

마이크로 컴퓨터와 전문가 체제

한국과학기술원 이 재 규
경영과학과(工博)
한국과학기술원 이 호 근
경영과학과

I. 序論

전문가 체제(Expert System: 이하 ES라 부름)란 희귀하고 값비싼 전문가의 知識과 經驗을 체계화하여 컴퓨터에 기억시켜 둠으로써 전문가가 아닌 많은 사람들이 전문가를 만나지 않고서도 ES를 통하여 전문가의 능력을 빌릴 수 있도록 한 것이다.

전문가 체제는 오래전부터 컴퓨터 전문가들의 연구 분야였던 人工知能(Artificial Intelligence)을 그 모체로 하고 있으나 최근에는 경영정보체제(Management Information System)나 의사결정 지원체제(Decision Support System)분야에 종사하는 분들의 ES에 대한 관심이 날로 증가하고 있다. 왜냐하면 지금까지 경영과학 기법의 主種은 모델과 데이터였으나 다음 세대의 경영과학 기법은 그 위에 ES에서 다루는 Knowledge가 첨가된 형태가 그 대중을 이루게 될 것이라는데 인식을 같이 하기 때문이다.

따라서 본 지면을 통하여 전문가 체제의 원리와 그 開發, 특히 마이크로 컴퓨터를 사용한 ES의 개발에 대한 概括的인 내용을 소개함으로써 ES 응용의 보다 폭넓은 활성화에 도움을 주고자 한다.

II. 전문가 체제(Expert System)

초기 人工知能(Artificial Intelligence) 분야의

연구자들은 몇가지의 논리적인 推論方法(Reasoning Method)과 컴퓨터의 능력을 결합시킴으로써 모든 문제를 해결할 수 있는 범용 문제 해결 시스템(General Problem Solver)을 만들수 있으리라 믿었다. 그러나 이 분야에서의 연구경험이 축적되면서 General Problem Solver와 같은 시스템은 그 자체가 가지는 一般性으로 인하여 복잡한 현실문제를 해결하기에는 부적합하다는 결론과 함께 문제 해결의 중요한 要素로서 특정한 문제 영역에 대한 Knowledge에 관심을 모으기 시작하였다.

즉 일상생활에서 전문가가 자신의 전문분야에 관한 문제 해결을 非專門家보다도 월등하게 수행하는데, 이는 비전문가들이 가지고 있지 않은 Knowledge를 전문가가 가지고 있기 때문이라는 사실에 認識을 같이 하게 되었기 때문이다.

이러한 배경하에서 非專門家인 사용자가 마치 전문가에게 諮問을 구하듯이 컴퓨터에게 諮問을 구할수 있도록 전문가의 知識과 經驗을 컴퓨터가 처리할 수 있게끔 코드화하여 컴퓨터에 넣어 둔 것이 전문가 체제이다.

의사가 환자에게 자신의 지식체계에 따라 필요한 순서대로 질문하고 판단하여 병을 진단하는 과정을 컴퓨터가 실시할 수 있게 한 MYCIN이 1976년에 Stanford 大에서 개발된 이후로 ES는 여러 분야에서 開發되어 광택담사, 컴퓨터 기기 구성, 분자구조 설명, 의학적 진단 및 처방과 같은 기능을 성공적으로 수행하고 있다.

ES의 구조는 시스템에 따라 약간의 차이는

있으나 다음에 설명한 Knowledge 베이스, Inference Engine, Justifier와 Knowledge Acquisition 부분으로 구성되는 것이 보편적이며, 그 구조는 <그림-1>과 같다.

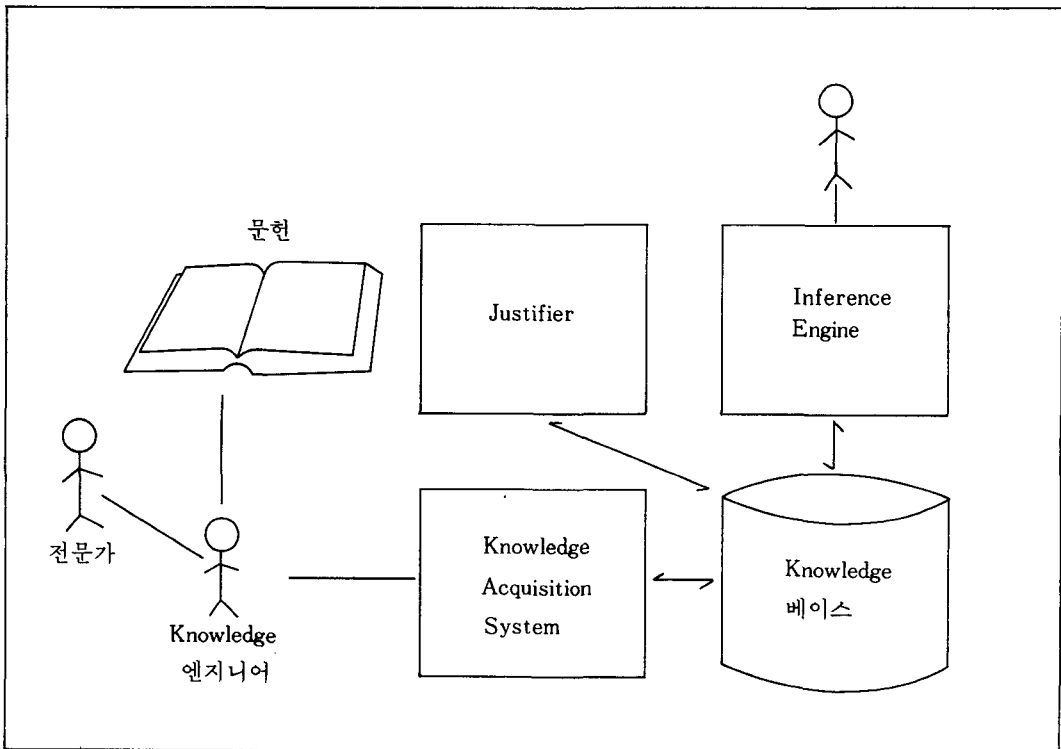
A. Knowledge 베이스

문제 해결을 위한 Knowledge를 모아 놓은 것을 Knowledge 베이스라고 하는데, Knowledge와 Knowledge 베이스의 관계는 데이터와 데이터 베이스의 관계와 유사하다.

