ICOT Technical Report: TR-204

TR-204

Knowledge Acquisition and Learning in Case Studies of Expert System Development

by
S. Kunifuji (Fujitsu Ltd.), S. Kobayashi (TIT)
M. Suwa (ETL), Y. Iwashita and N. Oka (ICOT)

September, 1986

©1986, ICOT

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03) 456-3191~5
Telex ICOT J32964

Institute for New Generation Computer Technology

KNOWLEDGE ACQUISITION AND LEARNING IN CASE STUDIES OF EXPERT SYSTEM DEVELOPMENT

By S. Kunifuji (IIAS-SIS,Fujitsu), S. Kobayashi (Tokyo Institute of Technology), Y. Iwashita, N. Oka (ICOT), M. Suwa (Electro Technical Laboratory)

1. Introduction

In 1985, the Institute for New Generation Computer Technology (ICOT) received a contract to survey R&D trends on new-generation computer technology and its applications from the Japan Machinery Industry Association. In accordance with this contract, we researched various aspects of knowledge acquisition in expert systems. The purpose of the research is to solve "knowledge acquisition" problems that are "bottlenecks" in building expert systems. We held meetings on knowledge acquisition using developing expert systems as case studies and summarized the results. This paper presents these results and our views on them, and considers their relationship with learning.

As expert systems progress, the tools for supporting knowledge representation and utilization are hastily being prepared. However, human knowledge of problem-solving in the real world is extremely diverse and uncertain. To use this knowledge as input information for tools, basic problems concerning knowledge acquisition must be solved before problems concerning knowledge representation and utilization are solved. Research on solving the former problems lags behind research on the latter, but it is one of the most important tasks.

As we pointed out in references *1 and *2, knowledge acquisition problems fall into the following three categories corresponding to its :

- Stage 1: Knowledge extraction problem

  In this stage, the problem is how to collect, classify, and analyze the knowledge that exists in the minds of experts in various fields and that is useful in solving problems, but that has never been expressed in symbols (language).

- Stage 2: Knowledge transformation problem

  In this stage, the problem is what representation form the knowledge processed in stage 1 should take and how it should be stored for processing in a computer system.

- Stage 3: Knowledge base management problem*2,*3

In this stage, the problem is how to manage consistently and systematically the knowledge processed in stage 2 so as to allow modification and addition of new knowledge.

From the practical standpoint, our primary concern is with stages 1 and 2. One reason for this is that the computational difficulties *4, *5 have been found by our fairly systematic research*2,*3 into the problems in stage 3. To solve the problems in stage 1 and 2, we heard from the builders of expert systems about knowledge acquisition methods, putting aside our preconceived ideas. We heard from experts in fields as disparate as possible. The results are listed in 2 Table and analyzed.

2. Case Studies of Expert Systems

We were given descriptions of knowledge acquisition for building the following nine examples of expert systems:

(1) Production line expert system: Hitachi*6

(2) Expert system supporting electric power system analysis: Mitsubishi Electric*7

(3) Personal finance adviser: Fujitsu*8

(4) Large DC-motor diagnosis system: Toshiba*9

(5) Dam gate diagnosis system: Central Research Institute of Electric Power Industries*10

(6) HLSI wiring design expert system: NEC*11

(7) Expert system for designing lenses: Canon*12

(8) CAD for equipment layout in computer rooms: Hitachi*13

(9) Expert system for industrial property: Electro Technical Laboratory.*14

This section introduces these expert systems briefly and summarizes their features.

(1) Production line expert system *6

This sytem is a production line control software system developed according to a knowledge engineering paradigm to cope flexibly with model changes in products. The flexibility of the system is provided by easily changeable control rules written for it. The system is a pioneer rule-based control system for practical use rather than an expert system. Its features are improved modifiability and maintainability and reduced work steps for its development.

(2) Expert system supporting electric power system analysis*7

This system aims at providing a necessary environmet for efficiently scheduling an electric power system. The environment consists of a knowledge base for devising plans, user-oriented interfaces, a data base for managing hardware and test data, and high-speed interactive functions. One feature of the system is short analysis time. That is, a problem that a human expert has to spend one hour to solve can be solved in two minutes by this system with its knowledge base.

