WORKING PAPER

KNOWLEDGE-BASED SYSTEMS

Overview and Selected Examples

E. Weigkricht
L. Winkelbauer

December 1987
WP-87-101

IIASA
International Institute
for Applied Systems Analysis

KNOWLEDGE-BASED SYSTEMS

Overview and Selected Examples

E. Weigkricht
L. Winkelbauer

December 1987
WP-87-101

INTERNATIONAL INSTITUTE FOR APPLIED SYSTEMS ANALYSIS
A-2361 Laxenburg, Austria

Preface

The Advanced Computer Applications (ACA) project builds on IIASA's traditional strength in the methodological foundations of operations research and applied systems analysis, and its rich experience in numerous application areas including the environment, technology and risk. The ACA group draws on this infrastructure and combines it with elements of AI and advanced information and computer technology to create expert systems that have practical applications.

By emphasizing a directly understandable problem representation, based on symbolic simulation and dynamic color graphics, and the user interface as a key element of interactive decision support systems, models of complex processes are made understandable and available to non-technical users.

Several completely externally-funded research and development projects in the field of model-based decision support and applied Artificial Intelligence (AI) are currently under way, e.g., *Expert Systems for Integrated Development: A Case Study of Shanxi Province, The People's Republic of China.*

This paper gives an overview of some of the expert systems that have been considered, compared or assessed during the course of our research, and a brief introduction to some of our related in-house research topics.

Kurt Fedra
Project Leader
Advanced Computer Applications

Acknowledgments

We would like to thank Sergei Golovanov for his assistance in obtaining bibliographic material from the Eastern European countries, and our Chinese collaborators for providing us with information on research in this field in the People's Republic of China.

Contents

KNOWLEDGE-BASED SYSTEMS

Overview and Selected Examples

E. Weigkricht and L. Winkelbauer

1. A General Introduction to Artificial Intelligence

Computers are dumb. They consist of nothing more than metal, glass, silicon and plastic and most of the software that is supplied only suffices to make them perform standardized tasks that are barely usable by non-specialists in the computer field.

The basic aim of the research branch of computer science called *Artificial Intelligence (AI)* is to improve the predicament of the human computer user by making the software smarter so that computers become easier to communicate with and more powerful in assisting users in various task areas. In other words:

> *Artificial Intelligence (AI) is the part of computer science concerned with designing intelligent computer systems, that is, systems that exhibit the characteristics we associate with intelligence in human behaviour – understanding language, learning, reasoning, problem solving and so on.* (Barr and Feigenbaum, 1982).

1.1. A Brief History

1.1.1. The Decade of the Prototypes: 1955 to 1965

John McCarthy of Stanford University coined the term **Artificial Intelligence (AI)** in 1956, when he was an assistant professor of mathematics at Dartmouth College. McCarthy, together with Marvin Minsky of MIT and others, organized a conference at Dartmouth in that year, bringing together the handful of people working on the task of building machines to emulate human brain functions.

In the beginning there was the belief that a single general-purpose problem solver with sufficiently powerful inference capabilities could solve all AI problems. The **General Problem Solver** (GPS) was developed by Newell, Shawn and Simon beginning in 1957 and continuing until 1969. The important contribution of this work was the idea of *means-end analysis* which searched the state-space in a, at the time, new manner. GPS turned out to be the wrong paradigm and the next decade concentrated on *less inference and more knowledge, because the power lies in the knowledge.* (Feigenbaum and Feldman, 1963)

On the mathematical side Slagle (1961) implemented a **Symbolic Automatic Integrator** in 1961 at MIT, which performed at *college freshman level* and was a progenitor of MACSYMA.

To facilitate the approach to **symbol processing** a new programming language was developed by McCarthy: **LISP (LISt Processing language).**

The first **natural language understanding systems** were built on the assumption that the *syntax* of a sentence sufficed to give its meaning. The first simple information retrieval parser (BASEBALL) was written at Lincoln Laboratories in 1963 (Green et al., 1963). Weizenbaum's famous **ELIZA** program (Weizenbaum, 1965) simulated a psychiatrist's dialog based on pattern-matching techniques. But all natural language understanding systems of the first decade were insufficient and it turned out that the basic assumption was wrong.

Research on **speech recognition** in the 1960s concentrated on the recognition of isolated words spoken by a known speaker.

1.1.2. The Decade of "Ivory-tower" Research: 1965 to 1975

During this period the research work was subdivided into three major fields. These were first, **computer vision** (Winston, 1975), secondly, **understanding natural language** (Winograd, 1972) and thirdly, the deeper (more fundamental) issues of **representation and understanding** (Bobrow and Collins, 1975).

The most well-known vision research of this time is that of Horn (Horn in Winston, 1975; pp.115-156) at MIT who did research on *shape from shading* in the mini-world of children's toy blocks.

This *blocks world* was also used in Winograd's program **SHRDLU** which could answer questions about the positions of the blocks on the tabletop and about some of SHRDLU's internal states.

Speech-input research was funded by DARPA in the early seventies, where, by the end of the decade:

- **HARPY** (whose pronunciation representation was an integrated network);
- **HEARSAY I & II** (pronunciation graphs and blackboard architecture);
- **HWIM** (segmented lattices);
- **DRAGON** (developed at Carnegie-Mellon using the experiences gained with the HARPY and HEARSAY projects; for commercial use)

had limited success rates (up to 94% at the word level, 17% - 60% at the sentence level).

New higher level AI languages were developed at the beginning of the seventies of which the best known and most commonly used one is **PROLOG** developed by A. Colmerauer and P. Roussel at the University of Marseille in 1972.

Symbolic Mathematics peaked in this decade with MIT's **MACSYMA** package (1974), which contains knowledge about symbolic algebra (for example about complex integral transformations) and is, by using an algebraic simplifier together with pattern matching techniques, able to solve algebraic equation problems input by the user, and so on.

1.1.3. The Decade of Real-world Applications: 1975 to the Present

In the last decade of AI research emphasis was placed on real-world applications of practical use.

Forerunners here were Shortliffe's **MYCIN** (Shortliffe, 1976) for diagnosis of infectious bacterial diseases such as meningitis, Hendrix's **LADDER** system for providing natural language access to databases (Hendrix et al., 1978), SRI's **PROSPECTOR** (Duda et al., 1978) which helps geologists in the early stages of investigating a site for ore-grade deposits, and the **DENDRAL** program (Buchanan and Feigenbaum, 1978; Lindsay et al., 1980; Feigenbaum and Lederberg, 1981) for handling mass-spectrometry analyses of chemical compounds.

Building on the experience gained during the development of these and other programs, computer companies began to develop in-house AI applications, such as DEC's **XCON (R1)** (McDermott, 1981, 1984) which helps configure computer systems from incomplete order information. Other applications were in computer hardware diagnosis (the

DART program for IBM; Bennet and Hollander, 1981) and oil-well analyses (the **DIPMETER ADVISOR** at Schlumberger; Davis et al., 1981).

In the last few years of this third decade a new type of commercial software has appeared on the market. These AI-based tools range from English language front-end for databases to expert-systems building tools.

Since the beginning of the eighties, it is not just software improvements that emerge from AI research. Special AI hardware is now available from LMI, Symbolics, Texas Instruments, Xerox, etc., and the Japanese have launched their well-known *Fifth Generation Project* which aims at the development of a new generation of computer hardware specially designed for *logic programming*, i.e., it will not be based on the traditional – and now ubiquitous – von Neumann hardware architecture.

AI is currently experiencing a boom such as never before, as both politicians and the public are beginning to believe that machines can be made to ***think***. This naive enthusiasm is dangerous; as Weizenbaum (1976) puts it ***"since we do not now have any ways of making computers wise, we ought not now to give computers tasks that demand wisdom"***; it will be a complicated task to reduce the expectations of the public to a feasible level, without reducing the belief in AI research.

1.2. Methods and Fields of Application

We do not intend in this paper to describe the methods and fields of applications in detail, because we focus our interest on the application field called *Knowledge-based Systems* (often also called *Expert Systems*) which is described in detail in the following section.

However, an overview including some of the terms dedicated to particular methods has been included so as to provide a starting point for gaining further information from the literature.

The **general methods** employed in AI are:

AI-Languages and AI-Systems: LISP, PROLOG, KRL, FRL, Smalltalk, ObjTalk, Flavors, etc.;

Knowledge Representation: Predicate calculus, Rules, Semantic Nets, Frames, etc.;

Heuristic Search: Breadth first search, Depth first search, Best first search, etc.;

Inferential Systems: Deduce inferences out of given or deduced knowledge;

Learning & Knowledge Acquisition: gain knowledge and generalize, i.e., gain knowledge about knowledge *(Meta-knowledge)*.

The **general application fields** in AI are:

Knowledge-based Systems (Expert Systems)

Natural Language Understanding Systems

Computer Vision

Robotics

Gaming Programs

Learning Systems.

2. Expert Systems

One of the major advances in computer science emerging from AI is the development of computer systems which separate the underlying knowledge from the procedural part of the system, thus allowing knowledge (which constitutes the abilities of the whole system) to be stored and edited (i.e., changed) without changing a single line of program code. These systems are called *Knowledge Based Systems (KBS)*.

Using this basic concept of knowledge/program separation, systems have been developed which incorporate the expertise of specialists as knowledge in order to make this expertise available to a broader group of people. These KBS are therefore called Expert Systems (ES). Although there are KBS that do not attempt to act as experts in the relevant problem domain they are applied to, the focus in this paper is on those systems that do claim to be experts, and take this to be the connotation of the term *expert systems (ES)*; but it should be borne in mind that conceptually all ES are KBS.

2.1. How Computer Systems become Experts

In addition to the usual problems that arise during software development (e.g., selection of the hardware, choice of the programming language and the programming tools) there is the need to incorporate expert knowledge about the problem domain into a software system to make it an expert system. The problem with this requirement is that the *computer experts* have to communicate with the *domain experts*. This communication problem often turns out to be the hardest problem during software development, especially when good software has to be written to perform non-trivial practical tasks.

In these cases the software developers should not only be computer experts, but also have enough knowledge about the problem domain so that they are able to incorporate the knowledge of the domain experts in their software. These people are called **knowledge engineers**.

But these "multi-experts" are not easy to find and are usually unaffordably expensive. Therefore another way had to be found, and this way led to the basic concept of KBS. The main rule behind this concept is as follows: *separate the knowledge in the software system from the program (i.e., the procedural part)*.

Then the domain experts are able to input their knowledge (in a special problem domain-oriented language) into the **knowledge base** of the software system in the form of *facts*[1)] and *heuristics*[2)] and the computer experts are then able to concentrate on their speciality, i.e., to write a (more or less) general problem-solving program (the **inference engine**) which deduces inferences based on the problem-oriented knowledge input by the domain experts.

2.2. Some Characteristics of Expert Systems

An **Expert System** has been defined as *"... an intelligent computer program that uses knowledge and inference procedures to solve problems that are difficult enough to require significant human expertise for their solution. Knowledge necessary to perform at such a level, plus the inference procedures used, can be thought of as a model of the expertise of the best practitioners of the field."* (Feigenbaum in Harmon and King, 1985).

As expert systems are to function like human experts, they should be able to do things that human experts commonly do. For example, experts consult with others to help solve problems. Thus, most expert systems ask questions, explain their reasoning if asked, and justify their conclusions. Moreover, they typically do this in a language that the user can easily understand. They allow the user to skip questions, and most can

1) knowledge generally agreed upon by experts in the problem field
2) mostly private, little-discussed rules of good judgement

function even when the user provides incomplete or uncertain data. However they are only able to assist human experts, and will never really be able to replace them.

What most obviously differentiates expert systems from standard database systems is that they are based on heuristics and are able to explain results and how they were deduced. In the following sections, further details about the special aspects of expert systems are discussed.

2.3. Requirements for Expert Systems

Although there are definitions like the one above there is no officially accepted and approved definition of what an expert system is and of which parts it has to consist of. Therefore, in what follows we will only list some requirements which, in our opinion, should be fulfilled if a software system is to be called an expert system (see Figure 1):

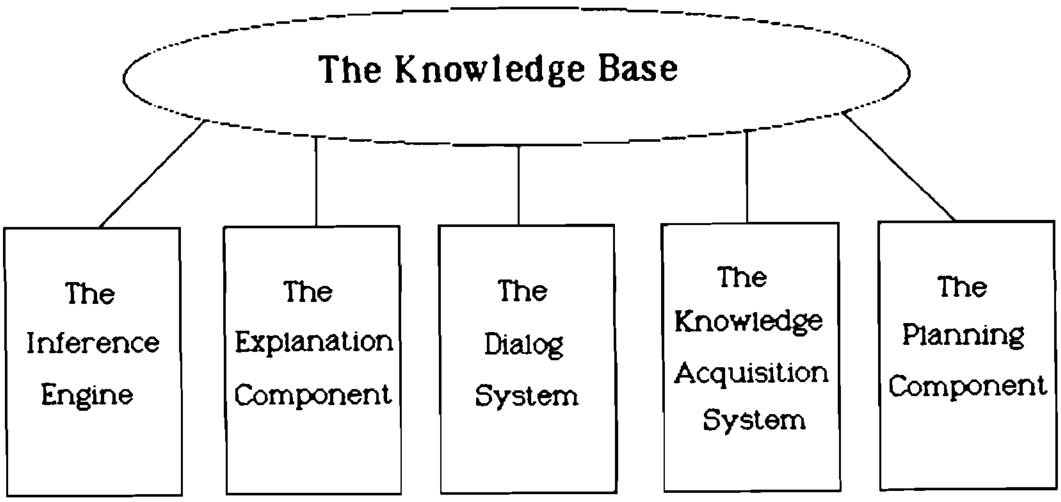

Figure 1: Components of an expert system

- An expert system should contain a **knowledge base** which represents human expertise about a specific problem area in the form of facts and heuristics.
- It should be able to deduce inferences based on the knowledge base with a program (the **inference engine**) which is totally separated from the knowledge base.
- The inference engine should be based on a problem-solving algorithm which is totally transparent to human experts. To achieve this transparency every expert system should incorporate an **explanation component** which is able to explain the final and interim results and to show the path of inferences which led to the results.
- It should be possible for the domain experts to build, modify and extend the knowledge base directly (i.e., without the need for a computer expert) in a **problem domain-oriented language**. The module which performs this task is called a **knowledge acquisition system.**

- The knowledge acquisition system should also be able to learn, i.e., modify its own knowledge base with experiences from previous runs.
- An expert system should incorporate an **intelligent dialog system**, which can be based on hardware features such as a ***touch sensitive screen*** or the now well-known *mouse* and use natural language, graphics and/or menu techniques to provide a convenient dialog-interface for the user. In any case there should be enough ***help*** facilities and explanation options in every stage of the dialog so that the user is always *in charge of* the system.
- To be able to fulfill the aim of acting like an expert an expert system should have the ability to ***plan*** strategies and to decide how they should be worked out. So far there do not appear to be any expert systems known to us that can provide such an ability.
- If the underlying knowledge represented in the knowledge base is inexact, i.e., based on likelihood and uncertainties, an expert system should be able to base its inferences on probability calculations and formulate its output as statements which express the system's uncertainty about the results.
- For educational and test reasons an expert system should allow the user to experiment with *what-if scenarios*, i.e., allow him to modify facts and heuristics in the knowledge base (usually a local copy of it) to show what effects these modifications produce and to improve the understanding of how the expert system works.

We do not claim that the above list of requirements is complete and concede that it is definitely subject to modification. Nevertheless, we think it encompasses at least the general guidelines which may be helpful in separating systems which are only *advertised* as being expert systems from *real* expert systems.

2.4. The Benefits of Today's Expert Systems

Expert systems do not display biased judgements (i.e., if the knowledge base is not biased), nor do they jump to conclusions and then seek to maintain those conclusions in the face of disconforming evidence. They are able to encompass the knowledge and experience of more than one expert. They do not have *bad days*. They always attend to details, and they always consider all the possible alternatives systematically.

The best (and only the best) of today's expert systems – equipped with thousands of heuristic rules and running on specialized hardware – are able to perform their specialized tasks better (i.e., at least in terms of speed and consistency) than a human specialist.

2.5. Current Limitations

All of the above is not to suggest that most of the existing expert systems are as good as human experts. The technology is new and just beginning to be applied to tough commercial problems.

Today's expert systems are confined to well-circumscribed tasks. They are not able to reason broadly over a field of expertise. They cannot reason from axioms or general theories. Most of them do not learn and, thus, they are limited to using the specific facts that they were *taught* by a human expert. They lack common sense, they cannot reason by analogy, and their performance deteriorates rapidly when problems extend beyond the narrow task they were designed to perform.

In short, no expert system will quite be able to *replace* a human specialist.

3. National Research in Artificial Intelligence

This section gives an overview of the general national activities of some countries in the field of Artificial Intelligence. The search was limited to literature in English, German and French; therefore this paper does not claim to include all the latest developments or to be complete.

3.1. The People's Republic of China

Several major research projects have been identified: e.g., ILEX, *an Intelligent Relational Database System* is undertaken by the Ministry of the Electronic Industry and was presented at the ACM (Association for Computing Machinery) Conference on Personal and Small Computers in December 1983. Two Expert Systems projects have been started recently, one at the Institute of Computing and Technology of Academia Sinica on *expert shells*, the other at the Shanghai Industry University. AI projects are under way at the Automation Institute of Shinyang of Academia Sinica, the Jilin University (see Wang Xiang-Hao, 1984), and at the Wuhan Air Force Radar Insitute (see Lu Hanwong, 1987).

Other important recent research is on:

- *Chinese medical diagnosis* at the AI Centre of the Beijing Institute of Technology and the Shanxi Institute of Automation, see also Wang Hongbing (1986), Qinghua University, and Liu (1986), Shanxi Institute of Automation;
- *Machine learning* at the Nanjing Institute of Technology;
- *Rule-based consultation systems* (Wang Shenkang, 1984);
- *Military command policy* (Wang Yuke, 1986) at the Beijing Insitute of Technology;
- *Knowledge Representation* by Guan Jiwen (1987), Zhang Chengai (1987), and Zhang Aidong (1987) at Jilin University, and by Yang Zhibao (1986) at the Wuhan University;
- *Machine Design* by Wang Qun (1987) at the Hua Zhong Institute of Technology;
- *Induction* by Muang Keming (1987) at the Nanjing Institute of Technology.

3.2. Eastern Europe

Hungary

The main work on Artificial Intelligence in Hungary is the development of different versions of the language PROLOG and PROLOG applications in many different fields (Szreredi and Santane-Toth, 1982):

- *Pharmaceutical research* (for drug-interaction see Darvas et al., 1976, 1978a, 1978c, 1978d, 1979b, 1980, Futo et al., 1978; for enzyme sequences see Matrai, 1979).
- *Information retrieval system* (for chemical information systems see Darvas et al., 1978b, 1979a; for query systems see Ban et al., 1979).
- *Computer-aided design* (for urban planning see Markusz, 1977a, 1977b, 1980a, 1980b, 1981, Markusz and Kaposi, 1982; for mechanical engineering see Molnar et al., 1981).
- *Software applications* mainly generation of COBOL programs (see Lang, 1978) and supporting computer systems.

USSR

There is a great deal of research at present in many areas. Some of the centers of activity are listed below:

- Computer Center of the Siberian Branch of the Academy of Sciences
- Leningrad Polytechnic Institute
- Institute for Applied Systems Analysis
- Institute for Problems of Information Transmission
- Institute for Physical Engineering
- Computer Center of the Academy of Sciences.

The fields of activity are mainly:

- *Plant surveillance and control;*
- *Robotics;*
- *Natural language processing and dialog systems;*
- *Pattern recognition;*
- *Automatic theorem proving and inferencing.*

Poland

Much attention is being paid to the *pattern recognition problem,* to *human voice synthesis* (in the Institute of Basic Technical Problems of the Polish Academy of Sciences in Warsaw), and *to medical diagnosis* (in the Medical Academy in Silesia, the Pedagogical Institute in Warsaw, the Institute of Applied Cybernetics of the Polish Academy of Sciences in Warsaw).

Czechoslovakia

The main application of AI is in *industry*. Research has been carried out mainly at the Technical University of Brno and at the Czechoslovak Academy of Sciences. Research in the area of expert systems, i.e., for diagnoses in various domains is carried out mainly at the Research Institute of Medical Bionics in Bratislava. Expert systems based on fuzzy methodology to deal with subjective, ill-defined and uncertain knowledge are developed at the Technical University of Brno in collaboration with the Helsinki University of Technology.

3.3. The European Community

Eighteen European countries, including the European Economic Community, contribute to the *EUREKA* program. In 1986, sixty-two projects were approved, for industrial collaboration in high technology undertakings.

The *ESPRIT* project was initiated by the European Economic Community to attempt to keep pace with other industrial powers, Japan and the United States in particular. Its primary objectives are to promote European industrial cooperation, to provide European industry with the basic technologies for the early 90s, and to work on agreements on international standards (see Esprit, 1986). Within this project, diverse areas are treated: there are several ongoing projects on expert systems (some of the developments are mentioned in section 5); there are also more general AI developments, such as *expert systems on an industrial scale* (Manucci et al., 1985), or further developments and applications of the language PROLOG (Esprit, 1986).

The *COST* (Co-operation in the field of Scientific and Technical Research) framework involves applied scientific research in various fields of common interest to the participants of COST, all the European OECD Member States.

France

France is deeply involved in the ESPRIT project.
Other fields are:

- *AI Languages;*
- *Image processing;*
- *Speech*: by GALF (Groupement des Acousticiens de Langue Française) and AFCET (Association Française pour la Cybernetique, Economique et Technique; interested in speech research).
- *Natural language understanding;*
- *Robotics;*
- *Pattern recognition*: especially by the French post office and telecommunications center and the Centre National d'Etudes des Telecommunication.

United Kingdom

The main British activities in Artificial Intelligence are concentrated in software houses, some of them strongly linked with American companies. One of the leaders in AI, Donald Mitchie, runs his own company, the Intelligent Terminals Ltd., and offers a large set of tools for ***knowledge engineering***. The other software houses are mainly building *Shells* and complete expert systems for a specific purpose or customer, or tools that are commercialized.

Research is also going on in the **PROLOG** language and in combinations of PROLOG with other languages: Salford University developed a package for PROLOG including LISP functions and LISP including PROLOG predicates (see Industry News, Jan. 1985); Sussex University developed POPLOG, a combination of PROLOG, LISP and POP-11.
Other main fields are:

- *AI languages;*
- *Image processing;*
- *Robotics;*
- *Automatic deduction and expert systems.*

FRG

Research groups at the GMD (Gesellschaft für Mathematik und Datenverarbeitung) are working on expert systems especially for practical applications in divergent fields (the automobile industry, for example). Other fields in the FRG:

- *Shells and tools* for the automatic construction of expert systems;
- *Medical image processing;*
- *Industrial applications and robotics;*
- *Aerial images.*

3.4. Japan

Research in Artificial Intelligence in Japan goes back for at least twenty years. The Japanese fifth-generation computers are expected to be the basis of knowledge-based information systems in terms of hardware and software. **PROLOG** is the main language adopted. A national research project concerning fifth-generation computers, sponsored by the Ministry of International Trade and Industry, began in 1982, involving all the major Japanese computer companies. The main fields of activity are (Ishizuka, 1984):

- *Medical diagnosis and consultation* at the University of Tokyo Hospital, the Science University of Tokyo and Tokyo Electric University.
- *Plant surveillance and control*, mainly undertaken by companies like Hitachi, Mitsubishi, Toshiba, which are not only large computer companies, but also manufacturing industries: expert systems are designed to work in their plants.
- *Industrial assessment* at the University of Tokyo.
- *Management and office systems* at the System Development Laboratory, N.T.T. (the public telephone company), ICOT (the Institute of New Generation Computer Technology).
- *CAD for VLSI* undertaken by NEC.
- *Image processing* at the Electrotechnical Laboratory.
- *Database access* at Osaka University.

3.5. The United States of America

In the United States, Artificial Intelligence research is being conducted in parallel in universities as well as in software and/or computer companies; in many cases companies and universities collaborate with great success.

The two main centers of AI research are Stanford University and the Massachusetts Institute of Technology (MIT): the first expert systems were built there (the DENDRAL project at Stanford University, considered the pioneering application of AI, by Feigenbaum–an acknowledged world leader in the application of AI–Lederberg, Buchanan and Lindsay; the MACSYMA project at MIT was originally designed by Engleman, Martin and Moses). Both (and several other American universities, e.g., Harvard University and the Information Sciences Institute (ISC) in automatic programming or the Carnegie-Mellon University in speech and human cognition) have numerous projects and departments working in different fields of AI, such as:

- *Image understanding;*
- *Logic and mathematics;*
- *Medicine and chemistry;*
- *Analysis;*
- *Heuristics and VLSI design;*
- *Automatic programming;*
- *Knowledge acquisition;*

as well as

- *Shells and practical applications..*

Their work is usually closely linked to companies. American companies are highly market-oriented and active in all kinds of software development, as well as in hardware production supporting AI (LISP machines, etc.).

4. AI Languages and Tools

To build an expert system that is to assist an expert in a particular domain a software package has to be created. No matter what software is used, ultimately the expert system will depend on some computer hardware – the machine on which the software is run. The most primitive level of computer software is termed *machine language,* i.e., the binary coded fundamental commands that flow into the central processing unit and direct the computer to make discrete physical responses.

At a slightly higher level, there is a software program that will direct by fundamental commands the basic operations of the computer. The *operating system* handles utility functions and can be written in or compiled into machine language.

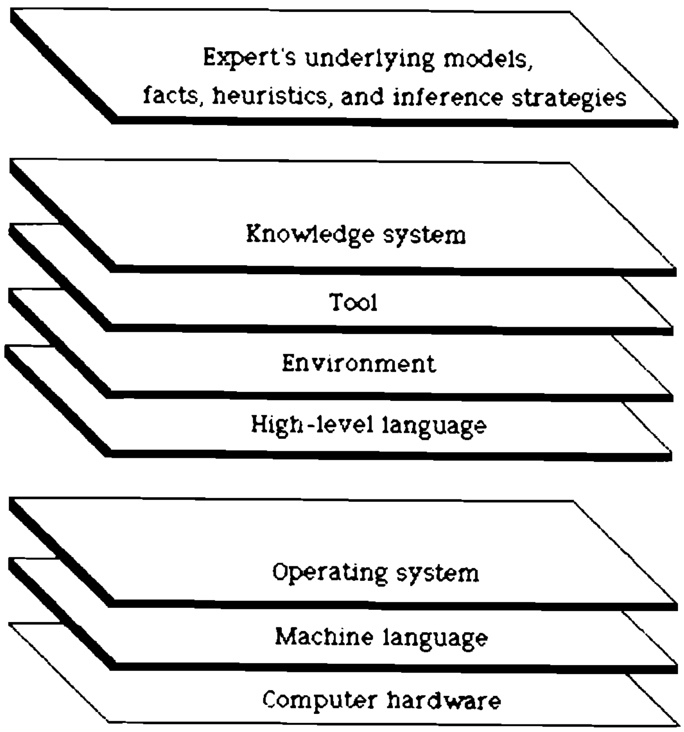

Figure 2: The six levels of software between human problems and computer hardware (after Harmon and King, 1985)

Most programming is done in one of a number of *high-level languages.* Well-known high-level languages include **BASIC, FORTRAN, COBOL, PL/1, PASCAL** and **C.** AI programmers commonly use high-level languages such as **LISP** and **PROLOG. PROLOG** contains constructs that make it easy to manipulate logical expressions, whereas **LISP** consists of operators that facilitate the creation of programs that manipulate lists. These constructs are useful for developing *symbolic computing programs,* just as iterative constructs like the *WHILE* loops of **PASCAL** are useful for *numeric programming.*

Just above the high-level languages are special packages of prewritten code that are typically called **programming environments.** An environment is usually closely associated with a particular high-level language and contains chunks of code written in that language that are useful for particular programming tasks. By analogy to conventional programming languages such as FORTRAN, environments are *libraries of subroutines* that can be chained together to develop specific applications.

Knowledge engineering tools are designed to facilitate the rapid development of knowledge systems. They represent the next level of software between human problems and computer hardware (see Figure 2). Knowledge engineering tools contain elementary constructs for modeling the world that determine the sorts of problems the tool can easily handle.

A tool has fewer applications than the language or environment in which it was written, but it is usually designed to facilitate the rapid development of expert systems that address a specific class of problems.

In this paper we do not consider AI languages and tools in detail. It is worthwhile, however, briefly describing the two major languages that dominate current work in AI and knowledge engineering and the general characteristics of current knowledge-engineering tools.

4.1. The LISP Family

Until very recently one could have said that **LISP** was the only AI language used by knowledge engineers. The language was created by **John McCarthy** in 1958. Of the languages still in use, only FORTRAN is older than LISP.

McCarthy describes LISP as follows: (McCarthy in Barr and Feigenbaum, 1982b)

1. *Computing with symbolic expressions rather than numbers; that is, bit patterns in a computer's memory and registers can stand for arbitrary symbols, not just those of arithmetic.*
2. *List processing, that is, representing data as linked-list structures in the machine and as multilevel lists on paper.*
3. *Control structure based on the computation of functions to form more complex functions.*
4. *Recursion as a way of describing processes and problems.*
5. *Representation of LISP programs internally as linked lists and externally as multilevel lists, that is, in the same form as all data are represented.*
6. *The function EVAL, written in LISP itself, serves as an interpreter for LISP and as a formal definition of the language.*

There is no essential difference between data and programs, hence LISP programs can use other LISP programs as data. LISP is highly recursive, and data and programs are represented as nested lists. It does not always make for easy-to-read syntax, but it allows for very elegant solutions to complex problems that are very difficult to solve in the various conventional programming languages.

There are only a few basic LISP functions; all other LISP functions are defined in terms of these basic functions. This means that one can easily create new higher-level functions. Hence, one can create a LISP operating system and then work up to whatever higher level one wishes to go to. Because of this great flexibility, LISP has never been standardized in the way that languages such as FORTRAN and BASIC have. Instead, a core of basic functions has been used to create a wide variety of LISP dialects (see Figure 3).