Knowledge를 컴퓨터에 저장하기 위해서는 Knowledge가 어떻게 구성되어 있는가를 把握하고 표현할 수 있어야 한다. 그래서 전문가가 소

유하고 있는 Knowledge를 표현하는 방법, 즉 Knowledge Representation에 대한 연구가 많이 진행되어 왔는데, 가장 널리 알려진 Knowledge Representation 방법으로는 Production System, Semantic Network, Frame과 같은 것들이 있다.

실제 전문가들의 지식은 “어떤 상황에서는 어떻게 한다”는 식의 규칙에 바탕을 둔 경우가 많으므로 條件(Condition)과 行動(Action)의 형태로 Knowledge를 표현하는 Production System (또는 Rule-Based System)이 가장 많이 사용되고 있다. Production System은 IF - THEN 규칙의 형태로 Knowledge를 표현하는데 Pro-



<그림-1> 전형적인 전문가 체제의 構造

duction System을 사용하여 Knowledge를 표현하는 간단한 예를 보이면 다음과 같다.

“육식동물은 포유류로서 고기를 주식으로 한다. 그리고 호랑이는 육식동물로서 갈색 바탕에 검은 줄무늬를 가지고 있다”라는 Knowledge를 IF-THEN 규칙을 사용하여 다음과 같이 나타낼 수 있다.

```
IF the animal is mammal and it eats meat
THEN the animal is canivore
IF the animal is canivore and it has
    tawny color and it has black stripes
THEN the animal is tiger
```

육식동물(Canivore)이 미리 정의된 후에 이 지식이 그다음 Rule의 조건부분에서 사용된 것은 특기할 만하다.

B. Inference Engine

전문가는 단순히 知識만을 가지고 있는 것이 아니라 자신이 가지고 있는 Knowledge에 근거하여 어떤 推論過程을 거침으로써 문제 해결을 위한 결론에 도달하게 된다. ES가 실제 전문가와 같은 기능을 수행할 수 있을려면 Knowledge 베이스 이외에, Knowledge 베이스를 이용하여 주어진 문제의 해결을 위한 결론에 도달하는 推論機能(Inference Engine)을 소유하고 있어야 한다.

推論方法으로는 Forward Chaining 과 Backward Chaining이 가장 많이 쓰이며 시스템에 따라서는 이 둘을 혼용하는 경우도 있는데 추론방법의 선택에 따라 같은 Knowledge 베이스를 사용한다 할지라도 구체적 문제에 대한 질문의 도출 및 결론의 추론 과정이 달라지게 된다.

Forward Chaining (또는 data driven control strategy)은 어떤 결론에 도달할 때까지 Knowledge 베이스를 처음부터 훑어 나가는 방법인데 반하여 Backward Chaining (또는 goal driven control strategy)은 어떤 결론을 목표로 정해두고 이 결론이 맞는가를 확인하기 위해 Knowledge 베이스를 사용한다.

C. Justifier

ES가 기존의 소프트웨어와 크게 다른 점중의 하나는 어떤 결론을 내렸을때 왜 그러한 결론에 도달했는지에 대한 이유를 설명할 수 있다는 점이다.

일상생활에서 전문가에게 諮問을 구할 경우 우리는 전문가로부터 그의 결론과 함께 그가 왜 그러한 결론에 도달했는가를 설명 받을 수 있다. 따라서 ES가 궁극적으로 전문가와 같은 기능을 담당할 수 있을려면 사용자가 요구할 경우 그의 결론에 대한 추론과정을 설명할 수 있어야 한다. 시스템에 따라서는 推論過程에 대한 설명뿐만 아니라 결론의 정확성에 관한 情報를 제공하기도 한다.

D. Knowledge Acquisition

Knowledge 베이스를 구축하기 위해서는 선택된 Knowledge Representation의 틀에 맞도록 전문가나 문헌 등으로부터 자료를 수집·정리하여 컴퓨터에 입력시켜야 한다. 이러한 과정을 Knowledge Acquisition이라 부르며 문헌고찰과 전문가와의 인터뷰를 통하여 Knowledge Acquisition을 수행하는 사람을 Knowledge 엔지니어라 부른다.

전문가의 지식이나, 문헌으로부터의 이론을 체계화하여 컴퓨터에 입력시키는 작업은 많은 시간과 노력을 요구하며 실제 ES의 개발 과정 중 가장 費用이 많이 드는 부분이다. ES가 얼마나 정확하고 충분한 Knowledge를 소유하고 있는가가 ES의 성공 여부를 좌우하는 중요한 要素라고 볼때 Knowledge Acquisition은 그만큼 중요하고도 힘든 작업이라고 할 수 있겠다. 뿐만 아니라 Knowledge가 靜態的이 아니라 시간에 따라 動態的으로 변하는 분야에서는 Knowledge 자체를 항상 유지 및 보수(Maintenance)해 주는 작업도 뒤따라야 한다.

따라서 최근에는 자동적으로 Knowledge Acquisition을 수행하는 능력과, Knowledge 자체를 관리하는 능력을 소유한 시스템에 관한 연구가 활발히 전개되고 있다. 실제로 많은 人工知能(AI) 연구들이 1990年代에는 시스템 자체가 學習(Learning)을 통하여 스스로 Knowledge

를 습득하는 Machine Learning이 AI분야의 중요한 연구과제가 될 것으로 믿고 있기도 하다.

이상은 ES의 원리에 관한 總括的인 내용들이다. 보다 세부적인 사항을 알고 싶은 讀者를 위하여 본 지면의 후반부에 ES에 관한 참고문헌을 첨부하기로 한다.

