
대학의 인적·지적 자원의 효과적 활용을 위한 지식관리시스템의 구현

최재원*

Implementation of the Knowledge Management System to Effectively Utilize the Human and Intellectual Resources of Universities

Jae-Weon Choe*

요 약

오늘날의 정보이용자는 단순 정보 차원 이상의 구체적인 지식을 필요로 하고 있다. 본 논문에서는 지식정보의 메카인 대학의 인적·지적 자원의 효과적 활용과 원활한 지식의 공유 및 교류를 가능하게 하는 지식관리시스템을 최소한의 비용으로 효과적으로 구축하기 위한 시스템의 구현방법에 대해 연구하고 이를 기초로 시스템을 구현하였다.

ABSTRACT

On these days concrete knowledge more than simple information is required in order to do something. In this paper we researched the development methods and design of the Knowledge Management System(KMS) to effectively utilize the human resources and various kinds of intellectual resources of the universities, which is the mecca of the knowledge and technology. We gave our focus to the methodologies that makes it possible to implement the KMS with the least costs based on the available databases and infrastructures.

키워드

지식관리시스템, KMS, Knowledge Management System, 종합정보시스템

1. 서 론

오늘날 정보화 시대에 살고 있는 정보이용자는 어떤 일을 수행하기 위해 단순 정보 차원 이상의 구체적인 지식을 필요로 하고 있다. 21세기 지식사회에서는 지식만이 유일한 자원으로 새로운 시대의 최대 경쟁무기가 될 것이다. 지식정보화 시대를 맞이하여 지식정보의 산실이라 불릴 수 있는 대학을 위시한 각급 교육기관의 전문인력과 지식정보를 기업, 교육기관, 연구소, 일반이 교류하고 활용할 수 있는 사회적 인프라의 구축이 시급히 요구되는 상황에 있다.

따라서 본 논문에서는 지식정보의 메카라 불릴 수 있는 대학의 인적·지적 자원을 효과적으로 활용할 수 있게 하는 지식관리시스템을 최소한의 비용으로 효과적으로 구축하기 위한 시스템의 구현방법에 대해 연구하고 이를 기초로 시스템을 구현하였다.

본 연구의 목적 달성을 위해 본 논문에서는 먼저 기존 정보시스템과 지식관리시스템의 차이점과 기존의 정보시스템이 안고 있는 문제점을 파악한 후, 시스템 개발방향을 설정하고 시스템의 요구사항을 분석하였다. 또한 지식관리시스템의 개발에 앞서 이에 대한 개념을 명확히 한 후, 최소 비용의

효과적 시스템 구축을 위한 구현방법을 제안하였고, 이에 따라 시스템을 구현한 후 향후 유사 시스템 개발에 도움이 되도록 시스템 구현을 위한 응용 프로그램의 핵심 소스를 상세히 기술하였다.

II. 지식관리시스템

지식정보의 메카라 불릴 수 있는 대학의 가능한 모든 유무형의 지식정보를 교수는 물론 기업·학생·일반인이 활용 가능하도록 하는 지식관리시스템을 최소한의 비용으로 효과적으로 구축하기 위해 기존의 정보시스템이 안고 있는 문제점을 파악하고 이의 개선을 위한 시스템 개발방향을 설정하고 시스템 요구사항을 분석하였다.

1. 지식관리시스템

지식이란 얻어진 정보를 바탕으로 새로운 것을 창조하는 능력으로 정보(Information) + 상황이해(Context) + 경험(Experience)으로 정의되며, 단순한 수치나 기록 등의 데이터나 이를 조직화한 정보에 상황이해와 경험이 더하여져 가치 부가적인 응용정보를 창출할 수 있는 것이 지식이다.[1]

지식은 존재형태에 따라 명시적 지식(explicit knowledge)과 암묵적 지식(tacit knowledge)으로 나눌 수 있다.[2] 명시적 지식은 논리적 추론이나 계산 등에 의해 검증된 문서로 표현될 수 있는 객관적이고 논리적인 디지털(이론적) 지식을 말하고, 암묵적 지식은 말이나 언어로 표현하기 어려운 주관적이고 경험적인 아날로그(실용적) 지식을 의미한다.

예전에는 정보의 희소성이 정보의 가치를 결정하는 요인으로 작용하였으나, 오늘날 정보쓰레기라는 용어가 등장할 정도로 정보과잉으로 인한 부작용이 심각한 상황에 와 있어, 이제는 정보의 홍수 속에서 가치있는 정보에 대한 선별이 불가피하게 되었다. 이러한 가치있는 정보에 대한 요구는 곧 지식에 대한 요구, 지식관리시스템에 대한 요구로 변화되고 있다.

지식관리시스템(KMS: Knowledge Management System)은 조직내의 인적자원들이 가지고 있는 각종 지식들을 체계화해서 쉽게 공유할 수 있도록 구성한 일종의 정보시스템으로 지금까지 구축된 객관적 정보 전달 체계 외에 주관적 의견 전달 및 수용을 위한 체계를 포함하고 있어야 한다. 최근 지식관리시스템의 가장 두드러진 특징은 어떠한 특

정 부분에 대한 전문성의 시스템화보다 조직 내에서 공유·교환·검증·수용의 절차를 거쳐 확신을 심어주는 과정을 단축시켜 궁극적인 경쟁력을 강화하려는 데 있다.[3]

2. 시스템 요구사항

지식관리시스템의 개발을 위해 기존의 정보시스템이 안고 있는 문제점을 파악하고 이의 개선을 위해 다음과 같이 시스템의 개발방향을 설정하였다.