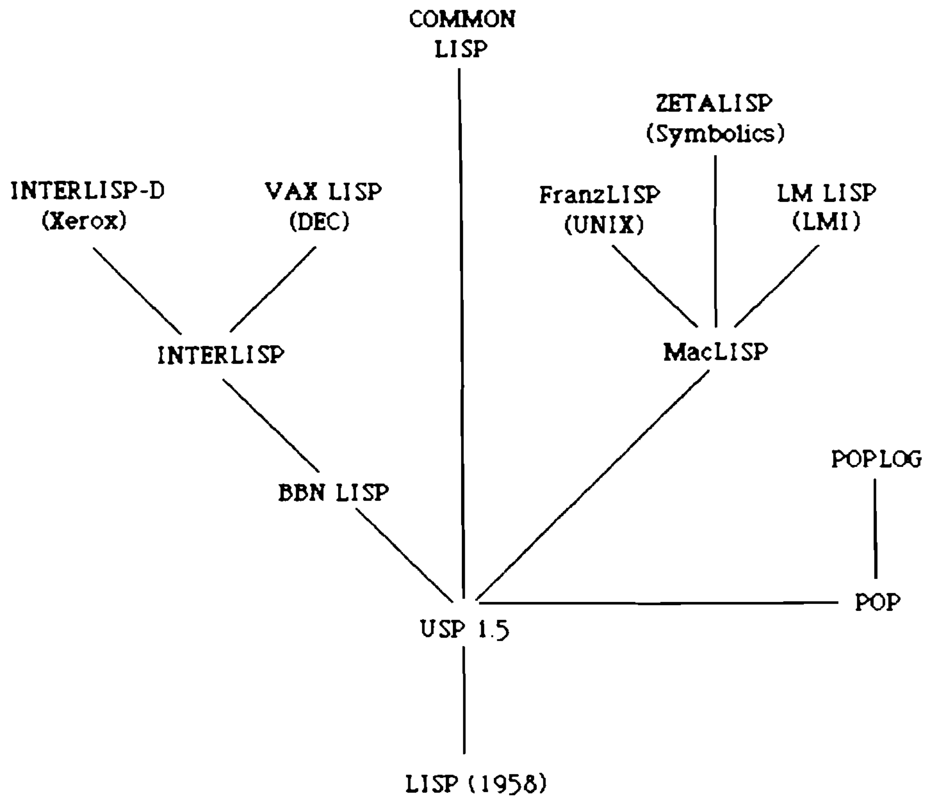

Figure 3: The LISP family

Among programming languages LISP is unique in that it stores its programs as structured data. The basic data structures in LISP are the *atom,* any data object cannot be further broken down, and the *CONS node.*

Each *atom* has an associated property list that contains information about the atom, including its name, its value, and any other properties the programmer may desire.

A *CONS node* is a data structure that consists of two fields, each of which contains a pointer to another LISP data object. CONS nodes can be linked together to form data structures of any desired size or complexity. To change or extend a data structure in a LISP list, for example, one need only to change a pointer at a CONS node (see Figure 4).

Elements of lists need not be adjacent in memory - it is all done with pointers. This not only means that LISP is very modular, it also means that it manages storage space very efficiently and frees the programmer to create complex and flexible programs.

Conventional programming languages normally consist of sequential statements and associated subroutines. LISP consists of a group of modules, each of which specializes in performing a particular task. This makes it easy for programmers to subdivide their efforts into numerous modules, each of which can be handled independently.

Suffice it to say that LISP is a very powerful language that is popular with programmers who routinely construct very large and complex expert systems.

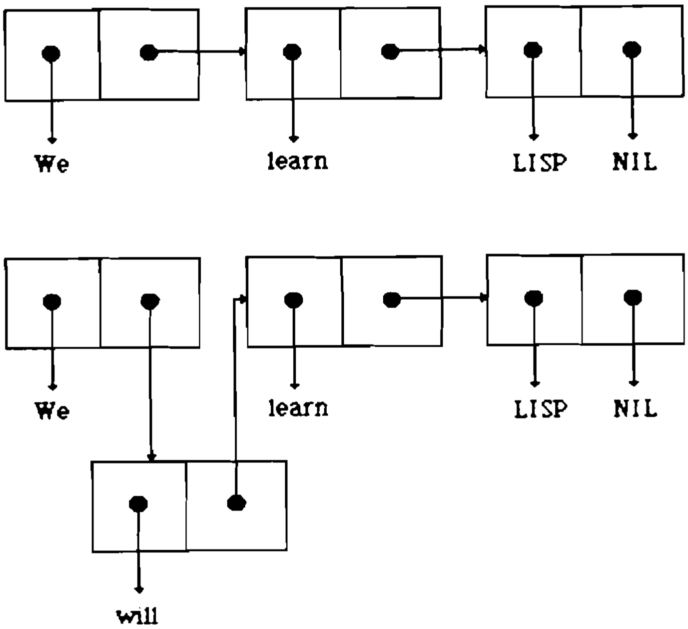

Figure 4: Modifying a LISP data structure

4.2. PROLOG

PROLOG, which is an acronym for *PROgramming language of LOGic,* was initially developed in 1972 by **A. Colmerauer** and **P. Roussel** at the University of Marseilles. PROLOG is a programming language that implements a simplified version of predicate calculus and is thus a *true logical language.* PROLOG has enjoyed great international popularity. The first efficient PROLOG compiler was developed at the University of Edinburgh. The Hungarian government has encouraged extensive industrial use of the language, and the Japanese fifth-generation project has adopted PROLOG as the fundamental language for the supercomputers they plan to build.

As with LISP, PROLOG is designed for symbolic rather than simply for numerical computation. PROLOG is very efficient at list processing. Similarly, PROLOG is an interpreted language and thus responds to any query by attempting to return an answer immediately.

To program in PROLOG, one does the following:

1. Specify some facts about objects and relationships.
2. Specify rules about objects and relationships.
3. Ask questions about objects and relationships.

Thus, if one entered the following fact:

works(john,mary).

i.e., "John works for Mary" and then asked:

?-works(john,mary).

i.e., "Does John work for Mary", PROLOG would respond by printing:

yes

The question "Whom does John work for" looks like this:

?-works(john,X).

To which PROLOG would reply:

X=mary

Rules in PROLOG are declared using the notation ":-", which can be read "if". The rule "X is the manager of Y if Y works for X" would look like this:

manager(X,Y):-works(Y,X).

After one has entered the above rule the question "Who is John's manager" could be posed to PROLOG as follows:

?-manager(X,john).

to which PROLOG would respond:

X=mary

In a sense, computation in PROLOG is simply controlled logical deduction. One simply states what one knows (i.e., *the facts)* and PROLOG responds with whether or not any specific conclusion can be deduced from those facts. In knowledge engineering terms, PROLOG's control structure is logical inference.

PROLOG is the best current implementation of logic programming, although it cannot begin to handle all the deductions that are theoretically possible in predicate calculus. At the same time, PROLOG's syntax is much less complex than most conventional programming languages of comparable power.

A programming language cannot be strictly logical, however, since input and output operations necessarily entail some extra-logical procedures. Thus, PROLOG incorporates some basic code that controls the procedural aspects of its operation. The procedural aspects of PROLOG are kept at a minimum and it is possible to conceptualize PROLOG strictly as a logical system.

PROLOG is not going to replace LISP or vice versa, although they have a similar application domain, but each language should be used to develop the parts of a system for which it is best suited.

4.3. Knowledge Engineering Tools

Knowledge engineering tools are designed to facilitate the rapid development of expert systems, therefore they are often also called *expert system shells.* To fulfill this purpose they incorporate specific strategies for representation, inference and control. They contain elementary constructs for modeling the world that determine the sorts of problems the tool can easily handle.

An appropriate analogy would be the tools a repairman uses. Rather than creating a new tool for each new task, the repairman collects a set of tools that have proved useful in past situations. Each tool is especially designed to perform a specialized task.

Knowledge engineering tools offer two advantages to expert system developers:

- they provide for rapid system development by providing a substantial amount of computer code that would otherwise need to be written, tested, debugged and maintained;
- tools provide specific techniques for handling knowledge representation, inference and control that help knowledge engineers to model the salient characteristics of a particular class of problem.

To create an expert system the knowledge engineer must study a particular problem domain. At first the knowledge engineer may focus on superficial behavior, but he soon moves on to question the expert in an effort to identify the underlying models, facts, heuristics and inference strategies that constitute expertise. Once the knowledge engineer understands the general characteristics of the expertise to be incorporated, he can judge if a suitable tool is available.

Generally the knowledge engineer finds that the expertise he is studying can be modeled effectively by means of one or other knowledge engineering tool. The knowledge engineer proceeds to formalize the expert's knowledge in the syntax of the tool. If there is a good match between the tool and the task, all will go smoothly. If the particular task varies in minor ways from the task the tool is designed to handle, the knowledge engineer needs to develop some special routines in the environment that lies below the tool.

Different companies have adapted different marketing strategies. Some tools are written in conventional languages to run on standard computers, whereas others are written in AI languages and designed to run on LISP workstations. Likewise some tools are narrowly focused, whereas others are designed to allow the user to develop systems appropriate to several different consultation paradigms (For an overview of currently available knowledge engineering tools ('shells') please refer to section 5.1).

The knowledge engineering field is new, and evolving very rapidly, therefore the only safe prediction one can make is that a wide variety of languages and tools will be used in the coming years.

5. An Overview of Existing Systems, Shells and Companies–Cross-references

This section gives an overview of existing Expert Systems, Shells, and companies specializing in Artificial Intelligence. This list is by no means exhaustive.

5.1. Table of Systems and Cross-references

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
1st-class	shell for PC's	Programs in motion Inc.
AALPS	plan optimal loading on aircraft	U.S. Army
ABEL	medicine	MIT
ABSTRIPS	robotics	E.D. Sacerdoti SRI International
ACE	diagnosis of telephone cable problems	G. Vesonder Bell Laboratories
ACLS	tool	Intelligent Terminal Ltd.
ACRONYM	image understanding	R.A. Brooks et al. * /Stanford Univ.
AGE	shell	H.P. Nii, N. Aiello * Stanford University
AI-SPEAR	failure diagnosis in tape drives and suggested preventive action	DEC / Billmers, M., and Swartout, M. *
AIDE	technical diagnosis	Bull(France) / Videcoq, J.-M. et al.
AIPS	graphical objects	Bolt Beranek and Newman Inc.
AL/X	knowledge engineering	J. Reiter / Intelligent Terminals Ltd.
ALPA	diagnosis in nuclear reactors	Piette, D. et al. *
ALVEN	medicine	University of Toronto
AM	concept formation in mathematics	D.B. Lenat * Stanford University
ANSWER	query system	Ban, P., Köhegyi, J., Suhai, G., Vespremi, A., Zsako, L. *
ANALYSER PLUS	tool	Business Information Techniques Ltd.
APEX	financial services industry	Applied Expert Systems, Inc.
APEX3	shell for fault diagnosis	Merry, M.
ARBY	diagnosis	D. McDermott, R. Brooks * /Yale University ITT, Smart Systems Technology

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
ART	tool	Inference Corporation
ASYL	synthesis tool	INPG/LCS(France) / Saucier, G., Crastes de Paulet, M., and Hanriat, S.
AUDITOR	aid external auditors in the field	Dungan, C.W. and Chandlers, J.S. *
AURA	logic and software design	Argonne National Laboratory
AUTODOT	inference	Martianov, V. *
Automated	Programming	Genkin, G., Hikin, A. *
BABYLON	tool	GMD *
BACON.5	scientific discovery	P. Langley et al. * /Carnegie-Mellon University
BATTLE	battlefield weapons assignment	J.R. Slagle, M.W. Gaynor * U.S. Navy
BDS	troubleshoot baseband distributions system of communications hardware	Lockheed
BETA	Battlefield Exploitation and Target Acquisition	TRW Corp. Defence Systems Division
BUGGY	computer aided instruction	J.S. Brown, R. Burton, K. Larkin * Bolt Beranek and Newman, Inc.
CAA	Casual Arhythmia Analysis	University of Toronto
CAA	advisory system for analysis management in chemical processes	Nuclear Research Center / Jaeschke, A. et al., Synergtech / Konrad, W. et al.,Technical University of Munich / Tjandra, O.
CADUCEUS	medicine	University of Pittsburgh
Cash Value	planning package	Hoskyns Ltd.
Callisto	management of large projects	Carnegie-Mellon University
CANSEARCH	online search intermediary	A.S. Pollitt *
CASNET	medicine	S.M. Weiss et al. * /Rutgers University
CATS-1	troubleshooting diesel-electric locomotives	General Electric Co.
CDx	analyse VMS dump files after crash	DEC
CENTAUR	medicine	J.S. Aikins * Stanford University
CHI	knowledge-based programming	C. Green et al. * /Kestrel Institute
Chinese	Traditional Medicine	Y. Liu * /Shanxi Institute for Automation
CLOT	medicine	Stanford University

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
CMUDA	CMU design automation	Carnegie-Mellon University
COAG	medicine	A. Lindgert and associates
CODEX	Computer-aided Diagnostic Expert System - medical aid	F. Gyarfas, M. Popper * / Research Institute of Medical Bionics, Bratislava
COMPASS	maintenance for telephone switching systems	Goyal. S.K. et al. *
CONAD	check and configure orders for computers	Nixdorf / Savory, S. *
CONCEPT	consumer goods marketing	Tymshare UK
CONGEN	chemistry	H. Brown, L. Masinter /Stanford University
CONPHYDE	CONsultant for PHYsical property DEcision	R. Banares, A.W. Westerberg and M.D. Rychener * /Carnegie-Mellon University Hungarian Acad. of Sc., Budapest, Hungary
Conphyde	physical property prediction	Banares-Alcantara, R. *
COPE	Case Oriented Processing Environment	George Washington University / Silverman, B.G. et al.
CRIB	fault diagnosis	T.R. Addis / International Computer Limited
CRITTER	evaluation of digital hardware design	V.E. Kelly, L.I. Steinberg, T. Mitchell, P. Schooley, J. Shulman, T. Weinrich *
CRYSALIS	chemistry	R. Engelmore, A. Terry * /Stanford University
CRYSTAL	tools	Intelligent Environments Ltd.
CRYSTAL II	shell	Intelligent Environments Ltd.
CSA	nuclear power plants	W.E. Underwood * Georgia Institute of Technology
CSS	planning reinstallation of IBM mainframes	IBM
DAA	Design Automatic Assistant	Carnegie-Mellon University
DANTES	real-time network trouble-shooting	CIG / Mathonet, R. et al.
DARE	Diagnostic And Repair Emulator	Symbolics Inc.
DART	diagnose hardware problems	M.R. Genesereth * /Stanford University
DAS-LOGIC	logic design	DEC
DATED	tool	Mazario, F.J.G. * Facultad de Informatica, San Sebastian, Spain

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
DEBUGGY	error detection	Xerox Corporation
DECADE	catalyst development	Narnares-Alcantara, R. *
DECGUIDE	tutor in design checking	Lockheed(Sunnyvale)
DEDALUS	automatic programming	Z. Manna, R. Waldinger * /SRI International
DELTA	diesel electric locomotive repair	P. Bonissone. et al. * /General Electric Company
Demeter	system design above register level	Carnegie-Mellon University
DENDRAL	chemistry	J. Lederberg, E. Feigenbaum * /Stanford University
DEREDEC -EXPERT	environment	Université du Québec / Paquin, L.C.
DESTINY	integrated structural design	Sriram, D. *
DEVISER	planning	Jet Propulsion Laboratory
DEX	error diagnosis	GMD *
DIAG8100	diagnose failure in DP equipment	Travelers Insurance
DIG VOLTAGE TESTER	aid troubleshooting digital voltage sources in testing lab	Lockheed
Digitalis Therapy Advisor	medicine	G. Gorry et al. * /MIT
DILOS	dialogue system	Briabrin, V.M., Pospelov, D.A. *
Dipmeter Advisor	interpreting oil well log data	R.Davis et al. * /Schlumberger-Doll Research and MIT
Discrete	Simulation	Futo, I., Szeredi, J. * / Inst. for Coordination of Computer Techniques, Budapest, Hungary
DISPATCHER	schedule dispatching of parts for robots	DEC
DIVA	technical diagnosis	CGE / David, J.-M., and Krivine, J.-P.
DOC	computer field service	Prime Computer
DUCK	logic programming	D. McDermott * /Smart Systems Technology
Dump Analysis		Software Architecture and Engineering, Inc.
EDAAS	advise on disclosure of confidential business information	Feinstein, J.L. and Siems, F. *

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
EDORA	Equations Differentielles Ordinaires Recurrentes Appliquees	P. Bernhard and C. Lobry / INRIA
EL	analysis	R. Stallmann, G. Sussman * /MIT
ELAS	wire line log analysis	C. Apte,M. Patchett,C. Apte * AMOCO and Rutgers University
EMUCS	datapath synthesis	Carnegie-Mellon University
EMYCIN	shell	W. van Melle * /Stanford University
ENGINE COOLING ADVISOR	diagnose cause of noise in automobile engine cooling system	Dourson, S. and Joyce, J. * Delco
EPM	Extended Program Model software representation for IPE	Advanced Information & Decision Systems Inc.
ES/P Advisor	shells	Expert Systems Ltd.
ESCE	consultation environment	IBM Corporation
ESCORT	process plant operators advice to handle and avoid crises	Sachs, P.A. et al. *
ESKORT	auditing VAT account	CRI A/S / Lethan, H.B.
ESDAT	medicine	University of Vienna
ESDE	development environment	IBM Corporation
ESIE	Expert System Inference Engine	Lightwave
ESP Advisor	shell	Expert System International Ltd.
ESPm	computer maintenance	NCR
ESS-S	simulation language selection	V. Jagannathan, and A.S.Elmaghraby * University of Louisville
EST	Expert Systems Toolkit	Mind Path Technologies
EURISKO	heuristics and VLSI design	D. Lenat * /Stanford University
ESPRIT	European AI Project	Lecompte, A. *
EXAMINER	medicine	University of Pittsburgh
EXCHECK	computer aided instruction	P. Suppes et al. * /Stanford University
ExMARINE	underwriting marine liability umbrella insurance policies	Coopers & Lybrand
EXPERT	shell	S.M. Weiss, C.A. Kulikowski * /Rutgers University
Expert Edge	shell	Helix Technology Ltd.
EXPERT4	shell –biological systems	Elsevier-Biosoft

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
EXPERTAXsm	corporate tax planning	Shpilberg, D. et al. *
EXPERTISE	spectra evaluation	Philips
Expert-Ease	Decision-Making	Export Systems Inc.
Expert Speller	spelling corrector	ADAT
EXSAT	configuring telecommunications satellites	Aérospatiale / Trousse, B.
EXSEL	computing	J. McDermott / Carnegie-Mellon University
EXSYS	shell for IBM PC's	Exsys Inc.
EXTASE	alarm in process control	Jakob, F. et al. *
EX-TRAN7	tool	A-Razzak, M. et al. *
FATEXP	analysis of accidents	Vaija, P., Järveläinen, M. Dohnal *
FAULTFINDER	diagnose failure in disc drive	Nixdorf /Savory, S. *
FFast	shell for financial applications	Coopers & Lybrand
Fiabex	risk control	CEP
Fieldserve	repair of electronic systems	Hofmann, M. et al. *
FOSSIL	palaeontology	Brough, D.R. and Alexander, I.F. *
FLOPS	shell including fuzzy sets	Kemp-Carraway Heart Institute / Dr. William Siler
FRODIRD-II	FRame Operating system for Design Integration with Relational Database	T. Tomiyama *
GA1	data interpretation	M. Stefik/Stanford University
GAMMA	spectral anal. for nuclear physics	D.R. Barstow * Schlumberger-Doll Research
GEM	interface management system	Bolt Beranek and Newman Inc.
Generator	control	Kitowski, J * / Inst. of Computer Sc., Univ. of Mining & Metall., Cracow, Poland
Genesis	genetic engeneering	IntelliGenetics Inc.
GEN-X	inference engin	General Electric Co.
Geomycin	geographic system	Davis, J.R., and Nanninga, P.M. *
GEOTOX	hazardous site evaluation	Wilson, J.L. et al. *
GEOX	identify earth surface minerals from image data	Chiou, W.C. * Lockheed
GEST	tool	Georgia Tech Research Institute
GODDESS	decision support	Pearl, J., Leal, A., Sdaleh, J. * University of California, LA
GRAPH-ES	network	Novosibisk / Computer Center of the Siberian Branch of the USSR Academy of Sc.

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
GUIDON	computer aided instruction	W.J. Clancey et al. * Stanford University
GUHA	knowledge acquisition	Pokorny, D. / Czechoslovak Acad. of Sc., Prague, Czechoslovakia
HAIL-1	configures circuit boards	Hazeltine Corp.
HARPY	speech understanding	B.T. Lowerre * / Carnegie-Mellon University
HASP	signal interpretation	Systems Controls, Inc.
HAVANE	videotex interface	P. Bosc / INRIA
HAZARD	environmental chemicals	Gottinger, H.W. *
Hazardous Waste	Risk Assessment and Management	P.F. Lynes et al. Craig-Lynes Chemical Management Inc.
HEADMED	medicine	Stanford University
HEARSAY II	speech understanding	D.R. Reddy et al. * /Carnegie-Mellon University
HEARSAY III	shell	B. Balzer et al. * /University of Southern California
HEATEX	networks of heat exchange	Grimes, L.E. et al. *
HI-RISE	building design	Maher, M.L. *
HODGKINS	medicine	C. Safrans et al. * / MIT
HYDRO	water resource problems	SRI International
ICLX	diagnose faults in heavy industries	Hakami, B. and Newborn, J. *
IDT	diagnosis of computer faults	H. Shubin, J.W. Ulrich * Digital Equipment Corporation
IKBM	shell	Oilfield Expert Systems Ltd.
IKE	Integrated Knowledge Environment	Lisp Machine Inc.
IKEE	Integrated Knowledge Engineering Environment for rule-base development	Nixdorf / Ludwig, A., Mellis, W., and Thomas, L.
ILEX	relat. DB system	Li, D.Y., Heath, F.G. * Min. of Elect. Industry, Beijing, China *
IMWS	Integrated Model for Water Strategy	Graillot, D. *
INCA	information handling	U.S. Department of Defense
Inform	rapid prototyping	University of Stuttgart
INFORMART ADVISOR	advise shoppers on computer purchases	Informat, Dallas
INDUCE	diagnosis	University of Illinois
Intellect	natural languages	Artificial Intelligence Corporation

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
INSIGHT	rule systems	Level % Research
INTERNIST	medicine	H. Pople, J.Myers (see Barr)* /University of Pittsburgh
IOTA	information retrieval	IMAC/LGI / Chiaramella, Y. et al.
IPE	Intelligent Program Editor analyses software	Advanced Information & Decision Systems Inc.
IPHIGENIE	software engineering instruction	ENSEEIHT / Coulette, B.
IRIS	medicine	M. Trigoboff, C. Kulikowski * /Rutgers University
IRIS	information retrieval intermediary	Dansk Datamatik Center / Larsen, H.J.
ISA	schedule orders for manufacturing and delivery	Orciuch, E. and Frost, J. *
ISIS	product. management and control	M. Fox, B. Allen, S. Smith, G. Strohm, R. Chak * Carnegie-Mellon University
Karnak	diagnosing wave solder process problems	Digital Equipment
KAS	knowledge acquisition	R. Reboh/SRI International
KBPA	Knowledge-based program automation	Kestrel Institute
KBSA	Knowledge-based software automation life-cycle support system	Kestrel Institute
KBS	modeling factory organization	Y.V. Reddy, M.S. Fox * Carnegie-Mellon University
KBS	military	J.A. Beloit, A.V. Lemmon, J.M. Selander * Mitre Corporation
KBVLSI	VLSI design	D. Lenat, W. Sutherland, J. Gibbons * Stanford University
KDS	Knowledge Delivery System	KDS Corporation
KEATS	Knowledge Engineer's Assistants tool	Open University and British Telecom
KEE	shell	G. Clemenson et al. /IntelliGenetics Inc.
KEPE	knowledge representation system	IntelliGenetics Inc.
KES	shell	Software Architecture and Engineering, Inc.
KES	rule environment	University of Maryland / Ahuya, S.B.
KIP	expert system building tool	Noesis SA

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
KMS	medicine	University of Maryland
KNOBS	mission planning	C. Engelman et al. * /Mitre Corporation
KS-300	inference system for industrial diagnostic applications	Teknowledge Inc.
LDS	laws	Rand Corporation
Learning	Machine	Lu Ji-Ren, Fang Yong-Sui * /Dept. of Radio Engn., Nanjing Inst. of Techn., China *
LES	electronic maintenance	Perkins, W.A. and Laffey, T.J. *
LHASA	chemistry	E.J. Corey, W.T. Wipke * /Harvard University
LIBRA	shell	E. Kant * /Stanford University
Linguistic	Processor	Shurov, Y.
Linguistic	Software	Korolev, E.I. (USSR)
LMS	algebra	University of Leeds
LOGIN	petroleum industry	Stanford University and Schlumberger, Ltd.
LOOPS	tool	Xerox PARC
M1	shell	Teknowledge, Inc.
MacExpert	shell for PC's	Mind Soft
Macpitts	hardware specification methodology for algorithmic VLSI design	MIT
Macro Expert	shell	Isis Systems Ltd.
MACSYMA	symbolic mathematic	C. Engleman et al. * (Mathlab)/MIT
Maintenance- Scheduler	technical diagnosis	La Trobe University(Australia) / Podbury, C.A., and Dillon, T.S.
MARS	Multiple abstraction rule-based simulator	Fairchild Laboratories
Maxwell	Knowledge base development	Silogic Inc.
MDS	mission planning	C.V. Srinivasan, D. Sandford /Rutgers University
MDX	medicine	B. Chandrasekaran et al. * /Ohio State University
MECHANIC	automobile mechanic	M. Fichtelman *
MECHO	analysis	University of Edinburgh
MECS-AI	shell	Tokyo University
MEDAS	medicine	University of Southern California
MEDIPHOR	medicine	Stanford University
MEDx	shell – medical systems	MEDx Systems Inc.
MENTAT	linguistics	ERLI / Ogonowski, A.

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
Meta-DENDRAL	chemistry	T.M. Mitchell
MICROPHEUM	medicine	Rutgers University
MIFASS	Marine Integrated Fire and Air Support System	J.R. Slagle et al. *
MIRAGE	MIcrocode/processor RAster Graphics Emulator	Hughes Aircraft
Micon	designing of single-board computers	Carnegie-Mellon University
Micro-Expert	shell	British Alvey project
Micro In-Ate	shell	Automated Reasoning Corporation
Micro-Synics	shell	British Alvey project
Micro Systems	shell	Isis Systems Ltd.
Microtrouble shooter	diagnosis faults	Mainwork Ltd.
Military	command policy	U. Wang * / Beijing Institute of Technology
MLSE	tool	Mizoguchi, F. * / Dept. of Industrial Administration, Sc. Univ. of Tokyo, Japan
MOLGEN	molecular biology	M. Stefik /Stanford University
MOTOR BRUSH DESIGNER	design of brushes and springs springs for small elect. motors	Delco
MOVER	airport transportation	Fenves, S. et al. *
MRS	Metalevel Representation System	Stanford University
MUDMAN	diagnose problems in composition of drilling	Kahn, G. and McDermott, J. *
MUSE	choosing data analysis techniques	INRA / Dambroise, E., IBM(France) / Masotte, P.
MYCIN	medicine	E.H. Shortliffe * /Stanford University
MXA	shell	SPL International / Stammers, R.A. *
Navex	monitor navigation console for Space Shuttle	Johnson Space Center
NEOMYCIN	medicine	Stanford University
NEUREX	medicine	University of Maryland
NEVAI	avalanche fall risk forecasting	CSI Piemonte / Massa-Rolandino, R.
Newton	analyzes effect of gravity	Xerox PARC

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
NEXPERT	modular newspaper system	Composition Systems Inc.
Nexus	shell	Helix Products and Marketing Ltd.
NIKL	handles structured predicates	Bolt Beranek and Newman Inc.
NOAH	planning	E.D. Sacerdoti * /SRI International
NTC	troubleshoot network problems	DEC
Nuclear Power	plants	B. Chandrasekaran * /The Ohio State University
Nuclear Power	plant	Kitowski, J., Moscinski, J. * / Inst. of Phys. and Nuclear Techn. Agh., Cracow, Poland
OCEAN	check & configure computer orders	NCR
Oil Well Drilling	advisor	Technowledge, Inc.
OLIMP	text synthesis	Gladun, V.P., Yavorsky, A.L. *
ONCOCIN	medical treatment	E.H. Shortliffe et al. * /Stanford University
OP-PLANNER	planning	RAND Corporation
OPS-5	shell	C.L. Forgy * /Carnegie-Mellon University
OPS-83	shell	Production Systems Technologies Inc.
OURCIN	diagnostic	E. Diday / INRIA and SEMA
PA	programming consultant	MIT
Palladio	VLSI design environment	Stanford, Xerox PARC
PAPLAN	tool	Futo, I., Szeredi, J. *
PARUS	design	
PECOS	automatic programming	D. Barstow * /Stanford University
PEDRO	technical diagnosis	CGE / Cornille, J.-M., and Meller, A., Avions Marcel Dassault - Breguet Aviation / Ruhla, J.
Personal Consultant	shell	Texas Instruments
PHOTOLITHOGRAPHY ADVISOR	troubleshoot photolithographic steps in circuit fabrication	Hewlett-Packard / Cline, K.L. et al.
Picon	industrial process control	Lisp Machines Inc.
PIERROT	diagnosis building alterations	Ermine, J.-L. and Pauly, J.-P.
PIES	diagnose problems on circuit fabrication line	Schlumberger
PIMS	project management	* /Esprit Project 814
PINE	help report analysis of software problems	IBM

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
PIP	medicine	S. Pauker, G.A. Gorry, J. Kassirer, W. Schwartz /MIT and Tufts Medical Center
PlanPower	financial planning	Apex Inc.
Plume	tool	DEC
POMME	helping apple growers	Virginia Tech / Drake, C., Roach, J., Virkar, R., and Weaver, M.
PREDIKT	design diagnosis and design synthesis	Oxman, R. and Gero, J.S. *
PREFACE-Expert	finance	Ecole Supérieure de Commerce de Paris / Senocourt, F.
PRIDE	new designs for copiers	Xerox
Prism	shell	IBM
PROBWELL	petroleum engineering	Tompkins, Stillman * AMOCO Production Company
PROCCIS	productivity control Information System	Kerkhoffs, E.J.H. Vansteenkiste, G.C. *
PROEXP	analysis of accidents	Vaija, P., Järveläinen, M. Dohnal, M.
Programmers Apprentice	automatic programming R. Waters * /MIT	C. Rich, H. Shrobe,
PROJCON	project management	Underwood, W.E., Summerville, J.P. * School of Information and Computer Science
PROPHET	tool	Futo, I., Szeredi, J., Redei, J. *
PROPS	shell	Expertech Ltd.
PROSPECTOR	geology	R.O. Dudaet al. * /SRI International
Prospection	in expert systems	Guan Jiwen * /Jilin University
Protein Diagn.	medicine	Helena Laboratories
PROTIS	multi-validation	Faculté de Médecine / Soula, G et al.
PROTON	expert system tool on the PSI machine	ICOT(Tokyo) / Nagai, Y., Kubono, H., and Iwashita, Y.
PROUST	analyzes Pascal programs	Yale University
PSI	knowledge-based programming	C. Green * Stanford University and Systems Control, Inc.
PSN	Procedural Semantic Network	University of Toronto
PSYCHO	medicine	University of Sheffield

