

# SYSTEMS APPROACH TO COMPUTING DEPENDABILITY IN AND OUT OF HITACHI

## *Concept, Applications and Perspective*

Hirokazu Ihara\* and Motohisa Funabashi\*\*

\* *Hiro Systems Laboratory, ihara@coral.ocn.ne.jp*

\*\* *Systems Development Laboratory, Hitachi, Ltd., funa@sdl.hitachi.co.jp*

**Abstract:** Authors and their colleagues have been contributing to dependable computing for the wide lines of Hitachi products since 1960. Fault-tolerance concept was not permitted under the company motto that no fault by fault-avoidance technology was one of engineering achievements. Suppose the bullet train system “Shinkansen” would have a failure, no need to explain social damage. Nowadays dependability concept seems accepted but failures are never permitted to keep our society safe and comfortable. In order to meet it, we took systems architecture and intelligence approaches chasing property of creatures from architectural and intellectual viewpoints though we never gave up fault-avoidance. Autonomous Decentralization Concept (ADC) was extracted in late 1970s. The basic principle is that the system state with no faulty part is only one of numerous states. Intelligent behavior is necessary to play autonomy and decentralization. We started AI technology around 1980. Hitachi Group has applied autonomous decentralization systems technology and AI technology to its computing systems and enjoyed their business with good reputation. The authors present their R&D activities namely at Systems Development Laboratory and practices in and out of the group. Finely future R&D horizon for dependable information systems is presented as the extension of the past and the present R&D.

**Key words:** fault-tolerance, dependability, autonomous decentralization, artificial intelligence, assurance

## 1. INTRODUCTION

Hitachi, Ltd. has had two major computer groups, that is, the computer systems business group which corresponds to business process and the

control systems business group which corresponds to control process. The former supplies the computer systems itself and, for the latter group, the computer system is only the attachments of industrial plants or apparatuses. The critical items of the computer control area compared to other as follows,

- (a) The faults of the control equipments or apparatuses cause large damage or disaster to the society or the business while they control on line and in real-time. They are mission-critical real-time systems with long MTBF. Thus, the reliability of the control systems is prior to its performance and cost.
- (b) The environment conditions such as ambient temperature, humidity, vibration, dust, etc. are very rigorous compared to those in normal environment.
- (c) The control functions are triggered by event-driven or data-driven means from the machines of processes. Therefore response time and processing time have to be within the limitation even if many tasks occurred at once. They can be the time-critical systems. In addition, both on-line and offline tasks exist together under one OS.
- (d) Manufacturers have to deliver the systems that are designed, implemented, tested, installed, and validated according to the approved specifications by the buyer. This business model may be designated as, so-called, those of the turnkey or the solution.
- (e) The systems used periodically to adapt their functions by changing demands and to metamorphose their hardware and software. Even various tests should be done without any interruption while the systems are working.

Therefore, we started our development on our own terms in 1960 and have developed many types of the computer control systems and delivered them to national infrastructure industries from our independent systems' approach standpoint.

Systems Development Laboratory of Hitachi, Ltd. has been taking the leadership in R&D of both business and control computer systems since its establishment in 1973. Here, we will namely describe on R&D activities with practical reality and its surrounding engineering of system dependability from its beginning and the coming perspectives regarding dependability. Dr. Ihara of System Development Laboratory had first acquaintance with Prof. Avižienis at UCLA in 1978. IFIP WG 10.4 was started after the working conference in 1979 in London [Ih 79]. At that time, Japanese had not accepted fault-tolerance approach since they had chased fault-avoidance in component level and get good product reputation as its result. Furthermore, the dependability concept published by the Working Group in 1992 took long time for Japanese industry, as well academia, to take in instead of reliability, until late 1990s [La 85] [La. 92].

## **2. FOOTPRINTS OF HITACHI'S CONTRIBUTIONS TO DEPENDABLE COMPUTING**

We have more than 40 years of research and development activity chasing dependable computing and supplied dependable systems to the market with high reputation.

### **2.1 Before 1970s**

The beginning of 1960s was when transistor as well as digital computers was rapidly introduced into business. Automation and systems engineering were the key words as advanced business. Several introduction of computer to real-time control were attempted in leading industries, such as iron and steel making process, electricity generation, electricity dispatching, chemical process, manufacturing automation, and so on. The digital computer control development at Hitachi started with development of the data logger from the beginning of 1960s after analog control apparatuses widely in industry. The reliability of the computer could not afford to industrial requirements, such as its ambient temperature, humidity, vibration, and shock as well as to its cost.

Semi-conductor as logics was in the rapid change to LSI. Hitachi developed three mini-computer systems named HIDIC according to the advent and applied them as infant systems to heavy industries, oxygen steel furnaces, slab mills, blast furnaces, thermal turbines, hot strip mills, chemical plants, electric dispatches, subway train dispatches, and so on. They realized fairly accepted reliability by fault avoidance technologies, for examples, selection of component, less components and connections, reduction of rate, and accelerated aging. However the reliability of computer was insufficient for critical applications. The system was single and stand-alone with centralized architecture with limited functions because of balance of hardware cost and reliability. It is gradually recognized that fault-avoidance could never achieve 100% operation.

### **2.2 In 1970s**

In 1970s, mini-computer systems were widely introduced in to the market. Mainstream of computer introduction to industry was still single architecture systems with manual back-up. However customers were gradually apt to accept redundant system architecture for critical system management and control. Software development cost emerged as one of constraints. The requirements of limited functions were strictly decided and



### **2.3 In 1980s**

In 1980s, the mini-computer steeply improved its size, number of component, performance, and cost in addition to reliability. The rapid advent of minicomputers made the systems widely available to be distributed as well as multiplex. The same and/or different functions were run respectively in assigned processors. The property of centralized and distributed architecture alike was namely fit to the needs from system manufactures. They could choose and connect their subsystems to meet their variety of product lines. The rigid communication protocol was standardized for local and wide area networks. The hierarchical and functional distributed architectures were widely applied to various industries. Hitachi delivered their unique architecture of HIDIC with global memory, task synchronization, and buss control for the dual or duplex architecture systems to life critical applications [Ka78].

Japanese economy rocketed up and the increasing mass production needed computerized lines. The larger and more complicated the system grew, the more difficult it became to construct and to isolate its faulty portions, which made service undependable and partially shut down. Sometimes the fault caused the domino effect. In addition, expandability and extensibility during on-line operation became necessary instead of replacement. The information systems of multi-national enterprises over the world introduced non-stop computer architecture. Advanced HIDIC were introduced to the transaction processing for its non-stop property and extensibility.

In order to break through these difficulties, we proposed new idea “Autonomous Decentralization Concept (ADC)” analogous to bio-structure. The first introduction as Autonomous Decentralized Systems (ADS) was the subway control mid 1980s when subways or new transportation systems were appearing in many major cities. Thereafter almost all subway control systems as well as suburban railways adopted ADS in addition to industrial applications [MiMo 84] [MoMi 84]. We proposed new measurement index, “Functionability” that shows the probability of functions working in real time [MoIh 82] [OrKa 84] [OrMo 92] [KeBe 01]. That is to say, it is a product of summation of the working durations of invoked tasks divided by their summarized mission time.

Multiplex architecture consisted of three off-the-shelf M-68000 bare microchips, produced by Hitachi, with stepwise-negotiating-voting (SNV) method was developed for aerospace [KaMa 89]. It was launched in January 1990 as the communication system on MUSES (HITEN) satellite to play moon-swing-by and attained healthy operation over three mission years and eight swing-bys. It was furnished the spatiotemporal design diversity and the

autonomous decentralized architecture both in hardware and software. Bare chips, a BPU, memories, controllers were mounted on a printed multilayer ceramic plate, 3 x 3 x 0.5 inches in size, three of which contained in anti-radiation metal. During its mission time, the system worked successfully despite 665 soft errors in three CMOS 64MB memories by cosmic rays.

We started research of artificial intelligence, knowledge engineering and fuzzy logics prior to national project “Fifth Generation Project” in 1980. Knowledge engineering was practically applied to nuclear power plant control, CMOS memory fabrication process and project management as the trials within the company. Fuzzy logics were applied to operation of the trains of Sendai municipal subway, which was the first commercial success in the world [YaMi 83]. Neural network technology was also in the scope of our research. It was applied to the crown control of cold strip rolling mills.