(3) Personal finance adviser*8

This is a prototype expert system providing financial advice for office workers. Based on the present structure and financial state of the family, the system forecasts the future household economy, evaluates plans such as purchasing a house, and gives advice. The system was intended for evaluating an expert system support tool called ESHELL.

(4) Large DC-motor diagnosis sytem*9

This system diagnoses large DC motors (for rolling machines in a rolling mill) developed under a knowledge - based system support tool called TDES2. The system locates errors and supports maintenance through interactions with the operator. The features of the system are modularity of knowledge, treatment of uncertain data, and text processing. The methods that the system uses to assign certainty factors to uncertain data and acquire observation data other than error data may require some improvemet.

(5) Dam gate diagnosis system*10

This system diagnoses a dam gate and evaluates its residual life from values produced by an analysis subsystem, values corresponding to quantitative data (such as data on displacement and stress) and qualitative data (such as data on appearance) in a data base. One feature of the system is the display of all conclusions obtainable from the quantitative and qualitative data given three attributes: safety evaluation, residual life estimation, and required maintenance time. The system may require the use of a more advanced system engineering methodology such as the multi-attribute effect utility theory in multi-purpose decision theory*15.

(6) LSI wiring design expert system*11

This system wires two LSI terminals that an automatic wiring program cannot wire. The system uses wiring know-how expressed as rules stored in a knowledge base. By using this system, even an unskilled worker can speedily and elegantly complete the wiring work. A feature of this system is that it has been developed under a PROLOG process or called CADLOG, which can be linked with FORTRAN in an existing CAD system. One problem that the system will confront is how to cope with an increase in the number of gates.

(7) Expert system for designing lenses*12

This system designs automatically a new lens system according to a patterned system design schedule and CAD knowledge when given parameter values for the specifications and framework of the lens system. The system has support functions such as initialization, CAD command generation, and design scheduling functions. One noteworthy point is an adoption of the apprenticeship method for knowledge acquisition.

(8) CAD for equipment layout in computer rooms*13

This system determines the layout of equipment in a computer room using a knowledge engineering technique. The system performs forward reasoning to satisfy semantic constraints in generating a plan. Its computation speed is high, but the scope of problems it can handle is narrow. The system basically solves search problems which the conventional technique can hardly handle because of the varieties of semantic constraints.

(9) Expert system for industrial property*14

This is a consultation system concerning procedural law in the Patent Law. The system offers an object-oriented and augmented logic programming language called KRIP/L for representing procedural law. The system consists of an expert support system that experts in law use to register legal knowledge in a knowledge base, and of a user support system that end users use to create a patent data base and judge the validity of registered patents. The system may find quite interesting future subjects, such as for finding references to judicial precedents and judging facts.

Table 1 shows an outline of the above nine expert systems. In Table 1, the systems are classified into the types diagnosis, control, planning, and design. The information below is entered in the columns of Table 1.

- "Development stage" column: Prototype or practical use. Other information such as the scope of problems which can be handled is also entered if obtained.
- "Development environment" column: The names of shells, languages, and machines.
- "Knowldedge base size" column: Number of rules and others. A rule has different meanings depending on the knowledge representation form and inference method. Therefore, consideration only of the number of rules is insufficient.
- "Inference engine " column: Forward and/or backward reasoning, uncertainty, etc.
- "Interface with externals" column: Names of external resources, I/O from/to external processors, names of graphic routines, etc.
- "Others" column: Remarks

3. Aspects of Knowledge Acquisition

Table 2 shows an outline of knowledge acquisition performed by the nine expert systems described in Section 2. The information below is listed in Table 2.

- "Participants" column: DE(Domain Experts) and/or KE (Knowledge Engineers). Distinction betwee DE and KE is not very clear.
- "Knowledge sources" column: Various knowledge sources (KS), e.g., know-how of experts, unformatted information such as graphic patterns, and text of symbols.
- "Support tools" column: Classification of support tools.
- "Basic policy" column: Basic policy of knowledge acquisition. This depednds on the types of problems given and the extent of difficulty in solving them.
- "Basic chart" column: Problem solving chart having feedback loops.

