

# WWW 기반 전문가시스템의 구조와 구현 기술의 비교 연구

송 용 옥\*

< 목 차 >

I. 서 론	VI. WWW 기반 전문가시스템의 구조 분석
II. WWW 기반 전문가시스템	V. 가설 검증 결과
III. WWW 기반 응용시스템 구현기술	VI. 결 론

## I. 서 론

전문가시스템은 인공지능의 여러 접근방법 중에서 가장 성공적인 방법 중의 하나로 알려져 왔으며, 특히 Engineering-Manufacturing, Business, Medicine, Environment-Energy, Agriculture, Telecommunications, Government, Law, Transportation, Finance, Production Management, General Management, Accounting-Auditing, Marketing-Sales, Electronic Commerce, International Business, Human Resource Management 등 여러 분야에서 성공적으로 응용되어 왔다.(Liebowitz, 1997) 최근에는 인터넷 사용의 확산과 WWW(World Wide Web)의 보급에 발맞추어 WWW에 기반을 둔 전문가시스템이 속속 개발되고 있다. 또한, 인터넷을 바탕으로 한 전자상거래의 등장에 따라 전자상거래를 지원하기 위한 지능형정보시스템에 대한 필요성이 제기되면서 이것을 전문가시스템 방법론을 이용하여 해결하려는 많은 연구와 개발이 진행되고 있다.

따라서, 인터넷과 WWW이라는 새로운 컴퓨팅 환경 하에서 전문가시스템의 구조도 기존의 Stand-alone 방식에서 벗어나 네트워크를 바탕으로 다수의 사용자를 지원하는 방식으로 개선, 발전될 필요가 있다. 그러나, WWW 환경 하에서의 데이터베이스 기반 응용시스템의 구조(김평철, 1966)에 대한 연구나 WWW 기반 전문가시스템의 구조 모형의 제시(임규건 외, 1997) 등에 대한 연구는 있었지만 전문가시스템의 구조와 구현 방법 및 장단점 비교 등에 대한 기술적 관점에서의 연구는 아직 없었다. 본 논문에서는 인터넷과 WWW이라는 컴퓨팅 환경 하에서 사용 가능한 전문가시스템 구현 기술들을 살펴보고 그에 따른 전문가시스템의

\* 경상대학교 경영정보학과 교수

구조들을 분류, 비교함으로써 WWW 기반 전문가시스템의 개발 시 알맞은 전문가시스템 구조를 선택할 수 있는 지침을 제시하고자 한다.

이를 위해 제 2 절에서는 WWW 기반 전문가시스템의 현황과 장단점을 살펴보고, 제 3 절에서는 WWW 기반 응용시스템을 구현하기 위한 인터넷 및 WWW 관련 기술들을 나열한다. 그리고 제 4 절에서는 제 3 절에서 소개한 기술들을 바탕으로 WWW 기반 전문가 시스템의 개발 방안과 구현 시스템의 구조를 살펴보고 제 5 절에서는 이 구조들을 비교, 분석함으로써 WWW 기반 전문가시스템의 개발 지침을 제시하도록 하겠다.

## II. WWW 기반 전문가시스템

기존의 전문가시스템들은 Stand-alone 방식으로 각 응용분야에서 전문가의 지식을 시스템 내부적으로 표현, 저장하고 적용하여 문제를 해결하였다. 그러나, 이러한 전문가시스템의 지식은 Stand-alone 방식하에서는 많은 사용자에게 제공될 수 없었고 단지 접근 가능한 몇몇 사용자들에게만 이용 가능하였다. 이러한 지식분배문제를 해결하기 위하여 Client/Server 구조를 채택하기도 하였으나 대부분 각 응용시스템의 특정 구조에 맞추어야 했고 또 Client의 비중이 큰 Fat-Client 구조로 인해 비용이 많이 들었기 때문에 이러한 Client/Server 전문가 시스템은 큰 실효를 거두지 못하였다. (임규진 외, 1997; Far, 1996)

이러한 상황 하에서 인터넷과 WWW의 등장은 전문가시스템의 개발을 위한 새로운 컴퓨팅 환경을 제공하였다. 인터넷에 연결된 전세계의 다수의 사용자가 표준화된 Web 브라우저를 이용하여 HTTP(Hypertext Transfer Protocol)(Berners-Lee, *et al.*, 1996)라는 인터넷 기반 프로토콜을 통하여 Web Server에서 제공하는 전문가의 지식을 향유할 수 있는 기반이 마련된 것이다. WWW 기반 전문가시스템의 장점으로는 다음과 같은 것을 들 수 있다. (Eriksson, 1996)

- (1) 여러 명의 사용자가 동시에 사용이 가능하다.
- (2) 24시간 언제든지 사용 가능하다.
- (3) 유연성을 제공한다.
- (4) 사용하기 쉽고 편리한 Web 브라우저를 사용한다.
- (5) HTML(Berners-Lee and Connolly, 1995)과 같은 공통 언어로 인터페이스 제작이 용이하다.
- (6) 멀티미디어를 이용한 다양한 사용자 화면의 구성이 가능하다.
- (7) 데이터 수준에서 사용자가 응용 프로그램에 접근 가능하다.
- (8) 동적이다.
- (9) 주식과 같은 실시간 데이터를 보여줄 수 있다.
- (10) 공용 전문가시스템이 가능하다.
- (11) 전문가시스템의 분배 문제가 간단해 진다.
- (12) 플랫폼에 독립적으로 다양한 플랫폼 지원이 가능하다.

물론, 아직까지 신뢰성이 있고 빠른 네트워크의 연결이 어렵고, 인터넷과 WWW의 개발성에 따른 사적 정보의 보안 문제가 있고, 인터넷 상의 요금 체제의 미정립으로 인해 과금에 문제가 있기는 하지만 이러한 단점들은 새로운 네트워크 기술의 개발, 암호화와 인증 체계의 도입, 과금 제도의 확립 등에 따라 차차 해결될 문제들이다.