## **2.4 In 1990s**

In early 1990, Japanese society turned to new era due to collapse of bubble economy and rapid inclination to the aging and information society as well as world wide environmental conservation. There came big innovational wave of business models by the computer networks and the personal computer. People’s needs diversified for their favorites, by which the product lines were encountered multiple-type and small lots instead of mass production. In addition, there occurred many disasters as well as failures by human-made faults in business, industrial, social and aerospace area, where dependability of human intelligence and behavior in computerized environment was taken up for discussion. User engineering was one solution to avoid human error and to develop huge systems including human being, either professional or nonprofessional.

Production cost threat forced for us to use PC as real-time control though the system nucleus still consisted of our original HIDIC. Programmable logic controllers (PLC) also controlled widely various machineries in production lines through LAN. We internationally proposed the interface of LAN by ADC to prevent domino effect and maintain assurance in operation, and legislated in 1996 by Open DeviceNet Association (OVDA) [ToSh 99]. All the efforts were integrated into an ADS product series named NX.

According to diffusion of Internet services, application of ADC was extended to the Internet service systems named Autonomous Decentralized Service Systems (ADSS). The essence of ADSS was an abstract model of Internet services comprising with three players; service requesters that request and use services, service providers that provide services, and service mediators that mediate services among requesters and providers. In ADSS, service mediators played a role of the data field in ADS with intelligent

brokerage functions that permit frequent joining and leaving of the system participants. The model was adopted into the OMG (Object Management Group) standard as TSAS (Telecommunication Service Access and Subscription), and was applied to Japanese and German Government Information Service Systems [KoAg 99][FuTo 01][ToFu 01].

We had initiative establishment of the international symposium of ADS (ISADS) of IEEE in 1993. One hundred and fourteen researchers of systems Science in academia set up the basic research project supported by Ministry of Education & Science for three years from 1990.

We can pick up COSMOS and ATOS for train management and control systems as our excellent results [SaKa 97].

According to the socio-economical change, COSMOS system based on Autonomous Decentralization Concept was developed as the descendant of COMTRAC for the Tohoku and the Jyoetsu Lines in 1995. The name stands for Computerized Safety, Maintenance and Operation System. The systems, HIDIC based, set up at each station and offices were connected through high speed transmission line played the leading part together with the station staffs provided with PC. The station systems took in safety function and other operations widely.

Autonomous Decentralized Transport Operation control and information System (ATOS) was introduced to phase in for the general management and control of high-density traffic railway transport in Tokyo metropolitan area [KiKa 99] [KeBe 00] [KeBe 03]. The experience of COMTRAC development considerably influenced ATOS.

ATOS has been continuously expanding since the first station started in operation in July 1993. The Chuo Line firstly began using this system in December 1996. The Yamanote Line and the Keihin Tohoku Line began using this system in July 1998, and the Sobu Line between Ochanomizu and Chiba began in May, 1999. The system installation is phasing in year by year, as shown in Fig.2.

Welfare and healthcare that had been out of the engineering became social big concerns. Dependability of medical treatment (radiation therapy, fiber scope, and so on) and advanced medical imaging apparatuses (X-CT, MRI, and USI) were closed up as life-critical systems [Ih 95].

Hospital Information Systems (HIS) and Radiology Information Systems (RIS) as business process, Picture Archiving and Communication System (PACS) as the medical image database became in real-time at sophisticated hospitals. Hitachi has delivered them based on ADC.

Intelligent Transport System (ITS) emerged as the means of environment conservation and safety. We proposed ADC applying to ad hoc grouping control of cars on the roads in the systems and experimental trial was done as a part of the national project [ShSa 03].

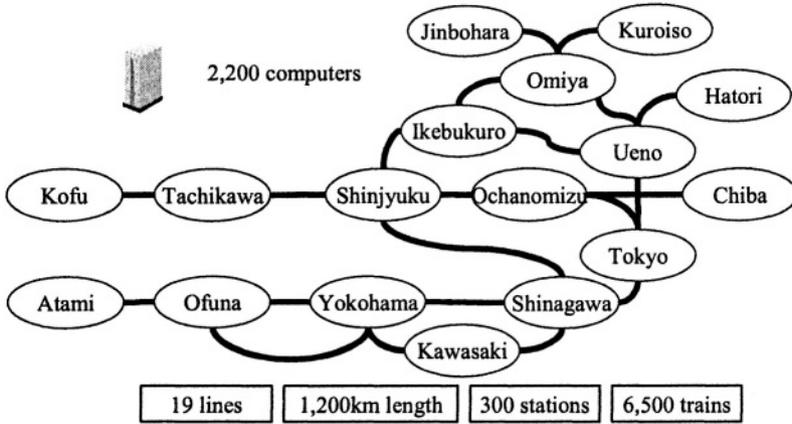


Figure 2. Managing area of ATOS

Progress of Artificial Intelligence in 1990s includes hybridization of fuzzy logics and neural networks as well as data mining technology for reducing cost for knowledge acquisition that was one of the most serious issues in developing intelligent systems. These knowledge acquisition methodologies significantly extended the application opportunity of intelligent systems such as public utility systems, financial dealing systems, CRM systems in telecommunication industries, and so on.

### 2.5 After 2000

After 2000, ADC has stridden to ubiquitous information environment [We 93] named Autonomous Super Distributed Objects or simply SDO (Super Distributed Objects) [FuKa 02]. The SDO are characterized two distinct features given by wireless technologies as well as micro-electronic device technologies such as Radio Frequency Identification (RFID), namely, (1) frequent and quick joining and leaving of the entities, (2) non-deterministic nature of services provided by relevant entities. These SDO features call for automatic services delivery according to the contexts of the system users with the constraints that come from the degree of trusts among the entities [KaSa 03][ToFu 03]. In contrast to ADC of which metaphor is biological cells constituting organs of life systems, the metaphor of SDO is placed on human society with multiple relationship that is characterized the following facts, (1) human belongs to multiple communities, (2) human recognizes identities of others by the degrees of belonging to the communities, (3) human interacts with others based on the recognition.

A model of SDO is developed that abstracts and wraps existing diversified application oriented standards such as UPnP, HAVi, ECHONET,

etc. The SDO model permits inter-operability among the existing technologies and is standardized by the OMG [SaKa 01] [KaSa 04].

The ADS product, NX series, is continuously spreading over social and industrial infrastructural systems. Among them, remarkable topic is that COMTRAC for Taiwan Shinkansen based on the NX series is scheduled in operation in 2005.

### 3. DEPENDABLE SYSTEMS APPROACH

We present two approaches to realize dependable computer systems in this chapter. One is systems architecture technology as system basis and the other is artificial intelligence technology as for information processing.

#### 3.1 Fault Tolerance and Assurance Architecture of Real time Systems

We classify the real systems to four domains from point of view of the system architecture in Fig.3. The vertical axis shows fault tolerance-fault avoidance and the horizontal centralization-distribution.

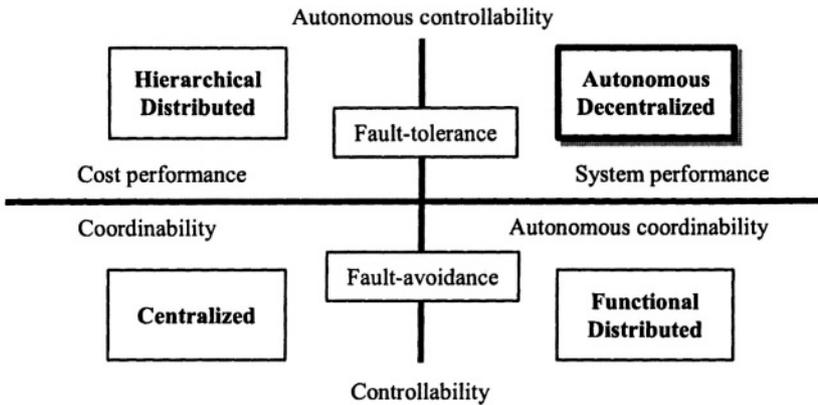


Figure 3. System architecture

As we describe in the previous chapter, it also shows the paradigm shift of system perception from centralization in 1970s to decentralization in 1990s with need shifts from the manufacture or technology to the user or demands. The centralized system structure, which was characterized by time-sharing process and fixed protocol for attaining best efficiency, had unacceptable weakness of flexibility and extensibility. The distributed

structure took over the centralized one to realize flexibility and extensibility. However this does not satisfy critical requirement from system dependability in on-line real-time operation. And high-assurance during system expansion became necessary since the total system is not defined in advance. The conventional centralized system is in the third quadrant and the distributed in the second and the fourth. All of subsystems forming Autonomous Decentralized System, placed in the first quadrant, upper right, have to satisfy autonomous controllability and autonomous coordinability as well as fault-tolerance and life cycle cost performance as the total system. Here we describe R&D and practices by Hitachi.