Ⅲ. ES의 開發을 위한 道具(Tools)

ES를 개발할 경우 표준화된 汎用 프로그래밍 언어를 사용하여 처음부터 문제 성격에 맞는 ES를 구축할 수도 있으나 ES의 개발을 위해 특별히 만들어진 소프트웨어를 사용함으로써 개발 시간과 노력을 단축 시킬 수도 있다.

A. 범용 프로그래밍 언어(General Purpose Language)

ES는 표준화된 범용 프로그래밍 언어인 LISP, Prolog, PASCAL, FORTRAN, ASSEMBLER 등을 사용함으로써, 문제의 성격에 가장 적합한 Inference Engine 과 Knowledge Representation을 설계하고 코드화하여 개발할 수 있다. 범용 프로그래밍 언어 가운데 LISP가 가장 많이 사용되는데 이는 LISP가 Symbol을 처리하는데 용이하기 때문이다. 그러나 LISP를 사용할 경우 ES의 개발이 편리하다는 것이지 반드시 LISP로 ES를 개발하여야 하는 것은 아니다. 실제로 EXPERT는 FORTRAN을 사용하여 개발한 ES이며 M-1은 ASSEMBLER로 되어 있다.

범용 프로그래밍 언어를 사용할 경우 비록 시간과 노력이 많이 소모되기는 하나 주어진 문제에 가장 적합한 Knowledge Representation 과 Inference Engine을 구축할 수 있는 장점이 있다.

B. 범용 표현 언어(General Purpose Representation Language)

OPS 5, HEARSAY-Ⅲ, AGE와 같이 특별히 Knowledge 엔지니어링(Knowledge 엔지니어가 Knowledge Acquisition을 하는 過程)만을 위해 開發된 범용 표현 언어를 사용하여 E

S를 개발할 수 있다.

특수하고 구체적인 문제 영역에서 사용될 수 있도록 개발된 Shell(추후 설명함)과는 달리 이들은 문제 영역에 무관하게 단지, ES의 개발을 쉽게 할 수 있도록 만들어진 소프트웨어이다. 예를 들어 OPS 5는 DEC이 VAX의 판매를 위한 시스템 구성도를 마련하기 위해 사용하는 전문가 체제인 R 1을 개발하는데 사용 되었는데 OPS 5가 Knowledge 엔지니어링에 필요한 기본적인 메카니즘을 제공하기는 하지만 어떤 특정한 문제 해결 전략이나 표현체계 (Representational Scheme)를 위해 만들어진 것은 아닌 것이다.

언어 이해 시스템인 HEARSAY-Ⅱ를 모체로 하고 있는 HEARSAY-Ⅲ도 선정된 문제 영역 내에서 잘못된 기능의 수정과 實驗을 통하여 ES를 구축할 수 있도록 표현과 제어체제를 제공한다든 점에서 OPS 5와 그 맥락을 같이 한다고 볼 수 있다.

AGE는 Knowledge를 이용하는 프로그램을 위한 여러가지 기저틀(Framework)에 대한 設計와 테스트를 통하여 ES 개발가가 자신의 문제에 적합한 Knowledge Representation과 Inference Engine을 선택할 수 있는 환경을 제공한다든 점에서 이 범주에 속한다고 볼 수 있다.

이 외에도 RLL, ROSIE 등을 들 수 있는데, 범용표현언어는 대상 문제의 속성에 민감한 Shell 과는 달리 특정한 틀이나 문제 영역에 묶여 있지 않으므로 Shell 보다는 널리 사용될 수 있는 제어구조를 제공하고 있는 셈이다. 그러나 범용 프로그래밍 언어를 사용하여 ES를 개발하는 경우에 비하면 융통성이 부족하다고 하겠다.

C. Shell (또는 Skeletal System)

Shell은 Knowledge Acquisition과 Inference를 할 수 있도록 미리 마련된 소프트웨어이다. MYCIN은 MYCIN에서 사용한 Knowledge Representation 체계와 Inference Engine을, MYCIN과 유사한 환경을 갖는 새로운 문제 영역에 사용할 수 있도록 만든 Shell이다.

즉 MYCIN이 환자의 박테리아 감염 여부를 진단하는데 사용하기 위해 만들어진 ES이므로, 다른 병에 대한 진단지식을 MYCIN이 사용하는 Knowledge 표현체계를 통하여 Knowledge 베이스화하고 그 위에 MYCIN에서 사용하는 Inference Engine을 껍데기(Shell)처럼 씌움으로써 새로운 진단용 ES를 구축할 수 있도록 한 것이 EMYCIN이다. 따라서 EMYCIN은 그 자체가 Knowledge 베이스를 갖고 있지는 않으며 시스템 개발가는 단지 Knowledge Acquisition만 함으로써 새로운 ES를 개발할 수 있으므로 그만큼 ES의 개발에 시간과 노력이 단축될 수 있는 것이다.

이와 마찬가지로 물리브덴 광맥을 찾는 ES인 PROSPECTOR의 Knowledge Representation 체계와 Inference Engine을 그대로 사용하여 다른 광맥(예를 들면 우라늄)에 대한 Knowledge Acquisition만으로 새로운 ES를 만들 수 있도록 한 KAS도 Shell의 범주에 속한다. 이외에도 녹내장의 진단과 처방에 대한 ES인 C-ASNET으로부터 진화된 EXPERT도 EMYCIN이나 KAS와 그 맥락을 같이 한다고 볼 수 있다.

Shell은 특정한 문제를 대상으로 한 ES로부터 파생된 시스템이므로 새로운 대상문제가 가지는 환경, 즉 새로운 문제의 Knowledge 표현 체계나 推論機能이 Shell의 그것과 부합되어야만 사용할 수 있다는 제약조건을 가지고 있다. 그러나 새로운 문제 영역이 가지는 환경과 기존의 Shell이 부합될 경우 Shell의 사용으로 ES의 개발 시간과 노력을 상당히 단축시킬 수 있다.