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
Psychological	Experiments	Tancig, P., Bratko, I., Tancig, S. * / Josef Stefan Inst., Fac. of Elect. Engg., Univ. of Ljubliana, Yugoslavia
PTRANS	manufacturing and distribution management	P.Haley, J. McDermott Carnegie-Mellon University
PUFF	medicine	J. Kunz et al. * /Stanford University
Purposive	Thinking	Rabinovich, Z.L. (USSR) *
Optimisation of	dynamic systems	J.-P. Quadrat / INRIA
QUAL	electrical circuits	Xerox PARC
QUAL2E Advisor	advisor for stream water quality Model QUAL2E	Barnwell, T.O. Jr. et al.
R1	computer manufacturing	J. McDermott * /Carnegie-Mellon University
RAINBOW	shell	IBM
RAFFLES	fault diagnosis	T.R. Addis / International Computer Limited
RAYDEX	medicine	B. Chandrasekaran, S. Mittal, J.W. Smith * The Ohio State University
REACTOR	nuclear reactor accidents	W.R. Nelson * /EG&G Idaho, Inc.
Redesign	VLSI design	Rutgers University
RESEDEA	shell with biographical data management	G.P. Zarri *
REX	Regression EXpert	Bell Laboratories
RIDAM	RIsk/Decision Analysis Module	Chia Shun Shih, and Bernard, H.
RITA	automatic programming	Rand Corporation
RLL	knowledge engineering	R. Greiner, D.B. Lenat / Stanford
Robot	control	D.I. Novatchenko et al. *
Robotic	design	Vukobratovic, M. * / Glas. Srp. Acad. Nauka Leningrad Polytechnich. Inst., USSR
ROME	business	B. Allen, D. Kosy, B. Wise, D. Adam Carnegie-Mellon University
ROSIE	shell	J.E. Fain, F.A. Hayes-Roth, H.A. Sowizral, D.A. Waterman * RAND corporation
RuleMaster	shell	Radian Corporation

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
RUNE	shell for negotiation support	University of Ottawa / Szpakowicz, S. and Matwin, S., Carleton University / Kersten, G.E., and Michalowski, W.
RUP	Reasoning Utility Package	MIT
RX	medicine	R.L. Blum, G. Wiederhold * /Stanford University
S1	shell	Teknowledge, Inc.
SACON	analysis	J.S. Bennett, R.S. Engelmore * /Stanford University
SADD	digital designer	M.R. Grinberg * /University of Maryland
SAFE	experimental software automation system	University of Southern California, Information Science Institute
SAFIR	object-centered financial risk expert system	INRIA/IMAG / Rechenmann, F., Cap Sogeti Innovation / Doize, M.S.
SAGE	shell	SPL International Research Centre / Williams, G. *
SAM/WS	work station	Software Architecture and Engineering, Inc.
SAPFIR	inference	Ivanischev, V. *
SARYS	radar signature analysis	TASC, Inc.
Savoir	shell	ISI
Schema	VLSI design	MIT
SCHOLAR	instruction	J. Carbonell, A. Collins et al. Bolt Beranek and Newman, Inc.
SECOFOR	Advice on drill-bit sticking problems in oil wells	Teknowledge / Courteille, J.M. et al. *
SECS	chemistry	W.T. Wipke * University of California at Santa Cruz
SEEK	interactive advice on rule refinement	P. Politakis, S.M. Weiss * / DEC, Rutgers University
SENECA	fuzzy based universal expert system	M. Dohnal * /Technical University of Brno
SEPV	set of diagnosis expert systems for diseases of plants cultivated in France	INRA / Andro, T., Bachacou, J., Delhotal, P., Fayet, J.-C., Le Renard, J., Rellier, J.-P., and Thiolon, C.
Service Shell	diagnosis aid and KB	Noesis SA
Smart Expert Editor	technical publications and communications	J.M. Smart Smart Communications, Inc.

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
Smart Translator	language translation	J.M. Smart, A.J. LaMothe Smart Communications, Inc.
SMP	mathematics	Inference Corp.
Sniffer	software bugs	MIT
SOPHIE	instruction	J.S. Brown, R. Burton and colleagues Bolt Beranek and Newman, Inc.
SPADE	instruction	MIT
Spatial	Inferencing	Software Architecture and Engineering
Spear	field analysis of error logs	DEC
Spectra	Interpretation	Vida, M. * / Central NMR Lab. Slowak Techn. Univ., Bratislava, Czechoslovakia
SPERIL	analysis	Purdue University
SPEX	molecular genetics	Y. Iwasaki, P. Friedland * /Stanford University
SPIDER	image processing	Tamura / Electrotechnical Laboratory
SPILLS	hazardous spills	J.D.Allen,Jr. et al. * /Oak Ridge National Laboratory
SPHINX	medicine	Marseille University
STAMMER2	tactical situation assessment	R.J. Bechtel et al. * Naval Ocean Systems Center
STEAMER	instruction	Bolt Beranek and Newman
STOWAGE PLANNER	plan cargo storage for warehouse	Kawasaki Steel
SU/X	signal interpretation	Stanford University
Super Expert	shell on top of Expert Shell	Intelligent Terminals Ltd.
Symbolics Sage	document support	
SYN	design	Sussman, Steele, Dekleer/MIT
SYNCHEM	chemistry	H.L. Gelernter * State University of New York at Stony Brook
System-1	knowledge engineering tool	Teknowledge Inc.
Talib	synthesizes layouts for NMOS cells	Carnegie-Mellon University
TATR	tactical air targeting	M. Callero et al. * / The RAND Corporation
Tax Planning		Michaelsen, R.H. *
TAXMAN	legal reasoning and legal argumentation	L.T. McCarty Rutgers University
TECH	analysis	RAND Corporation and Naval Ocean Systems Center

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
TEIRESIAS	knowledge acquisition	R. Davis * /Stanford University
TEKNOWLEDGE	shell	Teknowledge, Inc.
Temperature	control	Wang, Y.A., Hsu, T.H., Chung, K.H. * Shanxi Mech. Eng. Inst., China *
Tess	shell	Helix Expert Systems Ltd.
Therese	tool	Pendibidu, J.-M. * Univ. de Picardie, France
Thoughtsticker	knowledge representation	British Navy and U.S. Army Research Institute
TIMM	shell	General Research Corporation
TOM	diagnostics for tomato plant pathology	INRIA
TQMSTUNE	Tune triple quadruple mass spectrometer	Lawrence Livermore Natl. Labs / Wong, C.
Translation		Medovyi, V.S. * /USSR Ac. of Sc.
Tropicaid	paramedical staff in tropical	INSERM
Tuner	adjusting signal-processing systems	TRW Corp., Defense Systems Division
TVX	Tutor for VMS systems	DEC / Billmers, M. and Carifio, M. *
TWAICE	shell	Nixdorf Computer AG
UNITS	tool	Stanford University
UTTER	speech synthesis	A.M. Serge et al. * / University of Illinois
VAX/VMS Adv.	instruction	CMU
VEGE	natural language	W.G. Lehnert, W.M. Bain * / Yale University
Vexed	VLSI expert editor	Rutgers University
VISIONS	image understanding	University of Massachusetts
VM	medicine	L. Fagan * /Stanford University
VT	configure orders for elevator system	Westinghouse
Waves	analyses of seismic data for the oil industry	Teknowledge Inc.
WEST	instruction	R. Burton, J.S. Brown
WHEAT- COUNSELLOR	control of disease in winter wheat crops	ICI / Eur. Digital Management Rep.
WHEEZE	medicine	Stanford University
WHY	instruction	A. Collins, A. Stevens, the ICIA research group * Bolt Beranek and Newman, Inc.

NAME	APPLICATION/AREA	AUTHOR/ORGANIZATION * see References
Wind forecast	expert system	Huang Reming * /Nanjing Institute of Technology
WIZDOM	shell with semantic network	Software Intelligence Laboratory Inc.
WUMPUS	instruction	I. Goldstein, Brian Carr * /MIT
XCEL	sales	Digital Equipment
XCON	computer manufacturing	J. McDermott, DEC engineers * Carnegie-Mellon University and Digital Equipment Corp.
Xi Plus	tool	Expertech / Forsyth, R. *
XSEL	salesforce tailor systems	DEC
XSYS	shell	California Intelligence
Xper	knowledge base engineering	A. Cappucio *
XTEL	design theater-wide telecommunications architectures	ICF / Feinstein, J.L., et al. Booz Allen & Hamilton / Siems, F. et al.
X-Ray	mineralogy	S.P. Ennis * /AMOCO Production Research
φNIX	modeling of oil well logs	D. Barstow, R. Duffey, S. Smoliar, S. Vestal Schlumberger-Doll Research
YAPES	shell written in PROLOG	T. Niblett * /The Turing Institute
YES/MVS	Monitor MVS operating system	IBM / Griesmer, S.K. et al. *

Sources: NTIS Database, Barr and Feigenbaum (1982), Bigger and Coupland (1983), Horn (1983), Miller (1984), Harman and King (1985), Expert Systems (1984, Vol.1, No.1), Expert Systems (1984, Vol.1, No.2), Expert Systems (1985, Vol.2, No.1), Expert Systems (1985, Vol.2, No.2), Expert Systems (1985, Vol.2, No.4), Expert Systems (1985, Vol.2, No.5), Expert Systems (1986, Vol.3, No.1), Expert Systems (1986, Vol.3, No.2), Expert Systems (1986, Vol.3, No.3), Expert Systems (1987, Vol.4, No.1).

5.2. Three Representative Examples of Expert Systems

In this section three of the most representative and well-known expert systems are presented: MYCIN (medicine), PROSPECTOR (geology) and R1 (hardware configuration).

MYCIN was the first large expert system that provided advice much like a human expert would. It was designed at Stanford University to provide consultative advice on diagnosis and therapy for infectious diseases of the blood and the use of antimicrobials. The positive identification of bacteremia takes 24 to 48 hours after the physician has taken samples, but there is a need to act quickly, even without complete results. Diagnosis and treatment are very complex and data are incomplete or inexact. Several human experts built up MYCIN's knowledge base: it is extremely detailed and combines their knowledge into one system. There are so many blood diseases that no human expert can retain each of them in memory: MYCIN considers every disease that is in the database, checks all the relevant details and is updated constantly by a medical center.

MYCIN gets the information it needs on the patient by asking the physician. It accepts the answer 'unknown' or 'uncertain' data qualified by a certain factor as an answer and will reason taking this into account. First it tries to identify the infectious organism, then prescribes drugs, which have to be compatible with each other and suitable for the specific patient. During all these queries, the physician has the possibility of asking for an explanation of the last step.

Given below is an example of a session with MYCIN that will show best how it really works (Davis, 1977):

1) Patient's name

** Fred Smith

2) Sex

** Male

3) Age

** 55

4) Have you been able to obtain positive cultures from a site at which Fred Smith has an infection?

** Yes

——— INFECTION-1 ———

5) What is the infection?

** Primary-bacteremia

6) Please give the date and approximate time when signs or symptoms of the primary-bacteremia (INFECTION-1) first appeared (mo/da/yr)

** May 5, 1980

The most recent positive culture associated with the primary-bacteremia (INFECTION-1) will be referred to as:

——— CULTURE-1 ———

7) From what site was the specimen for CULTURE-1 taken?

** Blood

8) Please give the date and time when this blood culture (CULTURE-1) was obtained (mo/da/yr)

** May 9, 1980

The first significant organism from this blood culture (CULTURE-1) will be referred to as:

——— ORGANISM-1 ———

9) Enter the identity of ORGANISM-1

** Unknown

MYCIN proceeds to ask questions that could help to identify the organism

10) Is ORGANISM-1 a rod, or coccus (etc.)

** Rod

11) The gram stain of organism-1:

** Gramneg

......

MYCIN proceeds with questions to get sufficient information to infer possible identification of the infection

My therapy recommendation will be based on the following probable infections and potentive causative organisms:

INFECTION-1 is PRIMARY-BACTEREMIA:
<item 1> The identity of ORGANISM-1 may be PSEUDOMONAS-AERUGINOSA
<item 2> The identity of ORGANISM-1 may be KLEBSIELLA-PNEUMONIAE
<item 3> The identity of ORGANISM-1 may be E.COLI
<item 4> The identity of ORGANISM-1 may be BACTERIOIDES-FRAGILIS
<item 5> The identity of ORGANISM-1 may be ENTEROBACTER
<item 6> The identity of ORGANISM-1 may be PROTEUS-NON-MIRABILIS

48) On a scale of 0 to 4, where a higher number indicates increasing severity, how would you rate Fred Smith's degree of sickness?

** 3

47) Does Fred Smith have a clinically significant allergic reaction to any antimicrobial agent?

** No

48) Fred Smith's weight in kg

** 70

My preferred therapy recommendation is as follows:

In order to cover for items <1 2 3 5 6>:
Give: GENTAMICIN
Dose: 119 mg (1.7 mg/kg) q8h IV [or IM] for 10 days
Comments: Modify dose for renal failure

In order to cover for item <4>:
Give: CLINDAMYCIN
Dose: 595 mg (8.5 mg/kg) q6h IV [or IM] for 14 days
Comments: If diarrhea or other GI symptoms develop, patient should be examined for possible pseudomembranous colitis

Do you wish to see next-choice therapy?

** yes

......

Rules are stored as a list of structures. Example of rules for the identity of organisms:

RULE037

IF: The identity of the organism is not known with certainty
And the stain of the organism is gramneg
And the morphology of the organism is ROD
And the aerobicity of the organism is aerobic
THEN: There is strongly suggestive evidence (.8) that the class of organism is enterobacteriaceae

MYCIN is built in a way that is easy to change: many other expert systems are based on its programs and structure. Every rule is a different module and can be removed or changed and the system is ready to run again. So it can be maintained and updated at the latest research level and stay effective.

PROSPECTOR is a rule-based expert consultant system for mineral exploration developed in 1981 by SRI International and the U.S. Geological Survey, to assist field geologists mainly in the evaluation of sites for certain deposits, the evaluation of geological resources in a region and the selection of favorable drilling sites.

The geologist applies a specific model matching the data he has on the region being investigated. But data and knowledge are, as with MYCIN, often *inexact, incomplete* and *uncertain*, so that at the heart of this system are techniques for evaluating measures, beliefs and suggestions. PROSPECTOR uses conditional probabilities and Bayes' Theorem to deal with the uncertainties. The probability values are low at the beginning of the run and are changed according to the propagated rules and data entered. It tries to find the model that matches best the data entered: More than fifteen specific models form PROSPECTOR's database, each of them consisting of some evidence or hypothesis (with a probability value), linked to a rule base, and forming a network altogether.

The user himself can take control of the run at any time, for example, to investigate a certain hypothesis or to ask for an explanation. He can stop PROSPECTOR whenever he wishes and provide additional information that is immediately inserted in the inference network. If PROSPECTOR is in control, it uses a depth-first strategy for traversing the inference network. It follows the rules according to their power for ruling out or increasing odds for a hypothesis.

An example of a simple rule:

IF: There is less pyrite than chalcopyrite
Or there is chalcopyrite
Or there is bomite-chalcopyrite
Or there is bomite
THEN: The mineralization is favorable for the potassic zone.

PROSPECTOR is a rule-based diagnostic system with a relatively simple construction and is an early development of how to handle indefinite knowledge and imprecise data.

R1 is used by Digital Equipment Corporation (DEC) for configuring VAX-11 computer systems. Given a customer's order, it determines what, if any, modifications have to be made to the order for reasons of systems functionality and produces a number of diagrams showing how the various components are to be associated. These diagrams are used by the technicians who assemble the system.

DEC does not sell preconfigured systems; it offers a selection of components that the customers can choose from. For the VAX-11/780, more than 400 components with up to eight properties each plus further information related to the components and assemblies have to be known. This means that a considerable amount of information is required to build up all possible systems. Another problem is that the technology is changing rapidly: efforts to develop a conventional program for configuration was unsuccessful.

R1 is an example of how to handle large search spaces through the use of "abstraction". Instead of evaluating all the possibilities here (an explosion of solutions), each part of the task should be performed as efficiently as possible without unnecessary searching, working with different levels of abstractions (successively more abstract descriptions).

R1 is data driven and works on three basic types of rules: sequencing rules, operator rules and information-gathering rules. An example of two rules where the premises of the second rule are a special case of the first is shown below (McDermott, 1980):

Assign-Power-Supply-2

IF: The most current active context is assigning a power supply
And an SBI module of any type has been put in a cabinet
And the position it occupies in the cabinet (its nexus) is known
And there is space available in the cabinet for a power supply for that nexus
And there is an available power supply
THEN: Put the power supply in the cabinet in the available space

Assign-Power-Supply-6

IF: The most current active context is assigning a power supply
And a unibus adaptor has been put in a cabinet
And the position it occupies in the cabinet (its nexus) is known
And there is space available in the cabinet for a power supply for that nexus
And there is an available power supply
THEN: Add an H7101 regulator to the order

Operator rules check special ways of treatment for the components, the information-gathering rules provide the other rules with information and handle the database. The sequencing rules divide the problem into a series of subtasks (or levels of abstraction), where the two main subtasks are:

- *correct the order:*
  Substitutions
  Components added
  Unnecessary components
  Possibly forgotten components
  Unused capacity
  Optimal ordering

- *step by step build up of the system:*
  put components into CPU cabinets
  put boxes into unibus cabinets
  put panels in unibus cabinets
  lay out system diagram
  work out the cabling

R1 has been continuously updated by DEC, it is being extended to cover PDP-11 configurations as well. It is now called XCON. In the most recent version there is a front end, the XSEL sales system, which helps salesmen to select and price the parts of an initial order. After the customer's agreement, XSEL passes the order to XCON, which configures the order.

5.3. Organizations Specializing in Expert Systems

Adaptive Technologies Inc.
600 W. North Market Blvd.
Suite 1
Sacramento, CA 95834, USA

ADAT
Bradford Science Park
1 Campus Road
Bradford BD & 1HR
England, UK

Advanced Computer Tutoring Inc.
701 Amberson Avenue
Pittsburgh, PA 15232, USA

Advanced Information & Decision
Systems
201 San Antonio Circle
Suite 286
Mountain View, CA 94040-1270, USA

Aion Corporation
101 University Avenue
Palo Alto, CA 94301, USA

AI research and Decision Systems
(AI & DS)
201 San Antonio Circle, Suite 286
Mountain View, CA 94040-1270, USA

Allen-Bradley
1201 South Second Street
Milwaukee, WI 53204, USA

All-Union Research Institute of
Systems Studies (VNIISI)
Prospect 60 Let Oktjabrja 9
117312 Moscow
USSR

AMAIA
Chemin Cazenave
ZI de St. Etiene
64100 Bayonne
France

American Auto-Matrix
One Technology Drive
Export, PA 15632, USA

Amoco Production Research
P.O. Box 591
Tulsa, OK 74102, USA

Analytic Sciences Corp.
1 Jacob Way
Reading, MA 01867, USA

Apex Inc.
Five Cambridge Center
Cambridge, MA 02142, USA

Apollo Computer Inc.
330 Billerica Road
Chelmsford, MA 01824, USA

Applicon-Schlumberger
4251 Plymouth Road
P.O. Box 986
Ann Arbor, MI 48106-0986, USA

Applied Expert Systems, Inc.
5 Cambridge Center
Cambridge, MA 02142, USA

ARCO Oil and Gas Company
P.O. Box 2819
Dallas, TX 75221, USA

Arity Corporation
358 Baker Avenue
Concord, MA 01742, USA

Arizona State University
Artificial Intelligence Laboratory
Computer Science Department
Tempe, AZ 85287, USA

Artelligence Inc.
14902 Preston Road
Suite 212-252
Dallas, TX 75240, USA

Arthur Andersen & Co.
33 West Monroe Street
Chicago, IL 60603, USA

Arthur D. Little, Inc.
Acorn Park
Cambridge, MA 02140, USA

Artificial Intelligence Corporation
100 Fifth Avenue
Waltham, MA 02254, USA

Artificial Intelligence Publications
95 First Street
Los Altos, CA 94022, USA

AT&T Bell Laboratories
2C278
Murray Hill, NJ 07974

Austrian Society for Cybernetic Studies
3 Schottengasse
A-1010 Vienna, AUSTRIA

Automated Reasoning Corporation
290 West 12th Street
Suite 1-D
New York, NY 10014, USA

Automatix Inc.
1000 Technology Park Avenue,
Billerica, MA 01821, USA

Batelle Institute
P.O. Box 900160
D-6000 Frankfurt 90, FRG

Battelle Memorial Institute
505 King Ave.
Columbus, OH 43201, USA

Bell Laboratories
600 Mountain Ave.
Murray Hill, NJ 07974, USA

Bell Laboratories
Artificial Intelligence Systems
Department
Whippany, NJ 07981, USA

Bitstream Inc.
Athenaeum House
215 First Street
Cambridge, MA 02142, USA

Boeing Company
Boeing Computer Service Company
Artificial Intelligence Center
Advanced Technology Application Div.
P.O. Box 24346
Seattle, WA 98124, USA

Bolt Beranek and Newman, Inc.
50 Moulton Street
Cambridge, MA 02138, USA

Booz, Allen & Hamilton Inc.
50 Moulton Street
Cambridge, MA 02138, USA

Boston University
Bay State Road
Boston, MA 02215, USA

Brattle Research Corp.
215 First Street
Cambridge, MA 02142, USA

Brigham Young University
Provo, UT 84602, USA

Brown University
79 Waterman Street
Providence, RI 02912, USA

Business Information Techniques Ltd.
Bradford University Science Park
20-26 Campus Road
Bradford BD7 1HR, UK

Bynas Division
Uny Co. Ltd.
5F, DAI-Nagoya Bldg.
28-12, 3-Chome, Meieki
Nakamura-ku, Nagoya, 450
Japan

California Intelligence
912 Powell Street 8
San Francisco, CA 94108, USA

Cambridge Consultants Ltd.
Science Park
Milton Road
Cambridge CB4 4DW
UK

CAM-I Inc.
611 Ryan Plaza Dr.
Suite 1107
Arlington, TX 76011, USA

CAP SOGETI INNOVATION
241 Rue Garibaldi
69422 Lyon Dedex 3
France

Carnegie Group Inc.
650 Commerce Court,
Station Square
Pittsburgh, PA 15219, USA

Carnegie-Mellon University
Intelligent Systems Laboratory
Schenley Park
Pittsburgh, PA 15213, USA

CCA Uniworks Inc.
20 William Street
Wellesley, MA 02181, USA

CEP
34 rue Rennequin
75017 Paris
France

COGNITECH
1 Rue J. Lefebvre
75009 Paris
France

Cognition
900 Tech Park Drive
Billerica, MA 01803, USA

Cognitive Systems, Inc.
234 Church Street
New Haven, CT 06510, USA

Columbia University
Computer Science Department
450 Computer Science Bldg.
New York, NY 10027, USA

Computer and Systems Science Dept.
University of Pavia
Strada Nuova 106/C
I-27100 Pavia, ITALY

Computer*Thought
1721 West Plano Parkway
Plano, TX 75075, USA

Consiglio Nazionale delle Ricerche
Piazzale Aldo Moro 7
I-00185 Rome, ITALY

Construction Engineering Research
Laboratory
U.S. Department of the Army
Champaign, IL 61820-1305, USA

Coopers & Lybrand
Management Consulting Services
Decision Support Group
1251 Avenue of the Americas
New York, NY 10020, USA

Czechoslovak Academy of Science
Narodni Tr. 3
111 42 Prague 1
Czechoslovakia

Daisy Systems Corporation
139 Kifer Court
Sunnyvale, CA 94086, USA

Dalek Corporation
2976 Wilson's Crossing Court
Decatur, GA 30033, USA

Dartmouth College
Hanover, NH 03755, USA

Data General
4400 Computer Drive
Westboro, MA 01580, USA

Defense Advanced Research
Projects Agency (DARPA)
U.S. Department of Defense
Strategic Computing Project
1400 Wilson Boulevard
Arlington, VA 22209-2308, USA

Delphi s.p.a
via della Vetraia 11
55049 Viareggio
Italy

Department of Chemical Engineering
Technical University of Brno
Kravi Hora 2
602 00 Brno
CSSR

Department of Computer Science
SUNY at Stony Brook
Stony Brook, NY 11794, USA

Department of Medical Informatics
University of Pitie Salpetriere
91, boulevard de l'Hopital
75634 Paris Cedex 13
FRANCE

Digital Equipment Corporation
AI Technology Center
77 Reed Road
Hudson, MA 01749, USA

Digital Equipment Corporation
Knowledge Engeneering Group
South Lawrence, MA 01843, USA

Digitalk Inc.
5200 West Century Boulevard
Los Angeles, CA 90045, USA

Dogwood AI Research
Northwest AI Ltd.
1826 W. 1st. Ave.
Vancouver, B.C. V6J 1G5
Canada

Dynamic Master Systems Inc.
P.O. Box 566456
Atlanta, GA 30356, USA

Eantech Inc.
325 East Middlefield
Mountain View, CA 94043, USA

EG&G Idaho, Inc.
P.O. Box 1625
Idaho Falls, ID 83415, USA

Elsevier-Biosoft
68 Hills Road
Cambridge CB2 1LA
UK

EMI Computer Software Ltd.
Boundary House
Boston Road
London W7 2QE
UK

ESL Inc.
495 Java Drive
P.O. Box 3510
Sunnyvale, CA 94088, USA

ESPRIT
200, Rue de la Lori
B-1049 Brussels
Belgium

European Space Agency
8-10 Rue Mario Nikis
75738 Paris Cedex 15
France

Expert-Knowledge Systems Inc.
6313 Old Chesterbrook Road
Mclean, VA 22102, USA

Expertech Ltd.
Expertech House
172 Bath Road
Slough
Berks SL1 3XE, UK

ExperTelligence Inc.
559 San Ysidro Rd.
Santa Barbara, CA 93108, USA

Expert Systems Corporation
100 Fifth Avenue
Watham, MA 02135, USA

Expert Systems, Inc.
868 West End Avenue, Suite 3A
New York, NY 10025, USA

Expert Systems, Ltd.
Expert Systems International
1150 First Avenue
King of Prussia, PA 19406, USA

Expert Systems International Ltd.
9 West Way
Oxford OX2 0JB
UK

Export Software International
Craigcrook Castle
Craigcrook Road
Edinburgh, Scotland, UK

Exsys Inc.
PO Box 75158
Albuquerque, NM 87194, USA

Fairchild Camera & Instrument Corp.
Lab for Artificial Intelligence Research
4001 Miranda Avenue
Palo Alto, CA 94303, USA

FMC Corp.
Artificial Intelligence Center
Central Engineering Lab.
1185 Coleman Ave.
Santa Clara, CA 95052, USA

Ford Aerospace & Communications Corp.
AI Lab.
P.O. Box 58487
1150 Gemini Avenue
Houston, TX 77258, USA

FRAMENTEC
Monte-Carlo Sun, Batiment E-F, 6e etage
74 Boulevard d'Italie
MC 98000, Monte Carlo

Franz Inc.
1151 Harbor Bay Parkway
Alameda, CA 94501, USA

Fuji Electric Company Ltd.
Shin Yarakucho Bldg.
12-1 chrome Yurakuchocho
Chiyoda-ku, Tokyo
Japan

Fujitsu Ltd.
1-chrome 6-1 Marunouchi,
Chikyoda-ku, Tokyo
Japan

GCA Corporation
Industrial Systems Group
One Energy Center
Naperville, IL 60566, USA

General Electric Company
Research and Development Center
P.O. Box 43
Schenectady, NY 12345, USA

General Motors Research Labs
Computer Science Department
Warren, MI 48090, USA

General Research Corporation
5383 Hollister Avenue
Santa Barbara, CA 93160-6770, USA

Georgia Institute of Technology
School of Information and
Computer Science
Atlanta, GA 30332, USA

Georgia Tech Research Institute
Artificial Intelligence Branch
GTRI-EML
Atlanta, GA 30332, USA

Gesellschaft für Mathematik und
Datenverarbeitung (GMD)
Postfach 1240
Schloβ Birlinghoven
D-5205 Sankt Augustin 1, FRG

Gold Hill Computers
163 Harvard Street
Cambridge, MA 02139, USA

Graphael, "Le Continental"
Avenue Descartes
BP 256
93153 Le Blanc, Mesnil Cedex
France

Graphics and Interactive Systems
Research
Center for Mathematics and Computer
Science
Kruislaan 413
1098 SJ Amsterdam
The Netherlands

GTE Laboratories Inc.
Box A, 40 Sylvan Road
Waltham, MA 02254, USA

Harvard University
Aiken Computer Laboratory
Cambridge, MA 02138, USA

Harwell Laboratory
Didcot, OX11 0RA
England, UK

Helena Laboratories
P.O. Box 752
Beaumont, TX 77704, USA

Heuristics Inc.
Sacramento, CA, USA

Hewlett-Packard Company
3000 Hanover Street
Palo Alto, CA 94304, USA

Hewlett-Packard
Computer Research Center
Applied Technology Laboratory
1501 Page Mill Road
Palo Alto, CA 94304

Helix Products and Marketing Ltd.
11 Ludgate Circus
London EC4M 7LQ, UK

Hitachi Ltd.
Energy Research Laboratory
C.P.O.B. 14
Chiyada-ku, Tokyo
Japan

Hitachi Ltd.
System Development Laboratory
5-1 Marunouchi 1-chrome
Chiyada-ku, Tokyo
Japan

Honeywell
MN17-2349
2600 Ridgway Parkway
Minneapolis, MN 55413, USA

Hoskyns Ltd.
Africa House
64-78 Kingsway
London WC2B 6BL
UK

Hughes Aircraft Company
Artificial Intelligence Center
23901 Calabasas Road
Calabasas, CA 91302, USA

