ISE-TR-89-78

EXPERIMENTAL STUDY ON A COMPUTER INTERCONNECTION ARCHITECTURE WITH INTER-APPLICATION COMMUNICATION FACILITY

by

Yoshihiko Ebihara

Tomoo Nakamura

and

Akira Sakaguchi

January 25, 1989

INSTITUTE
OF
INFORMATION SCIENCES AND ELECTRONICS

UNIVERSITY OF TSUKUBA

EXPERIMENTAL STUDY ON A COMPUTER INTERCONNECTION ARCHITECTURE WITH INTER-APPLICATION COMMUNICATION FACILITY

Yoshihiko Ebihara, Tomoo Nakamura and Akira Sakaguchi

Institute of Information Sciences and Electronics
University of Tsukuba,Ibaraki 305, Japan

Abstract

This paper describes a scheme to connect full scale computers of different architecture to a wideband local area network via a gateway process instead of a front end processor, and outlines the results of its application. This scheme has advantages that no reforming of the native operating system is necessary and that the system development is easy, fast and economical, though the system suffers slight performance reduction compared to direct network support. For a full scale computer which can run different operating systems concurrently by simulating virtual machine systems, this gateway process scheme offers an efficient way to connect computer systems of different network architecture without badly degrading the man-machine interface response or the performance.

Keywords: Local area network, Gateway process, Interprocess communication

(Presented at the International Computer Symposium 1988, Taipei, Taiwan, ROC, December 15-17, 1988)

1. Introduction

We discuss a scheme to connect a full scale computer of a different network architecture to a wideband local area network [1]-[3] via a gateway process, and the results of implementation of this scheme. An important feature of the gateway process is that most of the interconnection properties are primarily artifacts of host software in a general-purpose time-sharing system. The general-purpose time-sharing system provides reasonable response to any user program and many service facilities/utilities for implementing a gateway process. While Rochester's intelligent gateway and others [3]-[6] have usually been implemented on a smaller, less powerful front end processor, front end processors sometimes become bottlenecks for communication throughput for bulk data transmission. The proposed gateway scheme has advantages that the gateway process can accommodate a number of existing terminals attached to the time-sharing system with slight modification of the native operating system and can provide system designers with powerful debugging tools and the rich utilities of the existing system. This scheme is easy, fast and economical for gateway construction, compared to any other gateway technique, for support of virtual terminal services.

2. GAMMA-NET

GAMMA-NET is a local area computer network that connects several full and middle scale computer systems through a wideband LAN. Each computer system is specially tuned to specific types of processing such as conversational and batch processing. The LAN consists of a dual 32 MBPS optical fiber ring bus that forms a bidirectional line and specially designed transmission control processors - called ring bus processors (RBP's) - that control data link by hardware and firmware and are directly connected to the channel interface of host computers [1]. In GAMMA-NET, a set of communication protocols, whose structure is layered after the reference model of OSI, is specified.

3. Interface to connect a computer system to LAN

Each computer system from a different manufacturer has its own network protocol architecture. Access to a remote terminal follows an access method which is generally called a virtual terminal access method (VTAM). But each VTAM is different and has its own peculiarities which cause difficulties in the constituting of a heterogeneous network. In order to resolve such differences, the following schemes are adoptable:

(1) To unify these protocols and revise or newly develop a module that can accept the unified protocol.
(2) To translate and convert between the standard protocol and each system's own protocol.

The first scheme must be costly. Instead, we can find several interfaces to insert a module that performs the needed protocol conversion.