현재까지 WWW 환경에서 개발되어진 전문가시스템들의 예를 응용 분야와 응용조직, 이용 기술별로 분류하면 다음과 같다.(Intelligent Software Strategies, Winter 1997)

(1) Intelligent E-mail Interpretation, Classification, and Trouble Shooting

응용분야	이용 및 개발기관	개발기술
EZ Reader	Chase Manhattan Bank / Brightware	Rule-based parsing expert system, CBR
Bounced Mail Expert System	White House / MIT AI Lab.	Expert System
Internet Financial Service Mall	Swiss Bank / Brightware	Intelligent e-mail reader
Fielding of Product Inquiries	Amway / Brightware	Intelligent e-mail reader

(2) Smart Advisors for Marketing Products and Services, Training

응용분야	이용 및 개발기관	개발기술
Internet Loan Advisor	Security First Technologies / Security First Network Bank / Brightware	KBS loan advisor
Internet Site Guide	Amzi!, Inc.	Built using Amzi!'s own WebLS Prolog-based Web Server and development tools
Military Training System	US Army	IntelliNets Web Expert rule-based development/server tools
NASA Personnel Security Processing Expert System	NASA / Johns Hopkins University Applied Physics Lab	Rule-based expert system with Web interface
Personalized Entertainment Web Service	Open Sesame / Borders Inc.	Using Open Sesames neural net / KBS agent technology
Quartet	Illinois Chamber of Commerce	Using Brightwares A*E and A*E / Web to develop KBS / CBR-based job/applicant matching system

(3) Online Configuration Systems

응용분야	이용 및 개발기관	개발기술
PC Buying and Help System	Monorail, Inc.	KBS Web Product configurator and help desk based on Servicesofts Web Advisor and Knowledge Builder

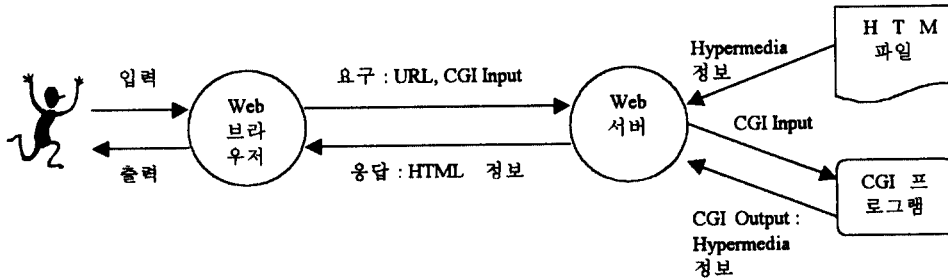
## (4) Help Desks for Technical Support

응용분야	이용 및 개발기관	개발기술
Customer Support	Broderbund Software	Web-based help desk build with Inferences CBR2 and CasePoint WebServer tools
FAQ Finder	University of Chicago	CBR with natural language interface
Netscape Help System	Netscape	Used Intellisystems SmartSite product to convert existing Call Center (phone-based) support system KBs to intelligent and automatically linked HTML files for deployment on a Web server
Taco Bell Internal Help Desk	Taco Bell	KBS built using Servicesofts Web Advisor and KnowledgeBuilder products, and Servicewares Knowledge-Pak prepackaged KBs
UltraNet Self Service Support	UltraNet	KBS Web help desk
Yodas Help Desk	LucasArts	Web-based customer support system build with Inferences CBR2 and CasePoint WebServer tools

### III. WWW 기반 응용시스템 구현기술

현재 가장 많이 사용되고 있는 WWW 기술로는 HTML, CGI, Java(Lemay and Perkins, 1996), External Helper 등이 있다. 먼저, HTML은 WWW이 제공하는 Hypermedia (Hypertext and Multimedia) 정보를 표현하기 위한 언어이다. HTML 언어에 의해 작성된 WWW 서비스용 정보 파일은 Web 서버에 의하여 읽혀져서 인터넷을 통해 그 정보를 요구한 Web 브라우저에게 보내진다. 이것을 받은 Web 브라우저는 HTML 언어를 해석하여 그 내용을 화면에 디스플레이 한다. 이 기술의 장점은 간단한 HTML 문법만으로 Hypermedia 정보를 풍부하게 표현할 수 있고, 또 Web 브라우저만 있으면 인터넷에 연결된 모든 Web 서버로부터 정보를 받아 볼 수 있다는 데 있다. 그러나, Hypermedia 정보가 미리 결정되어 파일의 형태로 저장되어 있으므로 각 사용자의 요구에 맞추어 동적으로 변화하는 정보를 제공하는데 한계가 있다는 단점이 있다.

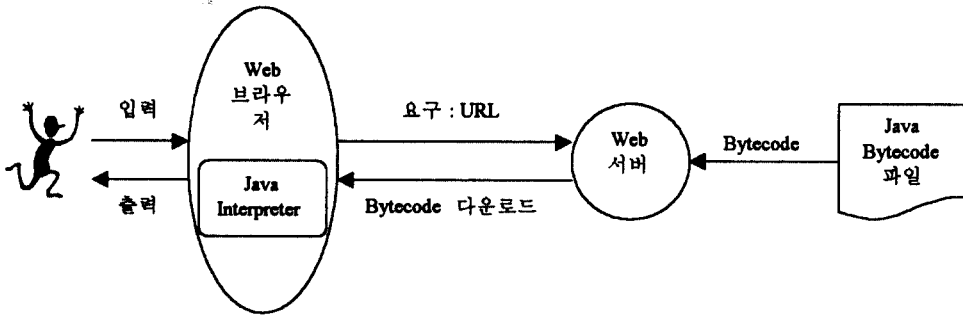
이러한 HTML의 단점을 보완하기 위하여 추가된 기술이 CGI (Common Gateway Interface) 기술이다. 이것은 HTML의 Form 기법에 의해 입력 받은 사용자의 정보를 Web 서버 시스템에 있는 특정 수행프로그램(이것을 보통 CGI 프로그램이라고 부른다.)에 전달하고, 다시 그 프로그램이 생성한 HTML 형식의 정보를 Web 브라우저에게 제공, 디스플레이 할 수 있도록 하는 자료 전달 통로를 추가로 정의한 것이다. 이 CGI 기술에 의해 WWW은 각 사용자의 요구에 맞추어 응답할 수 있는 동적 기능을 가질 수 있게 되었다. 지금까지 설명한 HTML과 CGI의 작동원리를 도식한 것이 <그림 1>에 나타나 있다.



