FIFTH GENERATION SYSTEMS

The Fifth-Generation Computer System (FGCS) project was proposed by Japanese researchers in the late 1970s. The word generation characterizes progress in computer development, usually relating to advances in hardware and referring to the elements used in building the computers. Thus the first-generation computers were based on vacuum tubes, the second generation on transistors, the third on Integrated Circuits (ICs). These were followed by computers using Large Scale Integration (LSI) and were denoted the third and a half generation. Today's computers use (Very Large Scale ICs) VLSI, and they constitute the fourth generation. The term *fifth generation* has two meanings: (1) a future computer beyond the fourth-generation computer due to appear in the near future, and (2) a logic-based computer different from conventional von Neumann-type computers which all computers to this date have essentially followed.

The Japanese Ministry of International Trade and Industry (MITI) organized the following committees and projects for the FGCS, where the year denotes the Japanese fiscal year (April through March):

- 1979–1981: Fifth Generation Computer Research Committee
- 1982–1992: FGCS Project
  - 1982–1984: Initial stage
  - 1985–1988: Intermediate stage
  - 1989–1992: Final stage
- 1993–1994: FGCS Follow-on Project

ICOT (Institute for New Generation Computer Technologies) was founded as the core research institute of the Project in 1982 and closed in March, 1995. There have been more than 100 research themes during the FGCS Project. As it is very difficult to explain all of them in detail, this article is restricted to an overview of their history.

PRELIMINARY STUDY FOR THE FGCS PROJECT

The circumstances prevailing during the preliminary stage of the FGCS Project, from 1979 to 1981, can be summarized as follows:

- The state of Japanese computer technology had reached the level of its most up-to-date overseas competitors.
- A change in the role of the Japanese national project for computer technologies was under discussion. It was proposed to work toward improving Japanese industrial competitiveness by catching up with the latest Western computer technologies, reaching a level where Japanese industry would make worldwide scientific contributions. This was to be achieved by the risky development of cutting-edge computer technologies.

In this environment, MITI started a study of a new project—the Fifth Generation Computer Project.

The Fifth Generation Computer Research Committee and its subcommittee (Fig. 1) were established in 1979. It took until the end of 1981 to decide on target technologies and a framework for the project.

Well over 100 meetings were held with a similar number of committee members participating. The following important near-term computer technologies were discussed:

- Inference computer technologies for knowledge processing
- Computer technologies to process large-scale databases and knowledge bases
- High-performance workstation technologies
- Distributed functional computer technologies
- Supercomputer technologies for scientific computation

These computer technologies were investigated and discussed from the standpoint of the important future technologies, social needs, and conformance with Japanese government policy for this national project and with the goal of making important international contributions by developing original Japanese technologies.

Through these studies and discussions, the committee decided on the objectives of the project by the end of 1980 and continued future studies of technical matters, social impact, and project schemes.

The committee's proposals for the FGCS Project are summarized as follows:

1. *Concept of the Fifth Generation Computer.* The Fifth-Generation Computer is to be based on parallel (non-von Neumann) processing and inference processing using knowledge bases as basic mechanisms. In order to realize these mechanisms, the hardware and software interface is to be a logic programming language (Fig. 2):
   - Computer for knowledge information processing system (KIPS)

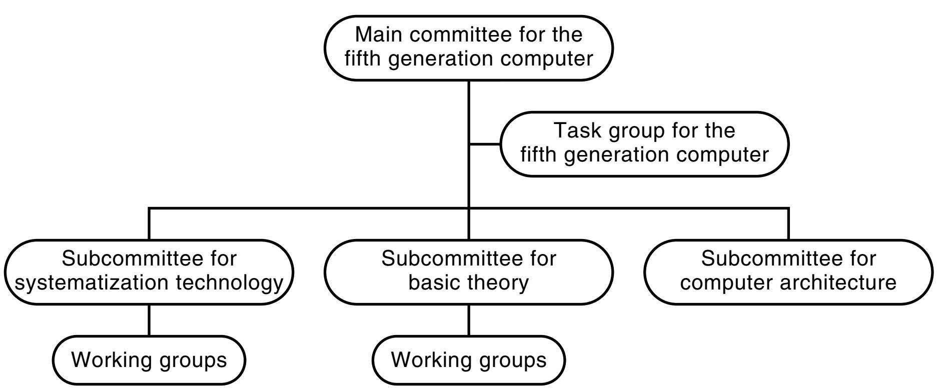

Figure 1. Organization for the Fifth Generation Computer Committee.

J. Webster (ed.), *Wiley Encyclopedia of Electrical and Electronics Engineering.* Copyright © 1999 John Wiley & Sons, Inc.

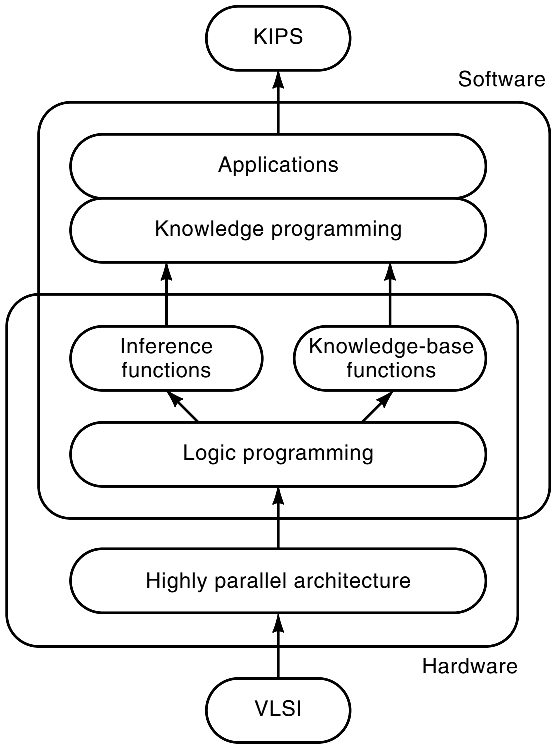

Figure 2. Concepts of the Fifth Generation Computer.

- Basic functions consist of inference using knowledge bases and ease of use as an intelligent assistant for human activities
- Basic mechanisms of hardware and software consist of logical inference processing based on logic programming and highly parallel processing.

2. *Objectives of the FGCS Project.* The objectives were to develop these innovative computers, capable of knowledge information processing, and to overcome the technical restrictions of conventional computers.
3. *Goals of the FGCS Project.* Project goals were to research and develop a set of hardware and software technologies for FGCSs and to develop an FGCS prototype consisting of one thousand element processors with inference execution speeds of between 100M LIPS and 1G LIPS (logical inference per second).
4. *R&D period for the project.* This was a 10-year project, divided into three stages:
   - 3-year initial stage for R&D of basic technologies
   - 4-year intermediate stage for R&D of subsystems
   - 3-year final stage for R&D of a total prototype system

MITI decided to launch the Fifth-Generation Computer System (FGCS) Project as a national project for new information processing and made efforts to acquire a budget for the project. It has been Japan's largest national project in the area of computer technology.

At the same time, the international conference on FGCS (FGCS'81) was prepared and held in October 1981 to announce these plans and to hold discussions on the topic with foreign researchers.

OVERVIEW OF R&D ACTIVITIES AND RESULTS OF THE FGCS PROJECT

Stages and Building in the FGCS Project

The FGCS Project was to investigate a large number of unknown technologies that had yet to be developed. Since this involved a number of risky goals, the project was scheduled over a relatively long period of ten years. Although it was initially designed as a ten-year project, it was later extended to eleven years, and was divided into three stages:

- In the initial stage (April 1982 to March 1985), the purpose of R&D was to develop the basic computer technologies needed to achieve the goal.
- In the intermediate stage (April 1985 to March 1989), the purpose of R&D was to develop small to medium subsystems.
- In the final stage (April 1989 to March 1993), the purpose of R&D was to develop a total prototype system. The final stage was initially planned to be three years. After reexamination halfway through the final stage, this stage was extended to four years to allow evaluation and improvement of the total system in 1992.