3. 1 Implementation issues

There are several implementation issues that must be taken into account when we add new features to existing systems. Important parameters to evaluate such implementation include:

(1) needed time, manpower and cost

(2) ease of system upgrading and revision (maintainability)

(3) execution efficiency, throughput and response time

3. 2 Selection of interface

Our problem is to offer and link a TSS server function to GAMMA-NET. This section discusses implementation issues of connecting the FACOM M-200 FNA (Fujitsu network architecture) to GAMMA-NET. Normally, TSS data flows in the sequence of TSS-VTAM-NCP-TERMINAL in FNA. If we don't want to modify an application module such as TSS, it is necessary to adhere to the protocol of the VT access method. We can find at least four levels of interfaces to link with GAMMA-NET: VT access method, VTAM internal interface, VTAM i/o, and physical i/o. The implications in using these four interfaces are as follows:

1) VT access method

1. 1) Direct VTAM support

In order to utilize the VT access method directly, it is necessary to incorparate the ring bus access method (RBAM) of GAMMA-NET into VTAM. In this scheme VTAM controls an RBP and GAMMA-NET as if these were VTAM's local devices. This scheme seems best from the viewpoints of efficiency, structural clarity and initial implementation. It will, however, cause troubles in the supporting and maintaining of non-standard devices in a VTAM which is a standard system product of a manufacturer.

1. 2) VTAM inter-application communication interface

An application process that uses VTAM can communicate with other application processes through the VTAM inter-application process communication facility. The other application process also uses VTAM macroinstructions as TSS. The gateway process can be executed in the normal user mode, thus there are no problems that might degrade system reliability.

2) VTAM i/o

In order to utilize the VTAM i/o side interface, it is necessary to intercept VTAM i/o requests. In this scheme the i/o requests that are to be executed by VTAM for terminal control are intercepted and executed by an interface module for an RBP. I/o commands and their control sequences are the basic interface for the terminal and are sometimes mixed up with the control scheme of VTAM. In addition, they are usually executed in a privileged processor mode; thus, the environment for execution and testing is rather complicated and must be prepared specially and carefully, and there is some danger of destroying the important area dedicated to system administration by bugs or failures.

3) Physical i/o

In GAMMA-NET the specially designed RBP is employed to control the data link, and we don't want to use a network control processor or a similar processor which will degrade throughput substantially; physical i/o intervention has no practical importance.

The best interface of the four is the inter-application process communication facility of VTAM.

4. Protocol conversion issues and requirements

The following functions are needed when the VTAM inter-application process communication is employed:

1) Compltete simulation of ordinary TSS processing executed between TSS processes and terminals beyond GAMMA-NET,

2) Conversion of the data stream of GAMMA-NET virtual terminals into FNA's terminal data stream and vice versa,

3) Acceptance of requests for interactive session from RBAM,
and

4) Control of links between TSS processes and terminals beyond GAMMA-NET.

5. System structure

5. 1 TSSP

The gateway process is located between VTAM and RBAM, which have almost the same VT access functions; it converts VTAM protocol to RBAM protocol and vice versa. As the gateway process is prepared to support TSS, it is called a TSS path-through process (TSSP). The conceptual structure of TSSP is shown in Fig. 1. Roughly speaking, TSSP is an application program which accesses (virtual) terminals of GAMMA-NET through RBAM file access protocol (FAP) macroinstructions. On the other hand, TSSP is also an application process that communicates with a VTAM application process such as the TSS process through the VTAM inter-application process communication facility.

1) TSSP as an RBAM application process

TSSP controls and accesses (virtual) terminals of GAMMA-NET through FAP macroinstructions of RBAM. In this respect, TSSP works as an ordinary user process to RBAM. In order to accept a session establishment request from a remote clients, a TSSP manager process is prepared. The TSSP manager process works as follows:

(1) The request for session establishment which is passed from RBAM to TSSP is directed to the TSSP manager process.

(2) The TSSP manager process initiates a TSSP user process for the requesting user.

(3) The TSSP manager process passes the logical link that was originally connected to the TSSP manager process to this TSSP user process.

Then, this TSSP user process can control the interaction with a terminal of GAMMA-NET. When a TSSP user process is notified of session termination by TSS, it releases the logical link of its process and closes the session. The structure has the following advantages:

(a) Both TSSP and user processes are kept independent of and consistent with each other.

(b) The structure of the program is simplified and the development of the system is easier.

