Overview of the INI Compound Local Area Network

---Bridge Protocol for Inter-Subnetwork Broadcasting---

Akihito Taguchi

Institute for New Generation Computer Technology
(ICOT)

Mita-Kokusai Bldg. 21F
4-28 Mita 1-Chome Minato-ku Tokyo 108 Japan
Phone:03-456-3192   Telex:ICOT J32964

Abstract

This paper is concerned with the bridge protocol of the INI (Internal Network in the ICOT Programming Office) local area network, currently under development by ICOT.

INI is a compound network consisting of several ETHERNETs (*1) and bridges. A bridge connects a pair of ETHERNETs. A simple and unique bridge protocol has been developed for interconnecting ETHERNETs. This protocol inherits the broadcast facility of ETHERNET protocol and realizes broadcasting the whole of INI.

*1: ETHERNET is a trademark of the Xerox Corporation .

1. Introduction

INI (Internal Network in the ICOT Programming Office) is a local area network which is being developed by ICOT. INI consists of ETHER-cables (called subnetworks), LIAs (LAN Interface Adaptors) and bridges. A bridge connects a pair of ETHER-cables.

INI is three-layered: physical, logical and application. INI has a broadcasting logical-layer protocol (called group communication), corresponding to that of layers 4 and 5 in the OSI reference model proposed by ISO. Scherr says [Sche1], "The ability for a satellite to broadcast simultaneously to a large number of stations has not been completely incorporated into the range of possibilities." Group communication incorporates the physical-layer broadcasting facility so as to reduce the volume of transmission; conventional session-oriented communication does not take advantage of this powerful facility. In addition, this is more useful for various applications, including network controls, than conventional session-oriented one.

Before long, various broadcasting physical communication media, which will support group-communication logical layer protocol, will be available; e.g. satellite communication and wireless communication [Gfe1]. As one of them, a simple and unique bridge protocol of the INI physical layer has also been developed for inter-subnetwork broadcasting and routing.

This paper primarily describes this bridge protocol. Refer to [Tag1] for the other aspects of INI, especially for the group communication facility.

2. Overview of INI

2.1 Objectives

INI is now under development primarily as a data-bus for PSI (Personal Inference Machine), the DELTA relational database machine [Yok1] and other intelligent terminals, and also as the basis of a programming support environment. INI is also intended for use as an experimental network in the following areas:

- Developing various network technologies, such as network management protocols, integrated network operating systems and protocol translation
- Developing advanced office automation and intelligent programming support environment
- Application of the Japan's FGCS software technology to network technology, e.g. an knowledge-based expert system for advanced network management and a knowledge-based consultation system for configuring a network system

2.2 Network Configuration and Architecture

(1) Physical Configuration

As shown in Fig.2.2-1, INI is a compound network for the following purposes:

- To localize experimental high-traffic communications within a subnetwork in order not to disturb other ordinary communications.
- To enhance the flexibility of coaxial cable for installation, thus economizing on cable requirements.

INI consists of ETHER-cables (called subnetworks), LIAs and bridges. A bridge connects a pair of ETHER-cables. An LIA interfaces locally with several attached intelligent terminals, and executes communication protocols as their front-end communication processer. One or more LIAs perform some aspects of network management, such as down-loading onto LIAs, and are therefore called NCLIAs (Network-Control LIAs). Gateway processers with protocol-translation capabilities will be required for interconnection with a global packet-switching network.

(2) Layers and Protocols

As shown in Fig.2.2-2, in contrast to the OSI reference model, INI has basically three layers: physical, logical and application. The application layer, resident in terminals, may be divided into sublayers according to the OSI. The protocols of the physical and logical layers are executed in LIAs, which also execute certain application-layer protocols, such as network managements.

- Physical-layer Protocol
  The physical layer corresponds to OSI layers 1,2 and 3. Its protocol is based on ETHERNET [DIX1] and is enhanced for inter-subnetwork routing and broadcasting, as described in Section 2.3. The inter-subnetwork broadcast facility, evolved from ETHERNET broadcast facility, is intended to be used for realizing group communications over the whole of INI.

