹 브라우저가 네트웍의 어디에 위치하든지 서로 동기화 될 수 있도록 한다. 이러한 장점들로 인하여 능동형 웹 서버는 다양한 인터넷 분야에 적용될 수 있는 잠재력을 가지고 있다. 특히 능동형 웹 서버가 전자 상거래 분야에서 자동화된 상품 선택 시스템 (automated product selection systems), 대리자 기반의 상거래 시스템 (agent-based commerce systems) 등에

적용될 때 기존의 전자 상거래 시스템은 더욱 다양하고 지능적인 시스템으로 발전될 수 있을 것이다.

참고문헌

[1] Berners-Lee, T. and D. Connolly, "Hypertext Markup Language - 2.0", *Request For Comments 1866*, 1995.

[2] Chang, T.S. and A.C. Ward, "Conceptual Robustness in Simultaneous Engineering: A Formulation in Continuous Spaces", *Research in Engineering Design*, Vol.7, No.2(1995), 67~85.

[3] Coleman, D., "Collaborating on the Internet and Intranets", in *Proceedings of the Thirtieth Annual Hawaii International Conference on System Sciences*, January 7-10, Maui, Hawaii, Vol.2(1997), 350~358.

[4] Cornell, G. and C.S. Horstmann, *Core JAVA*, SunSoft Press, California, 1996.

[5] Cristian, F., "Synchronous and Asynchronous Group Communication", *Communications of the ACM*, Vol.39, No.4(1996), 88~97.

[6] Desborough, J., *INTRANET Web Development*, New Riders, Indiana, 1996.

[7] Ellis, C., S.J. Gibbs, and G.L. Rein, "Groupware: Some issues and experiences", *Communications of the ACM*, Vol.34, No.1(1991), 38~58.

[8] Genesereth, M.R. and S.P. Ketchpel, "Software Agent", *Communications of the ACM*, Vol.37, No.7(1994), 48~53.

[9] Goodman, K.J., *Windows NT: A Developer's Guide*, M&T Books, New York, 1994.

[10] Hardwick, M., D.L. Spooner, T. Rando, and K.C. Morris, "Sharing Manufacturing Information in Virtual Enterprises", *Communications of the ACM*, Vol.39, No.2(1996), 46~54.

[11] Hinden, R.M., "IP Next Generation Overview", *Communications of the ACM*, Vol.39, No.6(1996), 61~71.

[12] Huh, S.Y. and D.A. Rosenberg, "A Change Management Framework: Dependency Maintenance and Change Notification", *Journal of Systems Software*, Vol.34, No.3(1996), 231~246.

[13] Hummon, N.P., "Organizational Structures and Exchange Processes", *Intelligent Systems in Accounting, Finance and Management*, Vol.2, No.4(1993), 235~246.

[14] Lamb, C., G. Landis, J. Orenstein, and D. Weinreb, "The ObjectStore Database System", *Communications of the ACM*, Vol.34, No.10(1991), 50~63.

[15] Migliarese, P. and E. Paolucci, "Improved communications and collaborations among tasks induced by Groupware", *Decision Support Systems*, Vol.14, No.3(1995), 237~250.

[16] Molina, A., A.H. Al-Ashaab, T.I.A. Ellis, R.I.M. Young, and R. Bell, "A Review of Computer-Aided Simultaneous Engineering Systems", *Research in Engineering Design*, Vol.7, No.1(1995), 38~63.

[17] Muralidhar, K., D. Batra, and P.J. Kirs, "Accessibility, Security, and Accuracy in Statistical Databases: The Case for the Multiplicative Fixed Data Perturbation Approach", *Management Science*, Vol.41, No.9(1995), 1549~1564.

[18] O'Leary, D.E., D. Kuokka, and R. Plant, "Artificial Intelligence and Virtual Organizations", *Communications of the ACM*, Vol.40, No.1(1997), 52~59.

[19] Petrie, C.J., "Agent-Based Engineering, the Web, and Intelligence", *IEEE Expert*, Vol.11, No.6(1996), 24~29.

[20] Quadrel, R.W., R.F. Woodbury, S.J. Fenves, and S.N. Talukdar, "Controlling Asynchronous Team Design Environments by Simulated Annealing", *Research in Engineering Design*, Vol.5, No.2(1993), 88~104.

[21] Scheer, A.W., *CIM Computer Steered Industry*, Springer-Verlag, New York, 1988.

[22] Scherer, A., "Supporting Concurrent Engineering Using an Intranet Approach", in *Proceedings of the Thirtieth Annual Hawaii International Conference on System Sciences*, January 7-10, Maui, Hawaii, Vol.2(1997), 589~595.

[23] Second World Wide Web Conference '94, *Electronic Proceedings of the Second World Wide Web Conference '94: Mosaic and the Web*, October 17-20, Chicago, Illinois, 1994.

[24] Shen, H.H. and P. Dewan, "Access Control for Collaborative Environments", in *Proceedings of ACM 1992 Conference on Computer-Supported Cooperative Work*, October 31 to November 4, Toronto, Canada, pp.51-58, New York, 1992.

[25] Stroustrup, B., *The C++ programming language*, Addison-Wesley, Massachusetts, 1986.

[26] Varadharajan, V. and C. Calvelli, "An Access Control Model and Its Use in Representing Mental Health Application Access Policy", *IEEE Transactions on Knowledge and Data Engineering*, Vol.8, No.1(1996), 81~95.

[27] Wei, K.K., B.C.Y. Tan, C.L. Sia, and K.S. Raman, "Hypertext: A New Approach to Construct Group Support Systems", *International Journal of Information Management*, Vol.16, No.3(1996), 163~181.

[28] Wooldridge, M. and N.R. Jennings, "Intelligent Agents: Theory and Practice", *Knowledge Engineering Review*, Vol.10, No.2(1995), 115~152.

[29] Yuwono, B. and D.L. Lee, "WISE: A World Wide Web Resource Database System", *IEEE Transactions on Knowledge and Data Engineering*, Vol.8, No.4(1996), 548~554.