- "Procedure and no. of work steps" column: The entry depends on whether the system is a prototype or for practical use and, if the latter, on the difficulty of solving the given problem.
- "Changes in representation and quantity of knowledge" column: The changes involved with version-up.
- "Final classification of knowledge" column: The entry depends largely on the knowledge representation form which can be processed under a given development environment.
- "Knowledge acquisition method" column: Information in this column is the most important for our last attempt, that was insufficient thus requiring retry. Note whether the interview or apprenticeship method is entered.
- "Inquiry" column: Typical successful and unsuccessful events.
- "Comment" column: Problems unsolvable by the expert systems are mainly entered. Some of the systems have meager ability to represent knowledge including two-dimensional pattern.

From analysis of the above information about knowledge acquisition in the expert systems, it is clear that knowledge acquisition generally consists of the following steps:

1) Selecting problems :
   The possibility and value of building a system is evaluated, and problems are selected.
2) Evaluating current software technology :
   For selected problems, current AI software technology is evaluated, and necessity of the introduction of current AI software is inquired.
3) Identifying knowledge sources :
   The necessary knowledge for solving selected problems is evaluated, and so is the possibility of solving them. The quality and quantity of knowledge per knowledge source are analyzed.
4) Identifying expert models :
   If the main knowledge sources are human beings, then how the experts use their knowledge (their problem-solving techniques and inference methods) is clarified.
5) Identifying user models :
   What the users expect the system to do and how they use it are clarified.
6) Selecting knowledge representation form :
   Appropriate knowledge representation form is selected based on the results of steps 3) to 5).
7) Extracting knowledge :
   Knowledge in the form selected in step 6) is extracted from the knowledge sources.

8) Transforming knowledge :
The knowledge obtained in step 7) is transformed to a format usable in the computer and stored in the knowledge base.

9) Managing the knowledge base :
Consistency of knowledge added to the knowledge base is checked, and knowledge explaining added knowledge is automatically generated in the knowledge base.

Among the above nine knowledge acquisition steps, steps 1) to 6) are called system analysis in system engineering, and steps 7) to 9), modeling. Note that system analysis is more important than modeling for a problem requiring a more advanced expert system than other problems. The most important task at present is to combine the system analysis and modelig methods for a given problem most appropriately.

4. Knowledge Acquisition Methodology*16, *17

This chapter discusses the basic nature of problem in knowledge acquisition in the following four application fields of expert sytems :

- Data interpretation
- Diagnosis
- Monitoring and control
- Planning and design

(1) Knowledge acquisition in data interpretation

Data interpretation refers to analyzing data obtained by measuring equipment and sensors, estimating system states, and providing them with physical meaning. Acoustic, video, digital, and spectral data are analyzed.

Knowledge acquisition in data interpretation divides continuous analog data into meaningful segments, characterizes each segment symbolically, and extracts knowledge representing the higher-order correlation between segments. Conventional signal processing or data analysis techniques can only perceive lower-order correlations between data. They cannot perceive higher-order correlations because of the large amount of computation. Thus knowledge processing is required.

Knowledge used for data interpretation is divided into knowledge about system structure and properties, and empirical knowledge taken from expert human beings. The phases of knowledge aquisition depend on which kind of knowledge is more important. If the former kind is more important, knowledge acquisition lays emphasis on how to formalize the structure of the object. If the latter is more important, kowledge acquisition lays emphasis on how to formalize the knowledge obtained through pattern recognition by the human beings.

Knowledge representation and utilization in data interpretation must assume some missing data and the presence of noise. The introduction of uncertainty to knowledge representation is required, as is the introduction of cooperative problem solving such as blackboard modeling to knowledge utilization. If uncertaion knowledge and knowledge of cooperative problem-solving can be obtained from human experts, these kinds of knowledge must be restructured and optimized from the perspective of system engineering.

(2) Knowledge acquisition in diagnosis

Diagnosis finds the causes of errors in the system by using observed data and knowledge of causal relations in the system. Medical diagnosis, plant diagnosis, and equipment diagnosis are some examples.

The phases of knowledge acquisition in diagnosis depend on whether a natural or an artificial system is diagnosed.

When a natural system such as a living body is diagnosed, quantative knowledge of its macro structure is available, but its micro structure is almost like a black box, and the diagnosis has to depend on the experience-based surface knowledge of human experts on causal relations. Such knowledge contains uncertainty, thus treatment of uncertainty becomes an important task. To retain the consistencey of uncertain knowledge and the reliability of diagnosis results especially for multi-stage inference, a technique for integrating uncertain knowledge must be invented.