Hughes Aircraft Company
P.O. Box 1042
El Segundo, CA 90245, USA

HyperTech Associates
15775 North Hillcrest
Suite 508-407
Dallas, TX 75248, USA

IBM
Expert Systems Project
Palo Alto Scientific Center
1530 Page Mill Road
Palo Alto, CA 94304, USA

ICAM
1900 Sources Boulevard
Pointe Claire
Quebec H9R 4Z3
Canada

ICI Plant Protection
Woolmead House West
Bear Lane, Farnham
Surrey GU9 7UB, UK

Impediment Incorporated
333 Franklin Street
Duxbury, MA 02332, USA

Imperial College
Department of Computing
180 Queens Gate
London SW7 2BZ, UK

Inference Corporation
5300 W. Century Blvd., Suite 501
Los Angeles, CA 90045, USA

Informatics Section UNESCO
7, place de Fontenoy
75700 Paris
France

Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292, USA

Information System Support Division
World Health Organization (WHO)
20, avenue Appia
1211 Geneva, Switzerland

Institute of Gas Technology
4201 West 36th Street
Chicago, IL 60632, USA

Integral Quality
6265 Twentieth Avenue NE
Seattle, WA 98115

IntelliCorp
707 Laurel Street
Menlo Park, CA 94025-3445, USA

Intelligent Computer Research Institute
(ICS)
Applied Artificial Intelligence Reporter
P.O. Box 248235
Coral Gables, FL 33124, USA

IntelliGenetics
124 University Avenue
Palo Alto, CA 94301, USA

Intelligent Environments Ltd.
20 Crown Passage
St James's
London SW1 6PP, UK

Intelligent Terminals Ltd.
George House, 36 North Hanover Street
Glasgow G1 2AD, Scotland, UK

Intergovernmental Bureau of Informatics
PO Box 10253
I-00144 Rome
Italy

International Business Machines Corp.
1133 Westchester Avenue
White Plains, NY 10604, USA

International Business Machines
Palo Alto Scientific Cener
153 Page Mill Road
Palo Alto, CA 94303, USA

International Business Machine Corp.
P.O. Box 218
Yorktown Heights, NY 10598, USA

Isis Systems
11 Oakdene Road
Redhill
Surrey RH1 6BT, UK

ITMI
Chemin des Clos
ZIRST 38240 Meylan
France

ITMT Marketing Corp.
1000 Massachusetts Ave.
Cambridge, MA 02138, USA

Jeffrey Perrone & Associates
3685 17th Avenue
San Francisco, CA 94114, USA

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91103, USA

Josef Stefan Institute
Jamova 39
6100 Ljubljana
Yugoslavia

KDS Corporation
934 Hunter Road
Wilmette, IL. 60091, USA

Kemp-Carraway Heart Institute
Birmingham, AL 35234, USA

Kestrel Institute
1802 Page Mill Road
Palo Alto, CA 94304, USA

Kyoto University
Yoshida-Honmachi
Sakyo-Ku, Kyoto 606
Japan

Laboratoire d'Electronique et d'Etude
des Systemes Automatiques
University of Rabat
Boite Postale 1014
Rabat, Morocco

Lab. d'Epidemiologie, Statistique et
Informatique Medicales
Universite de Bordeaux II
146, rue Leo-Saignat
33076 Bordeaux-Cedex
France

Level 5 Research Inc.
4980 South A-1-A
Melbourne Beach, FL 32951, USA

Levien Instrument Co.
Sittlington Hill
PO Box 31
McDowell, VA 24458, USA

Lightwave
PO Box 290539
Tampa, FL 33617, USA

LISP Machines Inc.
Process Systems Division
6033 West Century Boulevard
Los Angeles, CA 90045, USA

Lithp Systems BV
Meervalweg 72
1121 JP Landsmee
The Netherlands

Lockheed Missiles and
Space Company Inc.
Lockheed Palo Alto
Research Laboratory
3251 Hanover Street
Palo Alto, CA 94304, USA

Logic Based Systems Ltd.
40 Beaumont Avenue
Richmond
Surrey TW9 2HE, UK

Logic Programming Associates Ltd.
Studio 4
The Royal Victoria Patriotic Building
Trinity Road
London SW18 3SX, UK

Logicware Inc.
1000 Finch Avenue West
Toronto
Ontario, M3J 2V5
Canada

Los Alamos National Laboratory
Group NSP/AWT MS F668
Los Alamos, NM 87545, USA

Lucid Inc.
1090 East Meadow Circle
Palo Alto, CA 94303, USA

Machine Intelligence Corporation
330 Portero Avenue
Sunnyvale, CA 94086, USA

Machine Vision International
325 E. Eisenhower
Ann Arbor, MI 48104, USA

Mainwork Ltd.
24 West Street, Porchester
Hants, England, UK

Massachusetts Institute of Technology
Artificial Intelligence Laboratory
or Clinical Decision Making Research
Group
545 Technology Square
Cambridge, MA 02139, USA

McDonnel Douglas
16441 Space Center Blvd.
Houston, TX, USA

MEDx Systems Ltd.
Springdale Court
PO Box 2000
Dover, MA 02030, USA

Microelectronics and Computer
Technology Corp.
9430 Research Blvd.
Echelon Building 1, Suite 200
Austin, TX 78759, USA

Microsoft Corporation
10700 Northup Way
Bellevue, WA 98004, USA

Migent Software
8300 N. Hayden Road
Scottsdale, AZ 85258, USA

Mind Path Technologies
Park Central VII
Suite 1002, LB 77
Dallas, Texas 75251, USA

Mind Soft
3 rue de l'Arrivee
75749 Paris Cedex 15
France

Mitre Corporation
McLean, VA 22101, USA

National Computing Centre
Oxford Road
Manchester M1 7ED
England, UK

Naval Ocean Systems Center
San Diego, CA 92152, USA

Navy Center for Applied Research in AI
Naval Research Laboratory
Washington, DC 20375, USA

Neuron Data Inc.
444 High Street
Palo Alto, CA 94301, USA

Nixdorf Computer AG
Vertriebsleitung Expertensysteme
Berliner Strasse 95
8000 München 40
Federal Republic of Germany

Noesis SA
1 rue du Tunnel
2996 Neuchatel
Switzerland

Northwest Computer Algorithms
PO Box 90995
Long Beach, CA 90809, USA

Oak Ridge National Laboratory
PO Box X
Oak Ridge, TN, USA

Oilfield Expert Systems Ltd.
Victoria Road, Burgess Hill
West Sussex, England, UK

Palladian Software
Four Cambridge Center
11th Floor
Cambridge, MA 02142, USA

PAR Technology Corp.
220 Seneca Turnpike
New Hartford, NY 13413, USA

Paradigm Associates Inc.
29 Putman Avenue, Suite 6
Cambridge, MA 02139, USA

Jeffrey Perrone & Associates
3685 17th Street
San Francisco, CA 94114, USA

Politecnico di Milano
Corso Duca degi Abruzzi 24
10129 Turin
Italy

ProCode International
15930 SW Colony Place
Portland, OR 97224, USA

Production Systems Technologies Inc.
642 Gettysburg Street
Pittsburgh, PA 15206, USA

Programming Logic Systems Inc.
31 Crescent Drive
Milford, CT 06460, USA

Programs in Motion Inc.
10 Sycamore Road
Wayland, MA 01778, USA

PROLOGIA
278 Rue St. Pierre
13005 Marseille
France

Purdue University
School of Computer Science
West Lafayette, IN 47907, USA

Queen's University
Kingston, Ontario K7L 3N6
Canada

Quintus Computer Systems Inc.
2345 Yale Street
Palo Alto, CA 94306, USA

Radian Corporation
8501 Mo-Pac Boulevard
P.O. Box 9948
Austin, TX 78766, USA

Reasoning Systems Inc.
1801 Page Mill Road
Palo Alto, CA 94304, USA

Research Institute of Medical Bionics
Jedlova 6
833-8 Bratislava
CSSR

Rohm and Haas Co.
Computer Application Research
727 Norristown Road
Spring House, PA 19477, USA

Rutgers University
Department of Computer Science
New Brunswick, NJ 08903, USA

Schlumberger-Doll Research
David Barstow
Old Quarry Road
Ridgefield, CT 06877, USA

Scicon Ltd.
43 Berners Street
London W1P 4AQ, UK

Science Applications
International Corporation
1710 Goodridge Drive
P.O. Box 1303
Mclean, VA 22102, USA

Science University of Tokyo
1-3 Kagurazaka
Shinjuku-ku, Tokyo 162
Japan

Scientific Computers Ltd.
Victoria Road
Burgess Hill
West Sussex, RH15 9LW, UK

Semantic Microsystems
1001 Bridgeway Suite 543
Sausalito, CA 94965, USA

Siemens Corporate Research
and Support Inc.
Research and Technology
Laboratories
105 College Road East
Princeton, NJ 08540, USA

Silogic Inc.
6420 Wishire Blvd.
Los Angeles, CA 90048, USA

SIRA
South Hill, Chiselhurst
BR7 5EH, UK

Smart Communications, Inc.
Box 963, FDR Station
655 Third Avenue
New York, N.Y. 10150, USA

Smart Systems Technology
Suite 421 North
7700 Leesburg Pike
Falls Chaurch, VA 22043, USA

Smart Systems Technology
6870 Elm Street
Suite 300
McLean, VA 22101, USA

Software Architecture and
Engineering, Inc., Suite 1220
1401 Wilson Boulevard
Arlington, VA 22209, USA

Software Arts Products Corp.
27 Mica Lane
Wellesley, MA, USA

Software Intelligence Laboratory Inc.
1593 Locust Avenue
Bohemia, NY 11716, USA

Softsync Inc.
162 Madison Avenue
New York, NY 10016, USA

Solution Systems
335-B Washington Street
Norwell, MA 02061, USA

Southwest Research Institute
6220 Culebra Road
San Antonio, TX 78284, USA

Spartacus Inc.
5 Oak Park Drive
Bedford, MA 01730, USA

SPL International Research Center
The Charter, Abingdon
Oxon., England, UK

SRI International
555 Ravenswood Street
Menlo Park, CA 94025, USA

Stanford University
Artificial Intelligence Laboratory
or Heuristic Programming Project
Stanford, CA 94305, USA

Starwood Corporation
P.O. Box 160849
San Antonio, TX 78280, USA

State University of New York
College at Buffalo
1300 Elmwood Ave.
Buffalo, NY 14222, USA

State University of New York
Department of Computer Science
Stony Brook NY 11794, USA

Strategy Consulting
1220 Edgewood Road
Redwood City, CA 94062, USA

Sun Microsystems Europe Inc.
Berkshire House
High Street
Ascot, Berkshire, SL5 7HU, UK

Sun Microsystems Inc.
2550 Garcia Avenue
Mountain View, CA 94043, USA

Symbolics, Inc.
4 Cambridge Center, 12th floor
Cambridge, MA 02142, USA

Syntelligence
800 Oak Grove Ave. Suite 201
Menlo Park, CA 94025, USA

Syracuse University
120 Hinds Hall
Syracuse, NY 13210, USA

Systems Control Technology Inc.
1801 Page Mill Road
Palo Alto, CA 94303, USA

Systems Designers Software
444 Washington Street
Suite 407
Woburn, MA 01801, USA

System Development Corporation
2525 Colorado Place
Santa Monica, CA 90406, USA

Systems Research Laboratories Inc.
2800 Indian Ripple Road
Dayton, OH 45440, USA

TECSI SOFTWARE
29 Rue des Pyramides
75001 Paris
France

Teknowledge Inc.
525 University Avenue
Palo Alto, CA 94301, USA

Tektronix Inc.
Wilsonville Industrial Park
P.O. Box 1000
D/S 63-393
Wilsonville, OR 97070, USA

Tektronix Inc.
Computer Research Laboratory
Knowledge-based System Group
P.O. Box 500
Beaverton, OR 97077, USA

Texas Instruments Inc.
P.O. Box 2909
M/S 2107
Austin, TX 78769, USA

Texas Instruments
P.O. Box 809063
Dallas, TX 75380-9063, USA

The Athena Group
575 Madison Avenue, Suite 1006
New York, NY 1022, USA

The Carnegie Group
650 Commerce Court
Stadion Square
Pittsburgh, PA 15219, USA

The Foxboro Company
Bristol Park
Foxboro, MA 02035, USA

The MITRE Corporation
P.O. Box 208
Bedford, MA 01730, USA

The Ohio State University
Dep. of Computer and Inf. Science
Columbus, OH 43210, USA

The Pentagon
Department of Air Force
Washington D.C. 20330, USA

The Rand Corporation
1700 Main Street
Santa Monica, CA 90406, USA

Thinking Machines Corporation
245 First Street
Cambridge, MA 02142, USA

Tokyo University of
Electrical Engineering
Kanda-Nishiki-cho
Chiyoda-ku, Tokyo
Japan

TRW Defense Systems
One Space Park
Redondo Beach, CA 90278, USA

Tufts University
Medford, MA 02155, USA

Tymshare Inc.
Knowledge Engineering
Products Division
20705 Valley Green Drive
Cupertino, CA 95014, USA

Tymshare UK
Kew Bridge House, Kew Bridge Road
Brentford TW8 0EJ, UK

Tymshare UK Ltd.
Heron House
Guildford Road
Woking, Surrey, UK

Ultimate Media Inc.
275 Magnolia Avenue,
Lakespur, CA 94939, USA

Ultramax Corporation
1313 E. Kemper Rd., Suite 3200
Cincinnati, OH 45246, USA

University of Aix en Provence
3 Ave. Robert Schuman
13626 Aix-en-Provence
France

University of Alberta
Department of Computing Science
Edmonton, Alberta T6G 2H1
Canada

University of California
Dept. of Electrical Engineering
and Computer Sciences
Berkeley, CA 94720, USA

University of California
Irvine, CA 92717, USA

University of California
Los Angeles
405 Hilgard Ave.
Los Angeles, CA 90024, USA

University of California at Santa Cruz
Board of Studies in Information Sciences
237 Applied Sciences Building
Santa Cruz, CA 95064, USA

University of Colorado
P.O. Box 7150
Colorado Springs, CO 80933, USA

University of Conneticut
Storrs, CT 06268, USA

University of Delaware
Newark, DE 19716, USA

University of Edinburgh
Department of Artificial Intelligence
Forrest Hill
Edinburgh EH1 2QL, UK

University of Florida
Computer & Information Science
Gainesville, FL 32611, USA

University of Heidelberg
Postfach 105760
Grabengasse 1
6900 Heidelberg
Federal Republic of Germany

University of Illinois at Urbana-Champain
Artificial Intelligence Group
Coordinated Science Laboratory
Urbana, IL 61801, USA

University of Kaiserslautern
Fachbereich Informatik
Postfach 3049
D-6750 Kaiserslautern
Federal Republic of Germany

University of Manchester
Oxford Road
Manchester M13 9PL, UK

University of Maryland
Department of Computer Science
College Park, MD 20742, USA

University of Massachusetts
Dept. of Computer and Information Science
Amherst, MA 01003, USA

University of Michigan
Dept. of Electrical Engineering
and Computer Science
3314 East Engineering Bldg.
Ann Arbor, MI 48109, USA

University of Minnesota
Minneapolis, MN 55455, USA

University of Missouri
School of Medicine
Columbia, MO, USA

University of Nebraska
School of Accounting
Lincoln, Nebraska, USA

University of Pennsylvania
Dept. of Computer and
Information Science
Philadelphia, PA 19104, USA

University of Pittsburgh
Computer Science Department
4200 Fifth Avenue
Pittsburgh, PA 15260, USA

University of Rochester
Wilson Blvd.
Rochester, NY 14627, USA

University of Southern California
Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90291, USA

University of Texas
Artificial Intelligence Laboratory
T.S. Painter 3.28
Austin, TX 78712-1188, USA

University of Tokyo
Hongo Bunkyo-ku
Tokyo
Japan

University of Tokyo
Institute of Medical Science
Minato-ku, Tokyo 108
Japan

University of Toronto
Department of Computer Science
Toronto, Ontario M5S 1A7
Canada

University of Utah
Department of Computer Science
3160 Merrill Engineering Bldg.
Salt Lake City, UT 84112, USA

University of Waterloo
Waterloo, Ontario N2L 3G1
Canada

University of Wisconsin
P.O. Box 2000
Kenosha, WI 53141, USA

University of Wollongong
Department of Computing Science
Wollongong, NSW 2500
Australia

U.S. Air Force
Fome Air Development Center
Griffiss Air Force Base
NY, USA

U.S. Army
Center for Artificial Intelligence
Fort Belvoir, VA 22060, USA

U.S. Department of Energy
1000 Independence Avenue SW
Washington D.C. 20585, USA

U.S. Environmental Protection
Agency
401 M Street SW
Washington D.C. 20460, USA

U.S. Navy
Applied Research in Artificial
Intelligence
Naval Research Laboratory
4555 Overlook Ave. SW
Washington D.C. 20375-5000, USA

VA Mental Health Clinical
Research Center
810 Vermont Avenue NW
Washington D.C. 20420, USA

Vanderbilt University
401 24th Avenue S
Nashville, TN 37212, USA

Verac Incorporated
9605 Scranton Road, Suite 500
San Diego, CA 92121, USA

Virginia Polytechnic Institute
and State University
Electrical Engineering Department
Blacksburg, VA 24061, USA

Wang Laboratories
One Industrial Avenue
Lowell, MA 01851, USA

Westinghouse Electric Corp.
Westinghouse Public Relations
One Maritime Plaza
San Francisco, CA 94111, USA

Westinghouse Research and
Development Center
1310 Beulah Road
Pittsburgh, PA 15235, USA

Wright-Patterson AFB
Air Force Logistics Command
Wright Patterson AFB, OH 45433, USA

Xerox Artificial Intelligence
Systems (AIS)
P.O. Box 7018
Pasadena, CA 91109, USA

Xerox Palo Alto Research Center
Knowledge Systems Area
3333 Coyote Hill Road
Palo Alto, CA 94304, USA

Yale University
Dep. of Computer Science
P.O. Box 2158 - Yale Station
10 Hillhouse Avenue
New Haven, CT 06520, USA

Zeta-Soft Ltd.
P.O. Box 139
Cambridge, MA 02141, USA

6. IIASA In-house Research

At IIASA the *Advanced Computer Applications Project* (ACA) develops and implements a new generation of interactive and model-based decision support systems, that combine methods and approaches of operations research and applied systems analysis with elements of AI and advanced information and computer technology.

These systems consist of *integrated sets of software tools*, designed for non-technical users in the field of industrial risk assessment, the management of hazardous substances, and integrated regional development. Their primary purpose is to provide easy access and to allow the efficient use of methods of analysis and information management normally restricted to a small group of technical experts and to permit a more comprehensive and interdisciplinary management in the problem domain. The use of advanced information- and data-processing technology, software engineering, and concepts of AI now permit a substantial increase in the group of potential users of advanced systems analysis methodology and thus provide a powerful tool in the hand of planners, managers, policy and decision makers and their technical staff.

Some of ACA's projects are listed below:

- *an Expert System for Integrated Development: A Case Study of Shanxi Province, the People's Republic of China* a collaborative project with Chinese academic, industrial, and governmental institutions in Shanxi province. An operational prototype level expert system is being developed for use by the regional government of Shanxi province for coordinated development planning, with simultaneous consideration of the numerous elements involved.
- *Advanced Decision-oriented Software for the Management of Hazardous Substances* under contract to the Commission of the European Communities, Joint Research Center (JRC), Ispra (IRIMS–The Ispra Risk Management System; for details see Fedra, 1985 and 1986).
- the *Interactive Risk Assessment for Chlorine Transportation in the Netherlands* for the Dutch Ministry of Housing, Physical Planning and the Environment: an interactive and graphics-oriented framework and post-processor for the risk assessment package SAFETI (Technica) which facilitates the quick generation, display, evaluation and comparison of policy alternatives and individual scenarios.

In the following we focus on the IRIMS project, touching on not only approaches and concepts but also on the implementation.

6.1. The Problem Area: Management of Hazardous Substances

The problems of managing hazardous substances are neither well defined nor reducible to a small set of relatively simple subproblems. The entire life-cycle of hazardous substances (Figure 5) (the industrial production sector, use and market, waste management, including treatment and disposal, the cross-cutting transportation sector, and finally man and the environment), involves numerous aspects and levels of planning, policy and management decisions, as well as technological, economic, socio-political and environmental considerations.

A modular design philosophy was adopted to allow the development of individual building blocks, which are valuable products in their own right. These were then provided with interfaces and integrated in a flexible framework which is easily modifiable, with increasing experience of use, and permits new methods and approaches to cover additional aspects of the problem. The computer is seen as a mediator and translator between expert and decision maker, between science and policy. The knowledge of human experts of different fields mentioned above is managed by the system and transmitted to the user in a convenient way, depending on his technical level and his information requirements. Expertise in the numerous domains is needed by the user and supported by the

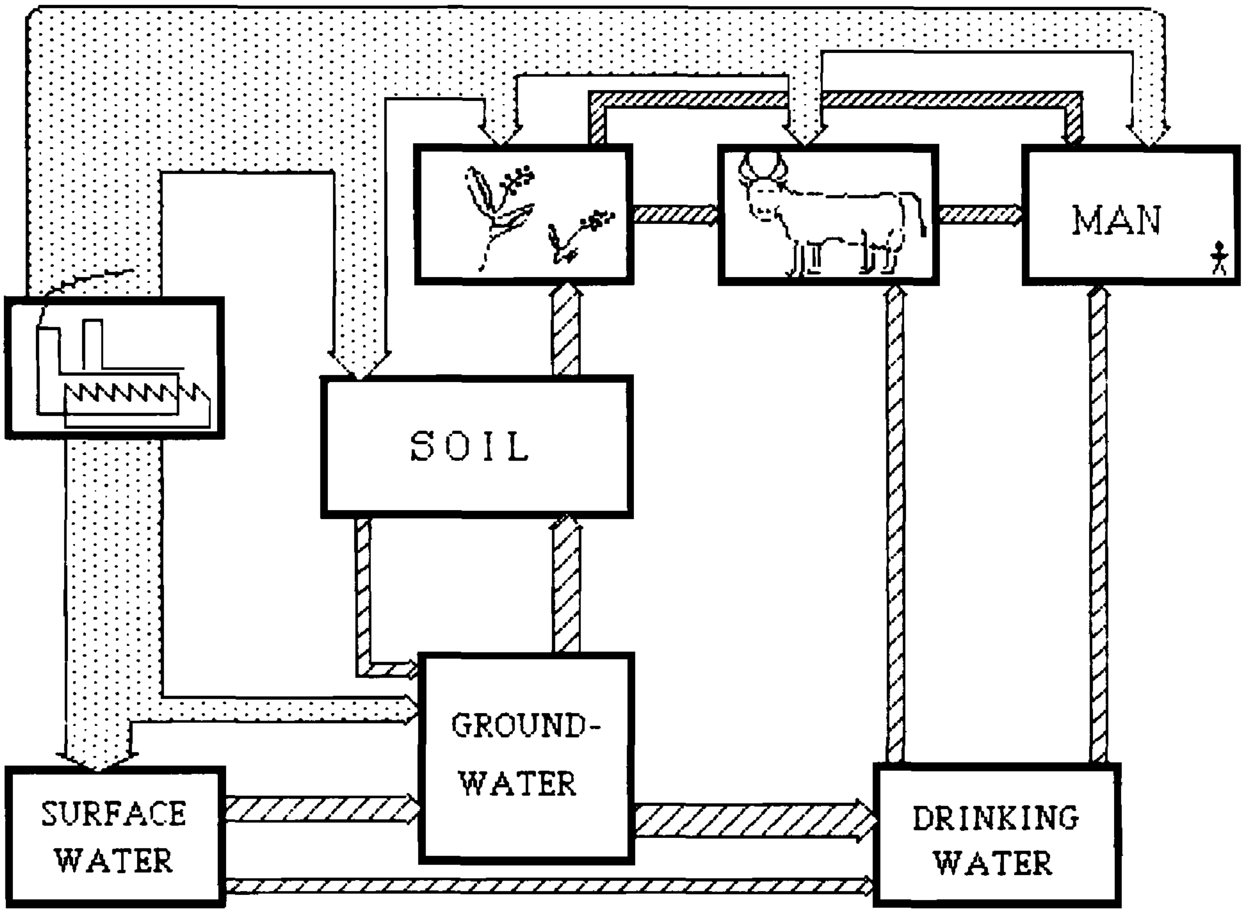

Figure 5: Life cycle of hazardous substances (after Fedra, 1985)

system: the user makes his information requests in any specific field and the system will come up with the corresponding module of knowledge acquired from human experts (eventually, former users).

Methods of applied systems analysis and risk assessment implemented using modern information processing technology with user friendly interfacing, can now support such a comprehensive, interdisciplinary approach to the management of industrial risk. This approach can provide a powerful interactive tool for planners and policy makers, because it makes access to a large number of relevant databases and problem simulation modules easy.

The IRIMS system consists of 12 main modules for information retrieval, simulation, optimization and multicriteria scenario evaluation. The user can choose a module by selecting one of the menu options of the master menu screen (Figure 6) and is then continuously guided through the system by subsequent submenus. The 12 main modules can be briefly described as follows:

Hazardous Substances Database: The information system on selected hazardous substances provides detailed information about chemical and physical properties of more than 700 hazardous substances, related production processes and cross-connections to the waste streams database, the regulations database and the production technologies database accessible via numerous entry strategies.

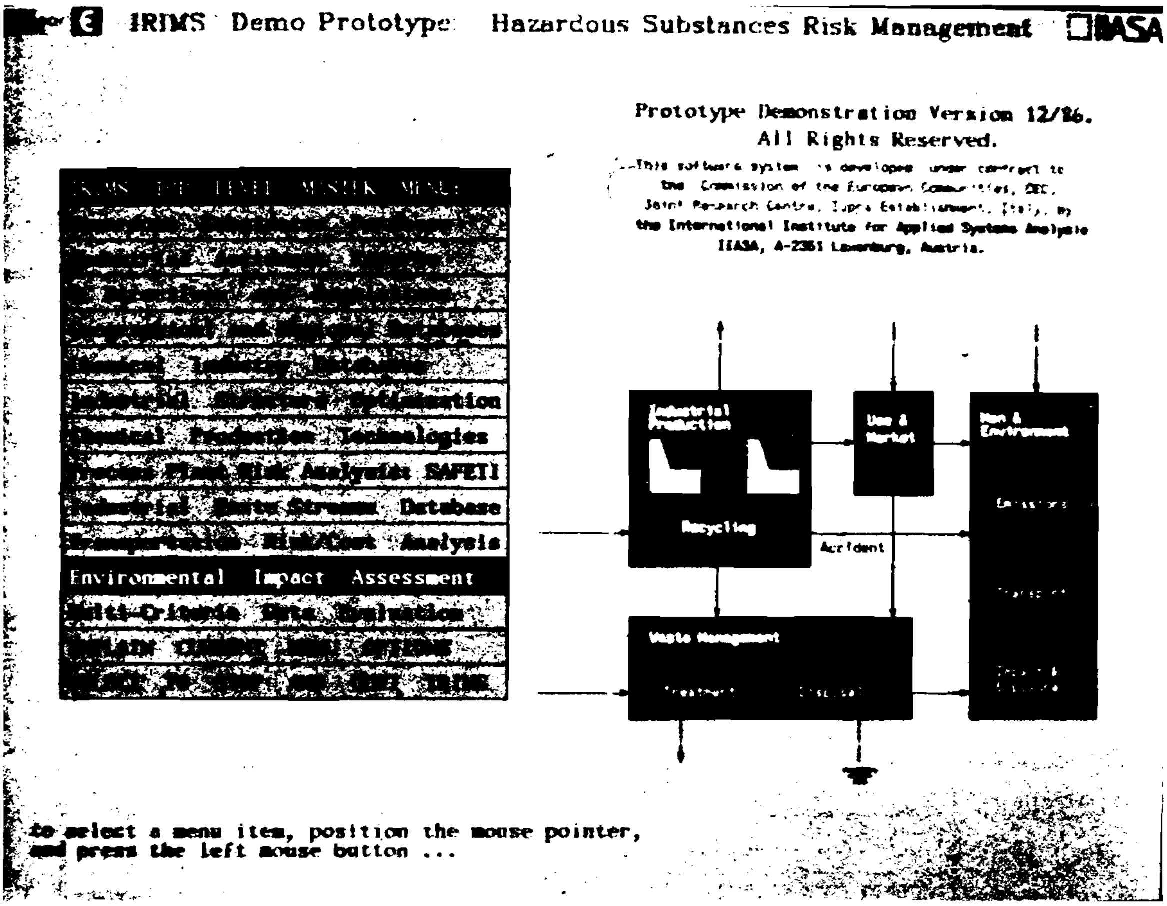

Figure 6: IRIMS top-level master menu

Industrial Accidents Reports: This database contains information on 'representative' industrial accidents in a picturebook-like form. It also provides cross-connections to the substance database and the regulations database.

EC Directives and Regulation: This database comprises directives and regulations of the EC related to hazardous substances and their production. It is accessible from many of the other modules for quick referencing.

Geographical and Regional Databases: This module provides, in a map-like form, informations on cities, country boundaries, important transportation ways (roads and the train-network) and a set of selected chemical plants and storage locations all over Europe.

Industrial Establishments Databases: This information system provides two major databases on industrial establishments and on storage locations which contain information about the produced or stored substances, the wastes and hazards emerging and offer cross-connections to the substances database, the waste streams database and the regulations and accidents databases.

Industrial Structure Optimization: Representing the *industry level* of the decomposition hierarchy of the industrial production system (see Zanelli, 1984) this optimization module performs linear optimization of the input/output behaviour of the aggregated elements of a pesticide industry.

Chemical Production Technologies: This information system allows the user to get detailed descriptions and symbolic simulation of selected chemical production technologies. This represents the system (Zanelli et al., 1984) or production process (Fedra et al., 1987) level of the decomposition hierarchy of the industrial production system and will be discussed in more detail below.

Process Plant Risk Analysis: SAFETI: This module provides an interface to the risk evaluation software package SAFETI (Technica, 1984).

Industrial Waste Streams Database: This information system contains detailed information on hazardous waste streams produced during the production of chemical substances. It provides numerous entry strategies (expert, non-expert, novice) and cross-connections to the substance database, the establishments database and the regulations database.