The TSSP processes are placed and executed in a single address space. This scheme is used because:

(1) Memory space and space management overhead are smaller than in a multiple space scheme.

(2) The TSSP procedure is transparent to users, and no user program runs in the TSSP space, thus there is no danger of sharing a common space after initial bugs are removed.

(3) About 250 users can be supported even in a single space scheme, and this is enough for a single host to serve.
(4) There is no significant difference in the development cost between single space and multiple space schemes.

2) TSSP as a VTAM application process

Each TSSP user process establishes a logical link between TSS, and controls the interactive session with TSS through VTAM macroinstructions.

5. 2 Structure of TSSP

TSSP is built in the host computer. TSSP consists of a TSSP manager process and a number of TSSP user processes. The TSSP manager process acts as a session controller against RBAM, while each TSSP user process is responsible for each user's session control.

5. 3 Function of TSSP

The functions that TSSP is responsible for are as follows:
1) Control of sessions and logical links as discussed in Section 6.
2) Conversion of terminal control protocol

TSSP recognizes the terminal type by the protocol, and connects a link to a VTAM terminal node. It also converts the code set and the protocol for controlling the terminal. It does not, however, handle actual data, but supports a transparent data path between RBAM and VTAM.

3) Data flow control

Basically, data transmission is performed in half duplex mode. RBAM supports 2048 logical links of multiplexed mode by using two physical data links. Each computer system is responsible for flow control. The basic flow control function is implemented in the logical link layer of the RBAM.

6. Protocol conversion issues

6. 1 GAMMA-NET protocol layer

The protocol layer of GAMMA-NET consists of four layers. Data link control layer controls the physical data links. Network control layer(NCP) controls the logical links. In Function control layer two types of protocol are specified: FAP and inter-process communication protocol (ICP). The FAP standardizes the access method to all resources, including even processes, regarding them as files. The ICP offers efficient inter-process communication facility for limited applications. Finally, several application protocols are specified in Application layer. Table 1 shows the usage of NCP/FAP commands by the virtual terminal protocol. For more details the reader should refer to [1].

6. 2 FNA protocol

In the TSS/VTAM protocol of FNA the following is specified:
(1) Half-duplex transfer mode is used.
(2) Maximum data size is 256 bytes, which may contain chained information to link longer messages which span more than two data messages or shorter messages which are accommodated in a data message.

6. 3 Protocol conversion

6. 3. 1 Basic policy of protocol conversion

The basic policies for protocol conversion are as follows:

(1) Protocol conversion is transparent to users. Thus, the data that are included in a message are not processed.

(2) Throughput must not be degraded badly.

(3) Each link is handled independently so that security and ease of error recovery are maintained.

6. 3. 2 Control sequences

Fig. 2 shows protocol correspondence between the GAMMA-NET virtual terminal protocol and the TSS/VTAM protocol of FNA, according to our protocol conversion policies.

6. 3. 3 Implementation issues

Several problems arise from the actual implementation. These are mainly related to timing or delay, i. e. interrupt handling, abnormal session termination and echo back.

1) Interrupt handling

Interrupts can be classified into two types; those caused in input mode and in output mode. Most interrupts are caused in output mode, like a BREAK to cause an immediate pause in a listing operation. In our system, response to such an interruption is not good. TSS/VTAM of FNA sends chained elements in a 256 byte data message in which several lines are accommodated, and there is no straightforward method to know how many lines are to be sent. In addition, there is some delay because of the BREAK request passing. When the protocol converter receives an interrupt it processes the interrupt as one element, but it can not discriminate the proper breaking point. Therefore the user feels that the processing of BREAK is not effective or not appropriate. To avoid such ambiguity, TSSP must de-block messages from TSS/VTAM of FNA and send them to a terminal one by one, but this will degrade throughput badly.