When an artificial system is diagnosed, knowledge of causal relations can naturally be extracted from knwoledge of the system design. This knowledge must be formed so that it can be used in diagnosis. If the system is large and if data collection by measuring instrument or an interaction with the operator is required, finding an error cause based only on knowledge of the system structure is likely to be redundant. In these cases, the experience-based knowledge of human experts must be used. Even if the knowledge is shallow, access to the vicinity of an error location by using it is efficient because intermediate inference paths are discarded. In short, diagnosis using only knowledge of sytem structure may be redundant though complete, and diagnosis using only the experience-based knowledge may be incomplete but efficient. Knowledge acquisition has to be considered taking the trade-off between them into consideration.

(3) Knowledge acquisition in monitoring and control

Monitoring and control monitors the system status and puts the system into a predetermined sequence of controls. Problems in control are extremely common in plant control, flight control, and production system control.

The phases of knowledge acquisition depend on whether a continuous system or a discrete system is controled.

A continuous system is a system to which the classic feedback control theory and modern control theory cannot be applied. It may be a non-linear system where large time lags exist.

Knowledge acquisition in a control system divides the state space into an appropriate number of sub-state regions and extracts control rules suitable for each sub-state region. The larger the number of sub-spaces, the more precise the control but the larger the cost for knowledge acquisition. Fuzzy control is useful in this case. Knowledge acquisition in a fuzzy control system obtains control rules for some representative points in the state space from human experts. For other points in the state space, it extracts control rules from those for the neighborhoods combined by fuzzy inference. This method greatly reduces problems in knowledge acquisition. However, selection of representative points is quite difficult, and the number of selected representative points must be appropriate. This task involves a trial-and-error phase.

If the system status depends largely on its history and requires multi-stage inference to determine appropriate control rules, fuzzy control is difficult and has to be replaced with rule-based control. However, if the system is highly autonomous and switches back automtically to its optimum state under coarse control, problems in knowledge acquisition are eased to some extent.

Sequential control was conventionally applied to a discrete system. However, rule-based control facilitates readability and maintenance and is advantageous from the perspective of knowledge acquisition. But it requires a support environment for placing the system in a complete state.

A control system must perform strictly real-time operations. Therefore, its inference must be made fast by compiling or reorganizing acquired knowledge.

(4) Knowledge acquisition in planning and design

A great deal of search work is required for building an optimum planning and design system. Determining a basic search loop is important for solving a combinatorial problem.

Many of the already developed expert systems for planning and design use conventional system engineering techniques with the new knowledge processing techniques to widen the degree of automation. Introduction of the system is decided when the cost for knowledge acquisition is less than the planning and design cost reduced by automation.

Planning and design are extremely creative activities. Therefore, the same problem is approached differently by planners and designers. However, the basic approach consists of a combination of the problem reduction method, hierarchical generation and test method , and top-down refinement method. Clarifying and evaluating these basic steps of problem solving are important for knowledge acquisition in planning and design. Knowledge acquisition by the apprenticeship method may be useful to the system designer in achieving these steps.

5. Knowledge Acquisition and Learning*18

This chapter discusses the introduction of learning functions into knowledge acquisition steps. Our research was intended to automate only the steps 7) to 9) among the nine knowledge acquisition steps described in Section 3. Generally, the following two tasks provide interfaces between an expert system and man:

a) Building a knowledge base :

This seeks to store knowledge for inference and problem solving. It must be easily performed by men without computer knowledge.

b) Supporting the user :

This provides the user with necessary information and inference results and solves the problem through interaction between the expert system and the user. Consistency of thought of the user is the key to the task.

Introduction of learning functions is also approached through these tasks. Typical approaches to the introduction of learning functions for supporting knowledge acquisition are listed below.

(A) Knowledge base editor :

An excellent program editor is indispensable for programming, as is an excellent knowledge base editor for building a knowledge base. Much effort has been expended on the development of this portion in presently commercial expert system shells. Users will be able to use better intelligent interfaces from now on. In the future, inductive inference such as editing by example*19 may be introduced.

(B) Knowledge base management :

The objectives of knowledge base management are maintaining the soundness and completeness of a given knowledge system, keeping the whole knowledge base consistent with any new knowledge added to it, and building a model for adjusting the knowledge base to explain the added knowledge. At ICOT , these objectives are strenuously investigated as knowledge assimilation and accommodation problems. However, they may be serious problems when the knowledge base is enlarged. The inductive inference system called Shapiro's Model Inference System is used to solve the knowledge accommodation problem.