### 3.1.1 Centralized systems and distributed systems

The computer control and management systems for the railway are suitable to explain for fine examples.

The first COMTRAC for the Tokaido Line in 1972 had the dual system structure, which runs programs synchronously in two computers shown in Fig.4. Specially tailored hardware compares two computed results, which are route control information selected as critical for safety. When some disagree arises between them, the comparator, Dual System Controller (DSC), cuts off both outputs and activates check programs in two computers. Then the results of check programs are compared to some pattern pre-stored in the ROM of the comparator. The either computer that produces its result, which meets with the pattern, is recognized as healthy one and its output is transferred as the right route control command. Therefore, when one of the dual computers is failed, the survival can run as single. The single system is not dependable from safety as well as reliability points of view.

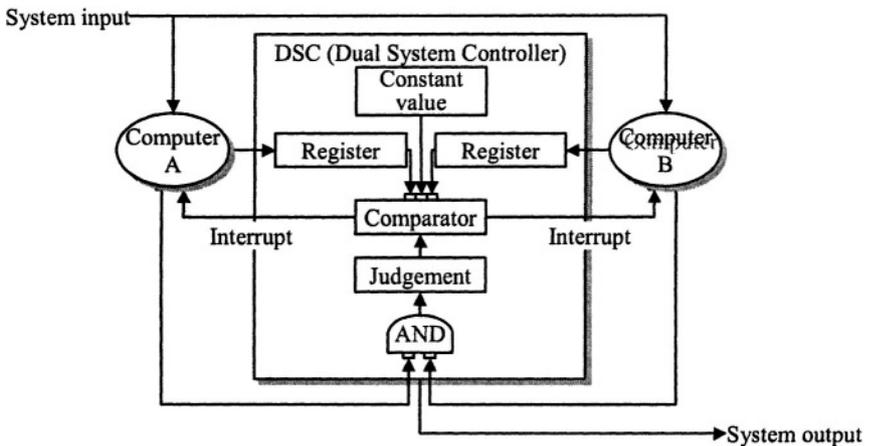


Figure 4. Dual system

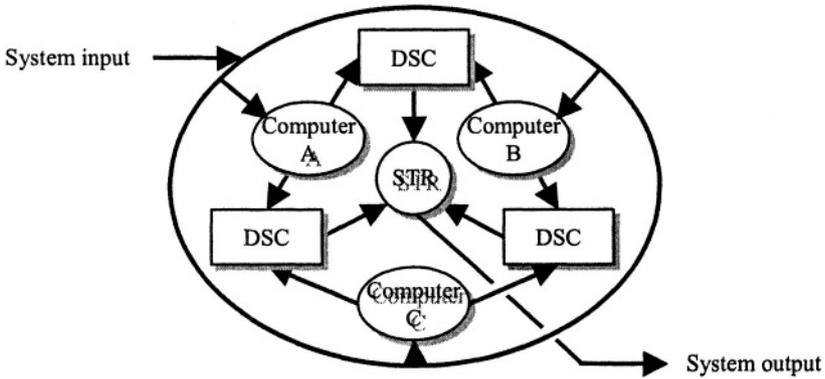


Figure 5. Dual-Duplex architecture

The next COMTRAC, shown in Fig.5 had a stand-by computer by the lessons learnt from the above [IhFu 78]. When the dual operation moves to single operation, the third one is connected to the surviving computer automatically. That is, the system is restructured again to dual operation. The command output is stopped about 2.5 seconds during reconfiguration to new dual system since the train control can permit time delay of three seconds. It may be called the system structure as symmetrical dual-duplex. It is reported more than 99.999% availability for 24-hour operation and no unsafe output in its operation life. This COMTRAC had the hierarchical system structure connected to another two main frame computers, so-called the duplex configuration, as the upper level system for planning schedule of train operation.

The same system structure was applied to the Tohouku and the Joetsu Lines later on. Of course, additional new functions were included according to economic-social demand for railway transportation.

The systems of the Tokaido and the Sanyo Lines were replaced by the up-dated hardware in 1988. The front-end computers for communicating with other computer systems, single, and message transfer computer for passenger, and duplex were separately added to the main processors of which structure was the same as its predecessor. Computers and 53 operator's consoles were connected with the main computers by double loop reliable network.

Although COMTRAC hardware systems structure has been changed according to social demand-pull and technology-push described in chapter 2.3, the system concept, that is, centralization, and fundamental program structure, that is, time sharing processing, were not changed. And the safety function for prevention of collision still remained in the traditional signal system assembled special components. This structure has great difficulties on reliability, construction, extension and replacement of the systems in

commercial operation from assurance viewpoint. We had to find out new paradigm for huge large real time computer control systems.

### 3.1.2 Autonomous Decentralized Systems [IhMo 84] [IhMo 87]

We grasped biological organism as a system with the following attributes,

- (a) It always includes inactive (temporary faulty, complementary or spare) parts
- (b) It always changes its conditions and states among operation, metabolism, generation and growth (plus or minus)
- (c) It always changes its objectives to the goal by alternatives selection, optimization and adaptation
- (d) It keeps accomplishing its objectives almost completely

This observation was opposite recognition against that system should be complete and stable.

Living thing is the integration of organs that are also integration of cells. We could consider that cell has uniformity of structure, equality of function and locality of information as cell level. Each cell has the same embedded programs as genes or DNA chain. Each cell has its autonomy, which means cell can live and function by itself following its environment. Organs or individuals are the integration of numerous cells, which have their autonomy with coordination around them. The system always survives despite it is not stable because of metabolism, growth, aging, and so on. Therefore, the system including faulty subsystem is quite normal. In other word, the system without any fault is only in one state of the system. This is our basic perception on systems.

Initially subsystems autonomously exist, and a system is defined as the result of the coalition assembly of subsystems when a certain objectives appear. Two attributes are applicable to not only the hardware system but also the software system with the following definition. [MoSa 81]

- (a) **Autonomous controllability:** if any subsystem fails, the other subsystems can manage themselves.
- (b) **Autonomous coordinability:** if any subsystem fails, the other subsystems can coordinate their individual objectives among themselves.

It is clear that these two properties assure not only the fault tolerance but also the expandability and the maintainability of the system. They suggest that every subsystem requires an intelligence to manage itself and to coordinate with other sub-systems. That is, the subsystem consists of not only the application modules but also its own management module.

Moreover, these properties do not permit a common resource in order to isolate from the connected subsystems. Hence, a common file in the systems

is not acceptable in the autonomous decentralized software structure. For communicating and coordinating among the software subsystems without using the common file, the virtual data field (VDF) concept is proposed in Fig.6. All data is sent out to all other subsystems as soon as the subsystems originate the data. The subsystem does not continue to store the data without broadcasting it. The subsystem can access the data in the VDF when it passes over the subsystems. The data is removed after it transmitted through the VDF. The identical function of VDF is autonomously distributed in the subsystems though they have different control missions. [MoMi 85]

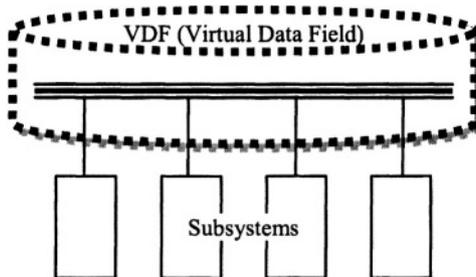


Figure 6. Virtual data field

Under this idea of VDF, the following conditions are required to satisfy the properties of the ADC.