<그림 1> HTML 과 CGI 의 작동원리

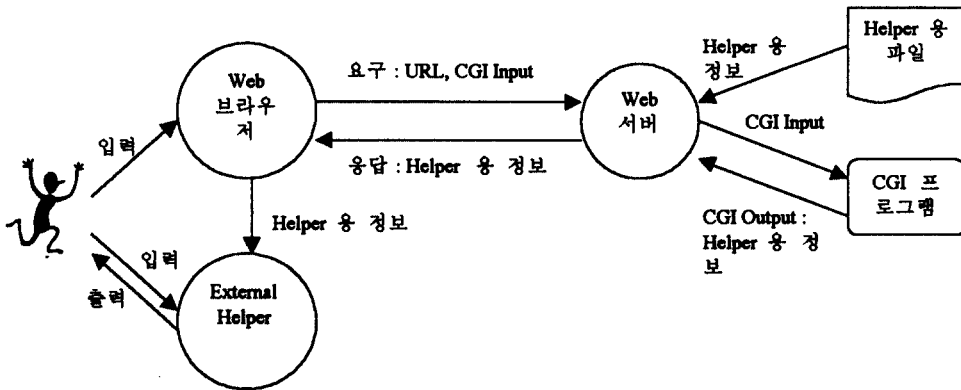
Java는 원래 WWW과 별도로 H/W 독립적인 프로그래밍 언어로서 출발하였다. Java의 언어적 특성은 C++ 언어와 유사하게 객체지향 개념을 제공한다는 것과 컴파일된 수행 코드가 그 수행 H/W에 상관없이 수행될 수 있다는 데 있다. 사실 컴파일된 수행 코드는 특정 H/W 용 기계어로 되어 있는 것이 아니고 Java 인터프리터라는 수행프로그램이 읽어서 수행할 수 있는 효율화된 이진코드이다. 이 코드를 Bytecode라고 부른다. Java는 인터프리터 방식으로 수행되는 것이지만 Bytecode가 매우 효율적으로 되어 있기 때문에 기계어로 된 수행프로그램보다는 느리지만 기존의 Text 파일 기반 인터프리터 방식에서보다는 훨씬 더 빨리 수행된다. 이러한 약간의 수행 비효율성을 감수한 대신에 Java 인터프리터가 있는 H/W에서는 재컴파일할 필요 없이 어느 H/W에서 컴파일된 Java Bytecode든 수행할 수가 있다. 이와 같이 Java 언어로 프로그래밍 되고 컴파일되어 수행되는 프로그램을 Java Application이라고 부른다.

그런데, WWW의 등장 이후 어떤 수행프로그램이 Web 브라우저에게 전달되어 수행될 경우 그것이 어느 H/W에 있는 Web 브라우저에서건 재컴파일 없이 수행될 필요가 생기게 되었고 이것을 지원할 수 있는 프로그래밍 언어로서 Java가 지목되었다. Java 인터프리터를 내장하고 있는 Web 브라우저가 Java Bytecode를 받으면 그것을 재컴파일할 필요 없이 바로 수행할 수 있기 때문이다. 이와 같이 Web 브라우저에 의해 수행되는 Java 프로그램을 Java Applet이라고 부른다. Java Applet의 장점은 HTML이나 CGI가 갖는 사용자 인터페이스의 단순성을 탈피하여 메뉴와 버튼 방식의 Windows 인터페이스가 가능하고, Web 서버로부터 Bytecode가 전달된 후에는 Web 서버와의 별도 정보교환없이 사용자의 컴퓨터에서 자체적으로 빠르게 수행될 수 있다는 데 있다. 그러나, Java Bytecode가 Web 서버로부터 Web 브라우저로 전달되는 초기 수행준비시간이 현재의 네트워크 환경하에서는 매우 길다는 데 단점이 있다. Java Applet의 작동원리를 도식한 것이 <그림 2>에 나타나 있다.