(C) Problem formalization support :

This is an area untouched yet by current knowledge engineering. An attempt to express a certain problem by using a tool having a specific knowledge representation power encounters enormous obstacles. Establishing a procedure for formalizing it and developing a system supporting its execution are important. The first thing is to establish the knowledge system methodology per application field, then a problem formalization support system should be developed by utilizing the concepts of a software reuse system. The support system may have analogical reasoning functions.

(D) Semi-automation of knowledge acquisition :

Automating knowledge acquisition is an ideal of the researchers in artificial intelligence. At present, however, some efforts are being devoted to the development of a semi-automatic knowledge acquisition system with intervention by man. Such a system may have the advanced learning functions shown in (A) to (C).

6. Conclusion

This paper summarizes our view on knowledge acquisition and learning with various expert systems. It is based on a survey into "phases of knowledge acquisition in expert systems" that ICOT performed in 1985. As part of this research, we held meeting on knowledge acquisition with various expert systems to solve knowledge acquisition problems that are bottlenecks in building expert systems. The results were analyzed, our views on knowledge acquisition in expert systems were presented, and the relationship between knowledge acquisition and learning was discussed. Although the relationship between knowledge acquisition and learning was insufficiently discussed, our considerations on knowledge acquisition and learning are presented below as future objectives.

(1) Knowledge acquisition steps such as interview and protocol analysis are too direct and too experiential. Knowledge engineering and system engineering must have many things in common since both anayze large-scale complicated objects.

The advent of knowledge engineers versed in system engineering methodology is urgently required.

(2) Knowledge obtained from objects and that from experts should be clearly distinguished. The former should be used with advanced utilization, and its dependency on the latter minimized. The more complicated and versatile the problem, the more versatile the obtained knowledge. Advanced use of knowledge depends on the sources and types of the knowledge.

(3) Discussion of a knowledge-acquisition method is meaningless unless methods to represent and use knowledge are decided. A knowledge representation method and an inference engine should be flexibly selected to suit the type of knowledge obtained.

(4) Method for introducing learning functions to knowledge acquisition steps may be applied to the knowledge base editor, knowledge base management, problem formalization support, and semi-automation of knowledge acquisition described in Section 5.

(5) Some learning functions to be installed in the future are learning certainty factors, learning compiled knowledge by partial evaluation, learning strategies for knowledge acquisition, and introducing probabilistic inductive inference models .

In conclusion, we should go back to the starting point in an approach to a problem, that is to "First the problem, then the tool." The object to be obtained should be looked at squarely, an approriate system analysis methodology united with a modeling methodology, and a knowledge system methodology per application field established. These are the most important tasks given at present to persons interested in building a knowledge base for an expert system.

APPLICIATION: The following persons also participated in the research undertaken by ICOT: M. Ichikawa (Mitsubishi Heavy Ind.), T. Kawano (Toshiba), S. Goto (NEC), H. Sanada (Osaka Univ.), H. Tanaka (Tokyo Univ.), T. Terano (Central Res'ch Inst. of Elec. Pow. Ind.), R. Nagata (Petroleum Resource Dev.), A. Fusaoka (Mitsubishi Elec.), H. Monzen (Fujitsu), and T. Watanabe (Hitachi). We thank these fellow researchers for their useful opinions. Also, we profoundly appreciate the kind cooperation of the compainies and personnel at our meeting.

References

*1 S. Kunifuji, H. Kitakami, T. Miyachi, K. Furukawa: Fundamentals and applications of knowledge engineering (4th - Knowledge base management in PROLOG-,Vol.24 No.6, pp.539-548, June 1985 (in Japanese).

*2 S. Kunifuji, H. Kitakami, T. Miyachi, K. Furukawa: Knowledge base management in logic programming language PROLOG, Proc. of the Logic Programming Conference '85, pp.141-154, July 2nd 1985 (in Japanese).

*3 H. Kitakami, S. Kunifuji, T. Miyachi, K. Furukawa: Knowledge base management system in logic programming language PROLOG, Journal of Information Processing Society of Japan, Vol.26 No.11, pp.1283-1295, November 1985 (in Japanese).