- (a) **Uniformity:** Each subsystem must be self-contained and independent of others, so that it can continue to operate even if others fail. That is, every structure is uniform to join VDF.
- (b) **Equality:** Any subsystem could behave correctly because of the receiver initiative protocol. Therefore, it is not permitted for each subsystem to direct any others. That is, all subsystems are equal in VDF.
- (c) **Locality:** Failure of one subsystem could mean a breakdown in communication and a failure to collect global information from other subsystems. Hence, every subsystem must be able to control itself and to coordinate with the other subsystems on a local-information basis only. That is, subsystem can access the data in VDF.

In order to realize these properties technically, asynchronous daisy chain broadcast communication with Content Code (CC) and data field (DF) scheme are invented. The subsystem generates some information broadcasts it with assigned code as the tag that means content of the information to all other subsystems. Any of them takes the broadcasted information selected by referring CC if it needs. There is not any special communication control process between sender and receiver. The receiver takes initiative in communication. Each subsystem is connected each other by Autonomous Control Processor (ACP) in duplex architecture. Thus, the database, named

DF, consists virtually of distributed data in the subsystems. The subsystems can access immediately their data accumulated in their memory when other subsystems that generate original transactions are out of order. Therefore the transaction rate of the connection lines remains on average even if any subsystem fails. In other words, the bottleneck of the communication does not emerge in the system.

The uniformity and the equality conditions mean that every subsystem has the same ACP and there is no master-slave relationship among them. The locality condition indicates that the subsystems must communicate synchronously only with neighboring ACPs without using the destination addresses. In a conventional system, as the exchange protocol is decided by hand-shake procedure, the destination address is a must since the sender subsystem recognizes in advance how the total system is structured and which subsystem must receive the data. Hence, the destination address restricts the autonomy of the subsystems to judge whether or not to receive the data. In the autonomous decentralized system, the packet with the content code protocol is used shown in Fig.7.

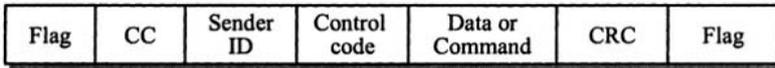


Figure 7. Content code

Every item of data attaching its corresponding CC is broadcasted into the DF and the ACP that selects only the necessary data by CC receives it. No data in the DF has priority. Each ACP judges autonomously how to process the data without being dependent on any priority and drives a corresponding software module when all the necessary input data, with which CC is previously specified within the ACP, is received from the DF and arranged by time stamp or serial number in the received data if necessary.

CC can include not only online information but also offline information such as test data and maintenance procedure. Thus, both online and offline functions are simultaneously carried on in the same hardware. It realizes easy modification, expansion and extension of the systems. This largely contributes to phase in implementation and operation [MoMi 84].

Systems Development Laboratory made up two types of the systems, loosely-coupled networks (Autonomous Decentralized Loop network: namely ADL) as the analogy of neural systems [MoIh 82] [IhMo 82] and tightly-coupled systems (Fault-tolerant Multi-micro Processor system by ADC, namely FMFA) as the analogy of the cerebellum [MiNo 83] to show by means of evidence. [MiMo 84] A NCP of ADL has two outlets and two inlets respectively to make the ladder loop DF shown in Fig. 8.

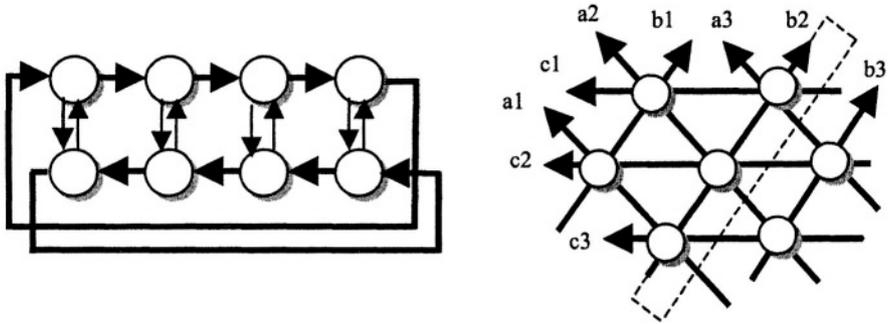


Figure 8. ADL and FMPA

Photo couplers isolated cells against electrical disturbance. FMPA consists of homogeneous cells. Each Cell has a microprocessor and three symmetrical pairs of bi-directional bus ports (FTB) and its controller, which make the hexagonal array structure shown in Fig. 8.

Each cell was connected with the adjacent six cells. FMPA has dynamic addressing and threshold voting for dependability. Two systems were connected and used to develop the control program for NCP in the laboratory. Pluralism easily form complex systems; dual, duplex and majority voting through DF. They showed high reliability by means of evidence and applied widely to the commercial systems by De-fact Standards [KaSa 99].

We have proved the property of high assurance of ADS by ATOS. ATOS is a large-scale system involving 19 train lines, 300 stations, and a total track length of 1,200 kilometers. It has been developed with the goal of improving safe and stable transport, passenger service, and management efficiency. First installation of ATOS was in 1993. Since then, the expansion of the service area and the functions of ATOS have been expanded by phase in installation procedure month by month. ATOS shows high assurance through the whole installation.[KeBe 00][KeBe 01][KeBe 03]

The station systems take charge of route control, passenger guidance, etc.; the central system notes the running status of trains and takes charge of general operation control work. A trunk network (100 Mbps) connects both systems. Ethernet as a branch LAN connects the railway-side equipments. For the hardware, general-purpose computers (WSs, PCs, PLCs, controllers, and real-time control servers namely RCSs) are used as open system architecture. The main sections of the station systems and central system are duplicated to secure reliability. Nowadays ATOS consists of more than 2,200 computers.

### 3.2 Artificial Intelligence

It is generally accepted that the computer program is following human problem solving procedure, that is, algorithm. We have to make computer programs by concept originally given by Von Neumann after deciding rigorously their boundary conditions.

Here we tried to clarify the step from human thinking way to the stored programming to justify our R&D target in artificial intelligent system for dependability of decision-making.

Fig.9 shows our understanding of intelligent activity [IhMo 87]. We choose two axes, the vertical one is structuring degree of problem and the other horizontal one is the extent of knowledge domain. We can assign the present programming technology in the first quadrant, and actual human thinking ability is assigned in the third quadrant. Human being has broad and emergent ways of subjective consideration and common sense. The problem in the second is universal and well structured. Knowledge is written deterministically and objectively in numerical expression and the structure of the problems is combinations of numerical formula. Non-linear problem and combinatorial problem are examples. The second area includes mathematics, result search, gaming, FFT, and so on. They have been investigated from ancient times as well as general problem solver (GPS) in early AI research. Fuzzy set and neural network are also placed in the second. The fourth has deductive reasoning in specific knowledge domains. Knowledge is written in the character trains. Its structure is non-deterministic. It is rather difficult to examine reasoning results that greatly depend on accumulated volume of the knowledge and on vagueness of necessary knowledge. The written knowledge itself is power.