2) Abnormal session termination

Whenever TSSP detects hardware errors by the terminal, the TSSP user process is terminated. TSS, however, does not close the session immediately after the hardware errors are detected, but watches status for a given period of time. When a TSSP user process is initiated, an identifier is assigned to it and notified to TSS. If a TSSP user process is abnomally terminated, the assigned identifier is freed by TSSP, but TSS still holds this identifier for a while. If TSSP receives a new request before TSS frees the identifier and if TSSP generates the same identifier, a double-assignment condition is caused. This condition is now avoided by reusing a freed identifier carefully according to the time elapsed since the identifier was freed.

3) Echo back

Current virtual terminal protocol does not have control over echo back, so that echo suppression for passwords is not available. This problem is left for future work.

7. Discussion of the gateway method and conclusion

Often a minicomputer is employed as a front end processor to connect a host computer to a network that has different network architecture.

The rationale for this scheme is as follows:

(1) To alleviate the load of the host computer for controlling data links.

(2) To separate control logics between different network architectures, so as to make connection easier.

However this scheme has several drawbacks:

(1) A separate minicomputer system is required.

(2) Productivity of software on a minicomputer is lower than that of a general-purpose full scale computer.

In GAMMA-NET, physical layers are supported by hardware and firmware of dedicated communication processors (RBP's). Then, it is preferable to support the rest of the control layers directly by software in the host computer. This scheme has the following merits:

(1) The existing native operating system can be used efficiently to support the gateway to convert protocol.

(2) The full set of advanced utilities for software development is available, thus productivity of software is excellent.

We discussed the way to connect a computer to a LAN. The system structure and operating conditions of the gateway scheme are discussed in detail, including selection of interface, system structure, issues of protocol conversion, function of protocol converter, correspondence and sequence of commands, etc.

Acknowledgements

The authors would like to acknowledge the contributions of Prof. Kazuhiko Nakayama of the University of Tsukuba and Prof. Katsuo Ikeda of Kyoto University in encouragement to carry out this research project.

References

[1] Y. Ebihara, K. Ikeda, T. Nakamura, M. Ishizaka, M. Shinzawa and K. Nakayama, GAMMA-NET: A Local Computer Network Coupled by High-Speed Optical Fiber Ring Bus --- System Concept and Structure -, *Computer Networks*, **7**, (1983), 375-388.

[2] Y. Ebihara, K. Ikeda, T. Nakamura, S. Nakatsuka and M. Ishizaka, Fault Diagnosis and Automatic Reconfiguration for a Ring Subsystem, *Computer Networks and ISDN Systems*, **10**, (1985), 97-109.

[3] Y. Ebihara, K. Ikeda, T.Nakamura, M. Ogawa and T. Takiguchi, Connection of Computers of Different Architecture to a LAN via a Gateway Process, *JIP*, **10**, (1987), 66-73.

[4] D. B. Boggs, J. F. Shoch, E. A. Taft and R. M. Metacalfe, Pup: An Internetwork Architecture, *IEEE Trans. on Comm.* , **COM-24**, (1980), 612-624.

[5] Y. Ebihara and M. Wilson, MENEHUNE DARPANET Driver, *ALOHA System Internal Document*, CCG/G-56, (1974).

[6] J. E. Ball, J. Feldman, J. R. Low, R. Rashid and P. Rovner, RIG, Rochester's Intelligent Gateway: System Overview, *IEEE Trans. on Soft. Eng.* , **SE-2**, (1976), 321-328.