첫째, 본 지식관리시스템은 대학 내의 가능한 모든 인적·지적 자원을 효과적으로 활용할 수 있도록 하는 것이 궁극적인 목적이므로 문서화되어 있는 명시적 지식은 물론이고 암묵적 지식 역시 활용 가능하도록 한다. 둘째, 본 지식관리시스템의 구축에 있어 새로운 데이터베이스를 재구축하기보다는 기존 구축되어 있는 데이터베이스를 연계시켜 최소한의 비용으로 효과적인 정보제공이 이루어지도록 한다. 셋째, 대부분의 DB가 처음 구축 시에는 최신 자료이어서 활용가치가 높으나, 일정기간 경과하게 되면 사장되는 실정으므로, 연계된 DB는 각 부처에 의해 스스로 up-to-date가 이루어지도록 하여 자동적인 정보 갱신이 유지되도록 한다. 넷째, 본 시스템의 이용자는 교수는 물론 기업·학생·일반이 될 수 있으므로 정보이용자의 요구에 맞게 여러 DB를 연계하여 통합 서비스를 제공함으로써 이용자 중심의 정보제공이 이루어 질 수 있도록 한다.

대학의 가능한 모든 인적·지적 자원의 효과적 활용이 가능하도록 시스템의 개발방향을 따른 시스템의 구체적인 요구사항은 다음과 같다.

- 1) 대학의 인적·지적 자원의 효과적 활용체제 구축
 - ① 분야별 전문인력 정보제공: 정보이용자에게 필요한 전문기술 인력을 학문분야 별로 검색할 수 있도록 하고, 전문인력에 대한 기본정보를 제공할 수 있도록 한다.
 - ② 학과별 전문인력 정보제공: 정보이용자에게 필요한 전문기술 인력을 학과·전공 별로 검색할 수 있도록 하고, 전문인력에 대한 기본정보를 제공할 수 있도록 한다.
 - ③ 전문인력 연구실적 정보제공: 분야·학과별 전문인력 찾기를 이용하여 검색한 전문인력의 연구실적 보기를 통하여 논문, 저서, 특허 등의 연구실적의 세부정보를 제공할 수 있도록 한다.
 - ④ 전문인력 개발실적 정보제공: 분야·학과별 전문인력 찾기를 이용하여 검색한 전문인력

의 개발실적 보기를 통하여 연구개발 및 사업화 실적 등의 세부정보를 제공할 수 있도록 한다.

- ⑤ 기술자문 정보 및 서비스 제공: 제품과 기술 개발을 위한 해당 분야의 기술자문 인력을 검색할 수 있도록 하고, 기술자문 인력에 대한 기본정보와 세부정보를 확인 후, 직접 전자메일 등을 이용한 자문요청이 가능하도록 한다.
- ⑥ 분야별 기술정보 제공: 제품과 기술 개발을 위해 요구되는 기술을 분야별로 파악하여 대상 기술에 관련된 다양한 기술정보를 제공하는 서비스가 필요하다. 능동적인 지식의 평가와 검증에 위한 메카니즘을 개발하여 지식 베이스를 구축하고, 분야별 카테고리 검색을 통해 실제 기업에서 제품 혹은 기술개발을 위해 요구되는 기술정보가 제공되어질 수 있도록 한다.
- ⑦ 분야별 교육정보 제공: 대학 혹은 연구소의 기업 지원을 위한 교육프로그램 및 기술지도 등의 교육정보를 분야별로 제공할 수 있도록 한다.
- ⑧ 애로기술 자문서비스 제공: 해당 분야의 전문가에게 애로기술에 대한 자문을 구할 수 있는 분야별 게시판이 제공되어야 한다.

- 2) 최소 비용으로 효과적인 정보제공 체제구축
 - ① DB의 자동적인 정보 갱신: DB의 자동적인 정보 갱신을 보장하기 위해 대학의 업적 DB와 연계하여 전문인력의 연구실적, 연구개발 및 사업화 실적에 대한 정보가 제공되어야 한다.
 - ② 시스템 및 개인정보 관리: 웹 페이지 제작을 위한 기본지식이 전혀 없는 초보자나 비전문가라 할지라도 간단한 마우스 조작과 최소한의 필요정보 입력만으로 시스템을 손쉽게 유지·보수할 수 있도록 한다.
- 3) 대학·기업 등 정보이용자 중심의 정보제공 체제구축[4]
 - ① 협력사업 안내: 기업의 기술개발을 위한 자금 및 지원 시책 등 유관 기관의 각종 협력사업에 대한 안내정보와 지원절차에 관한 정보, 기업 관련뉴스, 그리고 대학의 기업지원 프로그램의 안내정보와 지원절차에 관한 정보를 제공해야 한다.
 - ② 산학연 협력실적 정보 제공: 산학협력을 통해

수행한 연구개발에 대한 소개와 수행실적 등의 정보를 제공할 수 있도록 한다.

- ③ 구인·구직 안내: 정규직과 비정규직의 구인·구직 취업정보를 제공하고, 기업의 소규모 과제위탁과 학생 아르바이트를 통한 과제수탁이 가능하도록 정보를 제공한다.
- ④ 기업정보 제공: 각 지역의 업체를 분야별로 분류하여 해당 분야에 관련된 업체의 정보취득이 용이하도록 하고, 업체 홈페이지의 접근을 통해 세부정보의 취득이 가능하도록 한다.
- ⑤ 부가서비스 제공: 자료실, 게시판, 전자우편, 질의·응답을 위한 Q&A, 공지사항과 새 소식을 전하는 알림판 등의 부가서비스가 제공되어야 한다.

III. 지식관리시스템 구현방법

지식관리시스템의 개발에 앞서 이에 대한 개념을 명확히 하고 개발방법을 제안하였으며, 시스템의 구조를 설계하였고, 최소 비용의 효과적 시스템 구축을 위한 구현방법에 대해 연구하였다.