Further, a two-year project was added to disseminate the FGCS technologies:

- Follow-on Project (April 1993 to March 1995)

Each year the budget for the following year's R&D activities was decided. MITI made great efforts in negotiating each year's budget with the Ministry of Finance. The budgets for each fiscal year, which are all covered by MITI, are shown in Table 1. The total budget for each year for the three-year initial stage was 8 billion yen. For the four-year intermediate stage, it was about 22 billion yen. The total budget for 1989 to 1992 was about 25 billion yen. Consequently, the total budget for the thirteen-year period was about 57 billion yen.

R&D Subjects of Each Stage

At the beginning, it was considered that an R&D plan could not be decided in detail for a period as long as ten years. The R&D goals and the means to reach these goals also were not decided in detail. During the project, goals were sought and methods decided by referring back to the initial plan at the beginning of each stage. The R&D subjects for each stage,

Table 1. Budgets for the FGCS Project

Budget	1982	1983	1984	1985	1986	1987	1988	1989	1990	1991	1992	1993	1994
Yen	400M	2.7B	5.1B	4.7B	5.55B	5.6B	5.7B	6.5B	7.0B	7.2B	3.6B	1.3B	1.3B
US $	1.86M	12.6M	23.7M	21.9M	34.5M	35.0M	35.6M	40.6M	43.7M	51.4M	25.7M		

Table 2. Transition of R&D Subjects in Each Stage

Initial Stage	Intermediate Stage	Final Stage
Basic Software System • 5G kernel languages • Problem solving & inference module • KB management module • Intelligent interface module • Intelligent programming module	**Basic Software System** • 5G kernel languages • Problem solving & Inference module • KB management module • Intelligent interface module • Intelligent programming module • Experimental application system for basic software	**Experimental Parallel Application System**
Pilot Model for Software Development • SIM hardware • SIM software	**Development Support System** • Pilot model for parallel software development • Network system for development support	**Knowledge Programming System** • Knowledge construction & utilization • Natural language interface • Problem solving & programming (CLP, Prover) • Advanced inference method
Hardware System • PIM functional mechanism • KBM functional mechanism	**Hardware System** • Inference subsystem • KB subsystem	**Basic Software System** • Inference control module (PIMOS) • KB management module (KBMS: Kappa & Quixote)
		Prototype Hardware System

shown in Table 2, were decided by considering the framework and conditions mentioned below.

ICOT defined three groups out of nine R&D subjects at the beginning of the initial stage by analyzing and rearranging the five groups of ten R&D subjects proposed by the Fifth Generation Computer Committee.

At the end of the initial stage, the basic research themes of machine translation and speech, figure, and image processing were excluded from this project because computer vendors' efforts in these technologies had become very active.

In the middle of the intermediate stage, the task of developing a large-scale electronic dictionary was transferred to EDR (Electronic Dictionary Research Center), and development of the CESP (Common ESP system on Unix) was started by AIR (the AI Language Research Center).

The basic R&D framework for promoting this project is to have common utilization of developed software by unifying the software development environment (especially by unifying programming languages). By utilizing software development systems and tools, the results of R&D can be evaluated and improved. In each stage, the languages and the software development environment are unified as follows:

- *Initial stage.* Prolog on DEC machine
- *Intermediate stage.* ESP on PSI and SIMPOS
- *Final stage.* KL1 on Multi-PSI (or PIM) and PIMOS (PSI machines are also used as pseudo multi-PSI systems.)

Overview of R&D Results of Hardware System

R&D on hardware systems was carried out in each stage as listed below.

1. Initial stage
   (a) Functional mechanism modules and simulators for PIM (Parallel Inference Machine) of the hardware system
   (b) Functional mechanism modules and simulators for KBM (Knowledge Base Machine) of the hardware system
   (c) SIM (Sequential Inference Machine) hardware of pilot model for software development
2. Intermediate stage
   (a) Inference subsystem of the hardware system
   (b) Knowledge base subsystem of the hardware system
   (c) Pilot model for parallel software development support system
3. Final stage
   (a) Prototype hardware system

The transition is shown in Fig. 3.

The major R&D results on SIM were the PSI (Personal Sequential Inference machine) and CHI (High-Performance back-end Inference unit). In the initial stage, PSI-I (1(c)) was developed as the KL0 (Kernel Language Version 0) machine.

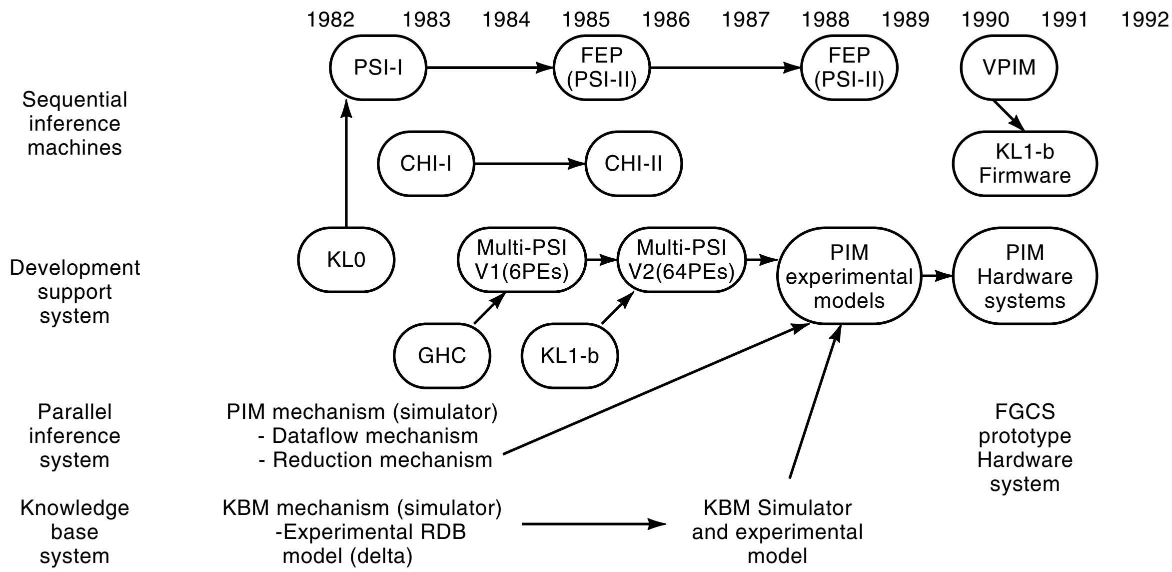

Figure 3. Transition of R&D results of hardware system.

PSI-I had around 35 K LIPS execution speed. Around 100 PSI-I machines were used as main WSs (workstations) for the sequential logic programming language, ESP, in the first half of the intermediate stage. CHI-I (1(c)) achieved around 200 K LIPS execution speed by using WAM instruction set and high-speed devices. In the intermediate stage, PSI was redesigned as the multi-PSI FEP (Front End Processor) PSI-II and has a performance of around 330 K LIPS to 400 K LIPS. CHI was also redesigned as CHI-II (2(c)), with more than 400 K LIPS performance. PSI-II machines were the main WSs for ESP after the middle of the intermediate stage and could be used for KL1 by the last year of the intermediate stage. PSI-III was developed as a commercial product by a computer company by using PIM/m CPU technologies, with the permission of MITI, and by using Unix. R&D on PIM had been continued throughout the project, as follows:

- In the initial stage, experimental PIM hardware simulators and software simulators with 8 to 16 processors were trial-fabricated based on data flow and reduction mechanism (3(a)).
- In the intermediate stage, ICOT developed multi-PSI V1, which was to construct six PSI-Is, as the first version of the KL1 machine. The performance of this machine was only several K LIPS because of the KL1 emulator (2(c)). It did, however, provide evaluation and experience by developing a very small parallel OS in KL1. This meant that ICOT could develop multi-PSI V2 with 64 PSI-II CPUs connected by a mesh network (2(a)). The performance of each CPU for KL1 was around 150 K LIPS, and the average performance of the full multi-PSI V2 was 5 M LIPs. This speed was a significant improvement that encouraged efforts at developing various parallel KL1 software programs, including a practical OS.
- After development of multi-PSI V2, ICOT promoted the design (2(a)) and trial-fabrication of PIM experimental models (3(a)).
- At the final stage, ICOT completed developed prototype hardware consisting of three large-scale PIM modules and two small-scale experimental PIM modules (3(a)). These modules are designed to be equally suited to the KL1 machine for inference and knowledge base management, and to be able to have installed all programs written by KL1, in spite of their using different architectures.