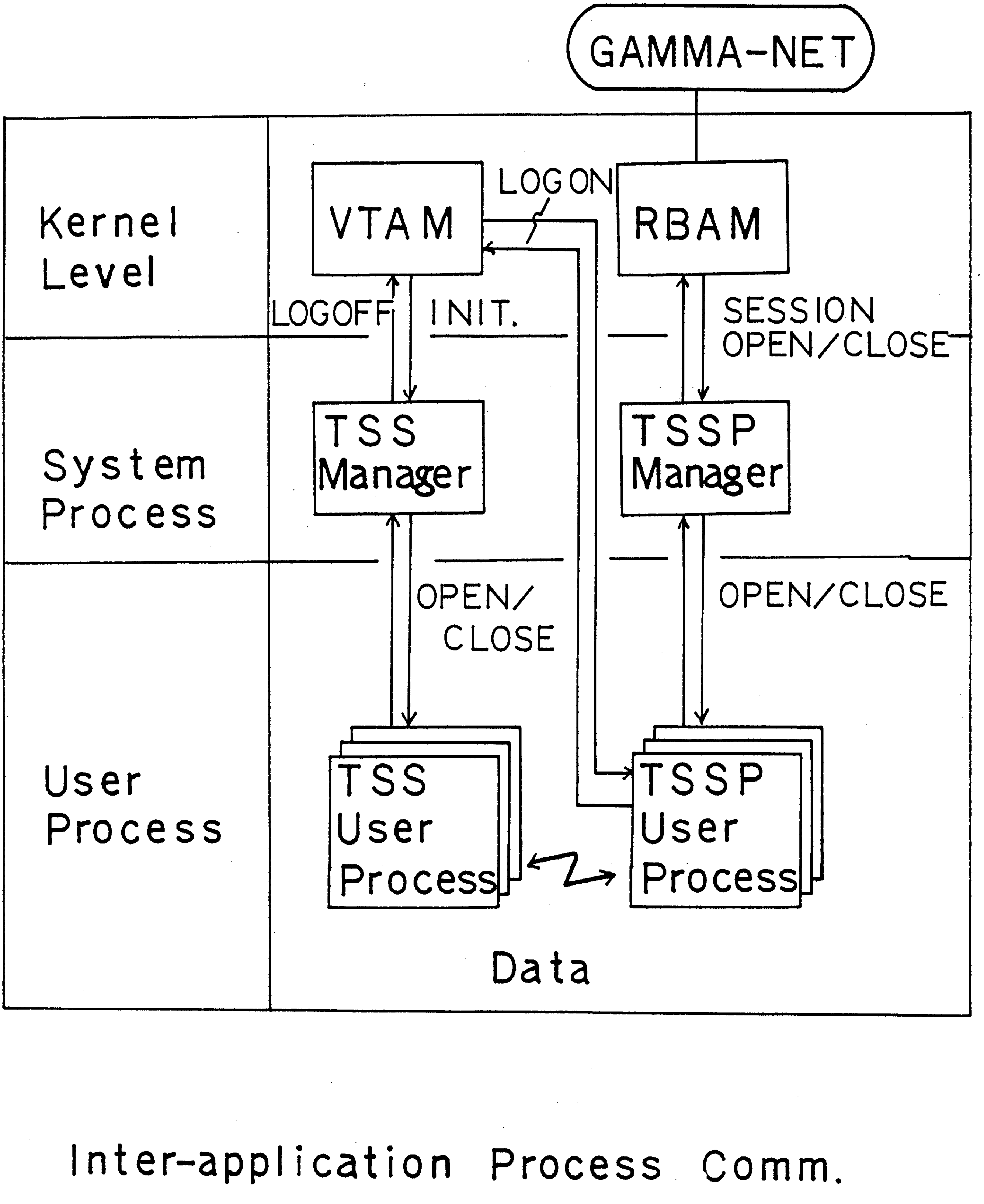