*4 S. Kunifuji, K. Furukawa: Present state and objectives of learning system research, Journal of the Society of Instrument and Control Engineers, Vol. 25 No.9, September 1986 (in Japanese).

*5 T. Sugawara, Y. Toyama: Theory of inductive inference, ibid (in Japanese).

*6 I. Tsushima, T. Tashiro, N. Komoda, K. Baba, S. Takakura: Application of rule-based control method to assembly-line control, Transactions of The Society of Instrument and Control Engineers, Vol.21 No.10, pp.1113-1120, October 1985 (in Japanese).

*7 R. Fujiwara, T. Sakaguchi, Y. Kohno, H. Suzuki: An intelligent load flow engine for power system planning, IEEE Proc. of Power Industry Computer Application Conference, pp.236-241, May 1985.

*8 H. Monzen, H. Usami: Commercial AI tool ESHELL using a blackboard model, Nikkei Computer, June 10th 1986 (in Japanese).

*9 S. Kawakita, N. Morita, M. Kohra: Analysis of diagnostic knowledge of large DC-motor and development of diagnosis system based on the result, Proc. of Knowledge Processing Symposium,Information Processing Society of Japan, pp.87-96, Septemer 1985 (in Japanese).

*10 T. Terano, Y. Shinohara, S. Matsui, H. Nakamura, S. Matsuura: Expert system for estimating the life of a dam gate at power plant, Preprint of 32nd Natioal Conference of Information Processing Society of Japan, Theses 21-3, March 1986 (in Japanese).

*11 S. Goto, K. Mori, T. Fujita, K. Mitsumoto: Expert system for designing LSI wiring and shortening interactive wiring processing time, Nikkei Electronics, November 4th 1985.

*12 K. Kikuchi, M. Asukai, E. Kato, T. Sasaki, T. Asano: Expert system for designing lenses, Preprint of Knowledge Engineering and Artificial Intelligence WG, material 44-4, Informatioin Processing Society of Japan, January 1986 (in Japanese).

*13 T. Watanabe, C. Yasunobu, K. Sasaki, T. Yamanaka: Application of knowledge engineering to CAD for equipment layout in a computer room, Preprint of Symbol in Processing WG, material 33-1, Information Processing Society of Japan, September 1985 (in Japanese).

*14 K. Nitta, J. Nagao: Problem solving functions of expert system for Industrial Property, Proc. of the Logic Programming Conference, '85, pp. 375-386, July 3rd 1985 (in Japanese).

*15 A. Ichikawa(Ed.): Theory and method of multi-purpose decision, The Society of Instrument and Control Engineers, 1985 (in Japanese).

*16 S.Kobayashi: Fundamentals and applications of knowledge engineering (5th) - Expert System (I) -, Journal of the Society of Instrument and Control Engineers, Vol.24 No.8, pp730-738, August 1985 (in Japanese).

*17 S.Kobayashi: Fundamentals and application of knowledge engineering (6th) - Expert System (II) -, Journal of the Society of Instrument and Control Engineers, Vol.24 No.9, pp838-840, September 1985 (in Japanese).

*18 M. Suwa: Knowledge engineering and applied artificial intelligence, Systems and Control, Journal of the Japan Association of Automatic Control Engineers, Vol.30 No.5, pp.269-273, 1986 (in Japanese).

*19 R. Nix: Editing by Example, Ph.D. Dissertation, Dept.of Computer Science, Yale Univ. 1983.