1. 시스템 개발방법

지식관리에 대한 개념과 시스템의 구축방법에 관해 선 의견이 분분하므로 지식관리시스템의 개발을 위해선 이에 대한 개념 설정을 명확히 할 필요가 있고, 지식의 형태에 따른 시스템의 구축방법과 정보의 홍수 속에서 지식의 선별을 위한 방법을 연구하였다.[5]

1) 지식관리시스템의 개념 정의

지식관리에 대한 개념 설정은 물론 실제 구축 방법론과 시스템 구축 형태에 대해서는 부정적인 의견을 포함해서 여러 가지 다른 견해가 있어 시스템 구축을 서둘러야 하는 개발자들을 혼란스럽게 하고 있다.

지식관리시스템은 궁극적으로 인공지능 기술과 결합하여 추론에 의한 지능형 시스템의 모습으로 발전할 것이나, 현재 지식관리시스템의 개념은 지적자산 관리에 더 가까우며, 지식관리시스템의 새로운 목표를 나타내는 대표적인 주제는 「Know What You Know」이다.[3] 이는 조직의 지식이란 이전의 전문가시스템처럼 특별하고 고도의 기술이 필요한 분야 보다 현재 알고 있는 바를 조직원 전

체가 공유해 고루 활용할 수 있게 만드는데서 더 큰 효용성을 찾을 수 있음을 의미한다.

2) 지식의 형태에 따른 시스템 구축방법

앞서 언급한 바 지식은 명료하고 체계화된 지식(explicit knowledge)과 애매하고 경험적인 지식(tacit knowledge)으로 나눌 수 있고, 지식의 존재 형태에 따라 지식관리시스템의 구축을 위한 접근 방법이 달라야 한다.

지식의 형태에 따라 시스템의 구축형태도 각기 다른데, 하나는 「배분 중심의 시스템」이고 또 하나는 「협력 중심의 시스템」이다. 배분 중심의 시스템은 명료한 지식을 다루는데 적합하며, 지식 저장소에 담겨있는 지식 자체에 관심이 집중된다. 이는 정보의 용이한 입출력, 정보의 효과적 습득·해석·결합 그리고 배분, 또 지식에의 유연한 접근 등이 강조된다.

반면 협력 중심의 시스템은 애매한 지식을 다루는데 적합하며, 애매한 지식을 보유하고 있는 사람들간의 상호교류에 관심이 집중된다. 예컨대 단순히 특정 지식과 관련된 동호회에 소속된 개인들의 주소록 형태가 될 수도 있다. 또는 전자회의실이나 토론장을 통하여 자유롭게 자신의 지식을 제공하거나 습득하게 하는 등의 이미 일반화되어 기본적으로 제공되고 있는 전자게시판이 대표적 사례라 할 수 있다.

3) 가치있는 정보 즉 지식베이스의 구축방법

인터넷 상에는 정보의 질이나 가치에 대한 평가나 검증이 이루어지지 않아 불필요한 쓰레기 정보들이 난무하고 있다. 즉, 정보의 과잉으로 정보의 홍수 속에서 가치있는 정보의 선별이 불가피하게 되었다. 가치있는 정보로 구성된 지식베이스는 그림 1에서와 같이 교수 개인 홈페이지 내의 지식정보, 전문가들에 의해 검증된 분야별 지식정보, 지식관리시스템 관련부서에 의해 유지되는 지식정보를 연계시켜 체계적으로 조직하여 형성하고 다양한 검색을 지원하는 지식검색엔진을 구현함으로써 개발되어질 수 있다. 이와 같은 지식정보는 한 교수 개인으로 본다면 제한적 일 수 있지만, 이를 학과, 단대, 대학 나아가 대학 전체로 확대시키면 엄청난 지식정보가 축적된 지식베이스의 구축이 가능하며, 이를 토대로 정보 이용자가 필요로 하는 실질적이고 가치 있는 정보를 다양한 검색을 지원하는 지식검색엔진을 통해 제공할 수 있다.

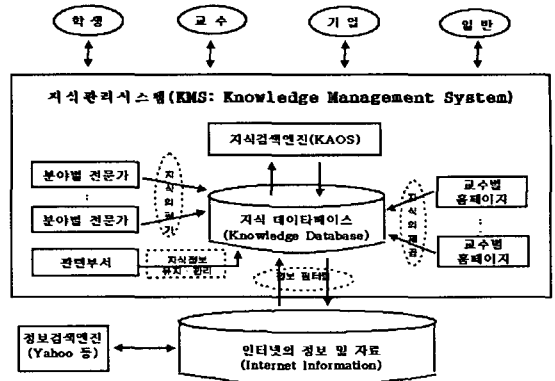


그림 1. 지식관리시스템의 개념도
Fig. 1. Conceptual drawing of Knowledge Management System

2. 시스템 구조

지식관리시스템의 구축을 위해선 반드시 통합 데이터베이스의 구축이 선행되어야 하며, 이는 상호 협력적인 분산처리가 이루어지도록 클라이언트/서버 구조로 설계하는 것이 바람직하다.

1) 시스템 구성

전형적인 2-tier 구조의 클라이언트/서버 시스템은 프로그래밍이 간단하다는 장점이 있지만 클라이언트 측의 어플리케이션이 SQL 명령을 직접 DBMS에 전달·실행하기 때문에 DBMS 노출로 인한 해킹의 우려가 크다. 또한 데이터베이스 구조에 변경이 가해지면 클라이언트 프로그램의 수정이 불가피하므로 유지보수에 있어 많은 어려움이 뒤따른다. 그리고 짧은 시간에 동시다발적인 서비스 요청으로 한 곳의 DBMS 서버에 많은 부하가 걸리는 경우에는 부하분산(load balancing)에 어려움이 있고, 여러 DBMS 서버를 접근하여 통합 데이터베이스 서비스를 제공하는 것이 어려워 시스템 확장 측면에 있어서도 문제가 있다.

그러므로 통합 데이터베이스 시스템은 시스템의 보안, 유지보수, 부하분산 및 시스템의 확장이 용이한 3-tier 구조를 가지도록 그림 2와 같이 시스템을 설계하고 구현하도록 하였다. 즉, 사용자는 웹 브라우저 단일 인터페이스만을 이용하여 웹서버에 서비스를 요청하고, 웹서버에 의해 구동되는 미들웨어의 데이터베이스 응용프로그램이 DBMS 서버를 제어하여 분산 데이터베이스 환경에서도 서비스가 제공되어질 수 있도록 시스템을 구현하였다.