Transportation Risk/Cost Analysis: Based on the European transportation network (roads and trains) this module simulates the transportation of hazardous substances between two user-defined cities, so evaluates a number of different transportation alternatives and gives the user the possibility to find 'his' optimal alternative using an integrated multi-criteria evaluation package (DISCRET; Zhao et al. 1986). In addition cross-connections to the substance database and (implicitly) to the geographical and regional databases are provided.

Environmental Impact Assessment: This module provides the user with three interactive simulation models for river water quality, long-range atmospheric transport, and groundwater quality management simulation. All comprise cross-connections to the substance database, which is used by all modules to extract the substance specific physical and chemical properties of the substances the user wants the models to run with.

Multi-Criteria Data Evaluation: Similar to the multi-criteria evaluation packages included in the transportation module (but independent of a specific simulation module) this module represents a powerful post-processor for discrete optimization of model-generated sets of alternatives.

EXPLAIN CURRENT OPTIONS: This option invokes a general-purpose context-driven explain function, which is included in all menus of the system.

STOP AND QUIT IRIMS.

Design guidelines and the overall structure of the system have been described in *Advanced Decision-oriented Software for the Management of Hazardous Substances: Structure and Design* (Study Contract No. 2524-84-11 ED ISP A), and *Advanced Decision-oriented Software for the Management of Hazardous Substances: A Prototype Demonstration System* (Study Contract No. 2748-85-07 ED ISP A), as well as in Fedra (1985) Fedra and Otway (1986), Zhao et al. (1985), Fedra et al. (1986), Peckham et al. (1986).

6.2. The Artificial Intelligence Application Modules

Detailed descriptions are given below of those modules of IRIMS which are AI applications, i.e., based on AI paradigms and using AI techniques. The first section describes a cross-module tool, and what follows gives detailed information about modules which can be used as standalone packages and also as a part of the integrated software system.

6.2.1. Direct Access to Specific Information through Parsing

When the system is in information retrieval mode it is hardly sufficient for the user only to be able to browse through the data and knowledge bases, guided by a list of identifiers which refer to the information entities. Direct access to specific information is a must for every information system which does not want to exasperate or bore its users.

But if one considers for example (as is the case in conventional direct access interfaces) the names of hazardous chemical substances (for which there are several taxonomies which often use different names for the same substance) or industrial waste stream descriptors (which are often very long, because they represent the substances the waste streams consist of and the production processes which have produced the waste streams) to be used as keywords for direct access interfaces, it can become a very difficult task for the user to specify a valid keyword and to type it in correctly.

Therefore we allow direct access to specific information even if the user is only able (or willing) to type in a part or a synonym of the keyword that in conventional interfaces would be required to give the user access to the information to be retrieved. This is done by *parsing* the specification the user has typed in.

6.2.1.1. The Basic Concept of Parsing

The symbols the user types in are compared with a sequence of patterns each of which refers to one or more keywords which then allow direct access to specific information entities. To be able to match as many user specifications as possible, the patterns are designed to be very general. On the other hand this means that only very few patterns refer to only one keyword. Usually a list of keywords would be the result of the matches and direct access would again be impossible.

Therefore a weighting coefficient has been assigned to each pattern which corresponds to the *importance* of the pattern with respect to the keywords it refers to. This leads to a weighting of the keywords which represents the probability with which the keyword could be deduced from the user's input specifications. This usually suffices to select one unique keyword to enable the system to retrieve and display the required information, e.g., information about the waste stream that would conventionally have to be referred to as *Liquid organics from hexamethylenediamine production by 1,6-hexanediol ammonolysis* can be retrieved from **one** of the following user inputs (for the sake of clarity separated by slashes) *ammonolysis / hexamethylenediamine / liquid organics from amonono / hexamethyl organics /*.

If the probability evaluation still leads to a list of keywords, the user is prompted if he wants to choose from among the keywords, or if he wants to respecify the input, which would lead to a new parsing and probability evaluation sequence.

6.2.1.2. Background and Details

The concept briefly described above is based on two approaches to probabilistic reasoning in rule-based systems, where the following assumptions have to be made:

- the *database entries* the user wants to obtain via the keyword patterns after entering his subjective specification of the entry from the parser correspond to ***hypotheses*** of rule-based systems which are to be confirmed (or unconfirmed) by the application of rules together with the user's interaction;
- the *keyword patterns* which reinforce the probabilities of the database entries correspond to *rules* of a rule-based system which confirm certain hypotheses;
- the *reinforcement coefficients* which are assigned to the keyword patterns and represent the importance of a keyword pattern correspond to *rule values* which are assigned (or computed) in rule-based systems and represent the importance of the rules.

The first approach is based on (Forsyth, 1984; pp.63-85) where he proposes the following steps (among others which have not been incorporated into our approach because they deal with special cases we can *a priori* exclude) for probabilistic reasoning. Extracting five of the proposed ten steps:

(i) *For each hypothesis establish a prior probability. ... This is held as P(H) to be later updated.*

(ii) *For each item of evidence establish a rule value. ...*

(iv) *Interrogate the user on that item of evidence... The user's response...is the variable R.*

(v) *Given R, recalculate all hypotheses which referenced that item of evidence in their knowledge base to find the a posteriori probability p(H:R).*

(x) *Send the system into a summary routine during which it announces...the exact details of all of the inferences it has made.*

If we translate the above sequence of steps into the context of our problem area the background of our approach becomes apparent: First we establish a prior probability **P(H)** for each database entry which represents the probability for each entry that just this one will be referred to by the user (*Step (i)*). This P(H) can be used to represent the preferences of specific users with respect to specific databases. As long as the system is used as a demonstration system mainly (i.e., by different users) we always reset the P(H) to zero before parsing is done to avoid confusing the current user with the preferences of a prior user.

Then a reinforcement coefficient **RC** is assigned to each keyword pattern. For reasons of transparency we use the reciprocal number of references to database entries of a keyword pattern as its RC.

$$\mathrm{RC} = \frac{1}{\mathrm{Refs}}$$

RC ..... reinforcement coefficient of a keyword pattern

Refs ... number of database entries referenced by a keyword pattern

The choice of the P(H) and the assignment of the RC represent the *tuning* of the parser, i.e., these coefficients are mainly responsible for the adaptation of the parser to a specific application domain and user group.

After this setup of the basic coefficients the system is ready for operation. But instead of being interrogated about evidences (*Step (iv)*) the user is asked to input his, subjective, specification of the database entry he desires to use. Then the keyword patterns are matched against the character string the user has specified and each matching pattern reinforces the database entries he is referring to by its RC, i.e., the P(H:R)'s are calculated (*Step (v)*).

After all keywords patterns have been tried to match and the final P(H:R)'s are calculated, a summary routine sorts all database entries according to their P(H:E)'s and lets the user choose among the list of database entries with the highest probability. If the user has input a specification which results in only one database entry to have the highest P(H:E) rating, the database is automatically accessed using this entry and the user is provided with the contents of the database referred by this entry without any further action.

The only problem not discussed so far is how the P(H)s are updated by the RCs to become P(H:R)s, i.e., how the reinforcement of the probabilities of the database entries is done. This is the point where the second approach incorporated comes in: the updating of certainty factors in MYCIN (Buchanan and Shortliffe, 1984). Although (or even because) this is an heuristic approach we found it very well suited to our requirements. It works as follows:

$$P(H{:}R) = P(H_{old}) + (1 - P(H_{old})) * RC_R$$

$P(H{:}R)$ ...... a posteriori probability of the hypothesis that database entry H is to be selected after the keyword pattern R has been matched successfully with a substring of the user's specification

$P(H_{old})$ .... a priori probability of the hypothesis that database entry H is to be selected before another keyword pattern has been tried to be matched with the user's specification

RC_R ..... reinforcement coefficient of the keyword pattern R which has been successfully matched with a substring of the user's specification.

As is the case in this approach there is no disconforming evidence in the rather simple (but nevertheless successfully used in numerous applications) algorithm which is sufficient for the parser. (For a detailed discussion on the advantages and limitations of MYCIN's certainty factor approach see Buchanan and Shortliffe, 1984, chapters 10-13).

To improve the efficiency and user-friendliness of the parser the following feature has been added to the approach so far described: Before all keyword patterns are matched with substrings of the user's specification, the specification is matched with each database entry as a whole to determine if a 100 per cent equality can be found. If this is the case, then this entry is the one to be used for database access and no further parsing or user interaction is required. In the case of experienced users this speeds up the parsing significantly (Figure 7).

6.2.2. The Symbolic Production Process Simulator

One of the main modules of the simulation system of IRIMS is the Symbolic Process Simulator, which simulates product-oriented production processes. It is designed to enable non-expert users to get an idea of how certain products are produced and where the hazard lies during the production process.

Each production process consists of ***Unit Activities*** (Unit Processes [Herrick et al., 1979] and Chemical Processes) and ***Units*** (Zanelli et al., 1984), where the Unit Activity takes place. The combination of a Unit Activity and a Unit, which is necessary if the process is to occur, is called an ***Operating Unit***. In order to satisfy a special production goal the Operating Units are linked by their input/output streams (direct or indirect recursive as well). Some input streams of Operating Units are connected to external* input streams, and some output streams of Operating Units are the waste and product output streams of the whole Production Process Module (Figure 8).

The production process starts as soon as input material is provided to the Operating Units which are connected to the external input streams. These Operating Units perform their Unit Activities depending on the input materials, the operating conditions of the

*) i.e., from outside the Production Process Module

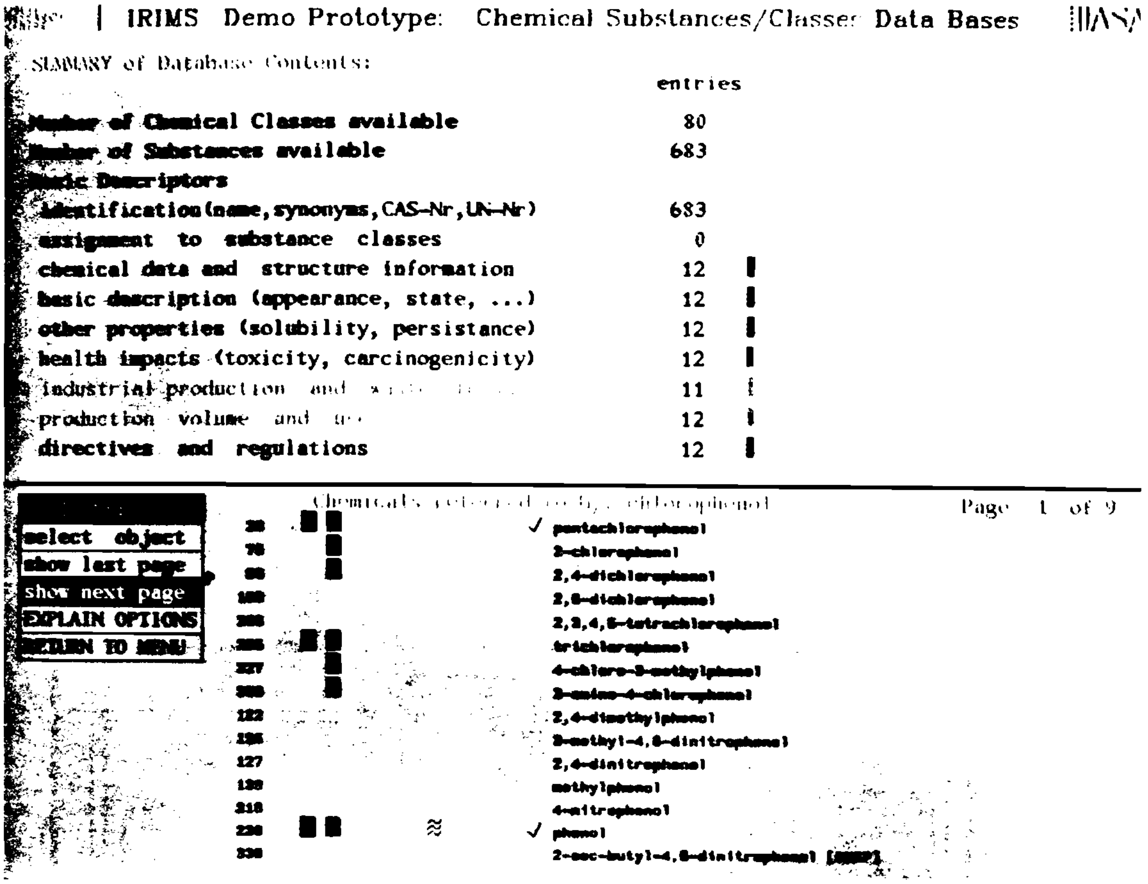

Figure 7: Parser screen of the chemicals database module of IRIMS

Unit and the constituents of the Unit, and by this produce some output material, which they send (via the linked input/output streams) to other Operating Units, which are activated on receiving input material. They too perform their Unit Activities and produce output, this activates other Operating Units and so on. After the production and the release of output material an Operating Unit is deactivated until it gets new input material. This sequence of activation and deactivation of Operating Units by materials terminates when there is no more input material for any of the Operating Units, e.g., all external input has been transformed to the desired products, byproducts and waste.

During the simulation of the production process the *Operating Hazards* of the Units and the hazards caused by the materials used and produced (e.g. input materials, interim products, end products, waste materials), the *Material Hazards,* are recorded and dynamically updated in the form of Hazard Ratings (NFPA, 1977; AICE, 1973; Sax, 1975).

The simulation is performed and controlled by forward chaining rules which operate on the simulation objects (the *Operating Units*).

6.2.2.1. The Components of the Symbolic Simulator

The symbolic simulator consists of ***knowledge bases***, which contain chemical expertise about Unit processes and Units as well as rules of control, and ***dynamic information tables*** (with dynamically instantiated simulation objects [Operating Units]) and an

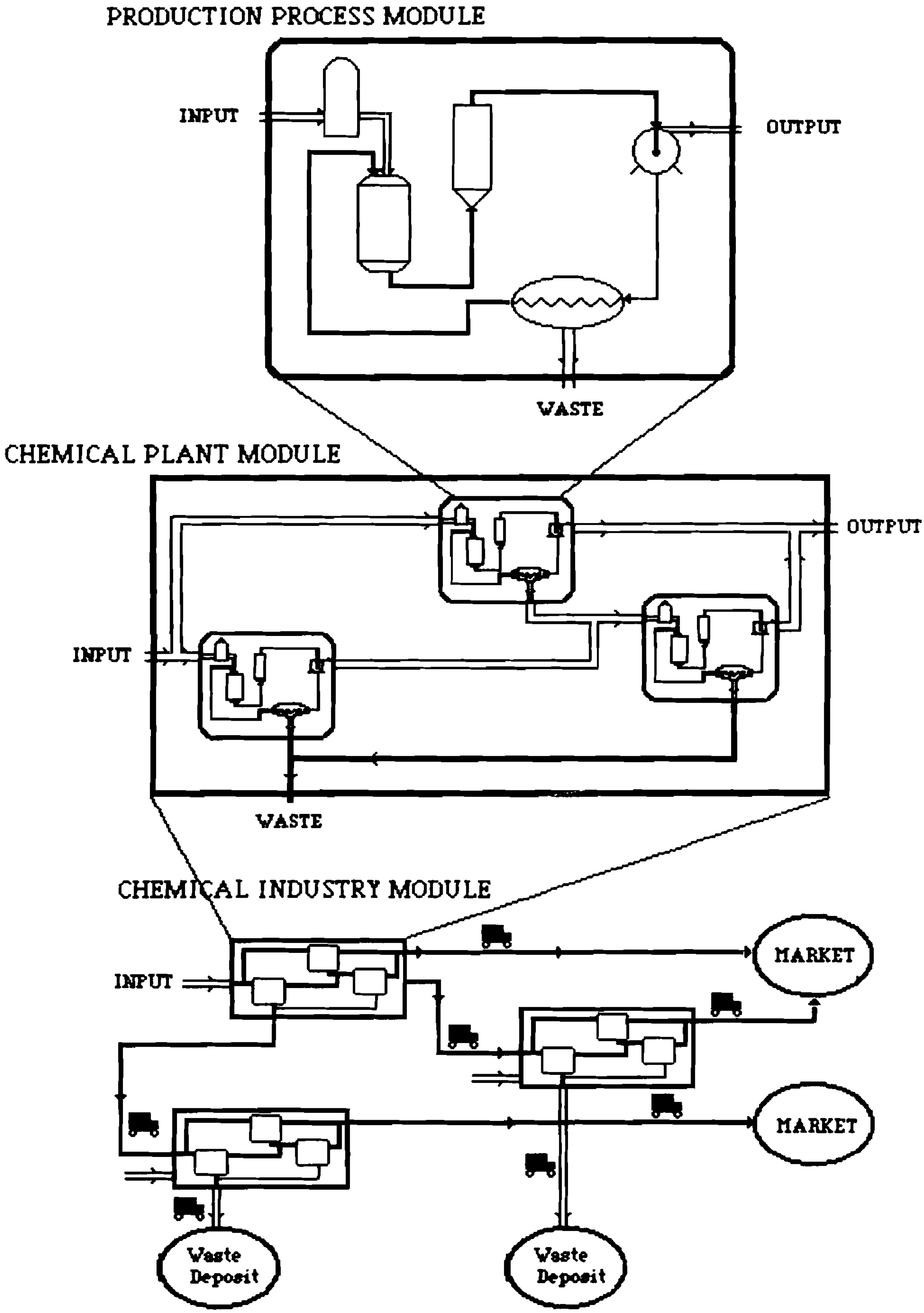

Figure 8: The production process module of IRIMS

inference engine which applies the rules of the knowledge bases and thus performs the simulation due to user requests.

a) Knowledge Bases

Unit Process KB:

All the rules required to simulate the Unit Processes of all production technologies provided in the production technologies database are stored here.

For our sample production process (chlorination of phenol) these rules comprise the following Unit Processes: *Halogenation* (8 rules), *Distillation* (2 rules), *Condensation* (3 rules), *Absorption* (1 rule), *Reflux* (2 rules) and *Pumping* (1 rule).

The rules represent input/output transformations under certain operating conditions of the simulation objects they are assigned to during a simulation run.

As an example, one of the *Halogenation rules* is listed below:

```
((IF ((in (Input (this Operating-Unit)) 'phenol)
    (in (Input (this Operating-Unit)) 'chlorine)
    (greater (temperature (this Operating-Unit)) '70C)
    (same (pressure (this Operating-Unit)) '1.3atm)))
((THEN ((material o_chlorophenol
        (From (this Operating-Unit))
        (To (port 0 (this Operating-Unit)))
        (Phase 'gas)
        (Hazard_ratings '(L L H H))
        (Status 'active))
    (material p_chlorophenol
        (From (this Operating-Unit))
        (To (port 0 (this Operating-Unit)))
        (Phase 'gas)
        (Hazard_ratings '(L L H H))
        (Status 'active))
    (material HCl
        (From (this Operating-Unit))
        (To (port 1 (this Operating-Unit)))
        (Phase 'gas)
        (Hazard_ratings '(L L H H))
        (Status 'active))
    (Status* '(phenol chlorine) 'inactive))))
```

In natural language:

If phenol and chlorine are supplied to the current operating unit and the temperature of the current operating unit is higher than 70°C and the pressure of the current operating unit is exactly 1.3 atm

then o_chlorophenol, p_chlorophenol and hydrogen chloride are produced in the current operating unit and sent to the operating unit which is connected with the current operating unit via the pipe starting at port 0 of the current operating unit and the status of phenol and chlorine is set to 'inactive' (i.e., they are marked as used up by this unit activity).

Unit KB:

This KB contains the information about the Units (i.e. the hardware) required to run all chemical production processes contained in the technology database of the framework system. The information covers the following properties: *Type of the Unit, Equipment description* (i.e. real hardware), *Operating conditions* (e.g., temperature, pressure,...), *Unit Activity* (i.e., the set of assigned production process rules - dynamically set by the appropriate Combining Rules) and *Operating Hazard Measurement* (i.e., the set of hazard rating rules for the dynamic evaluation of the hardware risk of the Operating Unit).

The Units required by our sample production process are: Stirred Batch Reactor, Batch Vacuum Distillation Column, Condenser, Codensing Trap, Absorption Tower, Recycle Pump, Reflux Drum and Flow Meter.

The Unit description of a *Stirred Batch Reactor* is as follows:

```
(Stirred_batch_reactor
  (Description:
  Type Stirred_batch_reactor
  Equipment_description
            (Features:
             ports (connected:)
             graphic_representation 'Reactor)
  Operating_conditions
            (Features:
             temperature 150C
             pressure 1.3atm)
  Unit_process nil
  Operating_hazard_measurement 'Reactor_Hazard_Rules))
```

In natural language:
The unit *stirred batch reactor* is described by its type which is 'Stirred_batch_reactor', its equipment, i.e. its ports (connections to other operating units via pipes, dynamically assigned by the production process rules) and its graphical representation on the screen, its operating conditions, i.e., the temperature and the pressure (currently supplied with default values), its unit processes (dynamically assigned by the combining rules) and the operating hazard measurement rules (in the current stage of development, only a placeholder for a rule package to be developed later).

Combining Rules KB:

Here all rules for combining Unit Processes with adequate Units–depending on various preference possibilities (for example: economic optimum, safety optimum, financial restrictions,...)–are included. These rules cause the selection of the appropriate Units (see Unit KB) for the Unit Processes that are used to perform the desired chemical production process and combine the Units with Unit Processes to create Operating Units (i.e., Instances of Unit Descriptions referred to by unique names with Unit Process rules assigned to the Unit_Process descriptor).
In case of our sample production process only seven combining rules are needed because the assignment of Units to Unit Processes used for the chlorination of phenol is unique.

Therefore the combining rules are as simple as the following one (although they may become extremely complicated in other cases):

```
((IF ((in (fact Unit-Processes) 'Halogenation)))
((THEN ((create Operating-Unit Stirred__batch__reactor Halogenation)
     (remove-fact Unit-Processes Halogenation))))
```

In natural language:
If one of the unit processes to be included in the simulation of the selected production process is 'Halogenation'
then a new operating unit will be created by combining the unit description of 'Stirred__batch__reactor' with the rule package of the production process 'Halogenation' and the fact that the production process 'Halogenation' is to be included is removed.

Production Process KB:

In this KB the process-specific rules of each implemented Production Process are compiled. These rules

- select the Unit Processes used for the simulation of the desired production process
- initiate the creation of the Operating Units by selecting the combining rules
- set up the linkage of the Operating Units by connecting the ports of the Units (see Unit KB, Equipment__description)
- distribute the external input materials using a set of input rules
- provide the default operating conditions for the Operating Units which enable a standard run of the production process
- activate Operating Units to which input material (external or from other Operating Units) has been sent, i.e. apply the Production Process rules of the Operating Units

For the sample production process 7 Production Process rules are implemented (3 to select the Unit processes, 3 to interconnect the Operating Units and one to activate the Operating Units to which input material has been sent).

As an example, one of the rules which select the required Unit processes depending on the desired products is shown below:

```
((IF ((same (fact Process) 'Chlorination__of__Phenol)
    (Or (same (fact Product) 'trichlorophenol__2/4/6)
       (same (fact Product) 'tetrachlorophenol__2/3/4/6))
    (empty (fact Unit-Processes))))
((THEN ((fact Unit-Processes
          ((Halogenation 2)
           (Distillation 2)
           (Condensing 1)
           (Codensation 2)
           (Absorption 1)
           (Reflux 2)
           (Pumping 1))))))
```

In natural language:
If the production process to be simulated is 'Chlorination_of_Phenol' and the desired product is either trichlorophenol_2/4/6 or tetrachlorophenol_2/3/4/6 and the unit processes which are to be used for the simulation are not already selected
then two halogenation processes, 2 distillation processes, one condensing process, two condensation processes, one absorption process, two reflux processes, and one pumping process are defined for the simulation of the production process.

Metarules:

To control the sequence of rule applications and to provide conflict resolution, Metarules are used. In case of our sample production process four Metarules have been implemented which can be used for other production process simulations as well. They schedule the application sequence of rule packages in four situations: start of the simulation, external material input, simulation of the production process and end of the simulation. An example of the Metarule for the start of the simulation is shown:

```
((IF ((not-empty Input-Table)
    (empty MIPT)))
((THEN (apply* Production_Process-Rules)
     (apply* Combining-Rules)
     (apply* Production_Process-Rules)
     (apply* Input-Rules))))
```

In natural language:
If there is input material to be supplied to the production process and there is no interim product produced so far,
then apply the following rule packages consecutively to the inference engine: Production Process rules, Combining rules, Production Process rules (again), Input rules.

b) Dynamic Information Tables

During a simulation run the simulation objects created and the deduced facts are represented by the following Dynamic Information Tables:

Operating Units:

This table includes all Unit Process/Unit combinations set up by the Production Process Rules via the Combining Rules. The Operating Units represent the simulation objects. They are the instances of the Unit KB (hardware) descriptions, i.e., there are values assigned to their slots and each Operating Unit.

Input Table:

In this table the descriptions of the the input materials which are required to run the chemical production process are included. They are external factors for the simulation and are automatically provided by the simulator to enable a standard simulation to be run.

Materials In Process Table (MIPT):

This table holds all materials (interim products) that are or were sent from one Operating Unit to another. The material descriptions consist of the name of the material (a unique name for each material; e.g., o-chlorophenol-1), the information from which to which Operating Unit the material is (or has been) sent, the phase of the material (gas, liquid or solid), the hazard ratings (high [H], medium [M] or low [L]) for relative pressure, flammability, toxicity and chemical burn risk and the status of the material (active, inactive) which indicates if the material-at the current stage of the simulation-is (*active*) or has been (*inactive*) present in the chemical production process.

A sample material representation (as created by the Unit Activity rules assigned to Stirred_batch_reactor-1) is as follows (also see the description of the Unit Activity KB above):

```
(o_chlorophenol-1
        (From (this Operating-Unit))
        (To (port 0 (this Operating-Unit)))
        (Phase 'gas)
        (Hazard_ratings '(L L H H)
        (Status 'active))
```

In natural language:
The currently viewed interim substance o_chlorophenol is referred to by its unique name 'o_chlorophenol-1', has been produced in the currently active operating unit, is to be transferred to the operating unit which is connected via port 0 of the current active operating unit, is in the gas phase, has low relative pressure, low flammability, high toxicity, high chemical burn risk, and its status is active.

Waste Table:

Here the information about all wastes produced during the production process is compiled. The descriptors and the internal representation are the same as for the materials in process and the wastes.

Products Table:

This table comprises all end products of the production process. The descriptors and the internal representation are the same as for the materials-in-process and the wastes.

c) The Inference Engine

To be able to apply the rules stored in the knowledge bases described above a so-called *inference engine* has been developed. An inference engine is a program (in this case a CommonLisp program) which evaluates premises of rules and generates the consequences if all the premises have been fulfilled.

The simulation is started by applying the Metarules, which control the scheduling of the rule packages by forwarding the name of the package to be applied to the *apply**-function* of the inference engine, which loads the referred rules from their knowledge base-if they are not already present-and hands rule after rule of the package over to the *rule-monitor* of the inference engine. The rule monitor evaluates the premises of the rule. If all premises have been fulfilled, the consequences of the rule are obtained (in other words *the rule is fired*).

A rule-package is applied as long as at least one rule of the package has been fired. When no more matching rules exist in a package the Metarules regain control and forward the next rule package to the inference engine. They are themselves applied by the same mechanism with the exception that the Metarule package is only once forwarded to the inference engine. When no more matching Metarule is found, no more inferences can be deduced and therefore the simulation is finished.

The Lisp code of the central part of the inference engine, the ***rule-monitor*** is shown below (in a simplified version):

```
(defun rule-monitor (rule)
    (let ((premise (get-premise rule))
        (consequence (get-consequence rule)))
     (cond ((member 'f (execute premise)) nil))
        (t (progn (execute consequence) 'Yes)))))

(defun execute (sequence)
    (mapcar 'eval sequence))
```

6.2.2.2. Performance

a) Starting up

Using one of the menu options of the technology database of IRIMS the user selects a specific production process (Figure 9).
Then he chooses the desired product from the products list of the database information displayed and activates the inference engine, which reads the rules for the specified production process from the Production Process KB (Figure 10).
Then the inference engine applies the rules for the selection of the unit processes. This in effect ensures that the rules of the selected unit processes are read from the Unit Process KB.

After this the rules for building the operating units are applied, which read and apply the rules for combining unit processes with adequate units from the Combining Rules KB, and create the operating units for the specified production process by adding the descriptions of the constituents from the Unit KB to the corresponding unit process rules.

Then the inference engine applies the rules for the linkage of the operating units, which connect the operating units by initializing the destination variables of the output rules of each operating unit.

After the set up is established the input distribution rules are activated, which then read the input material descriptions from the Input Table, give each of them a hazard rating (if not provided in the Input Table description), the description of its source (e.g., "external input") and its destination (an operating unit name), set the status to "active" (i.e., currently in the production process) and write this extended material description on the Materials In Process Table (MIPT).

b) The Iteration Sequence

After the start up of the simulation environment (drawing of the layout graphics and the display of the database information (also see Figure 10) the user is given the possibility to simulate the process either in ***single-step*** mode, i.e., the simulation is stopped after each step to let the user choose if (and in which mode) he wants to proceed or if he wants to terminate the simulation immediately (Figure 11), or in ***continuous mode***, i.e., the simulation is not stopped until the user interrupts by pressing a mouse button (he is then also shown the dynamic display menu) or it is finished.