The VPIM system is a KL1-b language processing system which gives a common base for PIM firmware for KL1-b developed on conventional computers.

R&D on KBM continued until the end of the intermediate stage. An experimental relational database machine (Delta) with four relational algebraic engines was trial-fabricated in the initial stage (1(b)). During the intermediate stage, a deductive database simulator was developed to use PSIs with an accelerator for comparison and searching. An experimental system was also developed with multiple name spaces, by using CHI. Last, a knowledge base hardware simulator with unification engines and multiport page memory was developed in this stage (2(b)). ICOT developed database and knowledge-base management software, called Kappa, on concurrent basic software themes. At the beginning of the final stage, ICOT thought that adaptability of PIM with Kappa for the various description forms for the knowledge base was more important than effectiveness of KBM with special mechanisms for the specific KB forms. In other words, ICOT thought that deductive object-oriented database technologies were not yet sufficiently mature to design KBM as a part of the prototype system.

Overview of R&D Results of Software Systems

The R&D of software systems was carried out on a number of subjects listed below in each stage:

1. • Basic software
   (a) 5G kernel languages
   (b) Problem solving and inference software module

(c) Knowledge base management software module
(d) Intelligent interface software module
(e) Intelligent programming software module
(f) SIM software module of pilot model for development support

2. 1(a)–(e) (as in the initial stage)
(f) Experimental application system for basic software module

3. • Basic software system
(a) Inference control module
(b) Knowledge-base management module
• Knowledge problem software
(c) Problem solving and programming module
(d) Natural language interface module
(e) Knowledge construction and utilization module
(f) Advanced problem solving inference method
(g) Experimental parallel application system

R&D Results of Fifth-Generation Computer Languages. As the first step in 5G language development, ICOT designed sequential logic programming languages KL0 and ESP (extended self-contained Prolog) and developed these language processors (1(a)). KL0, designed for the PSI hardware system, is based on Prolog. ESP has extended modular programming functions to KL0 and is designed to describe large-scale software such as SIMPOS and application systems.

As a result of research on a parallel logic programming language, Guarded Horn Clauses (GHC) were proposed as the basic specification for KL1 (Kernel Language Version 1) (1(a)). KL1 was then designed by adding various functions to KL0 such as a macro description (2(a)). KL1 consists of a machine level language [KL1-b (base)], a core language (KL1-c) for writing parallel software and pragma (KL1-p) to describe the division of parallel processes. Parallel inference machines, multi-PSI and PIM, are based on KL1-b. Various parallel software, including PIMOS, is written in KL1-c and KL1-p.

A'um is an object-oriented language. The results of developing the A'um experimental language processor reflect improvements in KL1 (2(a), 3(a)).

To research higher-level languages, several languages were developed to aid description of specific research fields. CIL (complex indeterminate language) is an extended language of Prolog that describes meanings and situations for natural language processing (1(d), 2(d)). CRL (complex record language) was developed as a knowledge representation language to be used internally for deductive databases based on nested relational database software (2(c)). CAL (*contrainte avec logique*) is a sequential constraint logic language for constraint programming (2(b)).

Mandala was proposed as a knowledge representation language for parallel processing but was not adopted because it lacks a parallel processing environment, and there were not enough experience with it in the initial stage (1(c)).

Quixote is designed as a knowledge representation language and knowledge-base language for parallel processing based on the results of evaluation by CIL and CRL. Quixote is also a deductive object-oriented database language and plays the key role in KBMS. GDCC (Guarded Definite Clause with Constraints) is a parallel constraint logic language that processes CAL results.

The transition is shown in Fig. 4.

R&D Results of Basic Software (OS). In the initial stage, ICOT developed a preliminary programming and operating system for PSI, called SIMPOS, using ESP (1(e),(f)). ICOT continued to improve SIMPOS by adding functions corre-

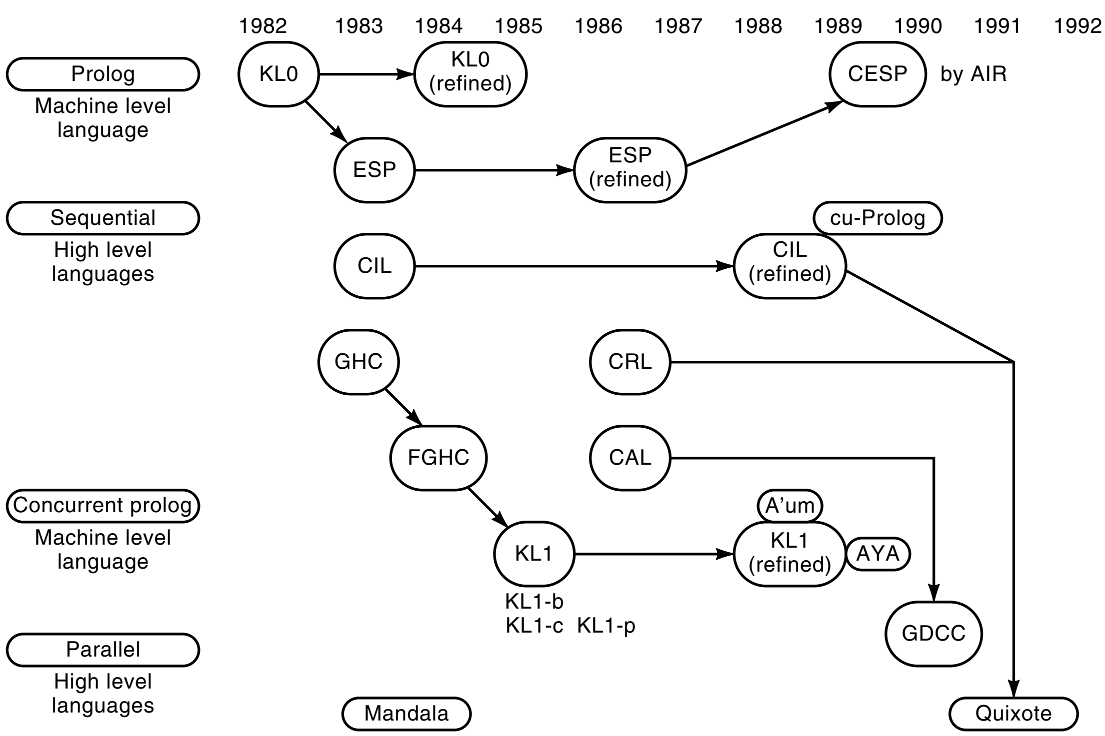

Figure 4. Transition of R&D results of 5G languages.

sponding to evaluation results. ICOT also took into account the opinions of inside users who had developed software for the PSI machine using SIMPOS (2(b),(f)).

Since no previous parallel OS suitable to its goals had been developed anywhere in the world, ICOT started to study parallel OS using our experiences with SIMPOS development in the initial stage. A small experimental PIMOS was developed on the multi-PSI V1 system in the first half of the intermediate stage (2(b)). Then, the first version of PIMOS was developed on the multi-PSI V2 system and was used by KL1 users (2(b)). PIMOS continued to be improved by the addition of functions such as remote access, file access and debugging support (3(a)).