IV. 마이크로 컴퓨터와 Shell

단 한번에 완벽한 Knowledge 베이스를 만든다는 것은 거의 불가능하기 때문에 1차적으로 Prototype을 開發하여 이를 시험하면서 점진적으로 Knowledge를 추가해 나가는 것이 ES 개발의 전형적인 형태이다. 이때 Inference Engine과 Knowledge Representation 체계가 문제 환경에 적합하다면 Knowledge를 점진적으로 추

가해 나가는 동안에도 초기에 설정된 Inference Engine에는 거의 修正을 가할 필요가 없게 된다.

따라서 문제 환경에 가장 적합한 Shell을 사용할 수만 있다면 ES 개발에 많은 시간과 노력을 절약할 수 있을뿐 아니라 이러한 여력을 Knowledge Acquisition에 집중시킬 수 있으므로 적합한 Shell의 선택과 사용은 매우 중요하다고 할 수 있겠다.

최근 마이크로 컴퓨터 분야의 급속적인 발전은 과거에 대형 컴퓨터에서만 가능하던 문제 해결 방법이 마이크로 컴퓨터에서도 가능케 함으로써 컴퓨터 産業에 새로운 變革을 일으키고 있다. 즉 마이크로 컴퓨터의 보급이 확산되면서 대형 컴퓨터에서 사용되던 많은 소프트웨어가 마이크로 컴퓨터용으로 변모되어 소프트웨어市場에 쏟아져 나오고 있는데 이러한 현상은 ES에서도 예외는 아니어서 마이크로 컴퓨터에서 사용할 수 있는 Shell이 많이 개발되고 있다.

이제는 人工知能(AI)의 전문가가 아니라도 마이크로 컴퓨터용 Shell을 잘 선택함으로써 과거 보다는 적은 비용과 노력으로 ES를 개발할 수 있는 환경이 조성된 것이다. 그러나 이러한 마이크로 컴퓨터용 Shell을 경영과학 기법을 위한 ES의 道具로 사용하기에는 아직 몇가지의 문제점이 남아 있다.

첫째는 이미 전술한 바 있는 Shell 자체가 가지는 融通性의 부족이다. 물론 EMYCIN, KAS, EXPERT와 같은 Shell은 특정한 문제 해결을 위한 ES로부터 派生되어 나온 반면 마이크로 컴퓨터용 Shell은 특정한 문제 영역으로부터 진화되어 나온 것이 아니라 개발 목적 자체가 汎用을 겨냥한 것이므로 前者보다는 一般性(Generality)을 가지고 있다고 볼 수 있다. 그렇다고 하더라도 Shell 자체가 가지고 있는 융통성의 제약에서 완전히 자유롭다고 할 수는 없으므로, 결국 문제 환경에 가장 적합한 Shell의 신중한 선택은 아무리 강조해도 지나치지 않는 ES 개발 담당자의 역할로 남아있게 된다.

둘째는 計算機能의 부족이다. ES가 경영과학 기법의 일부분으로 사용되기 위해서는 데이터나 모델과의 Interface를 위한 계산기능이 충분해

야 한다. 현존하는 마이크로 컴퓨터用 Shell 들이 필요한 계산기능을 충분히 제공하지는 못하지만 Language Interface 등을 통하여 이러한 결점을 보완할 수 있는 길을 어느 정도 보여주고 있다. 예를 들어 ES/P에서는 Prolog 언어와의 Interface가 가능하며 INSIGHT에서는 d-BASE와의 Interface가 가능하다. 따라서 經營科學 기법의 일환으로 ES를 개발하고자 할 경우에는 Shell의 특성을 충분히 활용할 수 있는 환경도 함께 이루어져야 한다.

컴퓨터 産業에서 마이크로 컴퓨터 사용의 급증과, ES 개발에서 Shell의 중요성을 감안할 때 마이크로 컴퓨터用 Shell에 대한 수요와 공급의 증가는 계속되리라 믿어진다. 참고로 가장 널리 쓰이는 범용표현 언어 (General Purpose Re-

presentation Language) 와 Shell, 그리고 잘 알려진 마이크로 컴퓨터用 Shell을 <표-1>에서 紹介한다.

다음 節에서는 대표적인 마이크로 컴퓨터 用 Shell중의 하나인 ES/P의 構造와 이를 이용한 ES의 開發事例를 소개함으로써 마이크로 컴퓨터를 사용한 ES의 개발을 원하는 經營과학자들에게 도움을 주고자 한다.

한가지 讀者에게 양해를 구하고 싶은 것은 이 분야의 역사가 美國에서도 얼마되지 않았으므로 우리나라에서 독자적으로 만든 Shell이 없기 때문에 영문으로 된 事例를 소개할 수 밖에 없다는 점이다. 우리 상황에 맞는 Shell의 開發과 함께 한글 Interface 기능의 첨가는 이 분야에 종사하는 모든 Researcher의 선결 과제이다.

<표-1> ES 開發을 위한 道具

특 성	범 용 표 현 언 어 (General Purpose Representation Language)						
	OPS5	ROSE	RLL	HEARSAY	AGE		
Inference Engine	Program	Backward Chaining Forward Chaining Change-driven	Program	Black Board	Black Board		
Uncertainty Handling	Program	?	Program	?	Program		
Knowledge Representation	Production Rules	Production Rules	Production Rules	Production Rules	Production Rules		
Language Interface	Anything	LISP	INTERLISP	LISP	LISP		
장 점	Generality	English-like Syntax Simulation Facility	Generality	General Purpose Control	Generality		
특 성	Shell			마이크로 컴퓨터 用 Shell			
	EMYCIN	KAS	EXPERT	EXSYS	ES/P	TIMM-PC	M-1
Inference Engine	Backward Chaining	Backward Chaining Forward Chaining	Forward Chaining	Forward Chaining	Backward Chaining Forward Chaining	Pattern-matching	Backward Chaining Forward Chaining
Uncertainty Handling	Probability Interface	Probability Interface	Probability Interface	Probability Interface	Program	Probability Interface	Probability Interface
Knowledge Representation	Production Rules	Production Rules	Production Rules	Production Rules	Production Rules	Production Rules	Production Rules
Language Interface	LISP	LISP	FORTRAN	-	PROLOG	-	ASSEMBLER
장 점	User Interface	User Interface	User Interface Multiple Database	User Interface	Language Interface	Pattern Matching	English-like Syntax
가 격	-	-	-	\$300	\$900	\$9,500	\$10,000