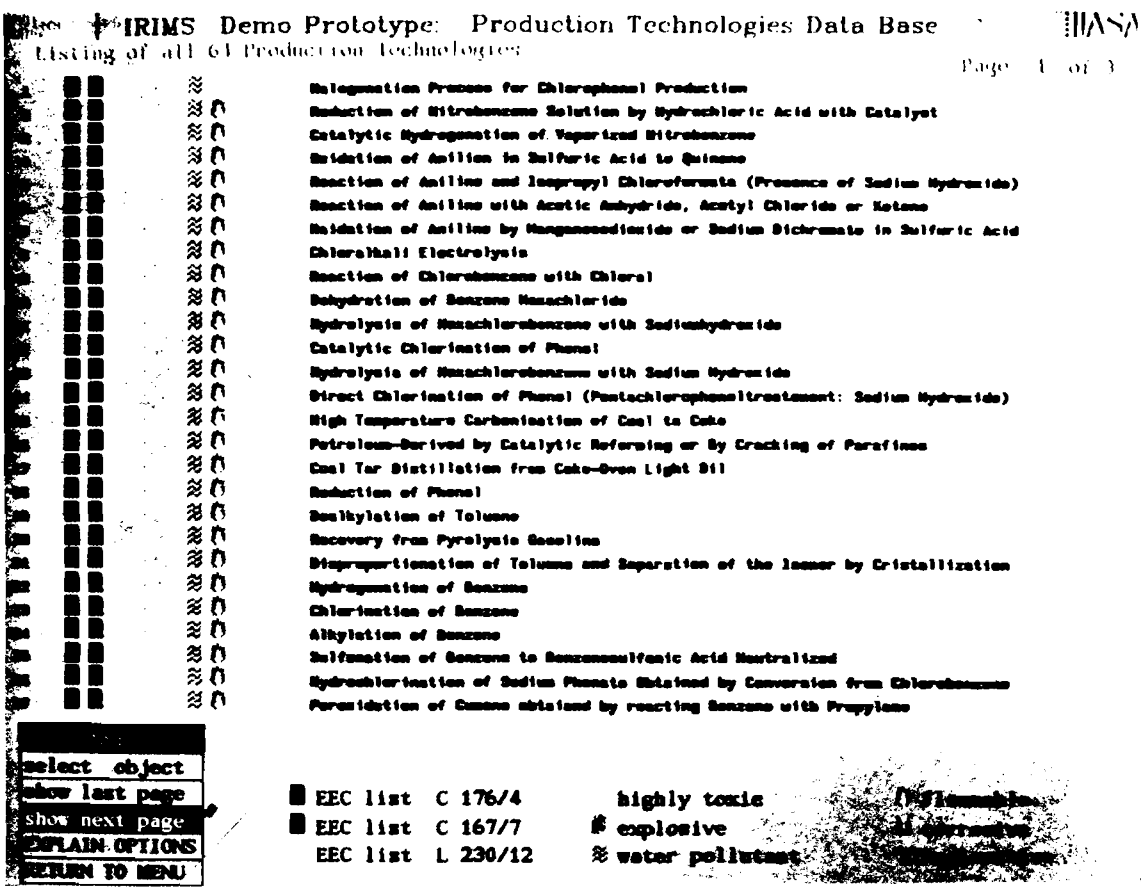

Figure 9: Top-level screen of the technology database system of IRIMS

Once the user has selected the simulation mode the inference engine reads the descriptions of the "active" materials from the MIPT, sets their status descriptor to "inactive" (i.e., had been in the production process) and activates (i.e., applies the rules of) the operating units, which are mentioned in the destination descriptions of the former "active" materials. Before the unit process rules of the activated operating units are applied, the rules for setting the operating conditions of the activated operating units are applied.

When the operating conditions are set–in future versions of the symbolic simulator with the active interaction of the user as well–the following occurs for each activated operating unit:

> The unit process rules are applied, which transform the input material descriptions to output material descriptions, depending on the operating conditions of the operating unit. The output material descriptions are then written on the MIPT (with the status descriptor "active"), to the Waste Table or to the Products Table, depending on the values of the destination variables of the output rules of the operating unit. Their material hazard rating is written to the Hazard Table as well, connected to the operating hazard estimation of the operating unit which produced the specific material. After this the operating unit is deactivated.

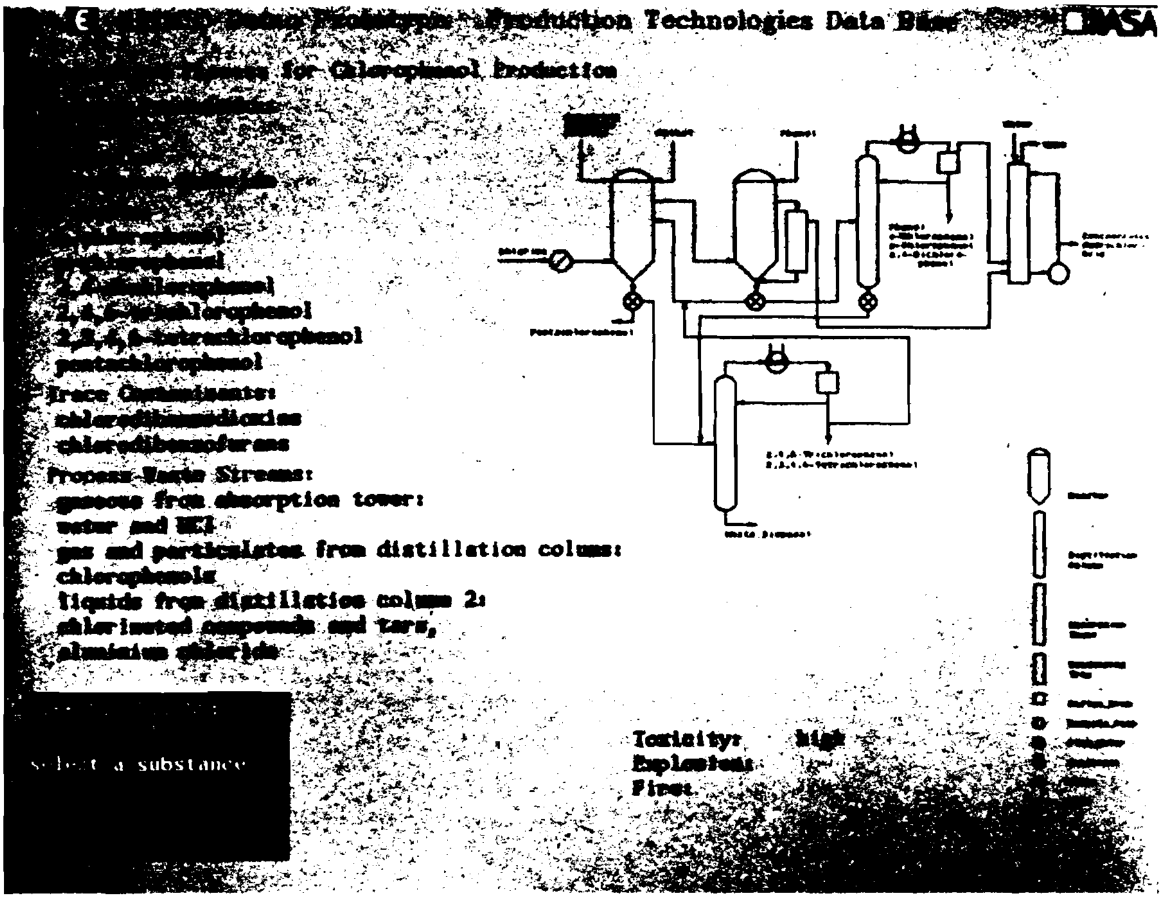

Figure 10: Product selection screen of the symbolic production process simulator

The new material descriptions on the MIPT are read by the inference engine and the sequence described above is repeated with new material descriptions, new active operating units, and so on.

When no further "active" material descriptions can be found in the MIPT by the inference engine, then there are no more operating units to be activated and the simulation of the production process ends.

6.2.3. The Heterarchical Object-oriented Information System on Hazardous Substances

Whenever any of the models in the simulation system of IRIMS is used, it is used for a given substance, substance group, or mixture of substances and substance groups. The classification of substances and substance groups, and the linkage between these groups and the physical, chemical, and toxicological properties of the substances are of critical importance.

With about 70,000 to 100,000 chemical substances on the world market, and about 1000 added to this list every year, any attempts at a complete or even comprehensive coverage within the framework of this project are illusory. Rather, we must provide information about a *representative subset* with an access mechanism that accounts for the ill-defined structure resulting from all the chemical nomenclature, trivial and trade names, and attribute-oriented cross-cutting groupings (e.g., oxidizing substances, water soluble toxics, etc.).

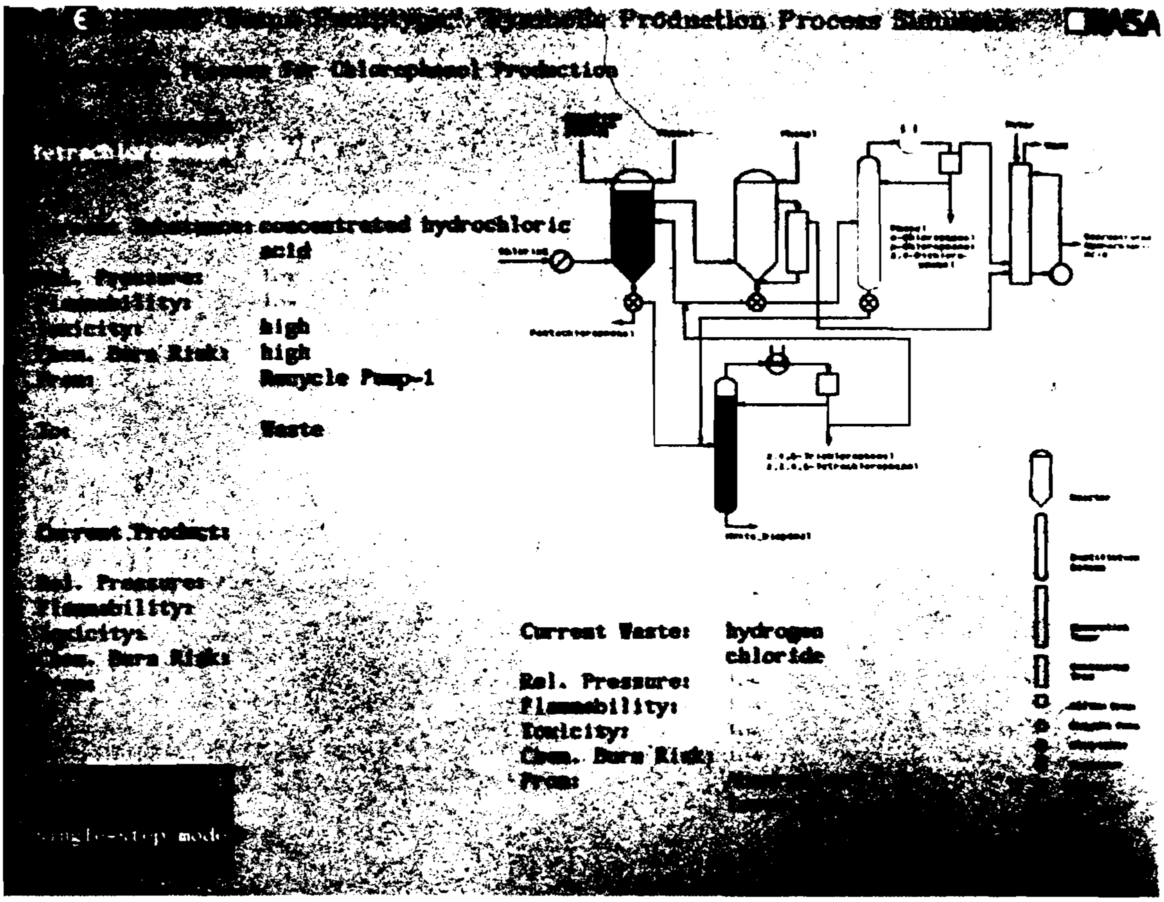

Figure 11: Symbolic production process simulator: dynamic display

The starting point for any attempts at classification is thus not organic chemistry or environmental toxicology, but a reflection on likely ways to formulate a problem. Entry points for substance identification are therefore *type of use* (e.g., agricultural chemical: pesticide) or industrial origin, i.e., production process or type of industry, implying an industrial waste stream (e.g., metal plating, pesticide formulation; a listing of 154 industrial waste streams that contain hazardous components is included in the EPA's WET model approach (ICF, 1984a,b)) rather than chemical taxonomy.

A common characteristic of all the elements in the information system on hazardous chemical substances (Figure 12) is the user interface: access to data and knowledge bases is through an interactive, menu-driven interface, that allows easy retrieval of the information stored without the need to learn any of the formal and syntactically complex query languages required internally.

6.2.3.1. Hybrid Knowledge Representation

Due to the diverse nature of the information required, we have chosen a hybrid approach to data/knowledge representation, combining traditional database structure and management concepts (e.g., relational databases) with knowledge representation paradigms developed in the field of AI. While most of the "hard", and often numerical or at least fixed-format data, are organized in the form of relational databases (using a relational database system developed at IIASA, see Ward, 1984), the knowledge bases again use a hybrid representation approach.

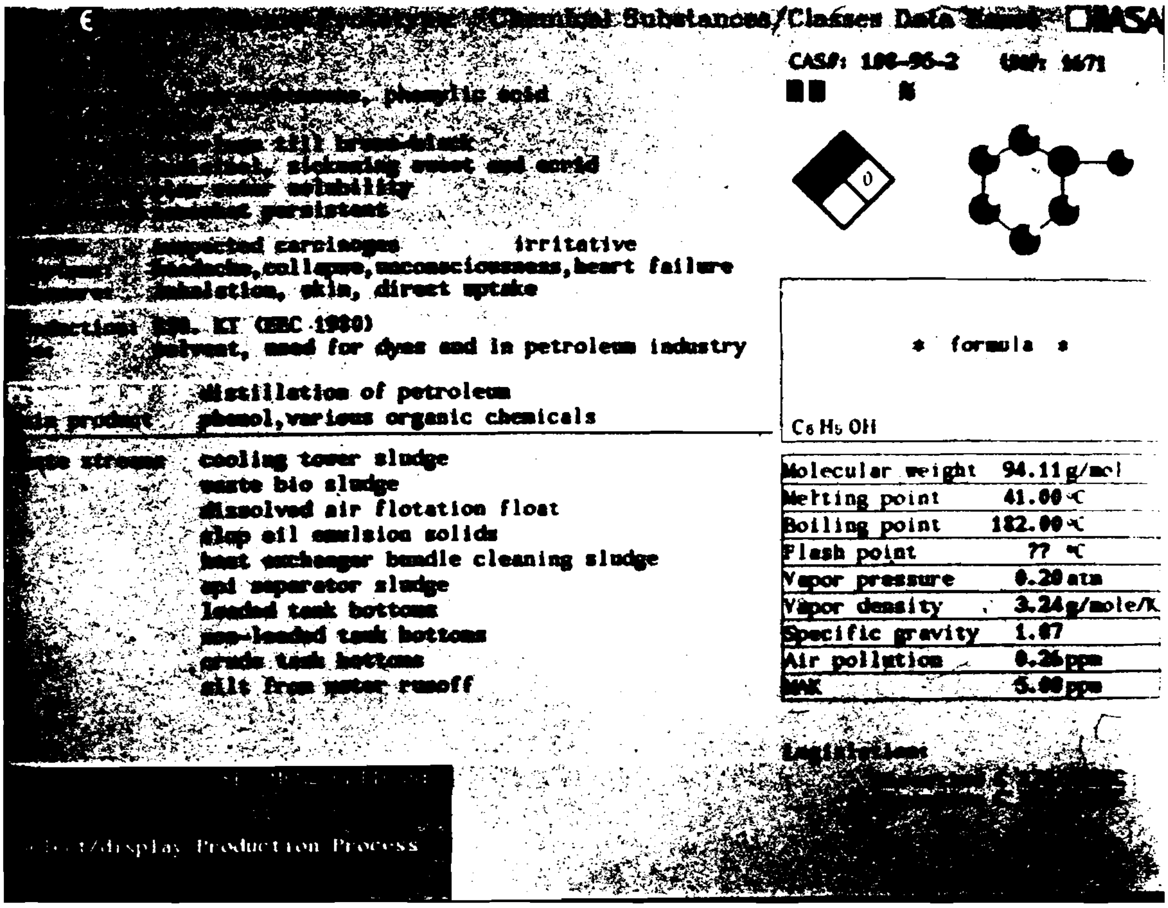

Figure 12: The hazardous substances database of IRIMS

Hybrid Knowledge Representation implies that within our information system, multiple representation paradigms are integrated. A knowledge base might therefore consist of term definitions represented as frames, object relationships represented in predicate calculus, and decision heuristics represented in production rules.

Predicate Calculus is appealing because of its general expressive power and well-defined semantics. Formally, a predicate is a statement about an object:

((property_name) (object) (property_value))

A predicate is applied to a specific number of arguments, and has the value of either TRUE or FALSE when applied to specific objects as arguments. In addition to predicates and arguments, Predicate Calculus supplies *connectives* and *quantifiers*. Examples for connectives are AND, OR, IMPLIES. Quantifiers are FORALL and EXISTS, that add some inferential power to Predicate Calculus. However, for the purpose of building up a representation structure for more complex statements about objects, Predicate Calculus representation becomes very complicated and clumsy, therefore in our system it has been integrated only to represent internal facts used by the inference module.

In *Object-oriented representation* or *frame-based knowledge representation*, the representational objects or *frames* allow descriptions of some complexity. Objects or classes of objects are represented by *frames* which form a hierarchy in which each object is a member of a class and each class is a member of a superclass (except the top-level classes). A frame consists of *slots* which contain information about the attributes of the objects or the class of objects it represents, a reference to its superclass and references to its members and/or instantiations, if it is a frame that represents a class. Frames are defined as specializations of more general frames, individual objects are represented by *instantiations* of more general frames, and the resulting connections between frames form *taxonomies*. A class has attributes of its own, as well as attributes of its members. An object *inherits* the member attributes of the class of which it is a member. The inheritance of attributes is a powerful tool in the partial description of objects, typical for the ill-defined and data-poor situations the system has to deal with.

A third major paradigm of knowledge representation are *production rules (IF - THEN decision rules)*: they are related to predicate calculus. They consist of rules, or *condition-action pairs*: "if this conditions occurs, then do this action". They can easily be understood, but have sufficient expressive power for domain-dependent inference and the description of behavior.

To combine the benefits of an object-oriented approach with those of condition-action pairs, a heterarchical frame structure for the chemical data and knowledge bases is being developed in CommonLisp, i.e., an object can be a member of *several* classes and each class can belong to several superclasses, and by adding "rule abilities" to a special slot called *actions*, i.e., this slot does not store information but performs *procedural tasks* which are defined as condition-action pairs. A detailed description of the heterarchical frame-structure is given below.

6.2.3.2. Heterarchical Structure for Information Management

Our approach foresees the use of a basic list of about 700 *substances* (or molecular substances, i.e., entities that do not have any sub-elements), constructed as a superset of EC and USEPA lists of hazardous substances. In parallel we construct a set of *substance classes* which must have at least one element in them. Every *substance* has a list of properties or attributes; it also has at least one *parent substance class* in which it is a member. Every member of a group inherits all the properties of this group. In a similar structure, all the groups are members of various other parent groups (but only the immediate upper level is specified at each level), where finally all subgroups belong to the top group *hazardous substances*.

While attributes of individual substances are, by and large, numbers (e.g., a flash point or an LD_{50}), the corresponding attribute at a class level will be a range (flash point: 18-30°C) or a symbolic, linguistic label (e.g., toxicity: very high).

The structure outlined below also takes care of unknowns at various levels within this classification scheme. Whenever a certain property is not known at any level, the value from the immediate parent_class (or the composition of more than one value from more than one immediate parent_class) will be substituted. The structure is also extremely flexible in describing any degree of partial overlap and missing levels in a hierarchical scheme (Figure 13).

6.2.3.3. Frame Syntax:

Each class-frame consists of the following six slots:

- **Explanation:** verbal information about the current frame, concerning the substance class which is represented by the frame, its attributes, the default values and/or indirect references and the position of the frame in the heterarchical structure; the main purpose of this information is for updating and editing the frame by a knowledge engineer
- **Superclasses:** references to the classes to which the current frame belongs

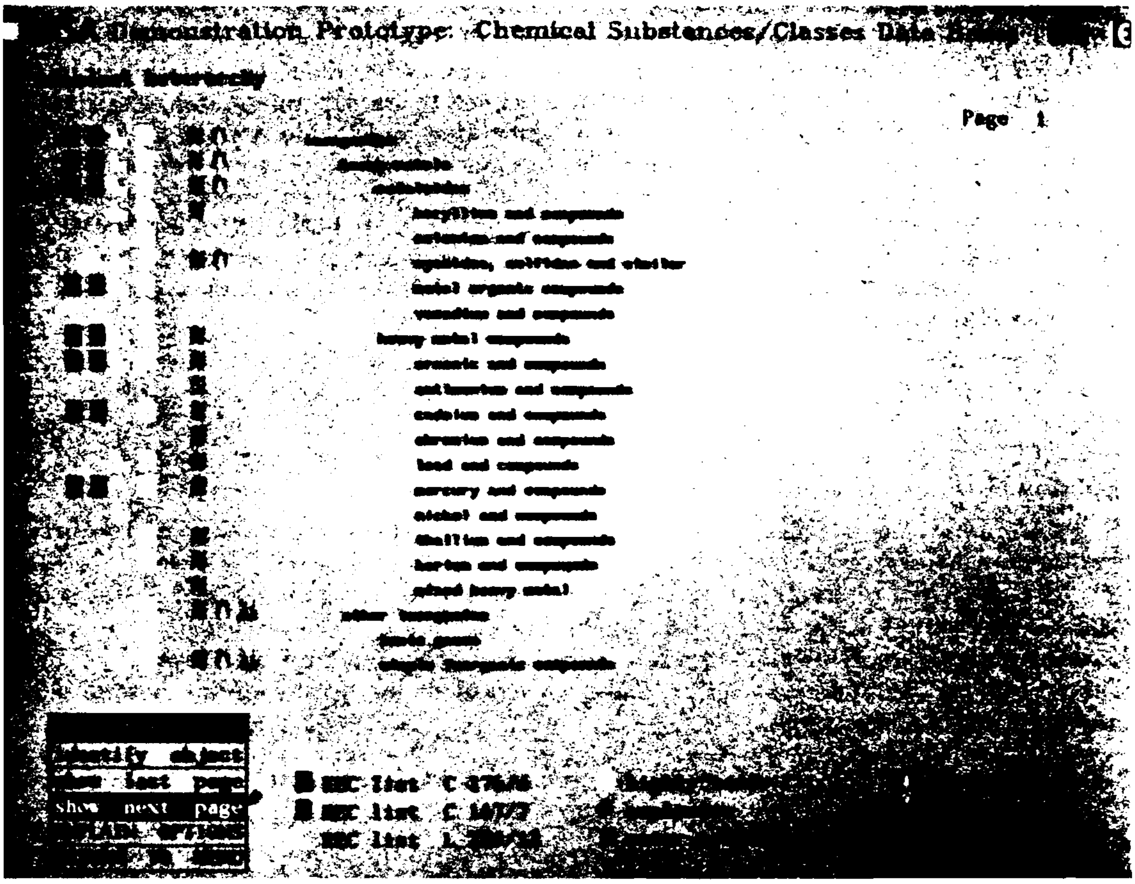

Figure 13: Listing of the heterarchical information structure

- **Description:** attributes with values and/or procedural attachments (i.e., procedures which calculate the values or refer to them or both) which describe the substance class represented by the current frame
- **Subclasses:** references to the classes which belong to the current frame
- **Instances:** references to the instances (i.e., substances) of the substance class represented by the current frame
- **Actions:** condition-action pairs, where the actions of an action part are carried out if the frame receives a message which matches the corresponding condition pattern.

The formal description of a frame is as follows:

```
(Class    classname
          (Explanation    (<Verbal Information>))
          (Superclasses   (<List of Classnames>))
          (Description    (
                          (<Slotname>-2 <Filler>-2)
                          ...
                          (<Slotname>-n <Filler>-n)))
          (Subclasses     (<List of Classnames>))
          (Instances      (<List of Substances>))
          (Actions        (If <Condition Pattern>
                           Then <Action Part>)))
```

<Verbal Information> = InfoText represented by a list of words

<List of Classnames> = classname | classname <List of Classnames>

<List of Substances> = substance | substance <List of Substances>

<Slotname> = attribute of the represented class

```
<Filler> =     (Class classname) |
               (Value value) |
               (default (Lisp s-expression)) |
               ($if-needed (Lisp s-expression)) |
               ($if-added (Lisp s-expression)) |
               ($if-changed (Lisp s-expression)) |
               ($if-deleted (Lisp s-expression))
```

<Condition Pattern> = (<Pattern>-1 <Pattern>-2 ... <Pattern>-m)

<Pattern> = constant | ?variable | #

<Action Part> = (Lisp s-expression)

As an example, two class-frames from the heterarchical knowledge base structure for phenols are given below:

(Class **aromatics**:

```
 (Superclasses   (Object))
 (Description    (attribute-1 .....)
                 (attribute-2 .....)
                 ...
                 (attribute-n .....))
 (Actions        (If (List your members)
                  Then (prog (ask self subclasses) (ask self instances))))
 (Subclasses     (aromatic_hydrocarbons aromatic_heterocyclics))
 (Instances      NIL))
```

(Class **mixed_hydrocarbons_substituted_with_two_chlorines**:

```
 (Superclasses   (aromatic_hydrocarbons_double_substituted
                  chlorinated_phenol
                  mixed_chlorinated_aromatic_hydrocarbons))
 (Descriptions   (attribute-o+1 .....)
                 (attribute-o+2 .....)
                 ...
                 (attribute-p .....))
 (Subclasses     NIL)
 (Instances      (2,4-Dichlorophenol 2,6-Dichlorophenol)))
```

6.2.3.4. Information Retrieval

To retrieve information the user may directly enter the name of a substance or of a substance class, or he may specify value ranges (numerical and/or symbolic) for one or more substance (class) attributes. The Information System transforms this specification into messages for the top-level classes (also called viewpoints). On receiving these messages, the frames which represent the viewpoints are activated.

The frames then check if they are selected by the user's specification, and if they are, then proceed to create messages for their subframes which again perform their matching operations and create messages, and so on. This recursive procedure does not need to search through the whole structure because it is directed by the rules in the *Actions* slots and the references in the *$if-* slots, supported by the inherited information.

This procedure results in a substructure of valid substance classes which represents the systems' view of the user's level of expertise. This substructure is from then on used to guide the user to the more detailed information, if he wishes to proceed with the interaction.

This recursive message sending and receiving can be applied again, starting from the current top level(s) of the substructure using the additional information provided by the user (based on the displayed status of the attributes of the classes level reached), until either the user is satisfied by the given information about the current substructure's attributes or until the level of instances (a single substance) terminates the user's attempts to get further information.

6.2.3.5. Updating the Knowledge Base

Updating is quite similar to the retrieval of information. First, the substructure which will be affected is localized by an interframe message sending/receiving sequence. Then the updates of the attribute values are entered and checked if they are consistent within the selected substructure by using the *$if-added* and the *$if-needed* slot fillers together with the rules of the *Actions* slot which deal with consistency tests.

After the consistency of the substructure has been proved, the same procedure is used for the next higher aggregation levels until the whole structure has been proved to be consistent with the new and/or changed attribute values.

6.2.3.6. Achievements

The heterarchical frame-based information system allows expert and non-expert users to retrieve general and detailed information, starting with the individual level of experience of the user, leading him down to the detailed information he might require, for example, to run a simulation model dealing with the environmental impact assessment of hazardous substances.

This is done by reducing the information structure of frames step by step to the current level of interest, because the more detailed the information provided by the user, the smaller the substructure for further search procedures and the better the performance of the information system will be.

Good system's performance is also guaranteed by the reduction of the search/match processes by using internal references instead of redundant value specifications and mainly by the search-directing rules, which are part of every frame.

The whole process of information retrieval and updating is carried out by the individual frames, which are activated by receiving messages, performing their tasks and finally sending messages to other frames.

The system also has an automatic consistency checking feature built in so that when updated it provides a flexible structure which can easily be extended, reduced or generally changed.

6.3. Automatic Learning–A Claim for the Future

The major bottleneck in the development and widespread use of expert systems is *knowledge acquisition*, i.e., the transfer of domain-specific knowledge from the human expert to the machine. Supporting knowledge acquisition by the machine itself, *automatic learning*, could (drawing on the knowledge and expertise of the users themselves):

- speed up the task of developing domain-specific expert systems, by providing a major productivity tool for knowledge engineers and systems developers;
- continuously improve performance through knowledge gained during the system's use and automatically incorporated into its knowledge bases;
- improve the user interface by learning about users and automatically adjusting to individual styles of interaction.

6.3.1. Background and State of the Art

As mentioned earlier, AI aims at developing computer systems that do things that would require intelligence if done by men. Fields of research include understanding natural language, robotics, computer vision, expert systems and machine learning.

We want to combine two of these fields–expert systems and machine learning–which have so far only been combined with overwhelming emphasis on the expert system part (see EXPERT-EASE, TIMM in Harmon and King, 1985).

The lack of extensive learning capability of most expert systems is not only caused by the current boom of expert systems development, but also by the fact that most "learning systems" either represent:

- *learning by being programmed,* as in every computer program;
- *learning by being told,* the usual knowledge acquisition components of expert systems (see TEIRESIAS in INFOTECH, 1981a and in Davis and Lenat, 1982), and expert system shells (see EMYCIN, ART, KEE in Harmon and King, 1985);
- experimental systems dealing with *learning by examples and discovery* (see Winston, 1975; Michalsky and Chilausky, 1980; Oakey and Cawthorn, Rissland and Soloway, Sammut, Selfridge, all 1981; Shapiro, 1982; Winkelbauer, 1984; Korf, 1985).

For our purposes the above terminology can be reduced to two levels: *standard knowledge acquisition modules* and *learning systems based on learning by examples.*

The standard knowledge acquisition modules normally do not produce any new knowledge. They reduce the burden on the user who builds and/or modifies the system's knowledge by performing context-sensitive tasks, which are e.g., rule and fact representation, storage and/or reorganization of the knowledge, consistency checks, etc.

Most of the above systems which are based on *learning by examples* consider only positive examples, i.e., they are built on the assumption that all examples which are given and/or discovered support the knowledge acquisition process by providing additional information to add to the knowledge gained so far. No negative examples are allowed.

We aim at the development of a truly combined system performing context-sensitive knowledge acquisition tasks with machine learning techniques based on *learning by examples* considering negative examples as well, in order to produce negative feedback to the already learned knowledge and allow limit estimations for the generality of the learned knowledge, to improve the link between machine-held knowledge representation in an expert system and the human expert, who builds up the knowledge base or consults the system.

6.3.2. Objectives and Approach

Within the framework of expert systems development, automatic learning aims at quantitative and qualitative performance improvement over more rigid, traditional approaches. An important prerequisite for automatic learning concepts of practical applicability is their development within a realistic, problem-oriented framework, i.e., an operational, domain-specific expert system.

The ongoing expert systems development study (Fedra, 1986), provides real-world material and application-oriented testing ground. The methodological research on machine learning will concentrate on:

- learning by examples in rule-based systems,
- metarule development and application strategies,
- and consistency checking of knowledge bases.