The Program Development Support System was also developed by the end of the intermediate stage (2(b)).

With regard to DB/KB management software, Kaiser was developed as an experimental relational DB management software in the initial stage (1(c)). Then, Kappa-I and Kappa-II were developed to provide the construction functions required to build a large scale DB/KB that could be used for natural language processing, theorem proving and various expert systems (2(c)). Kappa-I and Kappa-II, based on a nested relational model, were aimed at the database engine of deductive object-oriented DBMS.

In the final stage, a parallel version of Kappa, Kappa-P, was developed. Kappa-P can manage distributed databases stored on distributed disks in PIM (3(b)). Kappa-P and Quixote constitute the KBMS.

R&D Results of Problem Solving and Programming Technologies. Throughout this project, from the viewpoint of mathematical theorem proving and program specification, ICOT has been investigating proving technologies. The CAP (Computer Aided Proof) system was experimentally developed in the initial stage (2(e)). TRS (term rewriting system) and Metis were also developed to support specific mathematical reasoning, that is, the inference associated with equality (2(e)).

An experimental program for program verification and composition, Argus, was developed by the end of the intermediate stage (1(e) and 2(e)). These research themes concentrated on R&D into the MGTP theorem prover in the final stage (3(c)).

Meta-programming technologies, partial evaluation technologies, and the learning mechanism were investigated as basic research on advanced problem solving and the inference method (1(b), 2(b), 3(f)). The transition is shown in Fig. 5.

R&D Results on Natural Language Processing Technologies. Natural language processing tools such as BUP (bottom-up parser) and a miniature electronic dictionary were experimentally developed in the initial stage (1(d)). These tools were extended, improved and arranged into LTB (Language Tool Box). LTB is a library of Japanese processing software modules such as LAX (Lexical Analyzer), SAX (Syntactic Analyzer), a text generator and language databases (2(d), 3(d)).

An experimental discourse understanding system, DUALS, was implemented to investigate context processing and semantic analysis using these language processing tools (1(d), 2(d)). An experimental argument system, called Dulcinea, was implemented in the final stage (3(d)).

R&D Results on Knowledge Utilization Technologies and Experimental Application Systems. In the intermediate stage, ICOT implemented experimental knowledge utilization tools such as APRICOT, based on hypothetical reasoning technology, and Qupras, based on qualitative reasoning technology (2(c)). In the final stage, such inference mechanisms for expert systems as assumption-based reasoning and case-based reasoning were implemented as knowledge utilization tools to be applied to the experimental application systems (3(e)).

As an application system, ICOT developed, in Prolog, an experimental CAD system for logic circuit design support in the initial stage. ICOT also developed several experimental expert systems such as a CAD system for layout and logic circuit design, a troubleshooting system, a plant control system, and a go-playing system written in ESP (2(f), etc).

Small to medium parallel programs written in KL1 were also developed to test and evaluate parallel systems by the end of the intermediate stage. These were improved for application to PIM in the final stage. These programs are PAX (parallel semantics system analyzer), Pentomino solver shortest path solver, and Tsume-go.

ICOT developed several experimental parallel systems implemented using KL1 in the final stage, such as the LSI-CAD

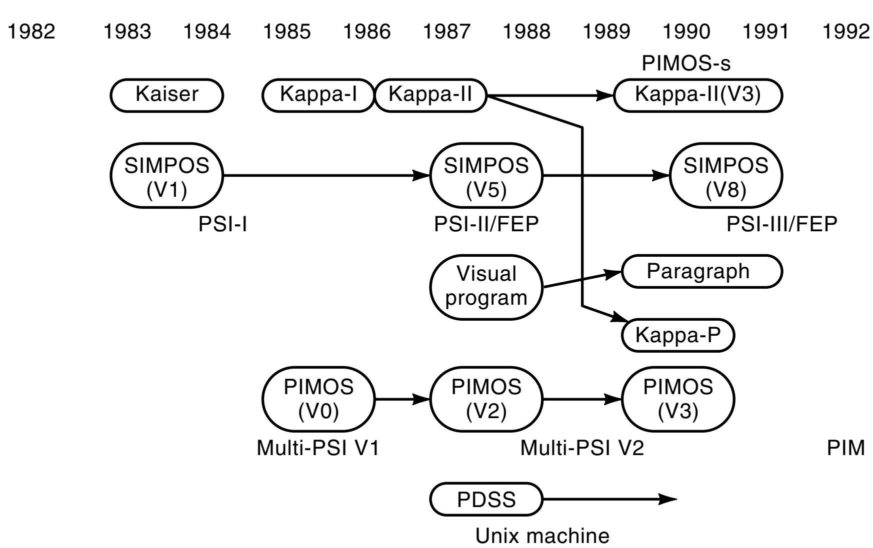

Figure 5. Transition of R&D results of basic software.

system (for logical simulation, wire routing, block layout, and logical circuit design), a genetic information processing system, a legal inference system based on case-based reasoning, and expert systems for troubleshooting, plant control, and go-playing (3(g)). Some of these experimental systems were developed from other earlier sequential systems in the intermediate stage, while others are new application fields that started in the final stage.

Infrastructure of the FGCS Project

As explained earlier, the main language used for software implementation in the initial stage was Prolog. In the intermediate stage, ESP was mainly used, and in the final stage KL1 was the principal language.

Therefore, ICOT used a Prolog processing system on a conventional computer and terminals in the initial stage. SIMPOS on PSI (I and II) was used as the workbench for sequential programming in the intermediate stage. ICOT used PSI (II and III) as a workbench and remote terminals to parallel machines (multi-PSIs and PIMs) for parallel programming in the final stage. ICOT also used conventional machines for simulation to design PIM and a communication (E-mail, etc.) system. With regard to the computer network system, a LAN was used as the in-house system, and a LAN was connected to domestic and international networks via gateway systems.

OVERVIEW OF THE FGCS FOLLOW-ON PROJECT

Motivations and Objectives

The parallel inference system built in the FGCS Project had the following drawbacks:

- PIMs had processing and interprocessor communication hardware specially devised for concurrent logic programming. It had many experimental features and consideration of their cost was premature. This prevented wider availability of the hardware.
- KL1 was the only high-level programming language available on the system. This prevented utilization of already existing software written in other languages.
- The operating system of PIMs had a user interface much different from commonly used operating systems. New users needed to get over this threshold before enjoying the benefits.
- Although KL1 was appropriate for the description of parallel symbolic processing programs, it does not have features of theorem-proving mechanisms for the full first-order logic, and higher level logic programming were needed.

These had been obstacles to broader utilization of the software developed in the FGCS Project. To overcome these obstacles, the following had to be achieved:

- An efficient and portable implementation of KL1 on computer hardware accessible to a wider range of researchers
- Language features to allow smooth linkage with already existing software
- A user interface consistent with widely used operating systems
- A higher level programming language which could provide more general theorem-proving capabilities: such as new common infrastructure for advanced research into computer science and technology

For this reason, ICOT placed all the major software developed in the FGCS Project in the public domain as ICOT Free Software. However, all the major software systems, including the PIMOS operating system, are written in KL1 and able to run only on PIMs. This is obviously an obstacle to their dissemination.

Around the end of the FGCS Project, general-purpose MIMD parallel machines started to appear in the market for large-scale numerical applications. They have the potential to provide greater processing power at low cost in the near future. Furthermore, they are equipped with a Unix-based operating system with some extension for parallel processing such as the software called "Parallel Virtual Machine" (PVM).

In view of this, ICOT and MITI decided to carry out the FGCS Follow-on Project as a two-year project in June 1992. The primary technical goal of the Follow-on Project was to develop a new KL1 and PIMOS environment on Unix-based parallel machines to overcome the above obstacles. This new environment was named KLIC because a KL1 program is compiled into a C program in this environment.