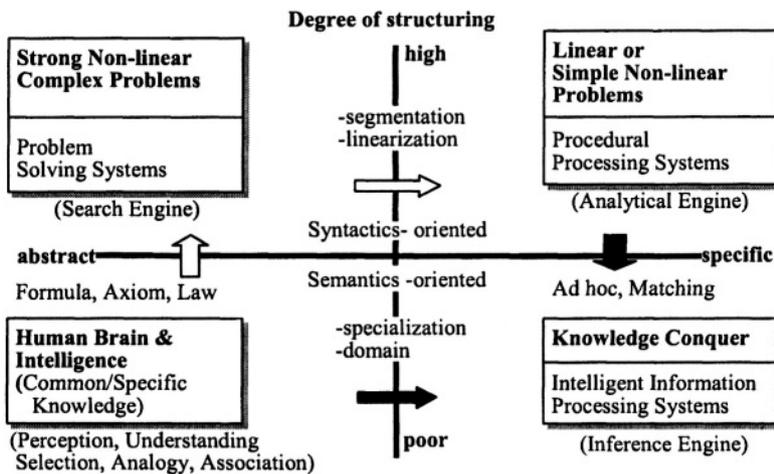


Figure 9. AI domains

Some of intellectual activity of human beings is transferred the third quadrant to the second by being distilled to axioms, natural laws and abstraction. Linearization and segmentation transform the problems in the second to the first. On the other hand, the forth cuts the expert knowledge of a selected domain out of the third. Knowledge Engineering begun by the Stanford Heuristic Programming Project (HPP). Knowledge Engineering aims to deduct the problem that is characterized by undefined structure and less-arithmetic knowledge. The combination of knowledge is dynamically varies by the contents of accumulated knowledge. In order to extract expertise, knowledge engineers and support environments were proposed. Stored program system in the first is made use with interpreting instead of compiling. We, having a lot of experiences in the first and the second went in the forth as the remained area in 1980.

Our first attempts to utilize linguistic knowledge targeted computerization of shallow expertise that consisted of pairs of situations and actions to be taken for higher level of goal attainments. The attempts covered diagnosis of nuclear power plants, quality control of semiconductor manufacturing [KuKo 89], project management for construction of fossil power plants [NiSa 84], and so on. It was found that applications in physical process should incorporate with existing mathematical models in addition to experts' linguistic knowledge. In parallel with these trials, a couple of AI development systems were developed including EUREKA that combined production and object oriented programming, LONLI that was based on Prolog, and KRIT for design and diagnosis of dynamic systems.

After number of trials, we formulated knowledge engineering approach for practical situations, but we needed powerful inference engines for the practices in real-time environment. We developed a fast pattern matching algorithm that permitted operation guidance by mini-computers in real-time situations. One of the most successful applications was a guidance system for blast furnace operation. The inference engine named EUREKA II was so powerful that enjoyed diversified applications including operation guidance of public utility plants, generation assistance of train diagrams, construction planning and project management support, financial investment assessment, and so on [TaMa 87][TsMa 88][TsKa 92][TsEg 96][TsEg 99].

Rich experiences of knowledge engineering provided further advancement of knowledge utilization. The guiding framework was predictive control where predictive simulation based on the knowledge of the controlled objects was conducted so as to select the best control alternative. Train operation control in the Sendai subway succeeded first commercial use of fuzzy logics in the world where fuzzy logics were used in evaluating control alternatives even though train dynamics were described by a set of traditional differential equations [YaMi 83]. The application

based on the train control idea was extended to controlling cranes that were definitely difficult for the traditional control theory. Much more thorough use of linguistic knowledge was realized in tunnel ventilation control where pollutant, air, and traffic dynamics in the tunnel were described by linguistic fuzzy knowledge that were used for prediction of the air quality in the tunnel with state filtering mechanism [FuAo 91].

Fuzzy logics were very effective in computerizing expertise knowledge, because the grade of fuzziness can be used for tuning outcome of the fuzzy logic inferences in contrast to the crisp logic inferences that require acquiring complete rigorous knowledge. However the adjustment of the grade values was not easy task for the system developers. In order to automate the adjustment of the grade values, the back propagation algorithm, namely chained algorithm for calculating partial derivatives, in neural network technologies was successfully introduced to the fuzzy logic network that was a network representation of fuzzy logics [MaSo 91][IcMa 93][Malc 93]. The development significantly reduced the burden in developing intelligent systems that resulted in expanding application domains that covered control systems for public utility plants as well as financial trading systems [FuMa 95].

In the late of 1990s, IT was becoming pervasive for human more and more so that capturing data that representing human behavior in IT could be simply realized. However it was not easy for systems analysts to utilize large amount of data in obtaining models of user behavior that were the determinants for delivering cost-effective superior services to the users. In order to cope with this problem, data mining technology was developed that permitted extracting knowledge on user behavior from large amount of data. Traditional statistical modeling technology was based on liner analysis, but extraction of knowledge from data mostly was assumed requiring nonlinear analysis. In order to deal with nonlinearity, categorizations for the continuous values were adopted and a powerful algorithm conscious to frequency in reading data from storage was developed that finds dependency among the data items. The technology was packaged as a software product named DATAFRONT and applied to customer relationship management systems in telecommunication operators and quality control systems in LSI manufacturers [MaMa 98][MoSa 98][AsMo 99][MaMa 99].

#### **4. PERSPECTIVE AND THE OFFERING FOR NEW HORIZONS**

In this chapter, we will try to write down several perspectives of dependable systems from our experiences and Japanese market demands.

### 4.1 Extended Autonomous Decentralization Concept [Ih 98] [Ih 99]

We have been doing research and development activities of ADC since 1980 as one of long term program at Hitachi and academia. Meanwhile, as described in previous chapters, the evolution of technology is greatly changing and renewing our society with information and visa versa. Rapid transition of the society requires the information systems must correspond to it.

Although neural system of living creatures looks as if the cerebral activity controls other organs, the normal cerebral functions are not generally ignited spontaneously without any stimuli from other organs or sensors. The organs are isolated from each other and permitted only to communicate with the brain neural cells in principle. Body system can consequently attain its survival objectives by the stimuli from sense organs. In addition, sensory subsystems change themselves according to their environmental conditions. Here we can recognize the hierarchical relation between the brain system and sensory subsystems.

Autonomous Decentralization Concept has put assumption that cell belongs to one layer with uniformity, equality and locality as subsystem level. An assembly of the cells that have a same object makes an organ. An assembly of the organs makes a human. A human makes family and so on. Each layer generally fulfills uniformity, equality and locality. And we can observe the hierarchy from the cell to the United Nations shown in Fig. 10.

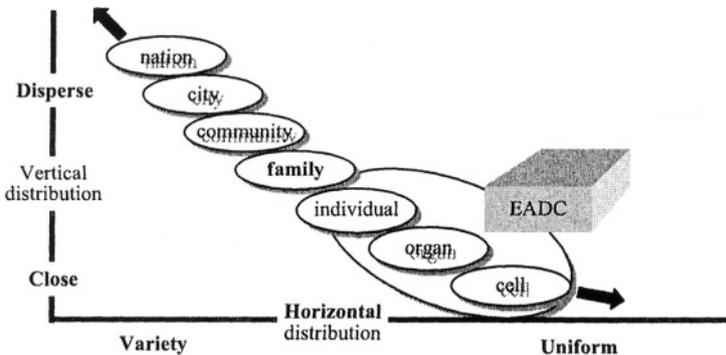


Figure 10. Hierarchical layer of EADC

Then we have to introduce the hierarchical layers in ADC. This hierarchy is adequate enough to attain the specific object of each layer. However, we have to recognize that each layer is not subjected to upper layer level although each layer consists of the lower level components, which are subsystems, from structure point of view. The subsystems of each layer

coordinate each other within their layers to achieve their objectives as ADC described in Chapter 3.1.

When we recognize the hierarchy of subsystem level, the upper level members are mostly influenced by the lower level members, their belonging level ones and their own conditions. We must notice that there are not certain master-slave relations between the layers.

After above observations, we proposed the extension of ADC (EADC) adding Autonomous Observability as the third attribute shown in Fig. 11.

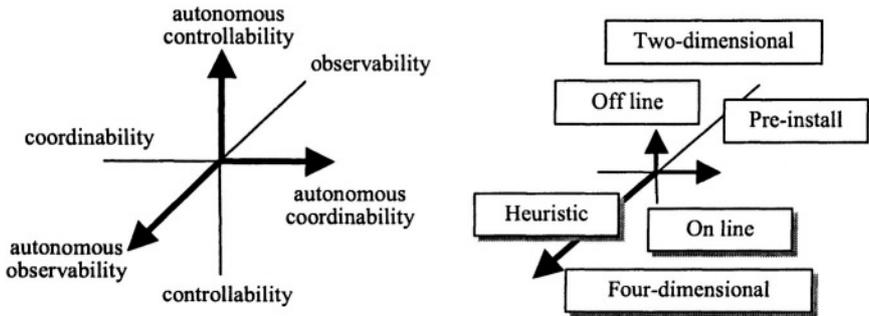


Figure 11. EADC

**Autonomous Observability:** if any change occurs in the surrounding of the subsystem, it can be observed immediately.

It aims at more dependable systems structure enhanced by our experiences until now and rapid advent of recent technologies. Though it is still infant level, the concept is now under R&D. One of the leading examples is the ad hoc grouping in ITS [ShSa 03] and entertainment robotics.