V. ES/P

A. ES/P의 構造

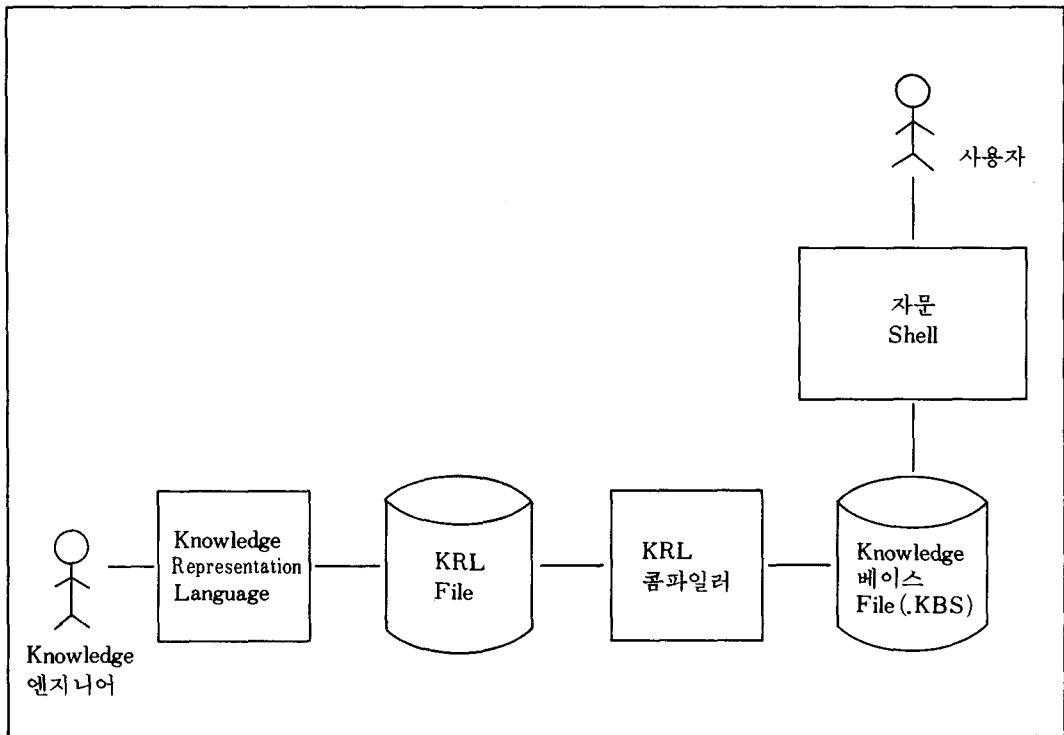
ES/P는 美國의 Expert Systems International社가 개발한 마이크로 컴퓨터用 Shell로서 컴퓨터로 하여금 전문가와 같이 사용자에게 諮問을 줄 수 있도록 만든 소프트웨어다. ES/P의 구조는 <그림-2>와 같이 KRL 콤파일러와 자문 Shell로 이루어져 있다.

ES/P를 사용하는 동안 컴퓨터는 사용자가 도움을 원하는 주제에 대해 일련의 질문을 던지며, 이 질문에 대한 諮問이나 정보를 화면에 디스플레이 하게 된다. 이를 위하여 Knowledge 베이스는 ES/P가 화면에 디스플레이하기 위해 필요로 하는 질문과 자문에 대한 내용을 모두

포함하고 있어야 한다. ES/P는 Knowledge 엔지니어가 Knowledge 베이스를 만들때 읽고 쓰기가 용이한 Knowledge Representation Language(KRL)을 제공하고 있으므로 Knowledge 엔지니어는 KRL의 Syntax에 맞도록 Knowledge 엔지니어링을 하면 된다.

KRL은 Knowledge 엔지니어를 위해 設計된 것이지 컴퓨터가 읽을 수 있는 형태는 아니다. 따라서 KRL로 이루어진 Knowledge 베이스 File을 諮問 Shell이 읽고 수행할 수 있는 기계어로 바꾸기 위해 KRL 콤파일러가 사용된다. 이때 콤파일된 File은 KBS라는 Extension을 가지게 되는데 자문 Shell은 이 File을 사용하게 된다.

예를 들어 PAYE. KRL은 KRL로 이루어진 Knowledge 베이스 File인데 이를 KRL 콤파일러로 콤파일을 시키게 되면 PAYE. KBS File



<그림-2> ES/P의 構造

이 생기게 되는 것이다.

KRL 콤파일러를 사용할 경우 Syntax Error Check뿐만 아니라 Knowledge 베이스의 一貫性(Internal Consistency)을 검토할 수 있다는 장점이 있다.

B. ES/P Knowledge 베이스

KRL을 사용하는 Knowledge 베이스는 여러 개의 Section으로 되어 있으며 각 Section 은 전형적으로 다음과 같은 사항을 포함하고 있다.

1. Paragraph

Paragraph는 사용자가 질문을 구하는 동안 컴퓨터 화면에 나타나는 내용을 담고 있다. 각 Paragraph는 무조건적으로 컴퓨터 화면에 디스플레이 될 수도 있으나 일반적으로 諮問을 돕기 위해 일정한 條件이 만족될 때만 화면에 디스플레이 되는 것이 보통이다.

2. Reference 명령문

이 명령문은 의사결정 추리(Decision Tree)에서 分枝(Branching)와 같은 역할을 수행한다. 즉 일정한 조건이 만족되면 그 다음 질문을 계속하기 위해 필요한 Section을 불러낸다.