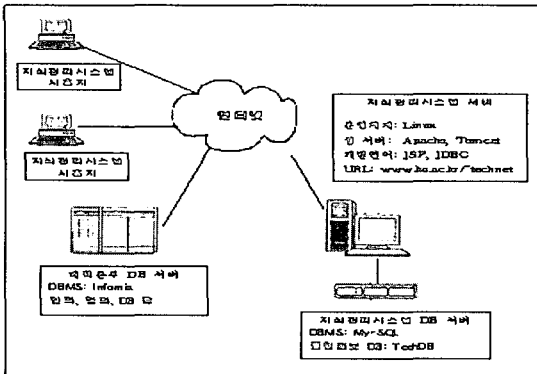


그림 2. 3-tier 클라이언트/서버 구조의 지식관리시스템 구성도

Fig. 2. 3-tier client/server structured Knowledge Management System

2) 소프트웨어 구성

대학 인적·지적 자원의 효과적 활용체제 구축을 위한 분야·학과별 전문인력 검색서비스와 이의 유지·보수를 손쉽게 하기 위한 홈페이지관리 기능의 소프트웨어 구성은 그림 3과 같다.

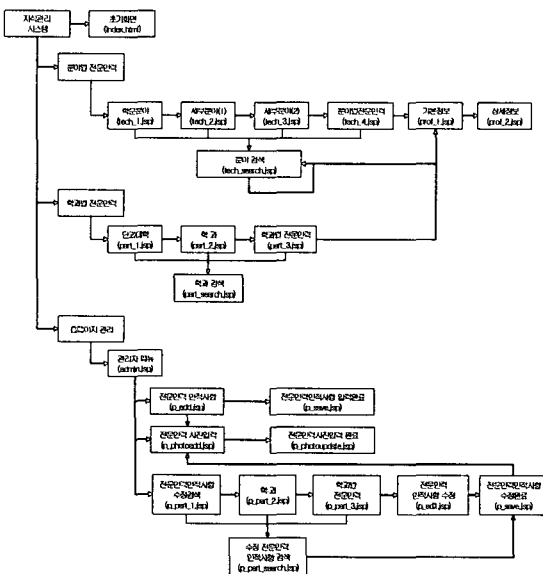


그림 3. 전문인력 검색서비스의 소프트웨어 구성도
Fig. 3. Software organization of experts searching services

3. 시스템 구현방법

빠른 실시간 서비스와 사용자의 서비스 접근이 용이한 시스템을 최소 비용으로 효과적으로 구축

하기 위한 구현방법에 대해 연구하였다.

1) 서비스 접근이 용이한 GUI 기반의 웹서비스 사용자의 정보 접근과 관리자의 유지·보수가 용이하도록 하이퍼텍스트와 GUI에 기반을 둔 멀티미디어 웹에 의해 서비스의 요청과 제공이 이루어지도록 사용자 인터페이스를 구현하도록 한다. 즉, 웹 페이지 제작을 위한 기본지식이 없는 초보자나 비전문가라 할지라도 간단한 마우스 조작과 필요정보의 입력만으로 사용자의 정보 접근과 관리자의 유지·보수가 이루어지도록 구현한다.[6]

2) 서비스 요청에 대한 빠른 실시간 서비스 제공 동적인 웹 애플리케이션을 제작하기 위한 일반적인 방법으로 CGI를 많이 사용해 왔으나 CGI는 사용자의 서비스 요청이 있을 때마다 자식 프로세스를 생성하여 서비스를 제공하므로 수많은 사용자가 동시에 서비스 요청하는 데이터베이스 시스템의 구현방법으로 적절하지 못하다.[7] 그러므로 다수의 사용자가 동시에 서비스를 요청하더라도 빠른 실시간 응답이 이루어지도록 쓰레드에 의해 서비스가 제공되도록 시스템을 구현한다.

3) 응용프로그램 개발언어와 방식

웹 응용프로그램은 C, PERL, PHP 등 다양한 언어로 구현되어질 수 있으나, 플랫폼 독립적인 실행이 이루어지도록 하기 위해서는 JAVA 언어로 시스템을 구현하고, 웹브라우저 단일 인터페이스를 통해 서비스가 이루어지도록 하는 애플릿(applet) 방식이 유용하다.

애플릿 방식은 클라이언트 측에서 실행되어지는 애플릿(client-side applet)과 서버 측에서 실행되어지는 서버릿(server-side applet: servlet)으로 구분되는데, 클라이언트 사이드 애플릿의 경우 프로그램을 서버로부터 다운 받아 클라이언트 측에서 실행하므로 서버에서 바로 실행하여 결과만을 전달하는 서버릿에 비해 실행속도가 느리다.[8]

그러므로 복수의 사용자가 동시에 접속하여 서비스를 요청할 경우에도 빠른 응답 서비스를 제공하고, 플랫폼 독립적인 실행이 가능하도록 하기 위해 쓰레드 기반의 자바 서버릿을 근간으로 하는 JSP(JavaServer Pages)로 시스템을 구현하도록 한다.

4) 최소 비용의 효과적 시스템 구축

최소 비용으로 효과적으로 시스템을 구축하기 위해서는 새로운 데이터베이스를 재구축 하기보다는 기 구축되어 있는 데이터베이스를 최대한 활용하도록 한다. 본 대학의 경우 교수의 인적자료 DB와 업적자료 DB가 Informix V.7.0.3 DBMS를 이용하여 기 구축되어 있으므로 DB 스키마 변환프로그램을 개발하여 기 구축된 DB를 전문인력의 표준 DB 양식으로 변환하고, 추가 부분만을 입력받아

전문인력 DB가 구축되어지도록 한다. 그리고 본 시스템에 연계된 교수의 인적자료와 업적자료 DB는 해당 부서에서 스스로 지속적인 up-to-date가 이루어지도록 하여 유지보수 상의 문제가 발생하지 않도록 한다.