- Logical-layer Protocol
  The logical layer corresponds to OSI layers 4 and 5. For this layer, INI has a unique new "group-communication" broadcasting

protocol, which is used in place of conventional session-oriented protocols. A logical communication medium, called a "communication group", is formed among several processes, precluding the necessity of establishing a session between each pair of processes, as required in session-oriented communications. A message sent by any process is broadcast to all member processes of the communication group. In this paper, the term "session" is used to a communication group consisting of only two processes. Group communication makes good use of the physical-layer broadcast facility, including the inter-subnetwork broadcast facility, so as to cut down on the volume of transmission data; session-oriented communication does not take advantage of this powerful facility. Even for interconnecting with global packet-switching networks, their broadcasting facility will also be utilized. Before long, various physical communication media, with the ability of broadcasting, will be available; e.g. satellite communication and wireless communication [Gfe1]. Scherr says [Sche1], "The ability for a satellite to broadcast simultaneously to a large number of stations has not been completely incorporated into the range of possibilities."

Programming with a communication group is powerful for applications, such as those listed below, because a communication group is an abstract data type representing an object consisting of several sessions.

- .Distributed database control [Yam1]
- .Distribution of electronic mail and online documentation
- .Distributed processing (Distributed algorithm) [Was1]
- .Knowledge gathering for future programming support environments, such as knowledge-programming [Tag1], visual-programming [Mac1] and so on

Especially for gathering and propagation of network management information, group communication based on broadcasting is more appropriate than conventional session-oriented communications which are primarily designed for one-to-one transmissions. For example, databases distributed within networks, containing network management information such as network configurations and program modules to be remotely loaded, can be simultaneously updated or queried by using a communication group. Moreover, group communication is useful for implementing basic integrated network operating system functions, such as resource-locking and process-synchronization.

Refer to [Tag1] for further details on the logical layer protocol.

(3) Addresses

Fig.2.2-3 represents the LIA, terminal and process addresses in INI. The LIA address consists of a subnetwork address and a node-address-intra-subnetwork. Table-2.3-1 shows LIA-addressing for intra- and inter-subnetwork communications.

(4) Logical Structure and Interface of LIA

Fig.2.2-2 represents the logical structure of LIA. The LIA logically contains dummy processes corresponding to application processes in local terminals; these execute logical- and physical-layer protocols. There is also an application process that manages the network. These processes communicate with each other through a pre-defined communication group for network managements.

There are physical- and logical-level interfaces between an LIA and its terminal. IEEE488 is available for the physical-level interface. The logical-level interface is conceptually located between an application process and its dummy process. It is analogous to the logical-layer protocol, but much simpler so as to reduce the burden on application processes. Dummy processes provide their application processes with reliable, transparent communication media like virtual circuits, so that application processes basically need not be concerned with data transmission control. In the logical-level interface, application processes are able to designate other application processes using symbolic names instead of process addresses. Name-resolution (translation from symbolic process names to process addresses) is one of network managements [Tag1].

(5) Reasons for Introducing LIAs

- The LIA executes logical-layer protocol so as to reduce the burden on its terminals. Even if this protocol must be modified, the terminals need not be changed. For example, the data-transmission command may be modified in order to superpose information required for network control. Such protocol modifications are occasionally necessary in experimental networks, such as INI.

- The LIA serializes simultaneous data-transmission requests from its terminals so as to reduce the possibility of collisions on ETHER-cable and the resultant reduction of INI efficiency.

- LIAs, where network-management processes are placed, will perform the processing required for intelligent networks and even for integrated network operating systems.

- The LIA may serve as an intelligent terminal controller for protocol translation and media conversion.

2.3 Bridge Protocols in Physical Layer

INI is a tree-structured compound network consisting of several subnetworks (ETHER-cables), as shown in Fig.2.3-1. Bridges are required to interconnect the subnetworks. A bridge must be as small a device as possible and not reduce ETHER-cable transmission capacity. So, bridge protocol needs to be as simple as possible. Conversely, a bridge must be sophisticated enough to detect ineffective packets passing through it, which otherwise tend to cause inefficiency when the broadcast facility is used for group communications in the logical layer, as mentioned above.

(1) Bridge Protocol and Bridge Processing

Table-2.3-1 shows the LIA addressing for intra- and inter-subnetwork communications. It should be noted that this addressing is consistent with that of ETHERNET. Table.2.3-2 shows the correspondence between the addressing schemes of INI and ETHERNET. The facility corresponding to ETHERNET's multicast group is not supported over the entire INI; this can be realized using the group communication facility of the logical layer.