Table 1. Outline of Expert Systems Being Surveyed

System \ Item	Classification	Development stage	Development Environment (Shell) (Language) (Machine)	Knowledge Base Size	Inference Engine	Interface with External World	Others
Large DC-motor Diagnosis System	Diagnosis	· Development of prototype completed. (50% of replies from expert) · The version for practical use is now under development.	TDES2 LISP-7 TOSBAC 7/70 (Minicomputer for control)	Rules: 200 (With certainty factor (With text/ modularization) No. of errors: 40 Input data: 1000 records	Forward and backward reasoning Conditional branching Certainty computation	Online data input from monitoring system	
Dam Gate Diagnosis System	Diagnosis	· Development of prototype completed. · This will be improved so that it can be used in the field.	KEE-2 ZETA-LISP SYMBOLICS 3670	Frames: 200 Rules: 40 Methods: 70	Forward and backward Scores for evaluation and multiple solutions	· Relational data base · Numerical analysis program in host computer	
Personal Finance Adviser	Diagnosis (Consultation on real-estate purchase)	· Development of prototype completed.	ESHELL UTILISP FACOM M380R	Frames: 200 Blackboards: 6 Rules: 90 (Knowledge sources: 16)	Forward reasoning Blackboard model	Graphic display on personal computer	
Expert System for Industrial Property	Diagnosis (Consultation on industrial property)	· Development of prototype completed. (Part of Industrial Property) · The whole industrial property will be stored in a PSI machine in the near future.	None KIHPO, Prolog-KABA PC9801	Objects: (50 classes) Interval logic expressions: 40 + Prolog	Backward reasoning as a principle Message passing	None	· Inference by a combination of provisions
Production Line Expert System	Control (assembly line control)	· Version for practical use completed.	(Unknown) (Unknown) HIDIC-V90/50 (Minicomputer for control)	Rules: 528 (With parameters) (User-defined procedures) (rules)	Forward reasoning	Detection of signals from real processes Output of control signals through a micro-computer	· Efficient development of control software · Improvement in modifyability and maintainability Understandable to production engineers
Expert System Supporting Electric Power System Analysis	Planning (Support of electric power system analysis/planning)	· Development of prototype completed. · Knowledge base will be perfected.	None FORTRAN 77, C-PROLOG VAX11/780 (Host) Apollo DN-420 (Work station)	· Rules: 150 · Program base Data base	Backward reasoning	Analysis software Graphic interface	· User-oriented man-machine interface · High-speed analysis
LSI Wiring Design Expert System	Design (With intervention by man)	· Development of a version for practical use completed. (For ordinary experts)	CADLOG Prolog NEC MS4100	Rules: 300 Meta-knowledge	Backward reasoning Control by procedure	CAD system (FORTRAN) CAD data base	
Expert System for Designing Lenses	Design (Support)	· Development of the version for practical use completed. · It will be tested.	Unique shell ZETA-LISP SYMBOLICS 3600	Frames + Rules (may be written in LISP): 500 to 600	Forward reasoning	CAD	
CAD for Equipment Layout in Computer Room	Design	· Development of the version for practical use completed.	None UTILISP M380H	Frames: 400 Rules: 400 Procedures: 1000	Forward reasoning	Graphic routine in FORTRAN	· Layout complying with semantic constraints

Table 2-1 Outline of Knowledge Acquisition(1/2)

Item / System	Knowledge Acquisition Environmentt: Participants	Knowledge Acquisition Environmentt: Knowledge sources	Knowledge Acquisition Environmentt: Support tools	Knowledge Acquisition Process: Basic policy	Knowledge Acquisition Process: Basic chart	Knowledge Acquisition Process: Procedure and Number of Work Steps	Knowledge Acquisition Process: Changes in Representation and Quantity of Knowledge	Knowledge Acquisition Process: Final Classification of Knowledge	Knowledge Acquisition Method	Inquiry	Comment
Large DC motor Diagnosis System	DE 1 KE 2	Experts, technical reports, test, theses, corporate data, field data, hearing	Knowledge base editor, Explanation function	Approximation to expert's knowledge and inference method using PS	Phases ranging from problem identification to test	Problem ID, development phase 1, development phase 2		Knowledge of error causes, error detection, and estimated error causes	Building a diagnostic knowledge system Emphasis placed on research to obtain certain knowledge from uncertain knowledge	Things went well because DEs were also KEs. Much labor went into correcting inconsistent knowledge	
Dam Gate Diagnosis System	DE 2 KE 3	Structure of objects, bearing forms, intrinsic information in DB and FORTRAN, experience	LISP Environment + KEE	Obtaining deep knowledge from existing systems Formatting empirical rules	3 to 4 years for DB and FORTRAN 6 months for ES	Phases for DB & FORTRAN, prototype ES,IPS on M200H, ES for practical use (on Symbolics)	Rules and methods: Refined gradually Frames: Scrap-and-build	DB & FORTRAN, ES program, knowledge base	Rules were added and modified while the system was used. Interview not conducted.	Experiences are well summarized, and an advanced method to use existing system is realised.	Importance of table information Necessity of knowledge representation method of handling numerical and graphic information
Personal finance adviser	DE 2 KE 1	Experts Magazines Textbooks Statistical data	ESHELL utilities, history collection feature, explanation function	Extraction and transformation to match ESHELL knowledge representation	6 phases ranging from user model determination to debugging	6 weeks for image creation, modeling, addition of goals, forecast, and judgment	Large change in hierarchy of frames	Frame BB KS	Extraction from textbooks, interviews, and protocol analysis	The interview method was successful. Advantages and drawbacks of frames, BB, and KS were clarified.	
Expert System Industrial Property	DE 2 KE 1	Experts Books	Editor, translator, inference engine, function explanation f	Reference to the results of jurist analysis	5 phases ranging from problem analysis to evaluation	Start: October '84 Stepwise development and improvement	Version-up from KRIPL-1 to KRIPL-2	Knowledge in KB is written in PROLOG according to internal expression	How provisions are used to solve problems was modeled, and the interpretations stored in KB.	Exceptions and time were expressed successfully. Options in the knowledge representation have problems.	Problems in fact recognition remain unsolved. Problems in processing of precedent DB also remain unsolved.
Production Line Expert System	DE ? KE ?	Experts	Conditions and rules were manually checked for omission before system operation	Unique method to divide control rules in the rule base.	4 phases ranging from system design to debugging	6 man-months in conventional way 1.9 man-months in AI method	Consideration for normal system Consideration for abnormal system Addition and improvement associated with timing control	4 classes such as input control rule and zone control rule	This is a forerunner rule-based system of efficiently developed control software having flexible changeability and high maintainability.	Required software development work steps is 1/3 of the conventional counterparts.	