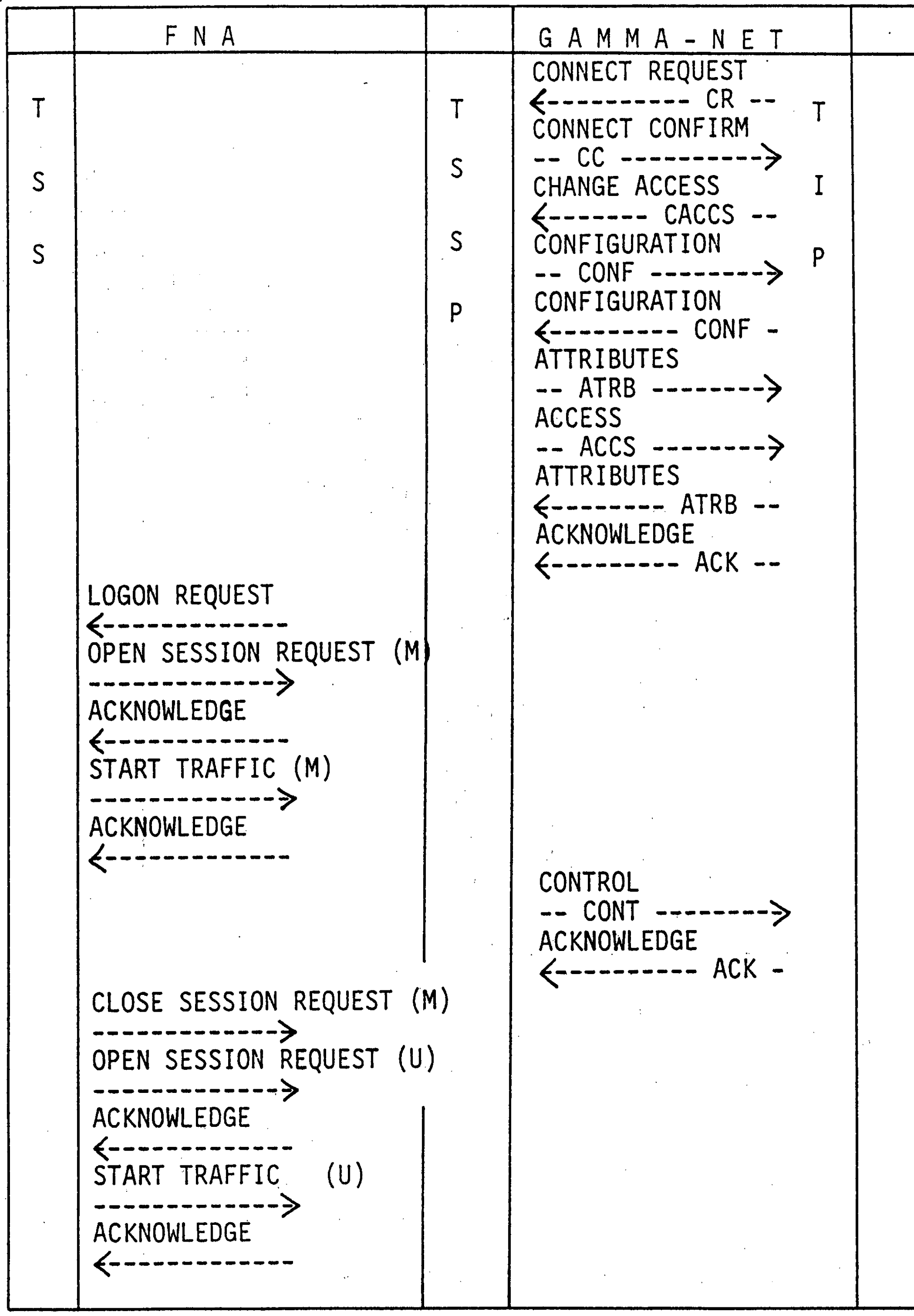