5) 시스템의 확장 및 호환성 유지

ODBC나 JDBC 드라이버는 특정 데이터베이스 관리시스템(DBMS)의 접근 및 제어를 위한 세부 내용을 모르더라도 드라이버에서 제공되는 API를 통해 DBMS의 접근·제어를 가능하게 할뿐만 아니라 DBMS가 바뀌더라도 해당 DBMS의 ODBC나 JDBC 드라이버만을 변경하면 응용프로그램은 수정하지 않아도 되는 장점을 갖게 된다.

그러므로 응용프로그램이 JDBC 드라이버에서 제공하는 API 함수만을 통해 DBMS를 접근·제어하도록 함으로써 DBMS가 바뀌더라도 응용프로그램은 수정하지 않아도 되는 DBMS 독립성이 보장 되도록 한다.

6) 프로그램 중복의 최소화와 재사용의 극대화

프로그램의 중복을 최소화하기 위해서는 우선 데이터베이스 응용프로그램의 구조 및 동작을 면밀히 분석하여 반복적으로 수행되는 루틴을 공통 루틴으로 독립시키는 것이 중요하다. 특히, 사용자 서비스를 위한 입출력 양식을 최소화하고 사용자 인터페이스를 간소화하는 것이 불필요한 노동력의 낭비를 막는 첩경이 된다. 그러므로 응용프로그램의 구조 및 동작 분석을 통하여 공통 루틴을 만들고, 최소화한 공통의 입출력 양식을 통해 서비스 제공이 가능하도록 시스템을 구현한다.

IV. 지식관리시스템 구현

제안한 구현방법을 따라 구현한 본 시스템은 하이퍼텍스트와 GUI에 기반을 둔 멀티미디어 웹에 의해 서비스가 이루어지고, 기 구축되어 있는 데이터베이스를 최대한 활용 최소 비용의 효과적 시스템 구축이 이루어지도록 하였고, 빠른 서비스를 위해 쓰레드 기반의 자바 서버릿[6]을 근간으로 하는 JSP로 구현하였고, 시스템의 보안과 확장이 용이하도록 3-tier 구조의 클라이언트/서버 시스템이며, 시스템의 확장과 호환을 용이하도록 JDBC의 API 함수를 이용하였고, 프로그램 중복의 최소화와 재사용의 극대화가 이루어지도록 구현하였다. 그리고 향후 유사한 웹 기반의 3-tier 데이터베이스 시스템 개발에 도움이 되도록 시스템 구현을 위한 응용프로그램의 핵심 소스를 상세히 기술하였다.

1. 데이터베이스 제어 프로그램

데이터베이스 제어 프로그램은 데이터베이스의 연결과 해제 기능을 수행하며, 데이터베이스 관리 시스템(DBMS)이 변경되더라도 JDBC 드라이버만을 변경하면 응용프로그램은 수정하지 않아도 되도록 하였으며, 공통 루틴으로 객체화하여 사용자의 데이터베이스 접근 시에 재사용이 가능하도록 하였다.

사용자의 데이터베이스 시스템에 대한 정보 요청은 다양하지만 이를 처리하는 데이터베이스 응용프로그램은 SQL 명령만을 달리할 뿐 실제 그 처리절차는 거의 동일하다. 그러므로 사용자의 정보 요청 별로 서비스 프로그램을 작성하는 것 보다 사용자의 정보요청을 파라메타로 전달받아 이를 기초로 SQL 명령을 만들고 공통 루틴을 통해 데이터베이스 관련 서비스를 제공하는 형태로 프로그램을 개발하면 프로그램의 중복을 최소화할 수 있다.

```
import java.sql.*;
import java.io.*;
import java.util.*;
public class Techdb {
    private Connection cont=null;
    Statement stmt=null; PreparedStatement pstmt
    null;
    ResultSet rs=null;
    public Connection getConnection() throws Exception {
        Class.forName("org.gjt.mm.mysql.Driver");
        String url = "jdbc:mysql://localhost/techdb?user=
        knowledge&password=manman";
        cont = DriverManager.getConnection(url);
        return(cont);
    }
    public Statement getStatement() throws Exception {
        cont = getConnection();
        stmt = cont.createStatement();
        return(stmt);
    }
    public PreparedStatement getPreparedStatement (St
    ring query) throws Exception {
        cont = getConnection();
        PreparedStatement pstmt = cont.prepareStatement
        nt(query);
        return(pstmt);
    }
    public void close() throws SQLException {
        rs.close();
        stmt.close();
        cont.close();
    }
}
```

그림 4. 데이터베이스 제어 프로그램(techdb.java)
Fig. 4. DBMS Control Program

2. 분야별 리스트 출력 프로그램

분야별 리스트 출력프로그램은 해당분야를 클릭하면 세부분야를 정의한 양식으로 출력하는 기능을 수행한다.