<그림 2> Java Applet 의 작동원리

마지막으로 External Helper 프로그램은 사용자의 컴퓨터에 미리 설치되어 있는 수행가능한 프로그램으로서 Web 브라우저의 요청에 의해 수행되는 프로그램을 말한다. Web 브라우저는 Web 서버로부터 응답받은 정보가 자신이 처리할 수 없는 형태일 때 그 내용을 External Helper 프로그램에게 전달하면서 External Helper 프로그램을 수행시킨다. 응답받은 정보의 형태는 HTTP에 의해 Content-type이라는 응답필드에 정의할 수 있도록 되어 있으며, Web 브라우저는 Content-type 필드의 값을 보고서 그것이 자신이 처리할 수 있는 것인지 아니면 어떤 External Helper를 수행시켜야 하는 것인지를 판단할 수 있다. Web 서버로부터 받아서 Web 브라우저가 External Helper에 전달하는 내용은 External Helper가 프로그램을 수행하기 위한 입력자료로서 사용된다. External Helper를 사용할 때의 장점은 External Helper가 Java Applet처럼 Windows 인터페이스를 제공하는 아주 빠른 자체 수행 프로그램이라는 데 있다. 또 Java Applet과는 달리 수행 준비 단계에 Web 서버로부터 Web 브라우저로 다운로드될 필요도 없다. 그러나, External Helper는 각 H/W에 맞게 기계어로 컴파일된 보통의 수행프로그램이므로 Web 브라우저가 있는 H/W에 맞추어 미리 프로그램을 컴파일하여 설치해 두어야 하는 불편함이 있다. 또, External Helper의 수행 내용이 바뀌어야 할 경우 모든 사용자가 다시 컴파일된 프로그램을 전달받아 재설치하여야 하므로 WWW이 지향하는 다중 사용자 지원환경에서 유지보수의 문제를 야기한다. 특히, WWW을 사용하는 사용자가 전문가가 아닌 일반 초보자일 때 프로그램의 재설치는 시스템의 사용 편리성을 매우 저하시키게 된다. External Helper의 작동원리를 도시한 것이 <그림 3>에 나타나 있다.



<그림 3> External Helper 의 작동원리

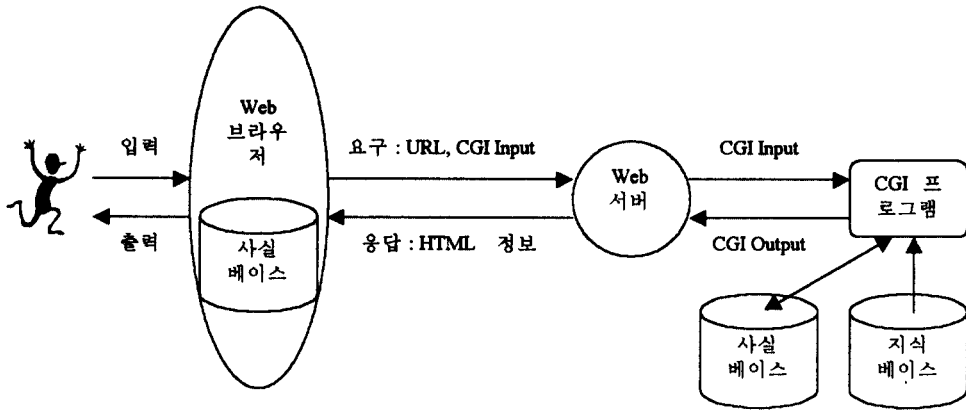
## IV. WWW 기반 전문가시스템의 개발 방안

앞 3 절에서 설명한 HTML, CGI, Java, External Helper 기술들을 이용하면 다양한 형태의 응용시스템을 개발할 수 있으며, 전문가시스템도 예외는 아니다. 본 절에서는 위 4가지 기술을 이용한 기본적인 형태의 전문가시스템 구조를 소개한다. 이 기본적인 형태의 전문가시스템을 응용, 조합하면 상황에 따라 더욱 복잡한 형태의 전문가시스템을 구성할 수도 있지만, 본 논문에서는 그 기초가 되는 기본 구조만을 설명한다.

WWW 기반 전문가시스템은 보통의 WWW 기반 응용시스템과 다른 큰 차이점이 두 가지가 있다.(임규건 외, 1997) 첫째, 보통의 응용 시스템들, 특히 데이터베이스 기반 응용 시스템들은 대부분 사용자의 요구에 따라 그 결과를 한두번 보내 주는 것으로서 전체 처리가 끝나게 된다. 반면에 전문가시스템은 추론과정이 필요하기 때문에 사용자와의 지속적인 상호작용이 필요하다. 그리고 이러한 상호작용결과 얻어진 정보들을 그때 그때 사실 베이스(Fact Base)에 추가하여야 한다. 둘째, 전문가시스템은 많은 양의 계산 능력을 요구하는 경우가 많다. 이러한 경우, 하나의 서버가 여러 명을 지원하는 구조를 갖는 전문가시스템은 서버에 과도한 부담을 줄 수 있다. 따라서, 이 경우에는 클라이언트의 계산 능력을 이용하는 전문가시스템 구조를 고려하여야 한다. 전문가시스템의 이러한 특징을 고려하면서 가능한 WWW 기반 전문가시스템의 기본 구조를 나열하면 아래와 같다.

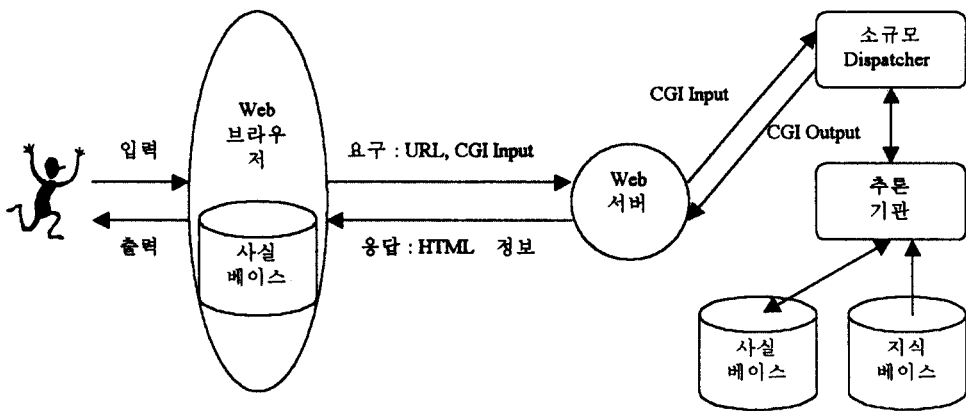
- (1) 추론기관이 서버 측에 있는 경우
  - i) 추론기관을 CGI로 구현하는 경우
  - ii) Web 서버가 추론기관을 내장하는 경우
  - iii) HTML만을 이용하는 경우
- (2) 추론기관이 클라이언트 측에 있는 경우
  - i) External Helper를 사용하는 경우
  - ii) 추론기관을 Java Applet으로 구현하는 경우