The definition of Autonomous Observability means that the system can autonomously observe its surrounding, recognize four dimensional changes (space and time), and report to the upper level of VDF.

It also includes a kind of the firewall against harmful interaction and malicious attack from the outside just like immunity. As for Autonomous Controllability previously defined, the subsystems have additionally the ability to change their control regions. And Autonomous Coordinability is added the ability of changing of information exchange for coordination. EADC implies autonomous adaptation of the structure and functions of the system to its environment fluctuation. In other words, the systems based on EADC will be built up by the demand sides oriented from quality of services instead of the supply sides oriented from quality of product. The boundaries and the conditions of large and complicated systems are getting hard for the supplier side to strictly decide prior to tailoring the systems. Proto-typing method or user engineering method will be introduced.

Autonomous observability can be assumedly intelligent sensor subsystems. We have already developed various smart sensor systems such as vision, hearing, touch and etc. for signal detection and processing. Radio wave watches, global positioning systems (GPS), gyrocompass, and RFID ( **$\mu$ -chips**) will be used to identify the accurate location and time of information source. Technologies of artificial intelligence and robotics make the observation intelligently autonomous.

The system structure with EADC dynamically expects the systems more flexible, dependable and secure than ever against environment changes.

## **4.2 Open Development and Operation Infrastructure**

Recently fatal failures and disasters often occurred in our daily life, for example transportation, public utility, production, medicine, administration and etc. The faults are often human-made by ignorance, careless mistakes and unskillful ness. Furthermore, it will be more difficult to call experienced experts and well-qualified project leaders together and to detain them for certain term in a project team. Lessons learnt from previous failures have not been reflected to the successive programs. Participating organizations or persons cannot be responsible for the system totally and have organized their peer group against criticisms outside. Project is still based on old-fashioned project management, not recursive one introducing rapid advent of information technologies [Ih 04].

Engineering and designing by IT seems getting important because the systems consist of tremendous number of the components, either hardware or software, is becoming difficult to develop and operate. The activities need deep specific knowledge and experience as well as ability of analysis and synthesis. Further more, know-how form human experience should be reserved in documentation or multimedia before the experts retire or remembrance diminishes.

In making information systems dependable for these 40 years, we have considerable experiences that the authors presented in the previous chapters.

From above observations, Open Development and Operation Infrastructure (ODOI) will be required for design and operation of sophisticated systems that furnish high dependability [TsUe 04]. The environment is opened to people who are especially interested in the applicable systems. They, including volunteers, can offer their knowledge, opinion, advice, verification, validation, criticism and work to the project through Internet. ODOI could serve:

- (a) The project members means to retrieve, accumulate, suggest, simulate, validate and synthesize comprehensive information and knowledge using suitable intelligent models for the phases of plan, design, validation and operation.

- (b) The outer project members and the experts outside opportunities to communicate, exchange, discuss, and interact with recursively among them.
- (c) The people interested in including citizens opportunities to watch recursively the on-going process and propose new idea or suggestion.
- (d) The target systems necessary dependability which is supported throughout their life cycles by sophisticated engineering systems.

The ODOI that the project teams maintain accumulation of necessary information in its database and knowledge base in addition to running project management, necessary analysis and synthesis with system technologies [KaTs 04]. The systems will gather the chronological multi-media record of the whole activity through the life cycle of development and operation, by which investigation of failures will make the target systems more dependable as defined Autonomous Observability in EADC.

The Emergent Synthetic Environment (ESE) for space development has been proposed since 2000, with the support of the National Space Development Agency, to acquire by advanced IT all information on development, design, verification, validation and operation of spacecrafts, in order to avoid human-made faults. It is a trial proposal of ODOI for the future [TsUe 04].

## **4.3 The Yaoyorozu Project: Marriage of Social Science and Information Technology**

### **4.3.1 Emerging technology affecting our social life**

M. Weiser advocated ubiquitous computing in 1988 where a lot of computers surrounding human help him/her to work without noticing their existences. Experiment of the ubiquitous computing by M. Weiser were conducted with 3 layered computer systems, namely the tab of Intel 8051 with 128KB, the pad of Motorola 68302 with 4MB, and the board of Sun workstation [We 93]. Nowadays our electronic surroundings are far different from those in the beginning of 1990s. For example, we might be surrounded by embedded controllers in many home appliances, mobile phones with a camera as well as a positioning sensor, and so on. In future, we will be surrounded a lot of intelligent devices based on nano-technology that function for monitoring and controlling our physical conditions, augmenting our sensing and memorizing capabilities, and information control systems interacting our environment where everything is equipped with smart tags.

This future picture suggests that we will be surrounded heterogeneous powerful intelligent devices based on the advanced super distributed objects

(SDO) technology which deals with dynamically forming trusted parties and their chains. In classical Japanese, the word “Yaoyorozu” (literally “eight million”) was used to refer to something that was countless in number, particularly in the phrase “Yaoyorozu no Kami-gami,” or “eight million gods.” The belief was that gods lived not only in the many old temples throughout Japan, but in the trees and the stones, in the sky and the water, constantly surrounding and protecting us. For this reason, we use “Yaoyorozu” for our coming heterogeneous information society in stead of ubiquitous.

The emerging technology as described in the above has great potential for changing our life style as well as our way of thinking about others and society. We have to be prepared for the new technological paradigm so that in 2002 an interdisciplinary research activity named Yaoyorozu Project formed under the sponsorship by Ministry of Education, Culture, Sports, Science and Technology, Japan.

#### **4.3.2 Yaoyorozu approach based on the trans-disciplinary science and technology**

Collaboration of humanity, sociological, and engineering knowledge is strongly required for attaining the Project goal, however this type of multi-disciplinary collaboration has been seldom realized so far. Some sophisticated mechanism shall be introduced for the collaboration. M. Gibbons et al. [GiLi 94] advocated so called Mode 2 knowledge production model that best utilizes IT for collaborations among multi-disciplines. For the efficient collaboration, Mode 2 claims (1) restoring interests in individual ordered knowledge, (2) emphasizing on design knowledge, and (3) modeling based on computer technology. H. Shimizu who conducts the social technology is proposing co-creation through sharing future vision among participants.

In the Yaoyorozu Project, the trans-disciplinary science and technology is the guiding principle for integrating knowledge diversified over multiple disciplines. Even though definition of the trans-disciplinary science and technology is not commonly agreed so far, but we assume that systems technology provides basic functions for integrating knowledge as well as producing new ideas. Many traditional disciplines have mainly been focusing on cognition of the internal and external entities but the trans-disciplinary science and technology calls for design of science in addition to traditional science and technology. This is the distinctive requirement to the trans-disciplinary science and technology. The systems technology including scenario generation, collective discussion over diversified disciplines, multi-agent simulation, and so on are expected respond to this synthetic requirement of our future [FuHo 03].

## **5. SUMMARY**

We, Hitachi group, have been chasing dependability as the company highest motto since it established in 1910. The motto seems considerably similar to the concept of “Dependability” though we did not recognize the word till IFIP WG 10.4 published firstly the book edited by J-C Laprie in 1991 including Japanese version by Y. Koga [La 92]. In the book, he did not describe Japanese word for dependability. Hence we have used phonetic spelling by Japanese Kana characters “dhi-pe-n-da-bi-ri-thi.”

Hitachi group has a wide product lines across from the semiconductor to the nuclear plant. Information technology is one of its important business area including dependable computer control systems. Here we described the activities concerning computer system dependability focusing on Systems Research Laboratory and its outskirts. In addition, we proposed new R&D horizon for dependable information systems.