Table 2-2 Outline of Knowledge Acquisition(2/2)

Item / System	Knowledge Acquisition Environmentt			Knowledge Acquisition Process					Knowledge Acquisition Method	Inquiry	Comment
	Participants	Knowledge sources	Support tools	Basic policy	Basic chart	Procedure and Number of Work Steps	Changes in Representation and Quantity of Knowledge	Final Classification of Knowledge			
Expert System Supporting Power System Analysis	DE 3 KE 2	Experience	DB analysis tool, explanation function, verbal protocol analysis	Extraction by discussion between experts	3 phases ranging from clarifying problems to debugging	New (analytical) knowledge extraction, verbal protocol analysis, feedback loop for adding problems		KB, program base, and DB	Knowledge is acquired from verbal protocol and system operation records	Analysis of verbal protocol clarified problem solving and inference steps.	
LSI Wiring Design Expert System	DE 2 KE 1	Experts Case study Test data	None	Analyzing, summarizing, and expressing problem solution methods used by experts	6 phases ranging from determining problems to improving the knowledge representation method.	400 rules in 2 years	Creation of rules: From simple to complicated rules Creation of metarules: Starts after basic rules are completed.	Procedural knowledge for problem solving, knowledge for circumstantial judgment, and knowledge for display	Creation of exprimental rules by DE and KE, addition of prototypical rules by DE, evaluation and resulting improvement and addition of rules by DE and KE, implementation and acquisition of final rules by DE.	Positive cooperation of DE was obtained because the approach allowed conventional software resources to be used.	Research into 2-dimensional patterns is required (for example, into the method to acquire knowledge automatically from the operation of figure by operater)
Expert System for Designing Lenses	DE 3 KE 3	Mainly experts	None	Listing all design steps Knowledge acquisition through Practical consistant design	Implementation of both interview and apprenticeship methods	The work load depends on whether the interview or apprenticeship method is used.			By the interview method, knowledge is acquired from DE. By the apprenticeship method, KE goes between DE and CAD, and extracts DE's empirical knowledge.	The advantages and drawbacks of the interview and apprenticeship methods were clafiried.	Some of the rules allow the program to be written in LISP
CAD for Equipment Layout in Computer Rooms	DE 3 KE 4	Experts Experts in information processing	Layout manual, knowledge base editor	Developing first a prototype then extending and reinforcing functions	4 phases of: Investigation, Prototype, Extended version, Version-up for field	Investigation: 3 to 6 months (2 men) Prototype: 9 to 12 months (3 men) Extended version: 6 to 9 months (3 men)	The scale of the extended version is almost twice that of the prototype.	Rule, frame, and procedure	A procedure is provided for each of the knowledge types: rule, frame, and procedure	Problems are amplified by writing rules about equipment groups. The knowledge representation form must be well considered.	A good representation form and a good editor are required.