먼저 추론기관이 서버 측에 있는 경우를 알아보자. 이것은 다시 세가지 경우로 나뉘는데 첫째는 추론기관을 CGI로 구현하는 경우이다. 이 구조에서는 추론기관이 CGI로 구현되고 추론기관인 CGI 프로그램은 Web 서버로부터 사용자의 입력을 넘겨 받아 지식베이스의 규칙과 사실베이스의 지식들을 바탕으로 추론을 진행하여 그 결과화면을 HTML 형식으로 Web 서버에 보내어 Web 브라우저에 전달되도록 한다. 앞에서 이야기한 바와 같이 추론과정은 사용자와의 지속적인 상호작용을 필요로 하나 WWW은 한번의 요구와 응답으로 한번의 처리가 끝나는 connectionless, stateless 특성을 갖고 있기 때문에 그 사용자와의 먼것번 대화의 내용을 저장하고 있는 사실베이스를 유지하기 위한 방안이 따로 마련되어야 한다. 현재, 이를 위한 방안으로 대표적인 것은 Form의 Hidden Field를 이용하는 방법, Cookie를 이용하는 방법, 데이터베이스에 사용자별로 대화내용을 저장하여 이용하는 방법 등이 있다. Hidden Field와 Cookie를 이용하면 사실베이스는 클라이언트 측에 있게 되며, 데이터베이스를 이용하면 사실베이스는 서버 측에 있게 된다. <그림 4>에서는 사실베이스가 서버 측에 있는 경우와 클라이언트 측에 있는 경우 각각을 한꺼번에 도시하고 있다.



<그림 4> 추천기관을 CGI 로 구현한 경우

위 CGI 방식의 경우 추천엔진이 많은 양의 계산을 필요로 하면 여러 명의 사용자가 동시에 전문가시스템을 사용하고자 할 때 서버에 과도한 부담을 줌으로써 시스템 자원의 부족과 성능저하를 초래하며, 특히 대규모 서비스의 경우 치명적인 영향을 줄 수도 있다. 이를 극복하기 위한 방법으로는 추천기관을 CGI 프로그램이 아닌 디몬으로 바꾸는 방법을 사용한다. 이 방식에서는 Web 서버의 CGI에 의해 소규모의 Dispatcher 프로세스가 구동되고, 이 Dispatcher 프로세스는 클라이언트로서 서버인 추천기관에 서비스를 요구하여 그 결과를 Web 서버를 통하여 사용자에게 보낸다. 이 경우 CGI 프로그램인 Dispatcher는 소규모 프로세스이므로 대규모 서비스에 의한 성능저하가 없게 된다. 그러나, 이 방식을 구현하기 위해서는 Dispatcher와 서버 추천기관 간에 추가적인 Protocol이 마련되어야 하고, 또, 추천기관의 서비스 일정을 계획하고 자원을 분배해주는 추가적인 프로그램이 필요하다. 이것을 도식한 것이 <그림 5>에 나타나 있다.

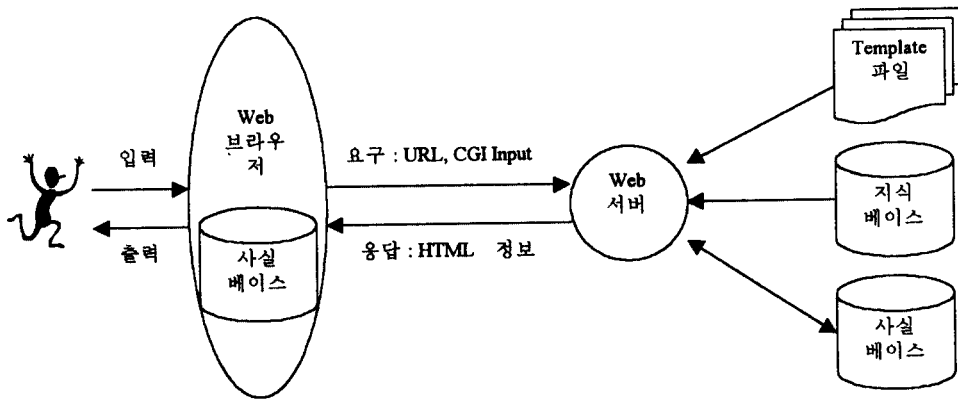


<그림 5> 추천기관을 CGI 로 구현한 경우 (대규모 서비스 용)

둘째로 Web 서버가 추천기관을 내장하는 경우를 살펴보자. 이것은 Web 서버 자신이 추천

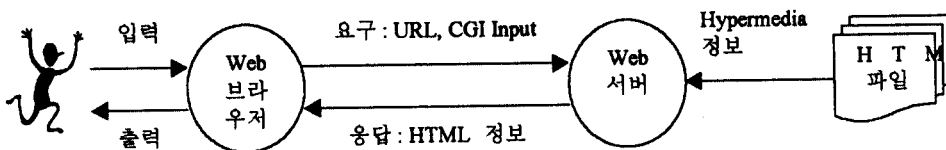


기관이기도 하기 때문에 Web 브라우저로부터 요구를 받아 바로 추론을 수행한 후 그 결과 화면을 HTML 형태로 응답을 보내는 방식이다. 다만, Web 서버의 추론을 제어하기 위한 Script 언어가 추가로 정의되어 있으며 Script 언어로 작성된 제어문장들은 Template 파일형식으로 Web 서버에 제공된다. Web 서버가 추론기관을 내장하는 경우는 두 가지 방법으로 구현이 가능하다. 하나는 추론기관을 내장한 전문가시스템 전용 Web 서버를 개발하는 것이고, 둘째는 기존의 Web 서버에서 지원하는 확장 API(예를 들어 Netscape의 NSAPI 등)를 이용하여 Web 서버에 추론기관을 추가하는 것이다. WWW의 connectionless, stateless 특성은 이 경우에도 마찬가지로 나타나기 때문에 사실베이스도 CGI 방식의 경우와 똑같이 구현된다. 이것을 도식한 것이 <그림 6>에 나타나 있다.