We, researchers of Hitachi groups, have been greatly influenced the phrase “Although our lifetime may not span a hundred years, we have concerns of a thousand years (from an ancient Chinese poem).” Dependability of information systems is based on the phrase.

## **6. ACKNOWLEDGEMENTS**

We express our sincere appreciation to Algirdas Avižienis for his valuable support for 25 years. Special thanks go to Prof. Alain Costes, Prof. Yoshihiro Tohma and the colleagues of IFIP WG 10.4 for their friendship.

The original suggestion on analogy of biology for ADC came from Prof. Takemochi Ishii during his talks and discussions. Prof. Atsunobu Ichikawa organized, and the late Prof. Masami Itoh promoted, the ADC research group in Japanese academia. Prof. Edward Feigenbaum gave us valuable suggestions and encouragements on AI research, Prof. Lotfi Zadeh on fuzzy control, Prof. Takeo Kanade on image understanding and Mr. Daniel Goldin on R/D management. We would like to put the late Mr. Arne O. Berglund of Sweden and the late Prof. Tohru Tanabe on record for their comprehension on ADC. Dr. Takeo Miura supported us with great patience from the beginning through the R&D mentioned here. We thank Dr. Michitaka Kosaka, General Manager of Systems Development Laboratory for his assistance too.

It is too hard for us to line up all of numerous excellent researchers, engineers and managers in and out of Hitachi, to whom we want to give our best regards and gratitude.

## REFERENCES

We referred mostly to the papers that were made only in English by the acquaintances of authors.

- [AsMo 99]Ashida,H. Morita,T. 1999: Architecture of Data Mining Server: DATAFRONT/Server, Proc. IEEE SMC'99, pp.882-887.
- [FuAo 91]Funabashi,M. Aoki,I. Yahiro,M. Inoue,H. 1991: A Fuzzy Model Based Control Scheme and its Application to a Road Tunnel Ventilation System, Proc. IECON '91, pp.1596-1601.
- [FuHo 03]Funabashi,M. Homma,K. Sasaki,T. 2003: Introduction to the Yaoyorozu Project, Proc. SICE 2003, pp.1332-1335.
- [FuKa 02]Funabashi,M. Kawano,K. Sameshima,S. Kato,H. 2002: Middleware technology for ubiquitous computing: AYA(context-Aware and Yet Another service) that permits autonomous collaboration on super distributed objects, Proc. 2002 IEEE SMC, pp.623-628.
- [FuMa 95]Funabashi,M. Maeda,M. Morooka,Y. Mori,K. 1995: Fuzzy and Neural Hybrid Expert Systems-Synergetic AI-, IEEE Expert, Vol.10, No.4, pp.32-40.
- [FuTo 01]Funabashi,M. Toyouchi,J. Kanai,A. Uchihashi,T. Kobayashi,T. Hakomori,S. Yoshida,E. Strick,L. Born,M. 2001: Development of Open Service Collaborative Platform for Coming ECs by International Joint Efforts, in Chin,W. Patricelli,F. Milutinovic,V. (eds.), "Electronic Business and Education: Recent Advances in the Internet Infrastructure," Kluwer Academic Publishers, pp.284-299.
- [GiLi 94]Gibbons,M. Limoges,C. Nowotny,H. Schwartzman,S. Scott,P. Trow,M. 1994: The New Production of Knowledge, Sage Publications of London.
- [IcMa 93]Ichimori,T. Maeda,A. Funabashi,M. Ohmori,K. Kometani,N. 1993: Improving Fuzzy Knowledge Using a Network Representation of Fuzzy Logic Inference Procedures, Proc. IEEE International Workshop on Neuro-Fuzzy Control, pp. 102-106.
- [Ih 04]Ihara,H. 2004: What do Fatal Failures in Japanese Space Programs Suggest Us?, Proc. PRDC-10, pp.343-352.
- [Ih 79]Ihara,H. 1979: Fault-Tolerance Through Autonomous Decentralized Systems, IFIP Working Conference on "Reliable Computing and Fault-Tolerance in 1980s", London.
- [Ih 95]Ihara,H. 1995: Dependable medical systems, Proc. FTCS-25 Silver Jubilee, pp.76-84.
- [Ih 98]Ihara,H. 1998: More Dependable Systems in Computing –Autonomous Decentralized Systems, Proc. IFIP-DCIA 98, pp.127-135.
- [Ih 99]Ihara,H. 1999: Proposal of Extended Autonomous Decentralization Concept and Dependable Medical Home Care System, Technical Report (R999-9) of IEICE, pp. 1-6.
- [IhFu 78]Ihara,H. Fukuoka,K. Kubo,Y. Yokota,S. 1978: Fault-tolerant Computer System with Three Symmetric Computers, Proc. IEEE, Vol.66, No. 10, pp.1160-1177.
- [IhMo 82]Ihara,H. Mori,K. 1982: Highly Reliable Loop Computer Network System on Fault-tolerant Decentralization Concept, Proc. FTCS-12, pp. 187-194 .
- [IhMo 84]Ihara,H. Mori,K. 1984: Autonomous decentralized computer control systems, IEEE Computer, Vol.17, No.8, pp. 57-66.
- [IhMo 87]Ihara,H. Mori,K. Miyamoto,S. 1987 : Evolution of Reliable Computing in Hitachi and Autonomous Decentralized System, in Avizienis,A. Kopetz,H. Laprie,J.(eds.) "Dependable Computing and Fault-Tolerant Systems Vol.1," The Evolution of Fault-Tolerant Computing, pp.77-99, Springer-Verlag Wien NewYork .

- [IhMo 87]Ihara,H. Mohri,S. Kawai,S. 1987: On Dependable Intelligent Systems in Space, Space Technol., Vol., No.3, pp.233-240, Maxell Pergamon Macmillan (38<sup>th</sup> IAF-82-32,1987).
- [Ka 78]Kamiuchi,T. et al 1978: A Load Sharing, N:1 Backup Multisystem, COMPCON Spring '78, pp.261-264.
- [KaMa 89]Kanekawa,N, Maejima,H. Kato,H. Ihara,H. 1989: Dependable Onboard computer Systems with a New Method: Stepwise Negotiating Voting, Proc. FTCS-19, pp. 13-19.
- [KaSa 03]Kato,H. Sato,Y. Fukumoto,T. Homma,K. Sasaki,T. Funabashi,M: Trust Network-Based Filtering to Retrieve Trustworthy Word-of-Mouth Information, The 2nd Workshop on Security in Ubiquitous Computing, UbiComp 2003.
- [KaSa 04]Kawakami,K. Sameshima,S. Kawano,K. Suzuki,J. Suda,T. Steglich,S. Shin,S. 2004: SDO model and its standardization activity, Proc. SAINT 2004 Workshops, pp.315-320.
- [KaSa 99]Kawano,K. Sameshima,S. Adachi,Y. 1999: An autonomous decentralized manufacturing system architecture and its trends for de-facto standards, Proc. IEEE SMC'99, pp422 - 427.
- [KaTs 04]Kanuf,R. Tsuruta,S. Gonzalez,A Kurbad,T Ihara,H. 2004: Improving AI Systems' Dependability by Utilizing Historical Knowledge, Proc.PRDC-10 pp.343-352.
- [KeBe 00]Kera,K Bekki,K Fujiwara,K. Kitahara,F Kamijyo,K. 2000: Assurance System Technologies based on Autonomous Decentralized System for Large Scale Transport Operation Control System,IEICE Trans.COMMUN.Vol.E83-B NO5.
- [KeBe 01] Kera,K. Bekki,K Fujiwara,K. Kitahara,F. Kamijyo,K. 2001: Assurance Technology for Growing System and Its Application to Tokyo Metropolitan Railway Network, IEICE Trans. Vol. E-84-D No.10.
- [KeBe 03]Kera,K. Bekki,K. Mori,K. 2003: High Assurance Autonomous step-by-step System Construction Technique based on Assurance Evaluation IFICE Trans. VOL.E86-D NO.10.
- [KiKa 99] Kitahara,F. Kamljou,K. Kakurai,Y. Bekki,K. Kera,K. Kawano,K. 1999: Phased-in construction method of ATOS, Proc. Integration of Heterogeneous Systems, pp.415- 424.
- [KoAg 99]Korner,A. Aghoutane,N. Ercan,G. Strick,R. Wunderlich,W. Funabashi,M. Kawano,S. Toyouchi,J. 1999: Efficient Brokerage in a CORBA-based Service Mediation Platform, Proc. ISADS 99, pp. 284-291.
- [Ku 97]Kuwahara,H. 1997: Experiences teach us the future of autonomous decentralized systems, Proc. ISADS 97, pp. 169-175.
- [KuKo 89]Kurihara,K. Kobayashi,T. Akashi,K. 1989: Knowledge Based Process Diagnosis Expert System for LSI Manufacturing, IPSJ Journal, Vol.30, No.7, pp.918-921.
- [La 85] Laprie, Jean-Claude 1985: Dependable Computing and Fault Tolerance: Concept and Terminology, Proc.FTCS-15, pp.2-11.
- [La 92]Laprie,J-C. 1992: Dependable Computing and Fault -Tolerant Systems, in Laprie,J-C. (eds.) "Dependability: Basic Concepts and Terminology," Vol.5, Springer-Verlag, Wien NewYork.
- [MaIc 93]MaedaA. Ichimori,T. Funabashi,M. 1993: FLIP-net: A Network Representation of Fuzzy Inference Procedure and its Application to Fuzzy Rule Structure Analysis, Proc. of 2nd IEEE International Conference on Fuzzy Systems, pp.391-395.
- [MaMa 98]Maeda,A. Maki,H. Akimori,H. 1998: Characteristic Rule Induction Algorithm for Data Mining, Proc. PAKDD-98.
- [MaMa 99]Maki,H. Maeda,A. Morita,T. Akimori,H. 1999: Applying data mining to data analysis in manufacturing, APMS'99, pp.324-331.