The second goal is to port several new and interesting systems developed in the FGCS Project to Unix-based machines using KLIC. To achieve this goal, these systems had to be made compact so that they could run on the Unix-based machines which are currently much smaller than the PIMs. Furthermore, new user interface portions have to be developed using standard software tools such as the X-windows Motif.

These goals were attained by the end of the Follow-on Project. Furthermore, even after the Follow-on Project was completed, many researchers at ICOT member universities refined and developed more software systems as ICOT Free Software (http://www.icot.or.jp/).

Overview of R&D Activities of the Follow-on Project

In the final stage of the FGCS Project, ICOT had more than 50 research themes, including about 20 parallel application systems. The general technical goal of the Follow-on Project was to make major software systems developed in the FGCS Project operational on Unix-based machines. Thus, the primary technical goal was the development of a new KL1 and PIMOS environment named KLIC on Unix-based machines. In choosing other themes, selection criteria included whether the theme would have a large impact on the future of computer science, and whether progress could be effectively accelerated by the use of parallel processing. The following research themes were chosen for the Follow-on Project:

- Parallel basic software
  1. KLIC system: a KL1 programming environment for sequential and parallel Unix-based machines
  2. Evaluation of PIM architectures and their KL1 language processors
  3. Parallel nested relational DBMS, Kappa

- Knowledge processing software
  1. Parallel theorem prover, MGTP
  2. Knowledge representation languages:
     - Deductive object-oriented language, Quixote
     - Parallel constraint logic programming language, GDCC
     - Heterogeneous distributed cooperative problem solving system, Helios
  3. Generic information processing systems
     - DNA and protein sequence alignment and editing system
     - New algorithms for sequence and structure analysis
     - Biological DBMS and KBMS
  4. Legal reasoning system, new Helic-II

KLIC System. The KL1 system consists of a KL1 compiler and a runtime library. The KL1 compiler is written in KL1 and compiles a KL1 program into a C program. The runtime library is prepared as a library of C programs which provide functions such as debugging, monitoring, parallel execution management, resource management, and so on. These functions are almost the same as the ones which PIMOS provides on the PIMs.

The development of KLIC was done in two steps: sequential version and parallel version. The parallel version uses the software called PVM which provides a standard interface for interprocessor communications over parallel or distributed machines.

Parallel Theorem Prover, MGTP. MGTP is a model generation theorem prover for full first-order logic. It is one of the most successful application programs in the FGCS Project. Generally, it has been well known that theorem provers have a very large search space and thus would be an interesting application of parallel processing. However, its computational structure is very irregular, and it is hard to predict how its search tree will extend its balances. Thus the computation has to be divided into parallel processable operations and allocated to many elemental processors.

MGTP successfully implemented this job division and allocation in the KL1 and PIMOS environment on the PIM model in late 1991. In the Follow-on Project, MGTP became the fastest theorem prover of this kind in the world and proved some open problem in quasi group theory. Furthermore, some of the MGTP provers were provided as tools for practical applications such as the rule-based engine of the Hellic-II legal reasoning system. This indicates that a theorem prover can be regarded as a higher-level inference engine and that it can be adapted for knowledge processing applications such as KBMS, natural language understanding, and software engineering.

Knowledge Representation Languages. In the FGCS Project, the research on knowledge representation language had been based on mathematical logic. The research started from nested relational databases and deductive languages. In the intermediate stage, research on constraint logic programming language started. On the other hand, research on Object-Oriented (OO) languages at ICOT was mainly done with system description languages such as ESP, which is extended Prolog developed as a programming language for PSI machines. ICOT started discussions on how to introduce the merits of OO databases into the above framework.

In the final stage of the FGCS Project, ICOT started the design of a deductive and OO databases language and DBMS, which resulted in the knowledge representation language, Quixote. As Quixote has rich OO based functions combined with a deductive language, it can fulfill the requirements for describing complex knowledge fragments, such as legal reasoning, biological reactions, and the semantic structure of natural language. In the Follow-on Project, intensive efforts were made to improve the language specification and system implementation, and the system was ported to Unix-based machines using KLIC.

Genetic Information Processing Systems. Research on genetic information processing was started with parallel processing of multiple alignment of protein sequences. This research topic had been continued in the Follow-on Project and extended to a sequence alignment and editing system. The system can handle both protein and DNA sequences. Its alignment algorithm is based on a DP matching algorithm. The use of genetic algorithms gives better alignments for some interesting cases. The use of constraints between some amino acids or nucleic acids was attempted in order to narrow the search space. The system was ported to Unix-based parallel machines and is available to biologists. Research on prediction of protein structure was made in the Follow-on Project. The use of the Hidden Markov Model (HMM) gave interesting results in some cases.

Research on biological DBMS and KBMS was made in connection with research on knowledge representation languages. Some biological reactions as well as the characteristics of biological and chemical materials were written in Quixote and stored in the knowledge-base.

Legal Reasoning System, New Helic-II. Research on Helic-II was started in the final stage of the FGCS Project. It was one of the application systems developed to generally evaluate the KL1 and PIMOS environment, knowledge representation languages, and other software tools. To analyze a given case and predict all the possible judgments, Helic-II uses two knowledge-bases and two inference engines (case knowledge-base with case-based reasoning and rule knowledge-base with rule-based reasoning). The rule-based engine was built on the MGTP theorem prover as its kernel. In the FGCS Follow-on Project, the new Helic-II added more sophisticated functions to Helic-II in order to simulate a debate between a prosecutor and a lawyer.

This system was very successful not only in demonstrating the usefulness of the FGCS technology, but also in showing possible uses for the FGCS technology which are beyond initial expectations. It also showed a better understanding of how FGCS technology can be used for applications in social-science areas.

PROMOTING ORGANIZATION OF THE FGCS PROJECT

ICOT was established in 1982 as a nonprofit core organization for promoting the FGCS Project and it began R&D work on fifth-generation computers in June 1982, under the auspices of MITI. Establishment of ICOT was decided after con-

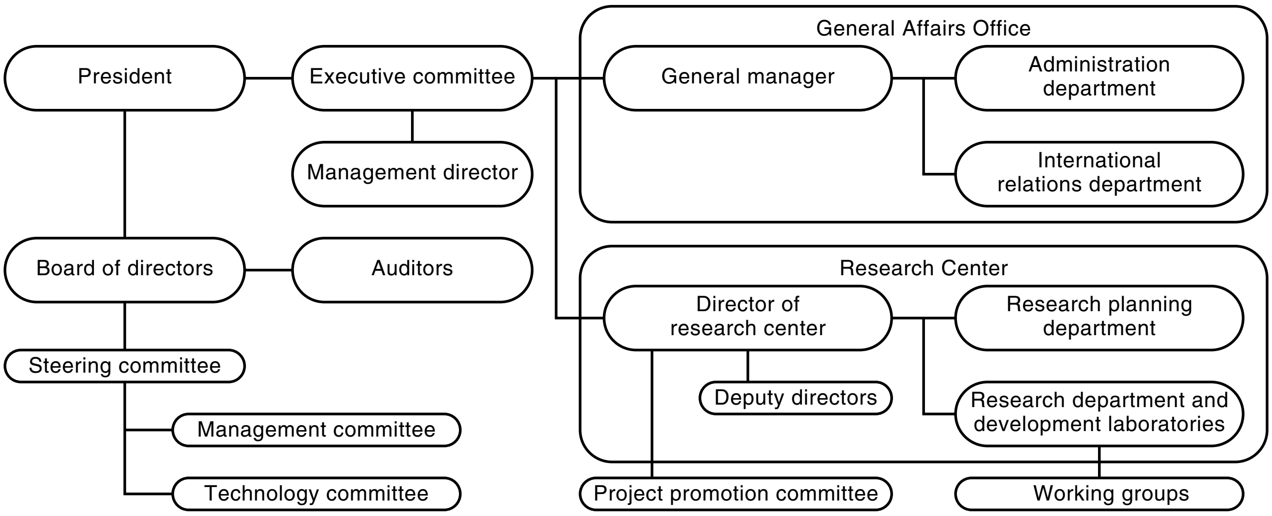

Figure 6. ICOT organization.

sidering the following needs and the efficiency offered by a centralized core research center for promoting original R&D:

- R&D themes should be directed and selected by a strong leadership, with the goal of hardware and software integration based on a unified framework of fifth-generation computers, throughout the ten-year project period.
- It was necessary to develop and nurture researchers working together because of the shortage of researchers in the field.
- A core center was needed to exchange information and to collaborate with other organizations and outside researchers.