3. 매개변수(Parameter)의 定義

諮問을 위해 사용되는 媒介變數는 다음과 같은 내용을 담고 있다.

a. Name

매개변수의 이름으로서 조건식이나 표현에서 매개변수를 지정할 때 사용된다.

b. 매개 변수의 類型(Type)

- (1) Number : 매개 변수가 수치 (numeric value)를 가질 때
- (2) Phrase : 매개 변수가 이름이나 주소와 같은 간단한 Text를 가질 때
- (3) Cotegory : 매개 변수가 일정한 수의 목록 중 하나를 가질 때(예를 들면 빨강·노랑·파랑 중의 하나)
- (4) Fact : 매개 변수가 참 또는 거짓중의 하나를 구분할 때

c. 설명부분(Explanatory text)

ES/P가 사용자에게 질문을 하였을 때 사용

자가 질문자체에 대해 더 많은 정보를 요구할 경우 화면에 나타나는 내용을 담고 있다.

d. 범위(Range)

매개 변수의 유형이 Number일 때 매개변수가 가질 수 있는 수치의 범위를 나타낸다.

e. 선택 목록(List of options)

매개변수의 유형이 Category일 경우에 매개변수가 가질 수 있는 가능한 Option들의 목록이다.

f. 규칙(Rules)

매개 변수의 유형이 Number일 경우에 매개변수의 값을 계산하기 위해 사용되는 數式을 뜻한다.

g. 質問(Askable)

ES/P가 매개 변수에 관련된 내용을 사용자에게 물어보기 위해 화면에 비추는 질문내용이다.

매개 변수가, 위에서 열거한 사항들을 모두 담고 있어야 하는 것은 아니지만 Name과 유형(Type), 그리고 질문(Askable)은 반드시 갖고 있어야 할 사항들이다.

VI. 세무관리 諮問에의 ES/P 應用

A. 事例 내용

조직의 규모가 클 경우 새로 고용하는 피고용자의 階層과 數는 적지 않으며 그 횟수 또한 빈번하다. 따라서 새로 고용하는 직원의 수입에 대한 항목별 세금을 결정하는 것은 그리 간단한 문제가 아니다. 뿐만 아니라 똑같은 조건으로 같은 職에 고용된 직원이라도 그들의 상황이 다르면 서로 다른 과세가 적용되어야 하므로 그 복잡성은 더욱 커지게 된다.

組織내에서 수년간 이 업무를 담당하여, 수입에 대한 항목별 세금 결정에 대한 規定이나 資料를 잘 알고 있는 전문가가 직장을 떠날 경우 새로운 직원이 전임자가 갖고 있던 Knowledge를 습득하는 데는 많은 시간과 노력이 필요하다. 이런 경우 전문가의 Knowledge나 세금산정의 규정에 관한 항목들을 컴퓨터에 입력시켜 돕으

로써 컴퓨터로 하여금 전문가(전임자)의 역할을 대신하게 한다면 신입자가 Knowledge를 습득하는데 시간을 낭비하지 않아도 된다. 게다가 전문가가 조직내에 남아 있다 하더라도 체계적으로 조직화된 Knowledge를 사용할 수 있으므로 일관성 있는 세금산정을 기할 수 있다.

이러한 背景하에서, 조직이 새로운 직원을 고용할 경우 피고용인의 수입에 대한 항목별 세금산정을 위해 Knowledge를 모아놓은 것이 PAYE. KRL File이다.

본 예제에서 사용하는 Knowledge 베이스(PAYE. KRL)는 Her Majesty's Stationary Office에서 발행한 "The Employer's Guide to PAYE"라는 책자로부터 발췌한 内容에 근거한 것이다.

B. Knowledge베이스 PAYE.KRL의 구조

본 예제에서 사용된 PAYE. KRL은 모두 6개의 주요 Section으로 되어 있다. (<표-2> 참조)

Control Section은 도입부분으로서, 모든 Section이 해당 Section에 대한 선언으로 시작되는 반면 Control Section에서는 Section에 대한 선언이 필요없다. 일반적으로 Control Section에서는 Knowledge 베이스의 제목을 비롯한 概括의인 内容을 화면을 통하여 보여준다.

Section Para-95에서는 새로운 피 고용자에게 문서 P45의 소지 여부를 물어본다. 여기

서 P45는 세금산정을 위해 필요한 문서의 이름이다. 피 고용자가 P45를 소지하고 있으면 Section Para-96으로 보내져 P45를 이용한 세금산정 지침을 화면을 통하여 사용자에게 보여지게 된다. 피 고용자가 P45를 소지하고 있지 않은 경우에는 Section Para-97로 보내져 피 고용자의 개인 상황에 대해 질문을 계속한다.

이때 피고용자의 고용기간이 얼마나 될 것인가에 따라 수입에 대한 항목별 세금산정이 달라지는데 고용기간이 1주일보다 짧은 경우는 Section Para-99에서 세금산정 지침을 보여주는 반면 고용기간이 1주일 이상이면 Section Para-98로 가서 해당 세금산정 지침을 보여지게 된다.

PAYE. KRL의 内容과 앞에서 언급한 ES/P Knowledge 베이스의 구조를 연결시킴으로써 ES/P에 대한 이해를 넓힐 수 있으리라고 본다. 지면상의 제약으로 PAYE. KRL 전체를 보여줄 수는 없으나, 일부인 Section Para-95에 나와 있는 内容을 Knowledge 베이스의 구조와 연결시켜 설명하기로 한다.

c. Section Para-95의 構造

Section Para-95의 内容은 <부록>에 나와 있다. "Section Para-95:"는 Section의 선언 부분으로서 Control Section을 제외한 모든 Section은 Section의 선언으로 시작하여야 한다. (/ *와 * / 사이의 문장은 Comment文이다.)

<표-2> PAYE.KRL의 Section과 内容

Section 이름	내 용
Control	도입부분
Para-95	문서 P45의 소지여부를 피 고용인에게 물어본다.
Para-96	피 고용인이 P45를 소지하고 있을 경우에 대한 지침(Instruction)을 보여준다.
Para-97	피 고용인이 P45를 소지하고 있지 않을 경우에 대한 지침(Instruction)을 보여준다.
Para-98	피 고용인이 P45를 소지하지 않고 있으며 고용기간이 1주일 이상일 경우에 대한 지침(Instruction)을 보여준다.
Para-99	피 고용인이 P45를 소지하지 않고 있으며 고용기간이 1주일 이하일 경우에 대한 지침(Instruction)을 보여준다.