- [MaSo 91]Maeda,A. Someya,R. Yasunobu,S. Funabashi,M 1991: A Fuzzy-based Expert System Building Tool with Self-tuning Capability for Membership Functions, Proc. the World Congress on Expert Systems, pp.639-647.
- [MiMo 84]Miyamoto,S. Mori,K Ihara,H. Matsumaru,H. 1984: Autonomous Decentralized Control and its Application to the Rapid Transit System, Computer in Industry, vol.5 no.2 pp.115-124, Elsevier Science Publishers BV.
- [MiNo 83]Miyamoto,S. Nohmi,M. Mori,K. Ihara,H. 1983: FMPA: Fault Tolerant Multi-Microcomputer System based on Autonomous Decentralization Concept, Proc. FTCS-13, pp.4-9.
- [Molh 82]Mori,K. Ihara,H. 1982: Autonomous Decentralized Network, Proc.COMPCON SPRING 1982, pp. 192-195.
- [MoMi 84]Mori,K. Miyamoto,S. Matsumaru,H. Ohshima,H. Hattori,A. 1984: Autonomous Decentralized Loop Network and its Application to Train Traffic Control System, Proc. IECON'84, pp.663-666.
- [MoMi 84]Mori,K. Miyamoto,S. Nohmi,M. Ihara,H. 1984: Online Maintenance in Autonomous Decentralized Loop Network ADL, Proc. COMPCON FALL'84, pp.323-332.
- [MoMi 85]Mori,K. Miyamoto,S. Ihara,H 1985: Autonomous Decentralized computer system and software structure, Computer Systems Science and Engineering, Vol.1, No.1, pp. 17-22, Butterworths.
- [MoSa 81]Mori,K. Sano,K. Ihara,H. 1981: Autonomous Controllability of Decentralized System Aiming a at Fault-Tolerance, 8<sup>th</sup> IFAC-World Congress, pp.183-189.
- [MoSa 98]Morita,T. Sato,Y. Maeda,A. Yamanaka,Y. Miller,R.M. Takeda,H. 1998: Techniques for Time-Series Data Processing in Plant Data Warehouse, PADD98, pp.177-182.
- [NiSa 84]Niwa,K. Sasaki,K. Ihara,H 1984: An Experimental Comparison of Knowledge Representation Schemes, The AI Magazine, Vol.5, No.2, pp.29-36.
- [OrKa 84]Orimo,M. Kawano,K. Mori,K 1984: Functional Reliability and its Utilizaion for Evaluating the Decomposition of System, Proc. IECON, pp.1197-1202.
- [OrMo 92]Orimo M. Mori,K. Ihara,H, 1992: Evaluation of System Division Based on Functional Reliability, Proc. the Automatic Control Conference, Vol. 28, No. 2,.
- [SaKa 01]Sameshima,S. Kawano,K. Funabashi,M. 2001: A trend to super distributed system and standardization activity in OMG, Proc. SAINT 2001 Workshops, pp. 5-8 .
- [SaKa 97]Sameshima,S. Kawano,K. Kumayama,J. Ito,T. Inoue,K. Fujishiro,S. 1997: An autonomous decentralized system architecture and techniques for on-line development and maintenance, Proc. ISADS 97, pp.121-128 .
- [ShSa 03]Shimura,A. Sakaibara,T. Hiraiwa,M. Aizono,T. 2003: Proposal of an autonomous group-management model and its application to intelligent transport system, Proc. ISADS 2003, pp.71-79.
- [TaMa 87]Tano,S. Masui,S. Sakaguchi,S. Funabashi,M. 1987: A Fast Pattern Match Algorithm for a Knowledge Based Systems Building Tool-EUREKA, IPSJ Journal, Vol.28, No. 12, pp. 1255-1268.
- [ToFu 01]Toyouchi,J. Funabashi,M. Kanai,A. Uchihashi,T. Hakomori,S. Yoshida,E. Strick,L. Born,M. 2001, Development of Service Integration Platform for One-stop Service Applications, Proc. WECWIS 2001, pp.123-125.
- [ToFu 03]Toyouchi,J. Funabashi,M. Honjo,S. Komoda,N. 2003: Web Service Integration Platform with Privacy Preferences for One-stop Application, ICWS'03.

- [ToSh 99] Tomita,N. Shibao,S. Omura,M. Oku,M. 1999 : Flow oriented approach for human-centered agile manufacturing systems, Proc. Integration of Heterogeneous Systems, pp.98-106.
- [TsEg 96]Tsuruta,S Eguchi,T Yanai, S. Ooshima, T. 1996:DGCAI: Dynamic Goal centered Coordination AI technology –For flexible computation solving real-time large-scale complex problems, Proc. AAAI-96, pp.490-495.
- [TsEg 99]Tsuruta, S. Eguchi,T. Yanai,S Ooshima,T 1999:A Coordination Technique in A Highly Automated Train Rescheduling System, Proc. IEEE SMC'99, pp.601-606.
- [TsKa 92]Tsuruta,S. Kakumoto, Y. 1992: Development of a Highly-Automated Train-Scheduling System using an Advanced AI Technology for Creating Very-High-Speed, Very-High-density Train-Schedules, Proc. OMPRAIL92, pp. 155-166.
- [TsMa 88]Tsuruta, S. Matsumoto, K. 1988: A Knowledge-based Interactive Train Scheduling System-- Aiming at Large-Scale Complex Planning Expert Systems, Proc. of IEEE Workshop on AIA 88, pp.490-495.
- [TsUe 04]Tsuruta,S. Uehara,K. Ihara,H. Tohma,Y Ooba,H. Oota,Y 2004: Design and Application of Emergent Synthesis Environment for Developing Dependable System, Proc. CITSTA 2004 (to appear in July, 2004).
- [We 93]Weiser,M. 1993: Some Computer Science Issues in Ubiquitous Computing, Communications of the ACM, Vol.36, Vol.7, pp.75-84.
- [YaMi 83]Yasunobu,S. Miyamoto,S. Ihara,H. 1983: Fuzzy Control for Automatic Train Operation System, Proc. 4<sup>th</sup> IFAC/IFIP/IFORS Conference, pp.33-39.
- [YaMi 85]Yasunobu,S. Miyamoto,S. 1985: Automatic Train Operation by Predictive Fuzzy Control," in M. Sugeno (ed.): "Industrial Applications of Fuzzy Control," Elsevier Science Publisher.