ICOT consisted of a general affairs office and a research center, as shown in Fig. 6.

The organization of the ICOT research center was changed depending on the progress being made, as shown in Fig. 7. In the initial stage, the research center consisted of a research planning department and three research laboratories. The number of laboratories was increased to five at the beginning of the intermediate stage. These laboratories became one research department and seven laboratories in 1990.

The number of researchers at the ICOT research center increased yearly, from 40 in 1982 to 100 at the end of the intermediate stage.

All researchers at the ICOT research center were transferred from national research centers, public organizations,

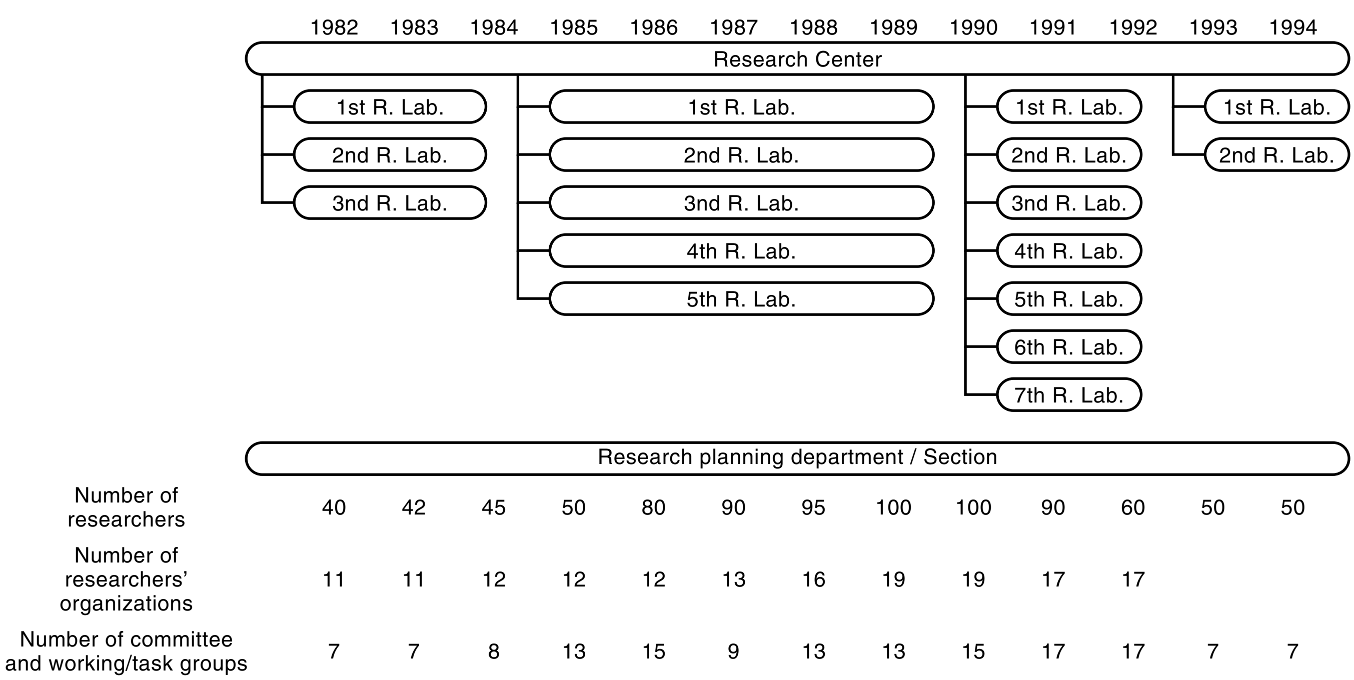

Figure 7. Transition of ICOT research center organization.

computer vendors, and the like. To encourage young creative researchers and to promote imaginative R&D, the age of researchers was limited to 35 years. Because all researchers were normally transferred to the ICOT research center for three to four years, ICOT had to continually receive and nurture newly transferred researchers. ICOT had to make considerable effort to continue to consistently advance R&D despite researcher rotation. This rotation meant that ICOT was able to maintain a staff of researchers in their thirties, and could also easily change the structure of its organization.

In total, 184 researchers had been transferred to the ICOT research center for an average transfer period of 3 years and eight months. The number of organizations that transferred researchers to ICOT also increased, from 11 to 19. This increase in participating organizations was caused by the habit of the approximately 30 supporting companies, to transfer researchers to ICOT midway through ICOT's intermediate stage.

The themes each laboratory was responsible for changed occasionally depending on the progress made. Figure 8 shows the assignment of research themes to each laboratory in the final stage.

Every year several visiting researchers were invited from abroad for several weeks at ICOT's expense for discussions and to exchange opinions on specific research themes with ICOT researchers. During the project, 74 researchers from 12 countries were invited in this program.

ICOT also received six long-term (about one year each) visiting researchers from foreign governmental organizations based on memorandums with the National Science Foundation (NSF) in the United States, the Institute National de Recherche en Informatique et Automatique (INRIA) in France, and the Department of Trade and Industry (DTI) in the United Kingdom (Fig. 9).

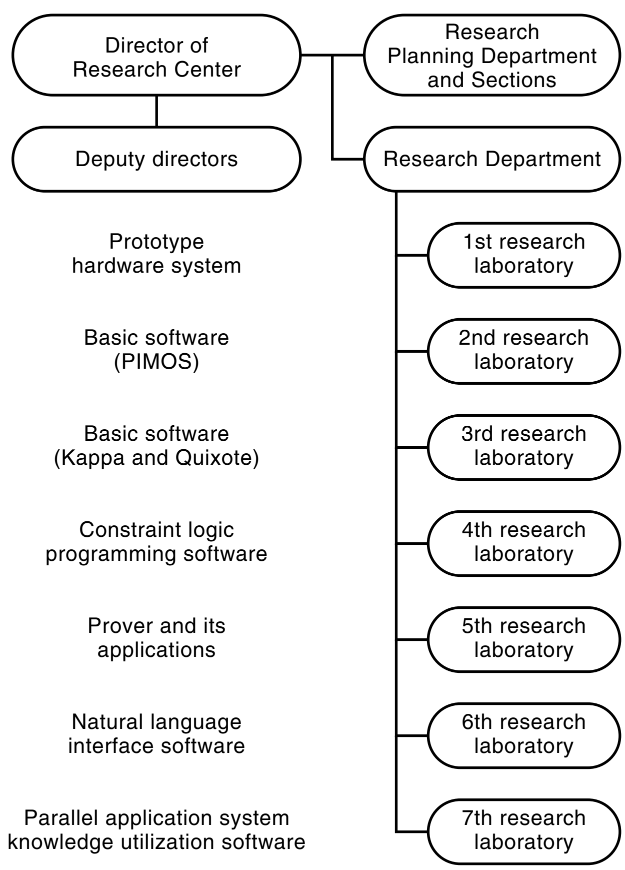

Figure 8. ICOT research center organization.

Figure 9 shows the overall structure for promoting this project. The entire cost for the R&D activities of this project was supported by MITI based on the agreement between MITI and ICOT. Yearly and at the beginning of each stage, ICOT negotiated its R&D plan with MITI. MITI received advice of this R&D plan and evaluations of R&D results and ICOT research activities from the FGCS Project advisory committee.