Fig. 1 Structure of Inter-application Communication

TSP		FAP COMMANDS		NCP COMMANDS	
PHASE		NAME	FUNCTIONS	NAME	FUNCTIONS
1	CREATE LINK			CONNECT REQUEST (CR)	ESTABLISH LOGICAL LINK
				CONNECT CONFIRM (CC)	LOGICAL LINK ESTABLISHED
2	CONNECT	CONFIGURATION (C O N F)	FACILITIES OF SYSTEM	DATA MESSAGE	DATA MESSAGE ON NORMAL FLOW
		ATTRIBUTE (A T T R)	FILE ATTRIBUTE INFORMATION		
		ACCESS (A C C S)	COMMAND OF ACCESS		
		CONTROL (C O N T)	COMMAND OF CONNECTION		
		CHANGE ACCESS (CACCS)	CHANGE DIRECTION		
		ACKNOWLEDGE (A C K)	RESPONSE OF ACCS/ CONT		
3	READY	CONTROL (C O N T)	COMMAND OF TRANSFER		
		DATA (D A T A)	FILE DATA		
		STATUS (S T A T)	RESPONSE		
		EXPEDITE (E X P)	ATTENTION	INTERRUPT MESSAGE	DATA MESSAGE ON EXPEDITE FLOW
4	DISCONNECT	ACCESS COMPLETE (C O M P)	END OF TRANSFER		
5	RELEASE			DISCONNECT REQUEST (D R)	RELEASE LOGICAL LINK
				DISCONNECT CONFIRM (D C)	LOGICAL LINK RELEASED

Table 1 FNA and NCP Commands

(1) CREATE LINK AND CONNECT



(M) : TSS MANAGER (U) : TSS USER

(continue)

Fig. 2 Controle Sequences

(2) CLOSE AND DISCONNECT

	F N A		G A M M A-N E T	
T S S	CLOSE SESSION REQUEST ----------------> ACKNOWLEDGE (M) <----------------	T S S P	COMPLETE -- COMP ---------> COMPLETE <-------- COMP -- DISCONNECT REQUEST -- DR ----------> DISCONNECT CONFIRM <----------- DC -	T I P

(continue)

Fig. 2

(3) DATA TRANSFER

	F N A		G A M M A - N E T	
T	CHANGE DIRECTION ----------------->	T		T
S		S	CONTROL -- CONT ------->	I
S	ACKNOWLEDGE <----------------	S	INPUT DATA (GET) <-------- DATA -	P
	INPUT DATA <----------------	P		
	OUTPUT DATA ONLY ----------------->		OUTPUT DATA (PUT) CONTROL -- CONT -------> OUTPUT DATA -- DATA -------> (EOD) ACKNOWLEDGE (OK) <------- STAT --	
	ACKNOWLEDGE <----------------			
	OUTPUT DATA (FIRST) ----------------->		OUTPUT DATA (PUT) CONTROL -- CONT -------> OUTPUT DATA -- DATA -------> (EOD) ACKNOWLEDGE (OK) <------ STAT ---	
	• • •		• • •	
	OUTPUT DATA (LAST) ----------------->		OUTPUT DATA (PUT) CONTROL -- CONT -------> OUTPUT DATA -- DATA -------> (EOD) ACKNOWLEDGE (OK) <------ STAT ---	
	ACKNOWLEDGE <----------------			

EOD : END OF DATA

Fig. 2

INSTITUTE OF INFORMATION SCIENCES AND ELECTRONICS
UNIVERSITY OF TSUKUBA
TSUKUBA-SHI, IBARAKI 305 JAPAN

REPORT DOCUMENTATION PAGE	REPORT NUMBER ISE-TR-89-78

TITLE

Experimental study on a computer interconnection architecture with inter-application communication facility

AUTHOR(S)

Yoshihiko Ebihara
Tomoo Nakamura
Akira Sakaguchi

REPORT DATE	NUMBER OF PAGES
January 25, 1989	12
MAIN CATEGORY	CR CATEGORIES
Computer Network	

KEY WORDS

Local area network, gateway process, interprocess communication

ABSTRACT

This paper describes a scheme to connect full sacle co-puters of different architecture to a wideband local area network via a gateway process instead of a front end processor, and outlines the results of its application. This scheme has advantages that no reforming of the native operating system is necessary and that the system development is easy, fast and economical, though the system suffers slight performance reductioncompared to direct network support. For a full scale computer which can run different operating systems concurrently by simulating virtual machine systems, this gateway process scheme offers an efficient way to connect computer systems of different network architecture without badly degrading the man-machine interface response or the performance.

SUPPLEMENTARY NOTES

Presented at the International Computer Symposium 1988, Taipei, Taiwan,ROC, December 15-17, 1988