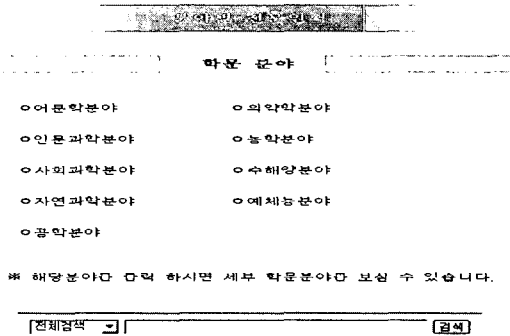


그림 5. 분야별 리스트 출력화면
Fig. 5. User interface of parts listing

```
public class partSelect extends Techdb {
    public Vector getPart() throws SQLException, Exception {
        Vector kv = new Vector();
        String query = "SELECT code, name FROM
        part where code like '%0000' RDER BY code
        sc";
        stmt = getStatement();
        rs = stmt.executeQuery(query);
        while (rs.next()) {
            PartItem pitem = new PartItem();
            pitem.code = rs.getString(1);
            pitem.name = rs.getString(2);
            kv.addElement(pitem);
        }
        close();
        return(kv);
    }
}
```

그림 6. 분야별 리스트 출력 프로그램1(PartSelect.java)
Fig. 6. Parts listing program 1

```
<%@ page contentType="text/html; charset=euc-kr" %>
<%@ page language="java" import="java.sql.*, java a.util.
*, java.io.*" %>
<jsp:useBean id="partselect" class="PartSelect"
scope="page" />
<%
    Vector v = partselect.getPart();
    PartItem pitem = new PartItem();
    int vsize = v.size();
    for (int i = 0; i < vsize; i++) {
        pitem = (PartItem) v.elementAt(i);
    %>
    <td><a href="tech_2.jsp? code=<%=pitem.e%>
    <%= pitem.name %></a></td>
<%
    }
%>
```

그림 7. 분야별 리스트 출력 프로그램2(tech_1.jsp)
Fig. 7. Parts listing program 2

3. 전문인력 조회 프로그램

분야별 리스트 출력프로그램에 의해 출력된 최

하위 세부분야를 선택하면 해당 분야의 전문인력 리스트가 출력된다. 해당 인력을 선택하면 전문인력의 기본정보와 상세정보를 조회하여 출력하는 기능을 수행한다.

인적 사항



● 이름	최지원
● 소속	컴퓨터공학전공
● 직위	조교수
● 전공	컴퓨터네트워크 및 인터넷(Computer Networks and Internet)
● 세부전공	정보통신, 인터넷응용, 운영체제
● 관심분야	컴퓨터네트워크 및 인터넷(Computer Networks and Internet)
● 전화	051-620-4786
● E-mail	choejw@ks.ac.kr
● 홈페이지	http://infonet.ks.ac.kr

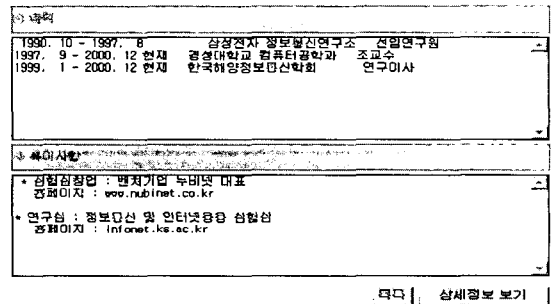


그림 8. 전문인력 기본정보 조회화면
Fig. 8. User interface for printing expert's basic information

```
public class profSelect extends Techdb {
    public ProfItem getProf(String r_id) throws SQLException,
    Exception {
        String query = "SELECT * FROM prof where r_id
        = '"+ r_id +"'";
        stmt = getStatement();
        rs = stmt.executeQuery(query);

        while(rs.next()) {
            ProfItem pitem = new ProfItem();
            pitem.r_id = rs.getString(1); // 사번/학번
            pitem.r_class = rs.getString(2); // 직위
            pitem.name1 = rs.getString(3); // 한글이름
            ..... 중략 .....
            pitem.resume = rs.getString(25); // 특이사항
            pitem.pub = rs.getInt(26); // 상세정보공개여부
            pitem.photo = rs.getString(27); // 사진화일명
        }
        close();
        return(pitem);
    }
}
```

그림 9. 전문인력 기본정보 조회 프로그램1
(ProfSelect.java)

Fig. 9. Expert's basic information retrieval program 1

```

<jsp:useBean id="profselect" class="ProfSelect" scope="page" />
<%
String r_id = request.getParameter("r_id");
ProfItem pitem = (ProfItem) profselect.getProf(r_id);
%>
<table>
<tr><td> </td>
<td> ● 이름 </td> <td> <%= pitem.name1 %> </td>
</tr>
<tr><td> ● 소속 </td> <td> <%= pitem.deptname %> </td>
</tr>
<tr><td> ● 직위 </td> <td> <%= pitem.r_class %> </td>
</tr>
<tr><td> ● 특이사항 </td> <td><textarea> <%= pitem.info %> </textarea></td></tr>
</table>

```

그림 10. 전문인력 기본정보 조회 프로그램2 (prof_1.jsp)

Fig. 10. Expert's basic information retrieval program 2

4. 전문인력정보 키워드검색 프로그램

사용자가 검색부문(분야/학과/이름)과 키워드를 입력하면 유사 문자열을 검색하여 해당 전문인력 리스트를 출력하는 기능을 수행한다.

```

<form name="form" method="get" action="tech_search.jsp">
<select name="searchpart">
<option value="part">학문</option>
<option value="dept">학과</option>
<option value="name">이름</option>
</select>

<input type="text" name="searchname">
<input type="submit" value="검색">
</form>

```

그림 11. 키워드검색 프로그램1(tech_1.jsp)
Fig. 11. Keyword search program 1

```

public Vector getProf(String code, String col) throws SQLException, Exception {
Vector pv = new Vector();
String query = "SELECT * FROM prof where
'+ col +' like '"+ code +%%' ORDER BY name asc";
stmt = getStatement();
rs = stmt.executeQuery(query);

while(rs.next()) {
ProfItem pitem = new ProfItem();
pitem.r_id = rs.getString(1); // 사번/학번
pitem.r_class = rs.getString(2); // 직위
pitem.name1 = rs.getString(3); // 한글이름
..... 중략 .....
}
close();
return pv;
}

```

그림 12. 키워드검색 프로그램2(ProfSelect.java)
Fig. 12. Keyword search program 2

```

<%
String title = null; String col = null;
String searchpart = request.getParameter("searchpart");
String searchname = request.getParameter("searchname");

if (searchpart.equals("part")) {
title = "학문"; col = "major1name";
}
if (searchpart.equals("dept")) {
title = "학과"; col = "deptname";
}
if (searchpart.equals("name")) {
title = "이름"; col = "name1";
}

Vector pv = profselect.getProf(searchname, col);
ProfItem pitem = new ProfItem();
int psize = pv.size();
for (int i=0; i < psize; i++) {
..... 중략 .....
}
%>

```

그림 13. 키워드검색 프로그램3(tech_search.jsp)
Fig. 13. Keyword search program 3

5. 인적사항 입력 프로그램

전문인력의 기본정보를 입력받아 데이터베이스에 저장하는 기능을 수행한다.