ICOT executed the core part of R&D and had contracts with eight computer companies for experimental production of hardware and development software. Consequently, ICOT could handle all R&D activities, including the developmental work of computer companies toward the goals of this project.

ICOT set up committees and working groups to discuss and to exchange opinions on overall plans, results, and specific research themes with researchers and research leaders from universities and other research institutes. Considering construction and the number of people in each working group, around 10 to 20 members, the total number in the committee and working groups was about 150 to 250 each year.

DISTRIBUTION OF R&D RESULTS AND INTERNATIONAL EXCHANGE ACTIVITIES

Because the project was a national project in which worldwide scientific contribution was very important, ICOT had made every effort to include its R&D ideas, processes, and project results when presenting ICOT activities. ICOT also collaborated with outside researchers and other research organizations.

These efforts had contributed to progress in parallel and knowledge processing computer technologies. The R&D efforts in these fields have increased because of the stimulation provided by the project. The R&D efforts continued to increase through distribution of R&D results. Many outside researchers have also made significant contributions to the FGCS Project through their discussions and information exchanges with ICOT researchers.

For example, GHC, a core language of the parallel system, was created through discussion with researchers working on Prolog and Concurrent Prolog. The Performance of the PSI system was improved by introducing the WAM instruction set proposed by Professor Warren.

ICOT had several programs for distributing the R&D results of the project, to exchange information and to collaborate with researchers and organizations.

1. The important way to present R&D activities and results was publication and distribution of ICOT journals and technical papers. ICOT published and distributed quarterly journals, which contain introduction to ICOT activities and technical papers to more than 600 locations in 35 countries. ICOT periodically published and sent more than 1800 technical papers to around 30 overseas locations. TRs (technical reports) and TMs (technical memos) were sent on request to foreign ad-

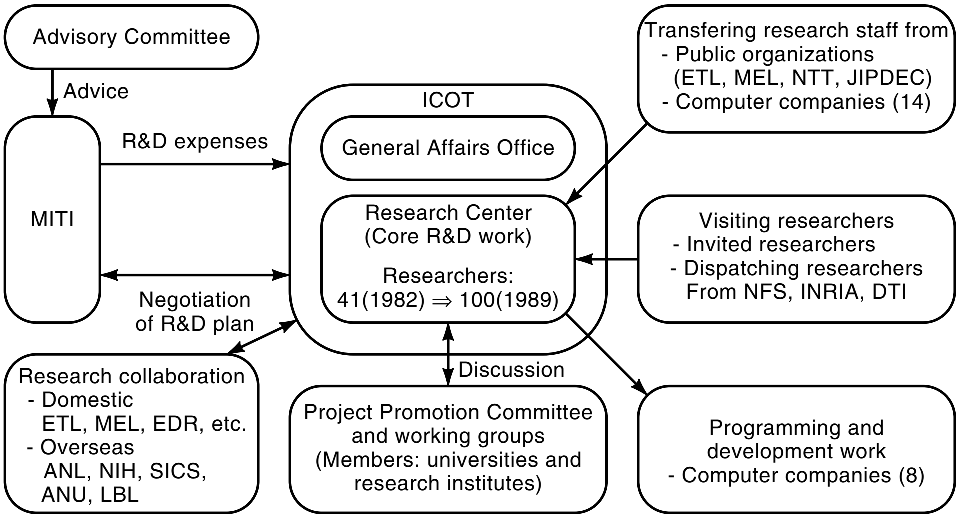

Figure 9. Structure for promoting FGCS project.

dresses. These technical papers consisted of more than 700 TRs and 1100 TMs published since the beginning of the project. A third of these technical papers were written in English.

2. In the second program, ICOT researchers had discussed research matters and exchanged information with outside researchers.
   - ICOT researchers had made more than 450 presentations at international conferences and workshops, and at around 1800 domestic conferences and workshops. They had visited many foreign research organizations to discuss specific research themes and to explain ICOT activities.
   - Every year, ICOT had welcomed around 150 to 300 foreign researchers and specialists in other fields to exchange information with them and explain ICOT activities to them.
   - As already described, ICOT invited 74 active researchers from specific technical fields related to FGCS technologies. ICOT also received six long-term visiting researchers sent from foreign governmental organizations based on mutual agreements. These visiting researchers conducted research at ICOT and published the results of that research.
3. ICOT sponsored the following symposiums and workshops to disseminate and exchange information on the R&D results and on ICOT activities:
   - ICOT hosted the International Conference on FGCS in November 1984 (1). Around 1,100 persons participated and the R&D results of the initial stage were presented. This followed the International Conference on FGCS in 1981 (2), in which the FGCS Project plan was presented. ICOT hosted the International Conference on FGCS in November 1988 (3). 1,600 persons participated in this symposium, and the R&D results of the intermediate stage were presented. Further, ICOT hosted the International Conference on FGCS in December 1992 (4), where the R&D results in the final stage were presented and demonstrated.
   - ICOT had held seven Japan-Sweden (or Japan-Sweden-Italy) workshops since 1983 (cosponsored with institutes or universities in Sweden and Italy), four Japan-France AI symposiums since 1986, four Japan-US AI symposiums since 1987 (cosponsored with NSF of the US), and two Japan-U.K. workshops since 1989 (cosponsored with DTI of the U.K.).
   - ICOT also hosted domestic symposiums on the project and logic programming conferences every year.
4. Because the entire R&D cost of the project was provided by the government, such intellectual property rights (IPR) as patents, which had been produced in the project, belong to the Japanese government. These IPRs are managed by AIST (Agency of Industrial Science and Technology). Any company wishing to produce commercial products that use any of these IPRs must get permission to use them from AIST. For example, PSI and SIMPOS were commercialized by companies licensed by AIST. The framework for managing IPR must impartially utilize IPR acquired through the project. That is, impartial permission to domestic and foreign companies, and among participating companies or others is possible because of AIST.
5. Software tools developed in the project that are not yet managed as IPR by AIST can be used by other organizations for noncommercial aims. ICOT started its World Wide Web server (http://www.icot.or.jp/) in October, 1994. These software tools were distributed through the above URL by ICOT until September, 1995, and by AITEC (Research Institute for Advanced Information Technology) from October, 1995, according to the research tools permission procedure. They include more than 20 software tools, such as PIMOS, PDSS, Kappa-II, the A'um system, LTB, the CAP system, the cu-Prolog system, and the TRS generator. In other cases, ICOT made the source codes of some programs public by printing them in technical papers.
6. On specific research themes in the logic programming field, ICOT has collaborated with organizations such as Argonne National Laboratory (ANL), National Institute for Health (NIH), Lawrence Berkley Laboratory (LBL), Swedish Institute of Computer Science (SICS), and Australia National University (ANU). Furthermore, in the Follow-on Project, ICOT set up tighter collaborative re-

search projects with the University of Bristol and the University of Oregon.

Around the end of the FGCS Project, MITI organized a high-level committee to assess the research results of the FGCS Project (5). One of its conclusions was that the results were considered to be still so far away from the market's needs that computer companies could not commercialize them in a few years although they can be of great value from an academic point of view.

After ICOT was closed, AITEC was established in October, 1995, as a subsidiary organization to the Japan Information Processing Development Center (JIPDEC). It has two roles:

1. Survey and assessment of R&D of information technology and forecasting of future important R&D topics
2. Dissemination of ICOT Free Software (IFS) and further development of parallel and knowledge processing software.

To realize the second purpose, AITEC not only distributes IFS, but also maintains the software, and promotes development of new software related to IFS. Through these activities, it intends to expand and disseminate parallel symbol processing and knowledge processing technologies.

CONCLUDING REMARKS

During the FGCS Symposium in 1994, the FGCS Project Evaluation Workshop (6) was held in Tokyo. After the FGCS Project, two special issues (7,8) were published, and after the FGCS Follow-on Project, one special issue (9) was published. From both academic and commercial points of view, most authors tried to assess the results and its future works.