As the central paradigm for software development, *learning by examples* will be taken to enable the learning system to acquire knowledge about various problem domains (and also about the user) directly from model- or user-generated examples. Based on concepts of pattern matching, similarities, and "gestalt", the examples will be transformed to rules from which generalized knowledge about the problem area will be derived in the form of higher-order rules. The generalization process can be repeated recursively for the next higher level of rules, as increasingly general views of the problem domain are constructed.

The knowledge the system holds about the problem areas will automatically be extended quantitatively as much as qualitatively, on the basis of the systems' functioning and use, i.e., by incorporating solutions from optimization programs. A rich model and knowledge base, that is also constantly enlarged through use of the system, holds an infinite number of combinatorial possibilities of "new" deduced knowledge and ways to organize this knowledge. Automatic learning can also help to get this continuous stream of information filtered and organized.

To add the acquired and the deduced or internally generated knowledge to the existing knowledge bases, a consistency-checking mechanism for knowledge bases will have to be developed to ensure the reliability and practical "common sense" usefulness of the generalized knowledge.

A coherent strategy for the integration and application of the learned knowledge in an expert system (e.g., performance improvement, automated knowledge acquisition, a self-adjusting user interface) is required. Therefore, the development of metarules, i.e., rules that govern the application of the problem-oriented knowledge of the system (learned or acquired from the user) is another important aim of the proposed research work.

By being run under interactive control, the system can learn from the user how to run, i.e., control in terms of e.g., operating temperatures, pressures, and feedstock ratios, a given set of production processes. Considering possible trade-offs between process risk and efficiency, the system can develop a knowledge base for the operational control of the entire production process, including shut-down and emergency relief operations.

The heterarchical information system on hazardous chemical substances (see section 6.2.3) and a rule-based diagnostic identification procedure for hazardous waste and appropriate waste treatment technologies are possible application areas for the development of metarules. A realistic system must handle cases of incomplete information at various levels of aggregation, substance classes and mixtures, etc. Since rules are more easily and obviously more precisely formulated at a single substance level, or at a high degree of disaggregation, developing metarules for higher-order aggregates, in particular in the diagnostic substance identification procedure, could greatly enhance the overall usefulness of the system.

For the chemicals database, it is necessary that the information management system is able to gain knowledge *about* its knowledge on hazardous chemical substances, i.e., to generate metarules, which contain information on the various possibilities of how chemical substances can be grouped and described. Since this grouping, or the possible entry points to this large body of information, are problem- and user-specific, an adaptive learning strategy seems by far preferable to a fixed thesaurus type strategy.

When the expert system is used to assist judgements on the impacts of a chemical substance on public and environmental health, it has to consider a large variety of legal and regulatory information, chemical and toxicological data, as well as institutional or personal preferences and perceptions. While user-generated preferences and perception-based assessment components are indispensable for any politically meaningful assessment, descriptors deduced from such sources and incorporated into the permanent knowledge bases will have to be checked for consistency with the existing regulations, directives and health standards.

In general, the interactive approach provides for numerous user-defined input values. Ultimately, they all have to be checked for feasibility. Since, in a realistic system of only moderate size, their number will be very large, and many of them will be symbolic and at a relatively high level of aggregation, it is impossible to keep feasibility bounds for all possible variables and input forms, especially because they are highly interdependent. Rather, it will be necessary to deduce such feasibility bounds from a set of hierarchical and context-oriented rules, which again should be adaptive and responsive to the system's use.

7. Conclusion

Expert systems are emerging as a new generation of software that holds great promise in shaping new information technology. Among their major potential are new ways to tackle extremely complex and "soft" problems that defy classical formalization through the use of e.g., heuristic approaches, as well as the extension of the group of potential users through new dimensions of user friendliness.

Application- and problem-oriented, rather than methodology-oriented systems, are most often *embedded systems*, where elements of AI technology are combined with more classical techniques of information processing and approaches of operations research and systems analysis. Here traditional numerical data processing is supplemented by symbolic elements, rules, and heuristics in the various forms of knowledge representation.

There are numerous applications where the addition of a quite small amount of "knowledge" in the above sense, e.g., to an existing simulation model, may considerably extend its power and usefulness and at the same time make it much easier to use. Expert systems are not necessarily purely knowledge driven, relying on huge knowledge bases of thousands of rules. Applications containing only small knowledge bases of at best a few dozen to a hundred rules can dramatically extend the scope of standard computer applications in terms of application domains as well as in terms of an enlarged non-technical user community.

Most of today's expert systems are mainly methodologically oriented and tend to underemphasize the role of the decision maker. Generally speaking, expert systems are not supposed to substitute human expertise by a program. They should be advisory programs that bring expertise to the user, who can then use his own expertise and experience to recognize patterns and symptoms, recall history, and exercise judgement.

8. References and Selected Bibliography

A-Razzak, M., and Hassan, T. (1984) EX-TRAN 7 User Manual. Glasgow: Intelligent Terminals Ltd.

A-Razzak, M., Hassan, T., and Pettipher, R. (1984) EX-TRAN 7 (Expert Translator): a FORTRAN-based software package for building expert systems. Research and Development in Expert Systems. Proceedings of the Fourth Technical Conference of the British Computer Society Specialist Group on Expert Systems, University of Warwick 18-20 December 1984, Cambridge, pp. 23-30

Abramson, B. (1984) Applied AI at Digital, The Artificial Intelligence Report, pp. 3-6

AICE (1973) Dow's Safety and Loss Prevention Guide, Hazard Classification and Protection. American Institute of Chemical Engineers, New York

Aikins, J.S. (1980) Prototypes and Production Rules: A Knowledge Representation for Computer Consultations, Report Number STAN-CSD-80-814, Computer Science Department, Stanford University

Allen, J.D.,Jr., Kelly, B.A., Johnson, C.K., and Oakes, T.W. (1982) Spills Problem and Applied Artificial Intelligence, UCC-ND and GAT Environmental Protection Seminar

Balogh, K., Futo, I., Labadi, K. (1977) The documentation of a PROLOG program verification system, NIM IGUSZI report, Hungary

Balogh, K. (1978) On an interactive program verifier for PROLOG programs, in: Mathematical Logic in Computer Science, B. Dömölki and T. Gergely [eds.] Coll. Math. Soc. J. Bolyai, Vol. 26 Budapest, pp. 111-142

Balzer, R., Erman, L., London, P., Williams, Ch. (1980) HEARSAY-III: A Domain-Independent Framework for Expert Systems, in Proceedings of the First Annual National Conference on Artificial Intelligence, AAAI, Stanford University, pp. 108-110

Ban, P., Köhegi, J., Suhai, Gy., Veszpremi, A., Zsako, L. (1979) Some questions of the information system of ANSWER, ELTE report, SOFTTECH D38, SZAMKI, Hungary

Banares-Alcantara, R. (1982) Development of a Consultant for Physical Property Predictions, Master's Thesis, Dept. of Chemical Engineering, Carnegie-Mellon University, May

Banares-Alcantara, R., Westerberg, A.W. and Rychener, M.D. (1985) Development of an expert system for physical property predictions, in Computers & Chemical Engineering, Vol. 9, No. 2, pp.127-142

Banares-Alcantara, R. (1985) DECADE: Design expert for catalyst development, PhD thesis, Carnegie-Mellon University, Dept. of Chemical Engineering. In preparation

Barnwell, T.O. Jr., Brown, L.C., and Marek, W. (1986) Development of a prototype expert advisor for the enhanced stream water quality model QUAL2E. Internal Report, September 1986, Environmental Research Laboratory, Office of Research and Development, U.S. Environmental Protection Agency, Athens, Georgia 30613

Barr, A., Feigenbaum, E. (1982) The Handbook of Artificial Intelligence, Volume I, William Kaufmann, Inc. and Pitman Books Ltd. London. pp.326-335

Barr, A., Feigenbaum, E. (1982) The Handbook of Artificial Intelligence, Volume II, William Kaufmann, Inc. and Pitman Books Ltd. London. p.15

Barstow, D.R. (1979) Knowledge Based Program Construction, Elsevier, Amsterdam

Barstow, D.R. (1980) Exploiting A Domain Model in an Expert Spectral Analysis Program, Proceedings of the First National Conference on Artificial Intelligence, pp. 276-279

Barstow, D., Duffey, R., Snoliar, S., and Vestal, S. (1982) An Overview of φNIX, Proceedings of the National Conference on Artificial Intelligence, pp. 367-369

Bechtel, R.J. (1981) STAMMER2: A Rule-Based Application, Proceedings of the Seventh JICAI, pp. 1053

Beloit, J.A., Lemmon, A.V., and Selander, J.M. (1982) KBS: An Expert Planning System for Crisis Response, Report Number MTR-82W0004, NTIS Number 64-83/Iss25

Bennett, J.S., and Engelmore, R.S. (1979) SACON: A Knowledge-Based Consultant For Structural Analysis, Proceedings of the Sixth IJCAI, pp. 47-49

Bennet, J.S., Hollander, C.R. (1981) DART: An Expert System for Computer Fault Diagnosis, in Proceedings of the 7th International Joint Conference on Artificial Intelligence, Univ. British Columbia, Vancouver, Canada

Bigger, C.J., and Coupland, J.W. (1983) Expert Systems: Bibliography, IEE

Billmers, M., Swartout, M. (1984) AI-SPEAR: Computer system failure analysis tool, in Proc. European Conf. on Artificial Intelligence, Italy, September 1984

Billmers, M., Carifio, M. (1985) TVX: An expert system advisor for VMS, in Proc. Conference on Human Interface. Forthcoming

Blum, R.L., and Wiederhold, G. (1978) Inferring Knowledge from Clinical Data Banks: Utilizing Techniques from Artificial Intelligence, Proceedings of the Second Annual Symposium on Computer Applications in Medical Care, IEEE, pp. 303-307

Blum, R.L. (1980) Automating the Study of Clinical Hypotheses on a Time-Oriented Database: the RX Project, in Lindenberg, D., Kaihara, S. [eds.], Medinfo 80, North-Holland, Amsterdam, pp. 456-460

Blum, R.L. (1982) Discovery, Confirmation, and Incorporation of Causal Relationships from a Large Time-Oriented Clinical Data Base: the RX Project, Comput. Biomed. Res., 15(2)164-187

Bobrow, D.G. (1962) A Question Answerer for Algebra Word Problems, Memo No. 45, Artificial Intelligence Project, Computations Center and Research Laboratory of Electronics, MIT, Cambridge, Massachusetts

Bobrow, D.G. and Collins, A. (1975) Representation and Understanding: Studies in Cognitive Science, Academic Press

Bonissone, P.P. and Johnson, H.E. Jr. (1983) Expert System for Diesel Electric Locomotive Repair, Proceedings of the IJCAI

Briabrin, V.M., Pospelov, D.A. (1976) DILOS - Dialog System for Information Retrieval, Computation and Logical Inference, in: Conference on Artificial Intelligence: Question-Answering Systems, International Institute for Applied Systems Analysis, CP-76-6, pp.11-19

Brooks, R.A., Greiner, R. and Binford, T.O. (1979) The ACRONYM Model-Based Vision System, Proceedings of the Sixth IJCAI, pp. 105-113

Brough, D.R. and Alexander, I.F. (1986) The Fossil Expert System. Expert Systems, April 1986, Vol. 3, No. 2.

Brown, J.S., Burton, R.R., and Bell, A.G. (1974) SOPHIE: A Sophisticated Instructional Environment for Teaching Electronic Troubleshooting, BBN Report No. 2790, Bolt Beranek and Newman

Brown, J.S. and Burton, R.B. (1978) Diagnostic Models for Procedural Bugs In Basic Mathematical Skills, Cognitive Science, 2, 2, April-June, pp. 155-192

Buchanan, B.G., Davis, R., Yu, V., Cohen, S. (1977) Rule Based Medical Decision Making: MYCIN, in Shires, D.B., Wolf, H. [eds.], MEDINFO 77, North-Holland, Amsterdam, pp. 147-150

Buchanan, B.G., Feigenbaum, E.A. (1978) DENDRAL and Meta-DENDRAL: Their Application Dimension, Artificial Intelligence, 11, 5-24

Buchanan, B.G., Mitchell, T.M. (1978) Model-Directed Learning of Production Rules, in D.A. Waterman and F. Hayes-Roth [eds.], Pattern-Directed Inference Systems, Academic Press, New York

Buchanan, B.G., Shortliffe, E.H. (1984) Rule-based Expert Systems. The MYCIN Experiments of the Stanford Heuristic Programming Project. Addison Wesley. Reading, Massachusetts. 748p.

Burton, R.R., and Brown, J.S. (1979) An Investigation of Computer Coaching for Informal Learning Activities, International Journal of Man-Machine Studies, pp. 5-24

Burton, R.R. (1982) Diagnosing bugs in a simple procedural skill, in D.Sleeman and J.S. Brown [eds.], Intelligent Tutoring Systems, Academic Press, New York, pp.157-183

Callero, M., Gorlin, D.M., Hayes-Roth, F.A. and Jamison, L. (1980) TATR: Tactical Air TaRgeting. Publ. No. N-1645-ARPA. The RAND Corpn., Santa Monica, CA

Cappucio, A. (1985) Xper: gestion de bases de cannaissances (database management), Micro-Systemes, 53, May, pp.158-160

Carbonell, J.R. (1970) AI in CAI: An Artificial Intelligence Approach to Computer-Aided Instruction, IEEE Transactions on Man-Machine Systems, MMS-11(4), pp. 190-202

Carhart, R.E. (1979) CONGEN: An Expert System Aiding the Structural Chemist, in D. Mitchie, [ed.], Expert Systems in the Micro Electronic Age, Edinburgh Univ. Press, Edinburgh

Carr, B., and Goldstein, I. (1977) Overlays: A Theory of Modeling for Computer Aided Instruction, AI Memo 406, MIT

Chandrasekaran, B., Mittal, S., and Smith, J.W. (1980) RADEX: Towards A Computer-Based Radiology Consultant, in Pattern Recognition in Practice, Gelsema and Kanal [eds.]. North-Holland

Chandrasekaran, B., Mittal, S., and Smith, J.W. (1981) MDX and Related Medical Decision-Making Systems, Proceedings of the Seventh JICAI, 1055p

Chandrasekaran, B., Sharma, D.D., and Milller, D.W. (1982) The Application of Knowledge-Based Systems to the Reactor Operation, Proceedings of the 1982 American Nuclear Society Winter Meeting

Chiou, W.C. (1985) NASA Image-based Geological Expert System Development Project for Hyperspectral Image Analysis, Appl. Opt. 24, 14, July, pp. 2085-2091

Clancey, W.J. (1979) Transfer of Rule-Based Expertise through A Tutorial Dialogue, Report No. STAN-CS-769, Stanford University (Doctoral Dissertation)

Clancey, W.J., Letsinger, R. (1981) Neomycin: Reconfiguring a Rule-Based Expert System for Application to Teaching, in Proceedings of the 7th International Joint Conference on Artificial Intelligence, Univ. British Columbia, Vancouver, Canada

CMA (1983) The CMA Hazardous Waste Survey for 1983. Final Report. Chemical Manufacturers Association, Washington D.C.

Cohen, P.R. and Feigenbaum, E.A. (1982) The Handbook of Artificial Intelligence, Volume III, William Kaufmann, Inc. and Pitman Books Ltd. London

Corey, E.J., and Wipke, W.T. (1969) Computer Assisted Design of Complex Organic Synthesis, Science, Vol. 166, pp. 178-192

Corkill, D., Lesser, V. (1981) A Goal-Directed HEARSAY II: Architecture Unifying Data-Directed and Goal-Directed Control, Comp. & Inf.Sc.Dept., Univ. of Massachusetts, Amherst, TR-81-15

Darvas, F., Futo, I., Szeredi, P. (1976) A program for the automatic filtering of drug interaction.s Proc. of the Coll. on the Application of Computing in Medicine and Biology, D. Muszka, [ed.] Szeged, Hungary, pp. 413-422

Darvas, F., Futo, I., Szeredi, P. (1978a) Some application of theorem proving based machine intelligence in QSAR: automatic calculation of molecular properties and automatic interpretation of quantitative structure-activity relationships. Proc. of the Symposium on Chemical Structure - Biological Activity: Quantitative Approaches, Suhl, GDR, Akademie Verlag, Berlin, pp. 251-257

Darvas, F., Futo, I., Szeredi, P., Bendl, J., Köves, P. (1978b) A PROLOG-based drug design system (Hungarian), Proc. of the Conf. Programming Systems'78, Szeged (Hungary), pp. 119-126

Darvas, F., Futo, I., Szeredi, P. (1978c) A logic-based program system for predicting drug interactions, International Journal of Biomedical Computing, Vol. 9, pp. 259, 271

Darvas, F. (1978d) Computer Analysis of the Relationship between the Biological Effect and the Chemical Structure, Kemiai Közlemenyek (Hungarian), Vol. 50, pp.97-116

Darvas, F., Szeredi, J., Futo, I., Redei, J. (1979a) A logic based chemical information system – theoretical considerations and experiences. Proc. of First National Conf. of von Neumann Computer Science Society, Szeged, Hungary, pp. 92-96

Darvas, F., Futo, I., Szeredi, P. (1979b) Expected interactions of spirololactions: predictions by computer, Proc. of the Conf. in Pathogenesis of Hyperaldosteronism. E. Glaz [ed.] pp. 219-220

Darvas, F., Lopata, A., Matrai, Gy. (1980) A specific QSAR model for peptides, in: Quantitative Structure Activity Analysis F. Darvas, [ed.], Akademiai Kiado, Budapest, pp. 265-278

Davis, J.R., and Nanninga, P.M. (1985) Geomycin: Towards a geographic expert system for resource management. J. Env. Management 21 , No. 4, pp. 377-390

Davis, R., Buchanan, B.G. and Shortliffe, E.H. (1977) Production rules as a representation for a knowledge-based consultation program, Artificial Intelligence Jnl. Volume 8, pp.15-46

Davis, R., Austin, H., Carlbom, I., Frawley, B., Pruchnik, P., Sneiderman, R. and Gilreath, J.A. (1981) The DIPMETER ADVISOR: Interpretation of Geologic Signals, Proceedings of the Seventh IJCAI, pp. 846-849

Davis, R., and Lenat, D.B. (1982) Knowledge-Based Systems in Artificial Intelligence, McGraw-Hill, New York

Diersch, H.-J. (1980) Finite-Element-Programmsystem FEFLOW. Program description (in German). Institute for Mechanics (AdW of GDR), Berlin

Diersch, H.-J. and Kaden, S. (1984) Contaminant plume migration in the aquifer: finite element modeling for the analysis of remediation strategies: a case study. Collaborative Paper CP-84-11, International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria, 68p

Dohnal, M. (1984) Application of Universal Expert Systems in Industry, Research Report, TUB, Brno, CSSR

Dourson, S., Joyce, J. (1985) Building of expert systems to diagnose noise in automotive engine cooling systems, in: Proc. of the Conf. on Surface Vehicle Noise and Vibration, May 1985, Soc. of Automotive Engineers, Michigan, pp. 19-26

Duda, R.O., Gaschnig, J., Hart, P.E., Konolige, K., Reboh, R., Barrett, P., and Slocum, J. (1978) Development of the PROSPECTOR Consultation System for Mineral Exploration, Final Report, SRI Projects 5821 and 6415

Dudak, M., Kleinova, N., Mikula, O. (1984) Application of Expert Systems in Chemical Industry, 8th International CHISA Congress, Prague

Dungan, C.W., amd Chandlers, J.S. (1985) Auditor: a microcomputer-based expert system to support auditors in the field. Expert Systems, Oktober 1985, Vol. 2, No. 4.

Energy & Technology Review (1985) Artificial Intelligence in Chemical Analysis, Feburary 1985

Engelman, C., Berg, C.H., and Bischoff, M. (1979) KNOBS: An Experimental Knowledge Based Tactical Air Mission Planning System, Proceedings of the Sixth IJCAI, pp. 247-249

Engelmore, R., Terry, A. (1979) Structure and Function of the CRYSALIS System, in Proceedings of the 6th International Joint Conference on Artificial Intelligence, Tokyo, Japan, pp. 250-256

Ennis, S. P. (1982) Expert Systems: A User's Perspective of Some Current Tools, Proceedings of the National Conference on Artificial Intelligence, Wm. Kaufmann, Inc., Los Altos, CA

Erdman, L.E., London, P.E., and Fickas, S.F. (1981) The Design and an Example Use of HEARSAY-III, JICAI, pp.409-415

Erman, L.D. and Lesser, V.R. (1975) A Multi-level Organization for Problem Solving Using Many Diverse, Cooperating Sources of Knowledge. Proc. of the 4th Int. Joint Conf. on Artificial Intelligence, pp.483-490.

Erman, L.D., Hayes-Roth, F., Lesser, V.R., and Reddy, D.R. (1980) The Hearsay-II Speech-Understanding System: Integrating Knowledge to Resolve Uncertainty, Computing Surveys, pp. 213-253

Esprit Project 814 (1986) PIMS: Project Integrated Management System: Helping Project Managers to Manage, Project Overview, Cap Sogeti Innovation, University of Amsterdam, BSO, London Business School, Pactel, London School of Economics, Turing Institute, University College London

Esprit '84 (1985) Status Report of Ongoing Work. J .Roukens and J.F. Renuart [eds.], Commission of the European Communities. Elsevier Science Publishers B.V. (North-Holland)

Esprit '86 (1987) Results and Achievments. Edited by the Directorate General XIII, Telecommunications, Information, Industries & Innovation. Commission of the European Communities. Elsevier Science Publishers B.V. (North-Holland)

Eur. Digital Mangement Report (1985) The answer lies in the shell. Expert System Development, ICI's Wheat Counsellor, June 1985, pp.2-6

Fagan, L. (1979) Knowledge Engineering for Dynamic Clinical Settings: Giving Advice in the Intensive Care Unit, Doctoral Dissertation, Computer Science Department, Stanford University

Fagan, L.M., Shortliffe, E.H., Buchanan, B.G. (1981) Computer-Based Medical Decision Making: from MYCIN to VM, Automedica, 3, 97-106

Fain, J.E., Hayes-Roth, F.A., Sowizral, H.A., and Waterman, D.A. (1982) Programming in ROSIE: An Introduction by Means of Examples, RAND Report No. N-1646-ARPA, 1982

Fedra, K. (1985) Advanced Decision-oriented Software for the Management of Hazardous Substances, Part I. Structure and Design, CP-85-18, International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria. 61p

Fedra, K. (1986) Advanced Decision-oriented Software for the Management of Hazardous Substances. Part II. A Demonstration Prototype System. CP-86-10. International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria. 98p

Fedra, K., Otway, H. (1986) Advanced Decision-oriented Software for the Management of Hazardous Substances, Part III. Expert Systems: Uses and Users. CP-86-14. International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria. 44p

Fedra, K., Weigkricht, E. and Winkelbauer, L. (1987) A Hybrid Approach to Information and Decision Support Systems: Hazardous Substances and Industrial Risk Management. Proceedings of the First Conference on Economics and Artificial Intelligence, Aix-en-Provence, 2-4 September, 1986. IFAC Publications, Pergamon. pp.169-175

Feigenbaum, E.A., Feldman, J. [eds.] (1963) Computers and Thought, McGraw Hill, New York

Feigenbaum, E.A., Buchanan, B.G., Lederberg, J. (1981) On Generality and Problem Solving: a Case Study Using the DENDRAL Program, in Meltzer, B., Michie, D. (eds.), Machine Intelligence 6, Edinburgh University Press, pp. 165-190

Feinstein, J.L., Siems, F. (1985) EDAAS: An expert system at the US Environmental Protection Agency for avoiding disclosure of confidential business information, Expert Systems, April 1985, Vol. 2, No. 2, pp.72-85

Fenves, S., Bielak, J., Rehak, D., Rychener, M., Sriram, D., and Maher, M.L. (1985) An expert system for diagnosis and repair of an automated transportation system, Technical Report, Robotics Institute, Carnegie-Mellon University, January 1985. In preparation

Fichtelman, M. (1985) The Expert Mechanic, Byte, 10, 6, June 1985, pp.205,206,208,211,212,214,216

Fieschi, M., Joubert, M., Fieschi, D., Soula, G., Roux, M. (1982) SPHINX - A System for Computer-Aided Diagnosis, Meth. Inform. Med., 21(3)143-148

Forgy, C.L. (1981) OPS 5 User's Manual, CMU-CS-81-135, Carnegie-Mellon University

Forsyth, R. (1987) Expertech Xi Plus, Expert Systems, February 1987, Vol.4, No. 1

Fox, M.S., Allen, B., Smith, S., and Strohm, G. (1983) ISIS: A Constraint-Directed Search Approach to Job Shop Scheduling, Proceedings of the IEEE Computer Society Trends and Application

Futo, I., Darvas, F., Szeredi, P. (1978) The application of PROLOG to the development of QA and DBM systems, in: Logic and Data Bases. H. Gallaire and J. Minker [eds.] Plenum Press, New York and London, pp. 347-376

Futo, I., Szeredi, J., Redei, J. (1979) The very high level language PROPHET and its application, in: Proc. of First National Conference of the Neumann Computer Science Society, Hungary, pp.146-157

Futo, I., Szeredi, J. (1980) PAPLAN - an AI language for parallel problem solving, in: Proc. of International Conf. on Artificial Intelligence and Information-Control Systems of Robots, Smolenice Castle, Czechoslovakia, pp. 74-76

Futo, I., Szeredi, J. (1982a) A Discrete Simulation System based on Artificial Intelligence Methods, in: A. Javor [Ed.], Discrete Simulation and Related Fields. Proceedings of the IMACS European Simulation Meeting, pp. 135-150

Futo, I., Szeredi, J. (1982b) A discrete simulation system based on artificial intelligence methods, in: Discrete Simulation and Related Fields, A. Javor [ed] North Holland, Amsterdam, pp.135-150

Gelernter, H.L., and Larsen, D.L. (1981) Experience with SYNCHEM 2, An Application of Artificial Intelligence to Organic Synthesis Discovery, presented at the Fall Meeting of the American Chemical Society

Genesereth, M.R. (1982) Diagnosis Using Hierarchicial Design Models, Proceedings of the National Conference on Artificial Intelligence, pp. 278-283

Genkin, G., Hikin, A. (1985) Automated Synthesis of Relational Logical Algorithm in Databases with Variable Function Structure, in: Technical Cybernetics, No.1, pp. 124-131

Gevarter, W.B. (1984) Artificial Intelligence, Expert Systems, Computer Vision and Natural Language Processing, Noyes Publications, Park Ridge, New Jersey

Gladun, V.P., Yavorsky, A.L. (1985) Text Synthesis in Thought Systems of Natural Language Dialogue, in: Kibernetika, No.1, p.98

GMD (1985) Der GMD-Spiegel 1/85, Expertensysteme – Vorboten einer neuen Technologie. (Expert Systems – Forerunners of a New Technology) Gesellschaft für Mathematik und Datenverarbeitung mbH, FRG. In German

Goodman, I.R., Nguyen, H.T. (1985) Uncertainty Models for Knowledge-Base Systems, North Holland, New York

Gorry, G.A., Silverman, H. and Pauker, S.G. (1978) Capturing Clinical Expertise: A Computer Program that considers Clinical Response to Digitalis, American Journal of Medicine, pp.452-460

Gottinger, H.W. (1984) HAZARD: an expert system for screening environmental chemicals on carcinogenicity, in Expert Systems, October 1984, Vol.1, No.2

Goyal, S.K., Prerau, D.S., Lemmon, A.V., Gunderson, A.S., and Reinke, R.E. (1985) Compass: An expert system for telephone switch maintenance, Expert Systems 2,3, pp.112-126

Graillot, D. (1986) Simulation in Support of Evaluation and Decision Making in Water Management Projects - IMWS: Integrated Model for Water Strategy. Preprints of the International UNESCO Symposium on Decision Making in Water Resources Planning, May 5-7, Oslo, Norway

Grauer, M., Lewandowski, A., Schrattenholzer, L. (1982) Use of the Reference Level Approach for the Generation of Efficient Energy Supply Strategies. WP-82-19, International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria

Green, C. (1977) The Design of the PSI Program Synthesis System, Proceedings of the IJCAI, pp. 380-381

Griesmer, J.H., Hong, S.J., Karnaugh, M., Kastner, J.K., Schor, M.I., Ennis, R.L., Klein, D.A., Milliken, K.R., and Van Woerkom, H.M. (1984) YES/MVS: A Continuous Real Time Expert System, in Proc. AAAI-84, AAAI, TX

Grimes, L.E., Rychener, M.D. and Westerberg, L.E. (1982) The synthesis and evolution of networks of heat exchange that feature the minimum number of units. In Chemical Engineering Communications, 14, 1982, Design Research Center, Carnegie-Mellon University, May

Grinberg, M.R. (1980) A Knowledge-Based Design System for Digital Electronics, Proceedings of the First National Conference on Artificial Intelligence, pp.283-285

Guan, J. (1986) The Development of an Expert System for Prospecting. Jilin University, The People's Republic of China

Guan, J. (1987) The Representation of Knowledge. Jilin University, The People's Republic of China

Gyarfas, F. and Popper, M. (1984) CODEX: Prototypes Driven Backward and Forward Chaining Computer-Based Diagnostic Expert System. Cybernetics and Systems Research 2, R. Trappl [ed.], Elsevier Science Publishers B.V., North-Holland

Hakami, B., Newborn, J. (1983) Expert system in heavy industry: an application, of ICLX in a British Steel Corporation works, ICL Technical Journal, 3,4, pp.347-359

Hamburger, H., and Slagle, J.R. (1983) Responding to a call of fire. NRL Rep. 8768, Naval Research Laboratory, Washington, D.C.

Harmon, P., and King, D. (1985) Commercial Tools, in Expert Systems - AI in Business, pp.92-134

Heiser, J.F., Brooks, R.E., Ballard, J.P. (1978) A Computerized Psychopharmacology Advisor, 11th Collegium Int. Neuro-Psychopharmacologicum, Vienna, Austria

Hendrix, G.G. et al. (1978) Developing a natural language interface to complex data, ACM Transaction on DBMS, Vol. 3, 1978, pp. 105-147

Heoser, J.F., Brooks, R.E. (1978) A Computerized Psychopharmacology Advisor, Proc. 4th annual AIM workshop, Rutgers University, pp. 68-69

Herrick, E.C., King, J.A., Quellette, R.P., and Cheremisinoff, P.N. (1979) Unit Process Guide to Organic Chemical Industries. Unit Processes Series, Organic Chemical Industries, Vol 1. Ann Arbor Science, Michigan

Hickam, D.H., Shortliffe, E.H., Bischoff, M.B., Scott, A.C., and Jacobs, C.D. (1985) A study of the treatment advice of a computer-based cancer chemotherapy protocol advisor, Technical Report KSL 85-21, Stanford University, Comp. Sci. Dept., accepted for publication in the Annals of Internal Medicine

Hofmann, M., Caviedes, J., Bourne, J., Beale, G., and Brodersen, A. (1986) Building expert systems for repair domains. Expert Systems, January 1986, Vol. 3, No. 1.