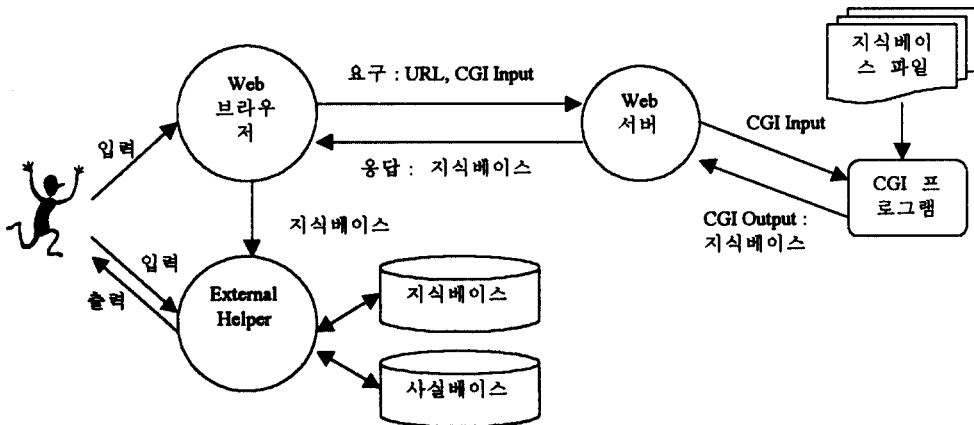
<그림 6> Web 서버가 추론기관을 내장한 경우

셋째로 HTML 파일만을 이용하는 경우는 추론 규칙이 Acyclic Digraph로 표현될 수 있는 역방향 추론용 규칙일 때만 가능하다. HTML의 Hyperlink가 방향성 있는 추론의 역할을 대신해 주기 때문에 Hyperlink된 HTML 파일들의 집합만으로 역방향 추론을 진행할 수 있다. 특히 이 구조의 전문가시스템은 진단, 법률자문 등의 분류문제에서 큰 힘을 발휘한다. 자세한 것은 송용욱, 이재규(1998)를 참조하기 바란다. 이 구조의 전문가시스템을 그림으로 표현하면 <그림 7>과 같다.



<그림 7> HTML 기반 전문가시스템

다음으로 추론기관이 클라이언트 측에 있는 경우를 살펴보자. 이것은 다시 두 가지로 나뉘는데 첫째는 External Helper를 사용하는 경우이다. 이 경우는 Web 서버 측에 있는 CGI 프로그램이 사용자의 문제 성격에 알맞은 지식베이스 파일을 골라서 External Helper를 구동시킬 수 있는 Content-type으로 보내면 그것을 받은 Web 브라우저는 추론기관을 갖고 있는 External Helper를 구동시키면서 Web 서버로부터 온 지식베이스를 넘겨주게 된다. 그 이후에는 사용자가 External Helper와 대화를 통하여 추론을 진행하면서 문제를 풀어나간다. 이 경우 사실베이스는 클라이언트에 만들어진다. 이 과정은 <그림 8>에 나타나 있다.



<그림 8> External Helper 를 사용한 전문가시스템 구조

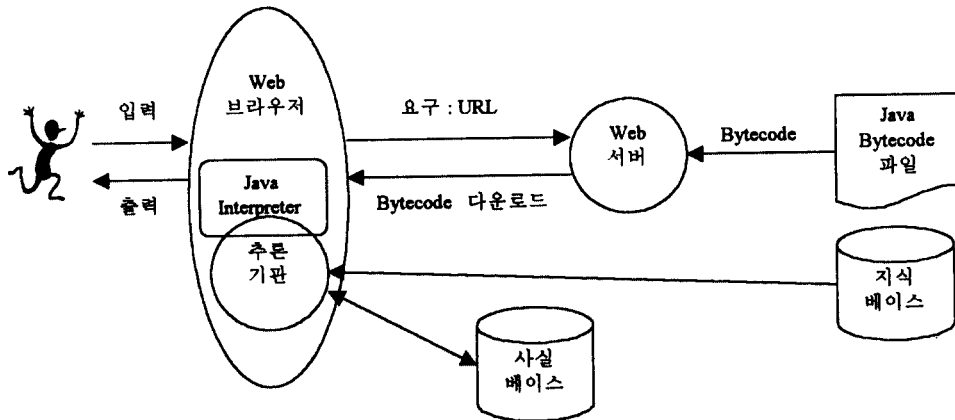
둘째로 추론기관을 Java Applet으로 구현하는 경우는 사용자의 요구에 따라 Java Bytecode가 Web 서버로부터 Web 브라우저에 전달되어 수행된다. 이 경우 지식베이스는 Web 서버가 있는 컴퓨터로부터 Java의 네트워크 기능을 이용하여 전달받는다. 사실베이스는 클라이언트 컴퓨터에 유지될 수도 있지만 Java Applet이 클라이언트 컴퓨터의 파일을 이용할 수 없도록 되어 있으므로 사실베이스의 용량이 큰 경우에는 지식베이스와 마찬가지로 Web 서버가 있는 컴퓨터에 유지되면서 Java의 네트워크 기능을 이용하여 주고 받을 수도 있다. <그림 9>에 이러한 내용이 도시되어 있다.