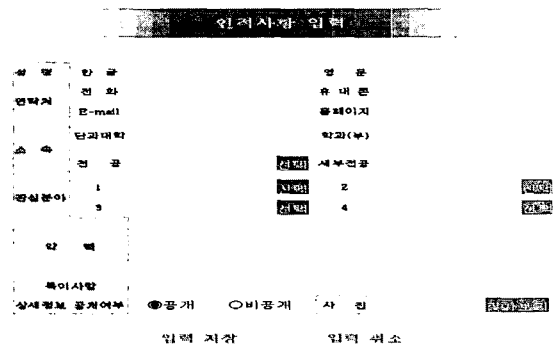


그림 14. 전문인력 기본정보 입력화면
Fig. 14. User interface for inserting expert's basic information

```

<form name="p_save" method="post" action="/p_save.jsp">
사번 <input type="text" name="r_id" >
성명 <input type="text" name="name1" >
전화 <input type="text" name="phone1" >
단대 <input type="text" name="partname" >
<input type="hidden" name="partcode" >
..... 중략 .....
<input type="submit" name="psave" value="입력 저장">
</form>

```

그림 15. 전문인력 기본정보 입력 프로그램1 (p_add.jsp)
Fig. 15. Expert's basic information insert program 1


```

<jsp:useBean id="profitemadd" scope="page" class="ProfItemAdd" />
<jsp:setProperty name="profitemadd" property="r_id" />
<jsp:setProperty name="profitemadd" property="name1" />
<jsp:setProperty name="profitemadd" property="phone1" />
<jsp:setProperty name="profitemadd" property="partcode" />
<jsp:setProperty name="profitemadd" property="partname" />..... 중략 .....
<% profitemadd.addProfItem(); %>
    
```

그림 16. 전문인력 기본정보 입력 프로그램2 (p_save.jsp)

Fig. 16. Expert's basic information insert program 2

```

public class ProfItemAdd extends Techdb {
    ProfItem pitem = new ProfItem();
    public void setR_id(String r_id) {
        pitem.R_id = r_id;
    }
    public void setName1(String name1) {
        pitem.name1 = name1;
    }
    ..... 중략 .....
    public void addProfItem() throws SQLException, Exception {
        PreparedStatement pstmt = null;
        String query = "insert into prof(r_id, r_class, ...)"
        values(?, ?, ...)";
        pstmt = super.getPreparedStatement(query);
        pstmt.setString(1, pitem.r_id);
        pstmt.setString(2, pitem.r_class);
        ..... 중략 .....
        pstmt.executeUpdate();
        pstmt.close();
    }
}
    
```

그림 17. 전문인력 기본정보 입력 프로그램3 (ProfItemAdd.java)

Fig. 17. Expert's basic information insert program 3

6. 인적사항 수정 프로그램

선택된 전문인력의 기본정보를 출력하고 기본정보의 수정사항을 입력받아 데이터베이스를 갱신하는 기능을 수행한다. 수정양식은 이전 기본정보를 포함하고 있는 것만 다르고 그림 14의 입력양식과

동일하다.

```

<jsp:useBean id="profselect" class="ProfSelect" scope="page" />
<%
String r_id = request.getParameter("r_id");
ProfItem pitem = (ProfItem) profselect.getProfR_id(r_id);
%>
<form name="p_save" method="post" action="/p_modify.jsp">
사번 <input type="text" name="r_id" value="<%= pitem.r_id%>" />
성명 <input type="text" name="name1" value="<%= pitem.name1 %>" />
전화 <input type="text" name="phone1" value="<%= pitem.phone1 %>" />
단체 <input type="text" name="partname" value="<%= pitem.partname %>" />
<input type="hidden" name="partcode" value="<%= pitem.part %>" />
..... 중략 .....
<input type="submit" name="psave" value="입력 수정" />
<input type="Button" name="pdel" value="삭제" onclick="delcheck()" />
</form>
    
```

그림 18. 전문인력 기본정보 수정 프로그램1(p_edit.jsp)

Fig. 18. Expert's basic information update program 1

```

<jsp:useBean id="profitemadd" scope="page" class="ProfItemAdd" />
<jsp:setProperty name="profitemadd" property="r_id" />
<jsp:setProperty name="profitemadd" property="name1" />
<jsp:setProperty name="profitemadd" property="phone1" />
..... 중략 .....
<% profitemadd.modifyProfItem(); %>
    
```

그림 19. 전문인력 기본정보 수정 프로그램2 (p_modify.jsp)

Fig. 19. Expert's basic information update program 2

```

public class ProfItemAdd extends Techdb {
    ProfItem pitem = new ProfItem();
    public void setR_id(String r_id) {
        pitem.R_id = r_id;
    }
    ..... 중략 .....

    public void modifyProfItem() throws SQLException, E
xception
    {
        PreparedStatement pstmt = null;
        String query= "update prof set r_id=?,
            r_name1=?, ... where r_id="+pitem.r_id;
        pstmt = super.getPreparedStatement(query);
        pstmt.setString(1, pitem.r_id);
        pstmt.setString(2, pitem.name1);

        ..... 중략 .....

        pstmt.executeUpdate();
        pstmt.close();
    }
}
    
```

그림 20. 전문인력 기본정보 수정 프로그램3(ProfItemAdd.java)