Section Para-95는 하나의 무조건 문장(Unconditional Paragraph)과 두개의 조건문장(Conditional Paragraph), 2개의 Conditional Section Reference, 그리고 5개의 매개변수 定義로 이루어져 있다.

1. 무조건 문장(Unconditional Paragraph)

ES/P에서는 중괄호 “{}”로써 조건문을 표시하는데 Section의 선언부분 바로 다음에 있는 문장은 조건문을 가지고 있지 않으므로 무조건 문장이 된다. 조건이 없으므로 ES/P가 Section Para-95를 호출하게 되면 Unconditional Paragraph 부분이 무조건 화면에 나타나게 된다. 일반적으로 무조건 문장은 그 Section이 어떤 기능을 담당하고 있는가를 서술하고 있다.

2. 조건문장(Conditional Paragraph)

{ } 안에 있는 조건문이 만족되는 순간 화면에 나타나는 내용을 담고 있다. <표-1>에 나와 있듯이 ES/P에서는 Knowledge 베이스가 Production Rules로 표시되므로 조건문({ }안의 내용)은 Production Rule의 IF부분에 해당하며 조건문장(Conditional Paragraph)은 THEN 부분에 해당한다고 볼 수 있다. 처음에 나오는 조건문장은 다음과 같은 Production Rule을 ES/P의 KRL로 나타낸 것이다.

```
IF    the employee cannot produce parts
      2 and 3 of form P45
AND  the employee was previously em-
      ployed
AND  the previous employment was in
      the current tax year
AND  the employee was not self-employed
AND  the employee has not lost the P45

THEN Ask the previous employer for the
      missing P45 parts and also follow
      the instructions given in paragraph
      (97) However, if the P45 is pro-
      duced at a later stage, then follow
      the instructions in paragraph(96).
```

조건문은 4개의 매개변수를 사용하고 있는데 매개변수의 定義에서 알 수 있듯이 3개의 類型이 Fact이며 나머지 하나는 Category이다. 따라서 매개변수 P45-given과 Got-P45가 거것이고 This-year가 참이며 History의 Option이 Prev-emp일 때 조건문 뒤에 나와 있는 내용이 화면에 나타나게 된다.

조건문이 만족되는가를 확인하기 위하여 ES/P는 각 매개변수의 定義에 나와 있는 Askable의 내용을 통하여 사용자에게 질문을 하게 된다. 이때 사용자는 Fact 유형 매개변수에 관한 질문에 대해서는 Y(yes) 또는 N(no)으로 답하고 Category 유형 매개변수에 대한 질문에 관해서는 해당 Option의 번호를 선택함으로써 질문에 대한 대답을 ES/P에게 알려주게 된다. 예를 들어 History(유형이 Category인 매개변수)를 물어보기 위해 ES/P는 화면에 다음과 같은 Option의 목록을 보여주는데 사용자가 “(2)”를 선택하면 History=Prev-emp의 조건이 만족되게 되는 것이다.

- (1) Self-employed
- (2) Previously employed full-time
- (3) Receiving full-time education
- (4) Claiming unemployment benefit
- (5) Any other occupation

3. Conditional Section Reference

조건문이 만족되면 Reference 명령문 다음에 오는 Section을 호출하여 諮問을 계속한다. 즉 P45-given이 참이면 Section Para-96으로 가서 자문을 계속하고 매개변수 P45-given이 거것이면 Section Para-97이 호출된다.

ES/P가 매개변수의 값을 알기 위해 질문을 할 경우 사용자가 “EXPLAIN”이라는 명령문을 키 보드를 통하여 입력시키면 매개변수의 定義 부분에 있는 Explanation 뒤의 문장이 화면에 나타난다. 이는 사용자가 질문에 대한 세부 사항을 알고 싶을 때 설명해 줄 수 있도록 Knowledge 엔지니어가 미리 만들어 둔 것이다 “EXPLAIN” 외에 “WHY”라는 명령문을 사용함으로써 ES/P 시스템 자체가 가지고 있는 질문에 대한 說明機能을 활용할 수도 있다.

이 외에도 ES/P는 사용자가 편리하고 쉽게 사용할 수 있도록 여러가지 기능을 제공하고 있는데 본 지면에서 이들을 다 다룰 수가 없으므로 보다 세부적인 내용을 원하는 讀者는 참고 문헌 안내에 나와 있는 ES/P manual을 참조하기 바란다.

VII. 結論

초기에 성공적으로 개발된 ES의 응용분야는 주로 의사의 진단, 화학 구조의 발견, 광택탐사, 자연어 처리 등으로 기업의 經營 문제와는 크게 연관이 없었다. 그러나 OPS-5를 이용하여 개발한 R1이 DEC의 VAX 판매를 위한 시스템 구성도를 제시하기 위해 이용되기 시작한 이후로 기업경영에의 성공 사례가 누적되면서 ES가 기업의 經營問題 해결에 사용될 수 있다는 가능성을 높여주게 되었다. 특히 최근에 들어서는 세무관리, 은행업무관리, 보험관리, 생산관리, 인사관리 등의 문제를 위한 ES가 폭넓게 연구 개발되고 있다.

초기의 ES 개발 때와는 달리 마이크로 컴퓨터의 기능과 보급이 확장되면서 상대적으로 적은 비용과 노력으로 ES를 개발할 수 있게 되었으므로 문제의 성격에 가장 적절한 마이크로 컴퓨터用 Shell을 잘 선택함으로써 개발시간과 개발비용을 동시에 단축시킬 수 있게 되었다는 것은 괄목할 만한 일이다.

경영과학 기법의 경우 실제문제를 해결할 수 있는 많은 計量 모델이 개발되었으나, 이를 사용해야 하는 상황과 사용방법에 관한 知識(Knowledge)의 부족으로 실용화 시키지 못한 경우가 非一非在하였다. 이제 이러한 모델들이 ES와 연결되어 사용된다면 모델에 대해 잘 모르는 사용자가 쉽게 모델을 사용할 수 있는 環境이 조성될 수 있으므로 경영과학의 현실문제에 대한 기여도를 크게 증가시킬 수 있으리라 믿는다. ORSA/TIMS의 최근 모임에 많은 Artificial Intelligence Section이 생겼다는 사실도 이를 대변해 주고 있다.