## V. WWW 기반 전문가시스템의 구조 분석

본 절에서는 앞 절에서 설명한 7가지 WWW기반 전문가시스템 구조들의 장단점을 파악하고 비교해 본다.

### 1. 추론기관이 서버 측에 있는 경우

CGI를 이용하는 경우는 WWW의 Form을 이용하여 사용자의 입력을 받아 처리하는 것이다. 이 방법은 기존의 Web 서버, Web 브라우저, URL, HTTP, HTML CGI 등의 기술을 변경 없이 사용할 수 있으며, 시스템의 개발, 시험이 용이하며 추후의 시스템 확장과 유지보수



<그림 9> Java Applet 으로 구현한 전문가시스템 구조

가 쉽다는 장점이 있다. 이 방식에서 문제가 되는 것은 WWW의 connectionless, stateless 특성 때문에 그 사용자와의 먼첫번 대화의 내용을 기억하기 위한 방안이 따로 마련되어야 한다는 것이다. 현재, 이를 위한 방안으로 대표적인 것은 Form의 Hidden field를 이용하는 방법, Cookie를 이용하는 방법, 데이터베이스에 사용자별로 대화내용을 저장하여 이용하는 방법 등이 있다. 또한, CGI로 구현된 전문가시스템이 추론기관과 지식베이스를 내장하였을 경우 그 프로세스의 크기가 크므로 다수의 사용자가 동시에 서비스를 요구할 경우 시스템 자원의 부족과 성능저하를 초래하며, 특히 대규모 서비스의 경우 치명적일 수 있다. 이를 극복하기 위한 방법으로는 추론기관을 CGI 프로그램이 아닌 디몬으로 바꾸는 방법을 사용한다. 이 방식에서는 Web 서버의 CGI에 의해 소규모의 Dispatcher 프로세스가 구동되고, 이 Dispatcher 프로세스는 클라이언트로서 서버인 추론기관에 서비스를 요구하여 그 결과를 Web 서버를 통하여 사용자에게 보낸다. 이 경우 CGI 프로그램인 Dispatcher는 소규모 프로세스이므로 대규모 서비스에 의한 성능저하가 없게 된다. 그러나, 이 방식을 구현하기 위해서는 Dispatcher와 서버 추론기관 간에 추가적인 Protocol이 마련되어야 하고, 또, 추론기관의 서비스 일정을 계획하고 자원을 분배해주는 추가적인 프로그램이 필요하다.

Web 서버가 추론기관을 내장하는 경우는 두 가지 방법으로 구현이 가능하다. 하나는 추론기관을 내장한 전문가시스템 전용 Web 서버를 개발하는 것이고, 둘째는 기존의 Web 서버에서 지원하는 확장 API(예를 들어 Netscape의 NSAPI 등)를 이용하여 Web 서버에 추론기관을 추가하는 것이다. 이 방식은 추론기관이 Web 서버에 내장되어 있기 때문에 개발, 시험, 시스템 확장, 유지보수 등이 용이하지 않고, 또 현재까지는 추론기관을 Web 서버에 내장하여 사용하는 방법에 대해 구체적으로 연구된 바가 없다. 특히, 전용 Web 서버의 경우는 특정 Web 서버에 종속되기 때문에 차후에 확장하거나 이식하고자 할 때 신축성이 떨어지게 된다.

HTML만을 이용하는 경우는 WWW의 기본 기술인 HTML만을 사용하기 때문에 초보자도 개발, 시험, 확장, 유지보수 등이 매우 용이하다. 또한, CGI를 사용할 때 나타나는 대규모 서비스 시의 시스템 부하 등의 문제도 전혀 없다. 다만, 이 방식이 문제가 되는 것은 HTML만을 사용하여 얼마나 일반적인 전문가 시스템을 구현할 수 있는가 하는 것이다.

## 2. 추론기관이 클라이언트 측에 있는 경우

추론기관이 클라이언트 측에 있는 경우는 다시 두 가지로 나누어 볼 수 있다. 첫째는 External Helper를 이용하는 경우이고, 둘째는 Java Applet으로 추론기관을 구현하는 경우이다.

External Helper 방식을 사용할 경우 추론기관은 External Helper가 갖고 있고, Web 서버로부터는 지식베이스가 전달되면 된다. 또, 추론 도중 추가 지식이 필요하다면 External Helper가 지식서버와 별도의 네트워크로 연결하여 필요한 지식을 가져올 수도 있다. 이 방식의 경우 추론기관은 전적으로 사용자의 컴퓨터에서 하나의 응용프로그램으로서 수행되기 때문에 서버와 네트워크에 추가 부담이 없으며, 따라서, 서버측에 추론기관을 둔 경우보다 빠르게 추론 결과를 얻을 수 있다. 또한, WWW의 connectionless, stateless 특성에서 오는 문제들도 나타나지 않게 된다. 그러나, 이 방식의 문제점은 사용자가 WWW 브라우저 이외에 External Helper라는 별도의 소프트웨어의 기능을 익혀야 하며, 또 External Helper를 사용하기 위하여 사전에 이것을 다운로드를 받아서 설치하여야 한다. 특히, External Helper는 External Helper가 설치되어 수행될 시스템 플랫폼에 맞추어 여러 가지 버전이 준비되어야 한다. 또, External Helper 프로그램이 변경되면 모든 사용자는 변경된 소프트웨어를 다시 다운로드 받아 설치하여야 하므로 시스템 확장과 유지보수가 어렵게 된다.

Java Applet을 이용하는 방식은 External Helper에서 나타나는 시스템 종속성, 설치 등의 모든 문제를 해결한다. 따라서, 추론기관의 개발, 시험, 확장, 유지보수도 용이하다. 동시에, External Helper의 장점을 그대로 향유하게 된다. 이 방식의 단점은 Web 서버로부터 Java 바이트코드를 전달 받는 시간이 길다는 것이다. 또한, HTML보다는 고도의 프로그래밍 능력을 필요로 하기 때문에 초보자가 쉽게 접근할 수 없다는 것도 상대적인 단점으로 지적할 수 있다.