In (7), Professor Ken Kahn states, "I do not believe the Fifth Generation Project is a failure because they failed to meet many of their ambitious goals; I think it is a great success because it helped move computer science research in Japan to world-class status and nudged computer science research throughout the world in a good direction." Many authors in (7) pointed out the discrepancy between the vision of the project that the promoters had popularized initially, and academic achievements in the related areas. Furthermore, from another point of view, Professor Evan Tick writes in (7), "ICOT did not create a revolution because it did not fundamentally change the manufacturers . . .. Either another project, or a radical restructuring of the diametric cultures of education and industry, will be required to propagate the advances made in the FGCS project," which is also quoted in Professor Edger Feigenbaum's article in (8).

Many of ICOT researchers who had been at ICOT for several years and led the FGCS Project have been welcomed by many universities. They have continued R&D on FGCS technologies: mainly parallel processing and knowledge information processing. Not only the results of the two projects but also their extensions and new results have been distributed as ICOT Free Software through AITEC's Web page (http://www.icot.or.jp). The number of visits have increased. The basic framework of FGCS technologies based on logic programming is likely to be more promising in the near future.

ACKNOWLEDGMENTS

It is very difficult to describe all R&D activities during the FGCS Project and the FGCS Follow-on Project, because there had been many research themes in various areas related to FGCS technologies. Takashi Kurozumi summarized the FGCS Project at FGCS'92 (10) and Shuichi Uchida summarized the FGCS Follow-on Project at FGCS'94 (11). This article is mainly based on the above two reports. The author especially wishes to thank Takashi and Shunichi for their permission to use their work.

BIBLIOGRAPHY

1. ICOT (ed.), *Proc. Int. Conf. of Fifth Generation Comput. Syst.* (FGCS'84), Tokyo: ICOT, 1984.
2. Toru Motooka et al. (eds.), *Proc. Int. Conf. of Fifth Generation Comput. Syst.* (FGCS'81), Tokyo: JIPDEC, 1981.
3. ICOT (ed.), *Proc. Int. Conf. of Fifth Generation Comput. Syst.* (FGCS'88), Tokyo: ICOT, 1988.
4. ICOT (ed.), *Proc. Int. Conf. of Fifth Generation Comput. Syst.* (FGCS'92), Tokyo: Ohmsha and IOS Press, 1992.
5. Committee for Development and Promoting of Basic Computer Technology, Fifth generation computer technology, *ICOT J.,* **40**: 2–24, 1994.
6. S. Uchida (ed.), *Proc. FGCS Project Evaluation Workshop,* Tokyo: ICOT, 1992.
7. E. Shapiro and D. H. D. Warren (eds.), Special Section on the Fifth Generation Project, *Commun. ACM,* **36** (3): 46–101, 1993.
8. R .P. van de Riet (ed.), Special Issue on FGCS Conference, 1992, *Future Generation Comput. Syst.,* **9** (2): 79–158, 1993.
9. Keiji Hirata (ed.), Special Issue on Results and Future Works of the Fifth Generation Computer Project, *J. Inf. Process. Society of Jpn,* **37** (5): 399–448, 1996 (in Japanese).
10. T. Kurozumi, Overview of the Ten Years of the FGCS Project, in *Proc. Int. Conf. of Fifth Generation Comput.Syst.* (FGCS'92), Tokyo, Ohmsha and IOS Press, 1992, pp. 9–19.
11. S. Uchida, General Report of the FGCS Follow-on Project, in *Proc. Int. Symp. of Fifth Generation Comput. Syst.* (FGCS'94), Tokyo: ICOT, 1994, pp. 1–9.

Reading List

A. Aiba, Constraint Logic Programming System: CAL, GDCC and Their Constraint Solvers, *Proc. Int. Conf. of Fifth Generation Comput. Syst.* (FGCS'92), Tokyo: Ohmsha and IOS Press, 1992, pp. 113–131.

T. Chikayama, Operating System PIMOS and Kernal Language KL1, *Proc. Int. Conf. of Fifth Generation Comput. Syst.* (FGCS'92), Tokyo: Ohmsha and IOS Press, 1992, pp. 72–88.

T. Chikayama, Parallel Basic Software, *Proc. Int. Symp. of Fifth Generation Comput. Syst.* (FGCS'94), Tokyo: ICOT, 1994.

E. Feigenbaum and P. McCorduck, *The Fifth Generation Artificial Intelligence and Japan's Computer Challenge to the World,* Reading, MA: Addison-Wesley, 1983.

T. Fujise et al., KLIC: A Portable Implementation of KL1, in *Proc. Int. Symp. of Fifth Generation Comput. Syst.* (FGCS'94), Tokyo: ICOT, 1994, pp. 66–79.

K. Furukawa, Summary of Basic Research Activities of the FGCS Project, in *Proc. Int. Conf. of Fifth Generation Comput. Syst.* (FGCS'92), Tokyo: Ohmsha and IOS Press, 1992, pp. 9–19.

R. Hasegawa and M. Fujita, Parallel Theorem Solvers and Their Applications, in *Proc. Int. Conf. of Fifth Generation Comput. Syst.* (FGCS'92), Tokyo: Ohmsha and IOS Press, 1992, pp. 132–154.

R. Hasegawa, Parallel Theorem-Proving System: MGTP, in *Proc. Int. Symp. of Fifth Generation Comput. Syst.* (FGCS'94), Tokyo: ICOT, 1994, pp. 51–55.

ICOT (ed.), *Proc. of the Int. Symp. of Fifth Generation Comput. Syst.* (FGCS'94), Tokyo: ICOT, 1994.

K. Kawanobe et al., Research and Development, in *Proc. Int. Conf. of Fifth Generation Comput. Syst.* (FGCS'84), Tokyo: ICOT, 1984.

T. Kurozumi et al., ICOT Research and Development, in *Proc. Int. Conf. of Fifth Generation Comput. Syst.* (FGCS'88), Tokyo: ICOT, 1988.

K. Nitta, K. Nitta, and N. Ichiyoshi, Experimental Parallel Inference Software, in *Proc. Int. Conf. of Fifth Generation Comput. Syst.* (FGCS'92), Tokyo: Ohmsha and IOS Press, 1992, pp. 166–190.

K. Nitta et al., Knowledge Information Software, in *Proc. Int. Symp. of Fifth Generation Comput. Syst.* (FGCS'94), Tokyo: ICOT, 1994, pp. 18–34.

K. Taki, Parallel Inference Machine PIM, in *Proc. Int. Conf. of Fifth Generation Comput. Syst.* (FGCS'92), Tokyo: Ohmsha and IOS Press, 1992, pp. 50–72.

Y. Tanaka, Natural Language Processing Software, in *Proc. Int. Conf. of Fifth Generation Comput. Syst.* (FGCS'92), Tokyo: Ohmsha and IOS Press, 1992, pp. 155–165.

S. Uchida, Summary of the Parallel Inference Machine and its Basic Software, in *Proc. Int. Conf. of Fifth Generation Comput. Syst.* (FGCS'92), Tokyo: Ohmsha and IOS Press, 1992, pp. 33–49.

S. Uchida et al., Outline of the FGCS Follow-on Project, *New Generation Computing,* **11** (2): 1993.

K. Yokota and H. Yasukawa, Towards an Integrated Knowledge-Base Management System: Overview of R&D on Databases and Knowledge-Bases in the FGCS Project, in *Proc. Int. Conf. of Fifth Generation Comput. Syst.* (FGCS&92), Tokyo: Ohmsha and IOS Press, 1992, pp. 89–112.

K. Yokota, From Databases to Knowledge-Bases—Kappa, Quixote, Helios, in *Proc. Int. Symp. of Fifth Generation Comput. Syst.* (FGCS'94), Tokyo: ICOT, 1994, pp. 35–50.

KAZUMASA YOKOTA
Okayama Prefectural University