〈참고문헌 안내〉

1. F. Hayes-Roth, D. A. Waterman and D.B. Lenat, Building Expert Exstems, Addison-Wesley Publishing Co. 1983.
2. R. Forsyth, Expert Systems : Principles and Case Studies, Chapman and Hall Computing, 1984
3. S. W. Weiss and C. A. Kulikowski, Practical Guide to Designing Expert Systems, Rowman & Allanheld Publishers, 1984
4. P. H. Winston and B. K. P. Horn, LISP, Addison-Wesley, 1981
5. B. G. Buchman and E. H. Shortliffe, Rule-Based Expert Systems, Addison-Wesley Publishing Co. 1984
6. Artificial Intelligence Applications for Business : Proceeding of 1983 NYU Symposium, May 1983
7. P. Kinnucan, "Software Tools Speed Expert System Development", High Technology, PP16-20, March 1985
8. M. Turner, Expert Systems : A Management Guide, PA Computers and Telecommunications, 1985
9. M. Serafin, "AI in Finance", Wall Street Computer Review, Oct. 1984
10. W. Rauch-Hindin, "Banker and AI experts team up for greater revenue", Systems & Software, Oct. 1984
11. M. Goul, B. Shane and F. Tonge, "Designing the Expert Components of a Decision Support Systems", Working Paper, 1984
12. 이재규, J. Hurst and 이재식, "The Third Generation Multiple Criteria Decision Support Systems", Working Paper, 1984
13. 이재규, "전문가 체제와 경영 : 그 원리와 현황", 경영과 컴퓨터, PP52-56, 1985년

8 월

14. ES/P Manual

<부록> Section Para-95의 構造

/ *-----* /
 / * * /
 / * SECTION FOR PARAGRAPH(95) * /
 / * * /
 / *-----* /

Section Para-95 :

' This section deals with the detail of ' &
 ' paragraph(95). This paragraph ' &
 ' establishes the basic details about the ' &
 ' new employee and obtains his P45 form ' &
 ' (or establishes the reasons why no ' &
 ' P45 is provided). The section then ' &
 ' references the rest of the knowledge ' &
 ' base depending upon whether a P45 was ' &
 ' provided or the reasons why one was ' &
 ' not given.' &
 ' Please ask' .. @employee..' to give you ' &
 ' parts 2 and 3 of form P45. (Note: this ' &
 ' is the old "Employers Certificate"). ' .'
 {Not P45-given and History=Prev-emp
 and This-year and not Got-P45}
 ' The tax office will require the form ' &
 ' for Inland Revenue purposes. ' &
 ' Therefore contact the previous ' &
 ' employer and ask for the form P45 to ' &
 ' be supplied as soon as possible. It may ' &
 ' also be a good idea to remind the ' &
 ' previous employer of his/her Duties ' &
 ' when an employee leaves.' .'
 {Not P45-given and History=On-dole and
 not Benefit-P45}
 ' Ask' .. @employee..' to notify his/her ' &
 ' Benefit office, either by going in person &

' or by completing and sending form ' &
 ' UB40, that he/she will no longer be ' &
 ' claiming unemployment benefit. This ' &
 ' should enable' .. @employee..' to obtain ' &
 ' parts 2 and 3 of form P45, which ' &
 ' he/she should then hand to you.' .'
 {P45-given}

Reference para-96.

{Not P45-given}

Reference para-97.

P45-given : ' Parts 2 and 3 of P45 are ' &
 ' in hand'

Fact
 Explanation

' The employee must supply you with ' &
 ' both parts 2 and 3 of form P45 ' &
 ' before you can proceed any further. ' &
 ' This official form is required by the ' &
 ' Tax office in keeping its records on ' &
 ' the tax paid by' .. @employee..' to date.' .'
 Askable

' Have you been able to obtain both ' &
 ' parts 2 and 3 of the formP45 from ' &
 ' ..@ employee..' ? ' .'
 Got-P45 : the P45 has been passed on ' &
 ' by the old employer'

Fact
 Explanation

' The previous employer should have ' &
 ' given' ..@employee..' a form P45 before ' &
 ' he/she left the previous place of ' &
 ' employment. This form should then be ' &
 ' available to you.' .'
 Askable

' Did the previous employer give ' &
 ' ..@employee..' a form P45 before or ' &
 ' on termination of the previous ' &
 ' employment? ' .'
 History : ' the previous situation of ' &
 ' ..@employee..' .'
 Category

Explanation

' This question is trying to establish' &
 ' what '..@employee..' was doing' &
 ' immediately prior to arriving for' &
 ' enrolement at your Place of work.' &
 ' This could include previous' &
 ' employment, education or' &
 ' unemployment.' .

Options

Self-emp-' Self-employed'

Prev-emp-' Previously employed full-time',

Education-' Receiving full-time education
 at school or college',

On-dole-' Claiming unemployment benefit',

Other-' Any other occupation'

Askable

' What situation was '..@employee..' in' &
 ' immediately prior to being taken on for' &
 ' employment?' .

This-year : ' the previous job ended this' &
 ' tax year'

Fact

Explanation

' This question wishes to establish' &
 ' whether the previous employment of' &
 ' ..@employee..' ended in a previous' &
 ' tax year, i.e. ended before the 1st' &
 ' April last.'

Askable

' Did the previous employment end' &
 ' during the current tax year?' .

Benefit-P45 : ' the P45 has arrived from' &
 ' the DHSS'

Fact

Explanation

' After termination of a claim for' &
 ' unemployment benefit, the Benefit' &
 ' Office should send the form P45 to' &
 ' the claimant so that he/she can present' &
 ' it to the new employer'

Askable

' Has '..@employee..' received the P45' &
 ' from the Benefit Office?' .