Horn, W. (1983) Expertensysteme, Funktionen und Systeme, Ein Überblick. (Expert Systems, Functions and Systems. An Overview) Rep.No.83-01, Dept. of Medical Cybernetics, University of Vienna, Austria

Huang, K. (1987) The Capacity of Machines for Induction. Nanjing University of Technology

Huang, K. (1983) An Expert System for Typhoon Track Forecasting. Nanjing University of Technology

ICF (1984a) The RCRA Risk-Cost Analysis Model. Phase III Report. Submitted to the Office of Solid Waste, Economic Analysis Branch, U.S. Environmental Protection Agency. March 1, 1984. ICF Incorporated, International Square, 1850 K Street, Northwest, Washington, D.C.

ICF (1984b) The RCRA Risk-Cost Analysis Model. Phase III Report. Appendices. Submitted to the Office of Solid Waste, Economic Analysis Branch, U.S. Environmental Protection Agency. March 1, 1984. ICF Incorporated, International Square, 1850 K Street, Northwest, Washington, D.C.

Industry News, Expert Systems, 1984, Vol. 2, No. 3, pp. 166-184

Industry News, Expert Systems, 1984, Vol. 1, No. 1, pp. 4-22

Industry News, Expert Systems, 1984, Vol. 1, No. 2, pp. 98-115

Industry News, Expert Systems, 1985, Vol. 2, No. 1, pp. 6-14

Industry News, Expert Systems, 1985, Vol. 2, No. 2, pp. 88-99

INFOTECH (1981a) The Fifth Generation. Infotech State of the Art Report Series 9, Number 1.

INFOTECH (1981b) Machine Intelligence. Infotech State of the Art Report Series 9, Number 3.

Ishizuka, M. (1984) Japanese work in Expert Systems, in: Expert Systems, Vol. 1, No.1

Ivanischev, V. (1982) SAPFIR: Computer System for Problem Presentation and Decision Formalization, Preprint, USSR Acadamy of Sciences, Scientific Research Center in Leningrad

Iwasaki, Y., and Friedland, P. (1982) SPEX: A Second Generation Experiment Design System, Proceedings of the National Conference on Artificial Intelligence, pp. 341-344

Jagannathan, V., and Elmaghraby, A.S. (1984) Computer-Aided Learning Tool for Simulation. Presented at the Third Annual Workshop on Interactive Computing: CAE: Electrical Engineering Education

Jagannathan, V., Elmaghraby, A.S., and Swenson, E.T. (1984) ESS-S: An Expert Support System for Simulation Language Selection, Working Paper, Univ. of Louisville

Jakob, F., Suslenschi, P., and Vernet, D. (1986) Extase: an expert system for alarm processing in process control. Proc. 7th European Conference on Artificial Intelligence, 21-25 July 1986, Brighton, pp. 121-132

Joubert, M., Fieschi, M., Fieschi, D., Roux, M. (1982) Medical Decision Aid: Logic Bases of the System SPHINX, 1st Int. Logic Programming Conference, G.I.A., Marseille

Kaden, S., Hummerl, J., Luckner, L., Peukert, D., Tiemer, K. (1985) Water Policies: Regions with Open-Pit Lignite Mining (Introduction to the IIASA Study). IIASA WP-85-04, International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria

Kant, E. (1979) A Knowledge-Based Approach to Using Efficiency Estimation in Program Synthesis, pp. 457-462

Kelly, V.E. and Steinberg, L.I. (1982) The CRITTER System: Analyzing Digital Circuits by Propagating Behaviors and Specifications, Proceedings of the National Conference on Artificial Intelligence, pp.284-289

Kent, D.L., Shortliffe, E.H., Carlson, R.W., Bischoff, M.B., and Jacobs, C.D. (1985) Improvements in data collection through physician use of a computer-based chemotherapy treatment consultant, J. of Clinical Oncology, 3, pp. 1409-1417, also Stanford Comp. Sci. Report KSL 85-22

Kerckhoffs, E.J.H. and Vansteenkiste (1986) The impact of advanced information processing on simulation - An illustrative review. In Simulation, January 1986

Kitowski, J., Moscinski, J. (1979) Computer Simulation of Heuristic Reinforcement-Learning Systems for Nuclear Power Plant Load Changes Control, in: Comput. Phys. Commun. Vol. 18, No. 3, pp. 339-352

Kitowski, J. (1983) Simulation of Thtr-Steam Generator Control using Artificial Intelligence Methods, in: Atomkernenerg. Kerntechn. (Germany), Vol. 42, No. 2, pp.124-125

Klix, F. (1979) On Interrelationships between Natural and Artificial Intelligence Research, in: Human and Artificial Intelligence, North-Holland

Korf, R.E. (1985) Macro-Operators: A Weak Method for Learning. Artificial Intelligence Journal,Spring 1985, pp. 35-77

Korolev, E.I. (1977) Linguistic Software of Automated Systems: Elements of Topology, in: Nauchino-Tekh. Inf. Ser. 2 (USSR), No. 9, pp. 1-6, Sept. 1977

Kunz, J. (1978) A Physiological Rule-Based System For Interpreting Pulmonary Function Test Results, Heuristic Programming Project Report No. HPP-78-19, Computer Science Department, Stanford University

Lang, I. (1978) On the generation of data processing ANSI-COBOL programs in PROLOG, Proc. of the Conf. Programming Systems'78, Szeged, Hungary, pp. 364-368

Langely, P., Bradshaw, G.L. and Simon, H.A. (1981) BACON.5: The Discovery of Conservation Laws, Technical Report, Dept. of Psychology, Carnegie-Mellon University

Lecompte, A. (1983) ESPRIT: Europe between America and Japan, in: Inf. et Gestion No. 140, March 1983

Lehert, W., and Bain, W. (1980) VEGE: Variable Processing in a Natural Language Query System, Research Report No. 193, Yale University

Lenat, D.B. (1976) Automated Theory Formation in Mathematics, Proceedings of the IJCAI, pp. 833-841

Lenat, D.B., Sutherland, W.R., and Gibbons, J. (1982) Heuristic Search for New Microcircuit Structures: An Application of Artificial Intelligence, AI Magazine, Summer 1982, pp. 17-26

Lesser, V.R. and Erman, L.D. (1977) A Retrospective View of the HEARSAY-II Architecture. In: Proc. of the 5th Intl. Joint Conf. on Artificial Intelligence, MIT, Cambridge, Mass. pp.790-800.

Lindsay, R.K., Buchanan, B.G., Feigenbaum, E.A. and Lederberg, J.(1980) Applications of Artificial Intelligence for Organic Chemistry: The DENDRAL Project, McGraw-Hill, New York

Li, D.Y., Heath, F.G. (1983) Ilex, An Intelligent Relational Database System, In: SIGPC Not. (USA), Vol. 6, No. 2, pp. 245-252

Liu, Y. (1986) The Design of an Expert System for Chinese Traditional Medicine. Shanxi Inst. for Automation, People's Republic of China

Lowerre, B.T. (1976) The HARPY Speech Recognition System, Ph. D. thesis, Computer Science Department, Carnegie-Mellon University, Pittsburgh, PA

Lu, J.-R., Fang, Y.-S. (1984) A Model of Second Order Learning Machine and its Realization by Software. In: Acta Electron. Sin. (China), Vol. 12, No.1, pp.87-93, Jan. 1984

Lu, H. (1987) System Structure Support for AI. Wuhan Air Force Radar Institute, People's Republic of China

Lynes, P.F., Cantwell, J.R., Shenoy, R., Kawanura, K., and Roth, J.A. (1987) An Application of Expert System Technology to Hazardous Waste Risk Assessment and Management. Craig-Lynes Chemical Mgt. Inc., Center for Intelligent Systems, Vanderbilt University, Nashville

Maher, M.L., (1984) HI-RISE: A Knowledge-Based Expert System for the Preliminary Structural Design of High Rise Buildings, PhD thesis, Carnegie-Mellon University, Dept. of Civil Engineering

Manna, Z. and Waldinger, R. (1978) DEDALUS - The DEDuctive ALgorithm Ur-Synthesizer, Proceedings of the National Computer Conference, pp. 683-690

Manucci, F., Gallagher, J., Swabey, M., and Fisher, C. (1985) Expert System Builder, in Esprit '84: Status Report of Ongoing Work. J. Roukens and J.F. Renuart [eds.], Commission of the European Communities, Elsevier Science Publishers B.V. (North-Holland), pp.183-202

Martianov, V. (1982) Specification and Computerized Inferences Theory, in: Novosibirsk, Development of a Software Application Package, Naoaka, Siberian Section of the Academy of Sciences, pp.5-23

Markusz, Z. (1977a) How to design variants of flats using programming language PROLOG, based on mathematical logic, Proc. of IFIP'77, Toronto, pp. 885-890

Markusz, Z. (1977b) The application of the programming language PROLOG for panel house design, Informacio-Elektronika (Hungarian) XI. No. 3, pp. 124-230

Markusz, Z. (1980a) The application of the programming language PROLOG for many storied panel house design, Informacio-Elektronika (Hungarian) XV. No. 5, pp. 256-263

Markusz, Z. (1980b) An application of PROLOG in designing many storied dwelling houses. Preprints of Logic Programming Workshop, Debrecen (Hungary), pp. 249-260

Markusz, Z. (1981) Knowledge representation of design in many-sorted logic. Proceedings of the 7th International Joint Conference on Artificial Intelligence, Vancouver, Canada

Markusz, Z., Kaposi, A. (1982) Engineering applications of logic programming under complexity control, Submitted for the MICAD-82, Paris, France

Matrai, G. (1979) The application of PROLOG for search of similar substructures enzyme sequence (Hungarian), MTA SZBK report

Mazario, F.J.G. (1982) DATED: A Language to Describe Plans and Strategies in Automatic Deduction Systems. Facultad de Informatica, San Sebastian, Spain

McCarthy, J., Abrahams, P.W., Hart, T.P., and Levin, M.I. (1962) LISP 1.5 Programmer's Manual. The Computer Center and Research Laboratory of Electronics, MIT, Cambridge, Massachusetts

McDermott, J. (1980) R1: A Rule-Based Configurer of Computer Systems, Carnegie-Mellon University Report CMU-CS-80-119

McDermott, J. (1981) R1: The Formative Years, AI Magazine, pp. 21-30

McDermott, J. (1984) R1 revisited: 4 years in the trenches, AI Magazine, Fall 1984, pp.483-534

McDermott, D., and Brooks, R. (1982) ARBY: Diagnosis with Shallow Causal Models, Proceedings of the National Conference on Artificial Intelligence, pp. 370-373

McDermott, D., and Brooks, R. (1982, 83) ARBY: Diagnosis with Shallow Causal Models, AAAI-82, AAAI-83

Medovyi, V.S. (1985) Translation from Natural Languages into Formal Language as Heuristic Task Search, Isvestia of the USSR Acad. of Sciences, in: Technical Cybernetics, No.1, pp.95-105

Merry, M. (1983) Apex 3: an expert system shell for fault diagnosis, GEC J. of Research, 1, 1, p.39

Michaelsen, R.H. (1984) An Expert System for Federal Tax Planning, in Expert Systems, Vol.1, No. 2

Michalsky, R.S. and Chilausky, R.L. (1980) Learning by being told and learning from examples: An experimental comparison of the two methods of knowledge acquisition. International Journal of Policy Analysis and Information Systems 4

Michalsky, R.S., Carbonnel, J.R. and Mitchell, T.M. (1983) Machine Learning: An Artificial Intelligence Approach Tioga Publ. Co., Palo Alto, 572p

Miller, R.K. (1984) The 1984 Inventory of Expert Systems, SEAI Institute

Minsky, M. [ed] (1968) Semantic Information Processing. The MIT Press, Cambridge, Mass.

Minsky, M. (1974) A framework for representing knowledge, MIT AI Memo 306, June.

Minsky, M. (1975) A Framework for Representing Knowledge. In: P.H. Winston [ed.] The Psychology of Computer Vision. McGraw-Hill. pp.211-280.

Mittal, S., Chandrasekaran, B., and Smith, J.W. (1979) Overview of MDX - A System for Medical Diagnosis, Proceedings of the 3rd Annual Symposium on Computer Applications in Medical Care, pp. 34-46

Mizoguchi, F., Maruyama, K., Yamada, T., Kitazawa, K., Saito, M., Kulikowski, C. A. (1979) A Case Study of EXPERT Formalism – Approach to the Design of Medial Consultation Systems through EXPERT Formalism, in Proceedings of the 6th International Joint Conference on Artificial Intelligence, Tokyo, Japan

Molnar, B.E., Markus, A. (1981) Logic programming in the modelling of machine parts, Proceedings of Compcontrol'81.Nov. 1981, Varna, Bulgaria

Moses, J. (1975) A MACSYMA Primer. Mathlab Memo No. 2, Computer Science Laboratory, Massachusetts Institute of Technology, Cambridge, Mass

Nelson, W.R. (1982) REACTOR: An Expert System for Diagnosis and Treatment of Nuclear Reactor Accidents, Proceedings of the National Conference on Artificial Intelligence, pp. 296-301

NFPA (1977) Fire Hazard Properties of Flammable Liquids, Gases, and Volatile Solids. NFPA 325M. National Fire Protection Association.

Niblett, T. (1985) Yet another PROLOG expert system, in CC AI, The Journal for the Integrated Study of Artificial Intelligence, Cognitive Science and Applied Systems, Intelligent Systems, Vol. 2, No 2, pp.3-30

Nii, H.P., and Aiello, N. (1979) AGE: A Knowledge-Based Program for Building Knowledge-Based Programs, Proceedings of the Sixth International Conference on Artificial Intelligence, pp. 645-655

Novatchenko, S.I., Pavlov, V.A., Yurevich, E.I. (1981) Specialized Modular Software System of Sensitized Robot Control Computer, in: Mech. and Mach. Theory (GB), Vol. 16, No. 1, pp. 41-48

Novosibirsk (1984) Problem-oriented Software package for Analysis and Synthesis of Network Structures, GRAPH-ES, Operation Information Material, Computer Center of the Siberian Branch of the USSR Academy of Sciences

Oleson, C.E. (1977) EXAMINER: A System Using Contextual Knowledge for Analysis of Diagnostical Behaviour, in Proceedings of the 5th International Joint Conference on Artificial Intelligence, MIT, Cambridge, USA, pp.814-818

Orciuch, E., Frost, J. (1984) ISA: Intelligent Scheduling Assistant, in Proc. IEEE/AAAI Conf. on Applications of Artificial Intelligence, Denver, IEEE Computer Society, pp. 314-320

Oxman, R., and Gero, J.S. (1987) Using an expert system for design diagnosis and design synthesis, Expert Systems, February 1987, Vol.4, No. 1

Pauker, S., Gorry, G.A., Kassirer, J., and Schwartz, W. (1976) Toward the Simulation of Clinical Cognition--Taking a Present Illness by Computer, American Journal of Medicine, pp. 981-996

Pearl, J., Leal, A., Saleh, J. (1982) Cognitive Systems Lab, School of Engin. and Appl. Sci., University of California, Los Angeles, CA, USA, IEEE Trans., Pattern Analysis and Machine Intelligence (USA)

Pendibidu, J.-M. (1982) THERESE: Presentation of the Project, in: Techn. and Sci. Inf. Vol. 1, No. 3, pp.235-237, May-June 1982

Perkins, W.A., Laffey, T.J. (1984) LES: a general expert system and its applications, SPIE Proc., 485, Applications of Artificial Intelligence, p.46

Piette, D., Roche, C., and Ianeselli, J.C. (1986) ALPA: Diagnosis expert systemfor supervision of nuclear reactors. Proc. 7th European Conference on Artificial Intelligence, 21-25 July 1986, Brighton, pp. 109-113

Polikarpov, V.S. (1977) Possibility of Simulating Intuition, in: Probl. Bioniki (USSR), No.18, pp.116-122

Politakis, P., Weiss, S. (1981) A System for Empirical Experimentation with Expert Knowledge, CBM-TM-91, Rutgers University, New Brunswick, NJ

Politakis, P., Weiss, S. (1984) Using empirical analysis to refine expert system knowledge bases, Artificial Intelligence, 22, 1 January 1984, pp.23-48

Pollit, A.S. (1984) A 'front-end' system: an Expert System as an online search intermediary, Aslib Proc., 36, 5, May 1984, pp.229-238

Popper, M. and Gyarfas, F. (1984) CODEX: a computer-based diagnostic expert system, in Artificial Intelligence and Information Control Systems of Robots, Elsevier Science Publishers B.V., pp.297-300

Pospelov, G.S., Ven, V.L., Litvinzev, P.I. (1980) The Experience of Dialogue Interactive System Development, Isvestia of the USSR Academy of Sciences, Technical Cybernetics, No.2

IIASA (1973) Proceedings of the IIASA Planning Conference on Computer Systems. IIASA-PC-7, International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria

Rabinovich, Z.L. (1979) Bionic Approach to Structural Simulation of Purposive Thinking, in: Kibernetika (USSR), Vol. 15, No. 2, pp.115-118, March-April 1979

Raphael, B. (1968) SIR Semantic Information Retrieval, in M.Minski [ed.]: Semantic Information Processing, MIT Press, Cambridge, Massachusetts

Rauch-Hindin, W.B. (1985) Artificial Intelligence in Business, Science and Industry, Prentice Hall, New Jersey

Reboh, R. (1981) Knowledge Engineering Techniques and Tools in the PROSPECTOR Environment, SRI International, Technical Note 243

Reggia, J.A. (1981) Knowledge-Based Decision Support Systems: Development Through KMS, TR-1121, Dept. Computer Science, University of Maryland, College Park, Maryland

Rich, C., Shrobe, H., and Waters, R. (1979) Overview of the Programmer's Apprentice, Proceedings of the Sixth IJCAI, pp. 827-828

Rissland, E.L. and Soloway, E.M. (1981) Constrained Example Generation: A Testbed for Studying Issues in Learning. Proceedings of the 7th International Conference on Artificial Intelligence, University of British Columbia, Vancouver, Canada, 1981

Reddy, Y.V., and Fox, M.S. (1982) KBS: An Artificial Intelligence Approach to Flexible Simulation, CMU-RI-TR-82-1, Robotics Institute, Carnegie-Mellon University

Roach, J.W., Virkar, R.S., Weaver, M.J., and Drake, C.R. (1985) POMME: a computer-based consultation system for apple orchard management using PROLOG, Expert Systems, Vol. 2, No. 2, pp.56-68

Sacerdoti, E.D. (1974) Planning in a Hierarchy of Abstraction Spaces, Artificial Intelligence, Vol. 5, pp. 115-135

Sacerdoti, E.D. (1977) A Structure for Plans and Behavior, Elsevier, N.Y.

Sachs, P.A., Paterson, A.M., and Turner, M.H.M. (1986) Escort - an expert system for complex operations in real time. Expert Systems, January 1986, Vol. 3, No. 1.

Safrans, C. Desforges, J., and Tsichlis, P. (1976) Diagnostic Planning and Cancer Management, Report No. TR-169, Laboratory for Computer Science, MIT

Sammut, C. (1981) Concept Learning by Experiment. Proceedings of the 7th International Conference on Artificial Intelligence, University of British Columbia, Vancouver, Canada

Savory, S. (1984) Faultfinder and Conad: 2 Working Expert Systems at Nixdorf Computer AG, in Proc. GDI Conference on Artificial Intelligence North Holland, Rueschlikon

Sax, N.I. (1975) Dangerous Properties of Industrial Materials, Van Nostrand Reinhold Company, New York

Selfridge, M. (1981) A Computer Model of Child Language Acquisition. Proceedings of the 7th International Conference on Artificial Intelligence, University of British Columbia, Vancouver, Canada

Serge, A.M., Sherwood, B.A., and Dickerson, W.B. (1983) An Expert System for the Production of Phoneme Strings from Unmarked English Text Using Machine Induced Rules, Proceedings of the 1st Conference of the European Chapter of the Association for Computational Linguistics, Pisa, Italy

Shapiro, E.Y. (1982) Algorithms Program Debugging. ACM Distinguished Dissertion. MIT Press, Cambridge, Mass.

Shortliffe, E.H. (1976) Computer-Based Medical Consultations: MYCIN, American Elsevier, New York

Shortliffe, E.H., Scott, A.C., Bishoff, M.B., Campbell, A.B., van Melle, W., and C. D. Jacobs (1981) ONCOCIN: An Expert System for Oncology Protocol Management, Proceedings of the Seventh IJCAI, pp. 876-881

Shpilberg, D., Lynford, E., and Schatz, G.H. (1986) ExpertTAXsm: an expert system for corporate tax planning Expert Systems, July 1986, Vol. 3, No. 3

Shubin, H., and Ulrich, J.W. (1982) IDT: An Intelligent Diagnostic Tool, Proceedings of the National Conference on Artificial Intelligence, pp. 290-295

Slagle, J.R., and Gaynor, M.W. (1979) Expert Systems Consultation Control Strategy, Proceedings of IJCAI, pp. 369-372

Slagle, J.R., and Gaynor, M.W. (1983) Expert System consultation control strategy. In Proceedings of the American Association of Artificial Intelligence 1983, Washington, D.C., Aug. 22-24

Slagle, J.R., Gaynor, M.W., and Hamburger, H. (1983) Decision support system for fire support command and control. NRL Rep. 8769, December

Slagle, J.R., Gaynor, M.W., and Halpern, E.H. (1984) An intelligent control strategy for computer consultation. IEEE Trans. Pattern Anal. Machine Intell. 6, 2 (Mar. 1984) pp. 129-136

Slagle, J.R., and Hamburger, H. (1985) An Expert System for a Resource Allocation Problem, Communications of the ACM, September 1985, Vol. 28, No. 9, pp.994-1004

Sleeman, D., and Hendley, R.J. (1982) ACE: A System which Analyses Complex Explanations, in Sleeman, D., Brown, J.S. [eds.], Intelligent Tutoring Systems, Academic Press, London

Sriram, D. (1985) DESTINY: A knowledge-based approach to integrated structural design, PhD thesis, Carnegie-Mellon University, Dept. of Civil Engineering

Stallman, R.M., and Sussman, G. (1979) Problem Solving about Electrical Circuits, in Progress in Artificial Intelligence, by Winston, P.H., MIT Press

Stammers, R.A. (1984) The MAX shell. Research and Development in Expert Systems. Proceedings of the Fourth Technical Conference of the British Computer Society Specialist Group on Expert Systems, University of Warwick 18-20 December 1984, Cambridge, pp.31-37

Steel, B. (1981) EXPERT – Implementation of a Data-independent Expert System with Quasi-natural Language Information Input, Res. Report DOC 81/23 (MSc Thesis), Imperial College of Science and Technology, London

Stefic, M. (1981) Planning with Constraints (MOLGEN: Part I), Artif. Intell., 16(2)111

Stefic, M. (1981) Planning and Meta-Planning (MOLGEN: Part II), Artif. Intell., 16(2)141

Stevens, A.L., Collins, A., and Goldin, S. (1978) Diagnosing Student's Misconceptions in Causal Models, BBN Report No. 3768, Bolt Beranek and Newman, Cambridge

Suppes, P. (1980) University-Level Computer Assisted Instruction at Stanford: 1968-1980, Stanford IMSSS

Swartout, W.R. (1977) A Digitalis Therapy Advisor with Explanations, in Proceedings of the 5th International Joint Conference on Artificial Intelligence, MIT, Cambridge, USA, pp. 819-825

Tancig, P., Bratko, I., Tancig, S. (1978) Mental Set Revisited in the Formal Manner, in: Current Topics in Cybernetics and Systems, pp. 68-69

Technica (1984) The SAFETI Package. Computer-Based System for Risk Analysis of Process Plants. Vol.I-IV. Technica Ltd., Tavistock Sq., London

Tomiyama, T. (1985) Beyond FRODIRD, in Third seminar of the joint Japanese-Hungarian project for expert systems on the history of science and technology, August 21-23, 1985, Veszprem, Hungary

Tompkins, J.W. (1982) PROBWELL: An Expert System Advisor for Determining the Causes of Problems with Producing Oil Wells, IBM Engineering and Scientific Study Conference Proceedings, IBM, Poughkeepsie, NY

Trigoboff, M., and Kulikowski, C. (1977) IRIS: A System for the Propagation of Inferences in a Semantic Net, Proceedings of the Fifth ILCAI, pp. 274-280

Underwood, W.E., Summerville, J.P. (1981) PROJCON: A Prototype Project Management Consultant, Proceedings of the International Conference on Cybernetics and Society, 149-55

Underwood, W.E. (1982) A CSA Model-Based Nuclear Power Plant Consultant, Proceedings of the National Conference on Artificial Intelligence, pp.302-305

Vaija, P., Järveläinen, Dohnal, M. (1985) Fuzzy Based Expert System for Analysis of Accidents. Helsinki Univ. of Technology, Finland and Tech. Univ. of Brno, CSSR

van Melle, W. (1979) A Domain-Independent Production-Rule System for Consultation Programs, Proceedings of the Sixth IJCAI, pp. 923-925

Vida, M. (1980) Computer-aided NMR Spectra Interpretation in an Artificial Intelligence System, in: Anal. Chim. Acta Comput. Tech. and Optimiz. Vol. 122, No. 1, pp.41-46, 1 March 1980

Vukobratovic, M. (1984) Computer-aided System for Robotics Design, Glas. Srp. Acad. Nauka, 1984, 339, pp. 19-34

Wang, H. (1986) An Analysis of EMYCIN and Investigation of Related Issues. Quinghua Univ., People's Republic of China

Wang, Q. (1987) An Investigation of the Strategies of Expert Systems in Machine Design. Hua Zhong Institute of Technology People's Republic of China

Wang, S. (1984) A Study of the Design of Rule-based Expert Consultative Systems. Zhejiang Univ. People's Republic of China

Wang, X-H. (1983) Artificial Intelligence. Jilin University, People's Republic of China

Wang, Y. (1986) The Design for an Expert System for Military Command Policy. Beijing Inst. of Technology, People's Republic of China

Wang, Y.-A., Hsu, T.-H., Chung, K.H. (1983) The Implementation of a High Accuracy Temperature Control System using a Microcomputer, in: Chin. J. Sci. Instrum. Vol. 4, No. 4, 350-6, November

Ward, J.R. (1984) User's Guide to the DB Relational Database Manager. Third Edition, LS7-1984, September, International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria

Waterman, D.A. (1978) Exemplary Programming in RITA, in Waterman, D.A., Hayes-Roth, F. [eds.], Pattern-Directed Inference Systems, Academic Press, New York

Weiss, S.M., and Kulikowski, C.A. (1979) EXPERT: A System for Developing Consultation Nodes, Proceedings of the Sixth IJCAI, pp. 942-947

Weiss, S.M., Kulikowski, C.A., Amarel, S., and Safir, A. A Model-Based Method for Computer-Aided Medical Decision Making, Artificial Intelligence, 11:145-172

Weiss, S., Kulikowski, C., Apte, C., Uschold, M., Patchett, J., Brigham, R., and Spitser, B. (1982) Building Expert Systems for Controlling Complex Problems, Proceedings of the National Conference on Artificial Intelligence, pp. 322-326

Weizenbaum, J. (1965) ELIZA - Computer Program for the Study of Natural Language Communication between Man and Machine, Communications of the ACM, Vol. 9, No. 1, January 1965

Weizenbaum, J. (1976) Computer Power and Human Reason. W. H. Freeman and Company. San Francisco. p. 227

Williams, G. (1984) SAGE, an expert system shell: Computer Aided Decision Making, London

Williams, M., Hollan, J., Stevens, A. (1981) An Overview on STEAMER: An Advanced Computer-assisted Instruction System for Propulsion Engineering, Behaviour Research Methods & Instrumentation, 13(2)85-90

Wilson, J.L., Mikroudis, G.K., and Fang, H.Y. (1986) GEOTOX: A Knowledge Based System for Hazardous Site Evaluation. Applications of Artificial Intelligence in Engineering Problems. Proceedings of the first International Conference, Southampton University, UK, April 1986, Volume II.

Winkelbauer, L. (1984) Lernen aus Beispielen. Ein Computersystem zur Simulation kindlichen Lernens. (Learning by Example. A Computer System for the Simulation of Learning during Childhood) MSc thesis. Faculty of Sociology and Economics, University of Vienna. 354p. In German. Unpublished

Winkelbauer, L. (1987) Symbolic Simulation of Chemical Production Processes, WP-87-45. International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria. 55p

Winograd, T. (1972) Understanding natural language, Academic Press

Winston, P.H. [ed.] (1975) The Psychology of Computer Vision, McGraw-Hill

Winston, P.H., and Horn B.K.P. (1981) LISP, Addison Wesley, Reading, MA

Wipke, W.T., Braun, H., Smith, G., Choplin, F., and Sieber, W. (1977) SECS: Strategy and Planning, Computer-Assisted Organic Synthesis, American Chemical Society, pp. 97-127

Yang, Z. (1986) The Representation of Knowledge. Wuhan University, People's Republic of China

Zanelli, S. Bello, G.C., and Frattini, B. (1984) Classification Scheme for Process Plants. Intermediate Progress Report, November. Joint Research Center, Ispra, Italy.

Zarri, G.P. (1984) Expert systems and information retrieval: an experiment in the domain of biographical data management, Int. J. Man-Machine Studies, 20, 1, January 1984, pp.87-106

Zhang, A. (1987) Elements of Knowledge and their Application within Expert Systems. Jilin University, People's Republic of China

Zhang, C. (1984) Expert Systems and Traditional Programs. Jilin University, People's Republic of China

Zhang, C. (1987) Knowledge-based Reasoning. Jilin University, People's Republic of China

Zhao, Ch., Winkelbauer, L., Fedra, K. (1985) Advanced Decision-oriented Software for the Management of Hazardous Substances, Part VI. The Interactive Decision Support Module. CP-85-50. International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria. 39p