Fig. 20. Expert's basic information update program 3

7. 인적사항 삭제 프로그램

선택된 전문인력의 기본정보를 출력하고 삭제요청에 따라 해당 전문인력을 데이터베이스로부터 삭제하는 기능을 수행한다.

```

<form name="del" action="p_del.jsp" method="get">
    <input type="hidden" name="r_id" value="<%=pitem.r_id%>" />
</form>
    
```

그림 21. 전문인력 기본정보 삭제 프로그램1(p_edit.jsp)
Fig. 21. Expert's basic information delete program 1

```

<jsp:useBean id="profitemadd" scope="page" class="ProfItemAdd" />
<jsp:setProperty name="profitemadd" property="r_id" />
    ..... 중략 .....
<%
    profitemadd.delProfItem();
%>
    
```

그림 22. 전문인력 기본정보 삭제 프로그램2(p_del.jsp)
Fig. 22. Expert's basic information delete program 2

```

public void delProfItem() throws SQLException, Exception {
    PreparedStatement pstmt = null;
    String query = "delete from prof where r_id="+pitem.r_id;
    pstmt = super.getPreparedStatement(query);
    pstmt.executeUpdate();
    pstmt.close();
}
    
```

그림 23. 전문인력 기본정보 삭제 프로그램3 (ProfItemAdd.java)

Fig. 23. Expert's basic information delete program 3

V. 결 론

본 논문에서는 지식정보의 메카인 대학의 인적·지적 자원을 데이터베이스화하여 이를 필요로 하는 사람은 누구나 활용할 수 있도록 하여 정보이용자들 간에 지식의 공유와 교류가 가능하도록 하는 지식관리시스템을 최소한의 비용으로 효과적으로 구축하기 위한 시스템의 구현방법에 대해 연구하고 이를 기초로 시스템을 구현하였다.

본 지식관리시스템은 하이퍼텍스트와 GUI에 기반을 둔 멀티미디어 웹에 의해 서비스의 요청과 제공이 이루어질 수 있도록 구현하였고, 기 구축되어 있는 데이터베이스를 최대한 활용하여 최소 비용으로 효과적 시스템 구축이 이루어질 수 있도록 하였고, 사용자요청에 따른 서비스가 쓰레드 기반의 자바 서블릿을 근간으로 하는 JSP로 구현하여 동시서비스 요청에 대한 응답이 신속히 이루어지도록 구현하였다.

또한 시스템의 보안, 유지보수, 부하분산 및 시스템의 확장이 용이한 3-tier 구조의 클라이언트/서버 시스템으로 구현하였으며, 응용프로그램은 플랫폼 독립적인 실행이 가능하도록 자바언어로 구현하였으며, 시스템이 확장성과 호환성을 갖도록 JDBC 드라이브 합수를 이용하여 데이터베이스 응용프로그램을 구현하여 DBMS가 바뀌더라도 응용프로그램의 수정을 최소화할 수 있도록 하였다. 그리고 프로그램 중복의 최소화와 재사용의 극대화가 이루어지도록 구현하였다.

본 시스템은 궁극적으로 정보이용자가 수행하고자 하는 과제에 대한 사양을 주면 이를 분석하여 정보 이용자가 필요로 하는 실질적이고 가치 있는 구체적인 지식-예를 들면, ① 무엇을 할 것인가?

(Know-what) ② 왜 해야하나? (Know-why) ③ 언제 해야하나? (Know-when) ④ 어떻게 개발할 것인가? (Know-how) ⑤ 사업추진을 위한 정보를 어디에 있는가? (Know-where) ⑥ 누가 개발할 것인가? (Know-who) 등-을 제공할 수 있는 꿈의 지식관리시스템으로 발전시키고자 한다.

끝으로 본 논문에서 공개한 응용프로그램의 핵심 소스는 이와 유사한 웹 기반의 3-tier 데이터베이스 시스템의 개발에 많은 도움이 되리라 생각된다.

- 제 7권, 제 2호, 2003년.
- [6] 최재원, 김진천, "고객 중심의 주문형 홈페이지 자동화시스템 구현", 한국해양정보통신학회논문지, 제 5권, 제 1호, 2001년.
 - [7] 최재원, "Java 서블릿을 이용한 웹 기반의 3-Tier 데이터베이스 시스템 설계 및 구현" 한국통신학회논문지, 제 27권, 제 9호, 2002년.
 - [8] 최재원, "PC 통신 DB 서비스의 인터넷 웹 DB 서비스로의 효율적 전환을 위한 구현방법 및 기술" 한국통신학회논문지, 제 27권, 제 8호, 2002년.

참고문헌

- [1] Brown J.S., Duguid P., "Organizing Knowledge", California Management Review, Vol. 40, No 3, 1998.
- [2] Nonaka, I. and Takeuchi, H., "The Knowledge Creating Company", Oxford University Press, 1995.
- [3] 광동철, "우리나라 과학기술산업 지식정보통합관리시스템의 모형 개발에 관한 연구", 정보관리학회지, 제18권, 제4호, 한국정보관리학회, 2001.
- [4] 이주승, "중소기업을 위한 지식관리시스템의 효과적인 적용방법에 관한 연구", 홍익대학교 정보대학원 석사학위논문, 2001.
- [5] 최재원, 이종혁, "대학의 인적·지적 자원의 효과적 활용을 위한 지식관리시스템의 개발 방법 및 설계", 한국해양정보통신학회논문지,

저자소개

최재원(Jae-Weon Choe)



1988년 2월 고려대학교 컴퓨터 공학과(공학사)
 1990년 8월 미시간주립대학교 컴퓨터 공학과(공학석사)
 1995년 8월 건국대학교 전자공학과(공학박사)

1990년 10월~1997년 8월 삼성전자 정보통신연구소 선임연구원

1997년 9월~2003년 현재 경성대학교 전기전자·컴퓨터공학부 부교수

※관심분야: 정보통신망, 이동통신, 인터넷 응용, 운영체제