지금까지 설명한 각 방식들의 장단점을 비교하면 <표 1>과 같다.

<표 1>에서 보듯이 전반적으로 우위를 갖는 것은 CGI 방식, HTML 방식, Java 방식임을 알 수 있다. Java를 이용한 개발에 비해 상대적으로 개발이 간편하고 초기 다운로드 시간이 없기 때문에 현재와 같이 느린 인터넷 속도하에서 상대적 우위성을 갖는 CGI 방식이 지금까지 널리 사용되어 왔다. (EXSYS) 그러나, 네트워크 속도가 개선되어 초기 다운로드 시간이 충분히 짧아지게 되면 대부분의 전문가시스템은 Java 방식으로 이행할 것으로 보인다. CGI나 Java에 대한 이러한 깊은 관심과는 대조적으로 HTML 방식에 대한 관심은 상대적으로 매우 적었던 것으로 보인다. 그 이유는 HTML 만으로는 일반적인 전문가시스템을 구현하는 것이 불가능하기 때문인 것으로 생각된다. 그러나, HTML의 Hyperlink 기능이 의사결정나무의 구현 및 일반적인 역방향 추론의 구현을 가능하게 할 수 있다는 점에 착안하면, 적어도 역방향 추론을 이용한 진단, 분류 등의 분석문제에 관한 한 HTML 방식의 전문가시스템은 충분히 구현 가능하며, 그 시스템은 <표 1>에서와 같은 장점을 향유하게 된다.

<표 1> WWW 기반 전문가시스템 구조 비교

	서버측 추론기관					클라이언트측 추론기관	
	CGI		WEB 서버		HTML	External Helper	Java
	내장식	디몬식	전용	API			
개발용이성	○	○			○	○	○
시험용이성	○	○			○	○	○
확장성	○	○			○		○
이식성	○	○			○		○
유지보수	○	○			○		○
대규모서비스		○	○	○	○	○	○
상태저장 불필요					○	○	○
부가프로그램 불필요	○		○	○	○	○	○
설치 불필요	○	○	○	○	○		○
수행준비속도	○	○	○	○	○	○	
수행속도					○	○	○
문제의 일반성	○	○		○		○	○

## VI. 결 론

이 논문에서는 WWW 기반 전문가시스템의 구현 구조들을 알아보고 장단점을 비교해 보았다. 현재는 CGI를 이용하는 방식과 Java를 이용하는 방식이 개발, 시험, 확장, 이식, 유지보수의 용이성 등의 장점 때문에 널리 사용되고 있음을 알 수 있었다. 이에 반하여 HTML을 이용하는 방식은 그 일반성의 부족 때문에 현재 그다지 각광을 받지 못하고 있다. 그러나, 역방향 추론에 초점을 맞추어 보면 HTML은 충분히 사용 가능하며, HTML을 이용하여 구현된 시스템은 CGI나 Java 보다 더 큰 장점을 갖는다. 따라서, 적어도 분석문제 분야에서는 HTML 방식이 앞으로 널리 사용되리라 생각한다. 앞으로의 이 방식에서 더 연구해야 할 것으로는 규칙의 일반성을 얼마나 더 높일 수 있느냐 하는 것과 추론을 위해 생성된 HTML 파일들을 효과적으로 관리, 유지, 보수할 수 있는 방안에 대한 것 등이 있다.

## &lt; 참고 문헌 &gt;

- 김평철, 웹을 위한 데이터베이스 통로의 분류체계, WWW 96-1(1996), 50-66.
- 송용욱, 이재규, WWW 기반 전문가시스템의 구조 분석과 HTML 기반 역방향 추론기관의 구현 방안, 한국경영정보학회/한국전문가시스템학회 '98 공동춘계학술대회, 1998.5.29-30.
- 임규전, 강주영, 이재규, Web 기반 전문가시스템의 구조 분석, 한국전문가시스템학회 97 추계학술대회, pp. 63-73.
- Berners-Lee, T. and D. Connolly, Hypertext Markup Language - 2.0, RFC1866, November 1995.
- Berners-Lee, T., R. Fielding, and H. Frystyk, Hypertext Transfer Protocol - HTTP/1.0, RFC1945, May 1996.
- Danesh, Arman, Teach Yourself JavaScript in a Week, Sams.net Publishing, 1996.
- Eriksson, Henrik, Expert Systems as Knowledge Servers, IEEE Expert, June 1996, pp. 14-19.
- EXSYS, Inc., Moving an EXSYS Application to the EXSYS Web Runtime Engine (WREN).
- Far, Behrouz H. and Zenya Koono, EX-W-Pert System: A Web-Based Distributed Expert System for Groupware Design, Expert Systems with Applications, Vol. 11, No. 4, 1996, pp. 475-480.
- Giarratano, Joseph and Gary Riley, Expert Systems: Principles and Programming, 2nd Edition, PWS Publishing Company, 1994.
- Intelligent Internet Systems: Tools and Applications Part II, Intelligent Software Strategies, Vol. 13, No. 1, Winter 1997, pp. 4-6.
- Lee, J. K., I. K. Lee, and H. R. Choi, Automatic rule generation by the transformation of Expert's Diagram: LIFT, Int. J. Man-Machine Studies, 32(1990), 275-292.
- Lemay, Laura and Charles L. Perkins, Teach Yourself JAVA in 21 Days, Sams.net Publishing, 1996.
- Liebowitz, Jay, Worldwide Perspectives and Trends in Expert Systems, AAAI, Summer 1997, pp. 115-119.