A bridge need not know the INI topology. For inter-subnetwork broadcasting and routing, it has only to know one given range of subnetwork addresses and the subnetwork addresses of two subnetworks connected to itself. However, this means constraining subnetwork addressing in such a way that all the subnetwork addresses of subnetworks beneath a bridge must be in the assigned range, as exemplified in Fig.2.3-1. LIAs and application processes need not be conscious of INI topology, either. In other words, both inter- and intra-subnetwork communications appear the same to LIAs and application processes. The LIA only needs to know the LIA addresses and the subnetwork address of its own subnetwork. It uses an individual-LIA-address-inter-subnetwork as the destination when sending a packet to another LIA in a subnetwork other than its own. An individual-LIA-address-intra-subnetwork is used when a packet is sent to another LIA in the same subnetwork. In all other cases, the broadcast-LIA-address is used.

The above addressing scheme minimizes the inter-subnetwork routing overhead, because it does not require bridges to distinguish between communication groups or to perform address conversion. Table 2.3-3 describes the bridge processing of packets.

Although broadcasting over the entire INI is naturally desirable, the last case of Table 2.3-3 (broadcasting-intra-subnetwork) is useful for partial re-transmission on communication groups. The sender process broadcasts a packet of data with the broadcast-LIA-address through the communication group, then waits for the responses (acknowledgments or negative-acknowledgments). The sender process may have to re-transmit the same packet to some member processes. Five cases are possible, re-transmission to each

- .process,
- .terminal,
- .LIA,
- .subnetwork, or
- .group of subnetworks.

The first is the simplest and has been adopted for the current version of INI, but here, the number of re-transmitted packets increases. The second and third could easily be implemented by making the LIA a little more sophisticated. The fifth is not desirable, since LIAs must necessarily know INI topology. The fourth, in which the broadcasting-intra-subnetwork is used, is now under study. This technique should be sufficiently effective to reduce the number of re-transmitted packets.

(2) Further Considerations of Bridges

When a bridge can receive no more packets because of its buffer-busy, it discards new packets without notifying the sending LIA. Therefore, the group communication end-to-end protocol of the logical layer must be responsible for detecting the loss of packets and for re-transmiting them. Because collision is the main cause of bridge buffer-busys, so the introducing of the priority ETHERNET [Kom1] might be effective to reduce the possibility of buffer-busys. The highest priority will be assigned to bridges. The online gathering protocol of statistics accumulated in bridges, such as the number of discarded packets and collisions, is also studied as one of network management functions.

Bridges are obviously weak point in INI from the viewpoint of reliability. Dual bridges and control protocols, such as switchover and echo-check, are also under study.

3. Design Guidelines for INI/CC Consultation System

Various knowledge-based systems, used for the installation and maintenance of computer or network systems, have been proposed, e.g., R1, for configuring VAX-11 computer systems [Mcd1], and ACE, for the maintenance of telephone-networks [Ves1].

INI/CC (Consultant for Configuring INI) is also a knowledge-based consultation system for determining the configuration of the INI compound network. It should be noted that logic programming system, such as PROLOG [Kow1], may be viewed as the first approximation to a KIPS (knowledge Information Processing System) [Yok1]. It possesses primitive but essential knowledge-processing capabilities, such as inference function and symbol processing, and it can also declaratively express various types of knowledge. The authors mean to feed it experiential knowledge gained in the course of using INI in order to construct INI/CC. The guidelines for constructing INI/CC are briefly described below.

(1) INI/CC is used to determine the following:

- .Location of LIAs
- .INI topology (location of bridges and repeaters)
- .Subnetwork addresses
- .Range of subnetwork addresses assigned to each bridge
- .Disposition of cables
- .Location of server processes

(2) INI/CC takes into consideration the following:

- .Characteristics of resources(LIA, Repeater, Bridge, Cable)
- .Layout of rooms and desks
- .Locality of communications
- .Load on server processes
- .Load on bridge
- .Number of collisions

4. Conclusion

This paper has shown a bridge protocol in the physical layer for inter-subnetwork routing and inter-subnetwork broadcasting. This protocol can be supported by a small bridge device and minimizes the inter-subnetwork routing overhead so as not to reduce ETHER-cable transmission capacity.

Compound local area networks, such as INI, are desirable for various reasons. And, more future applications will require the same group-communication facilities, based on the lower-layer broadcast facility, as the INI logical layer provides with. Therefore, inter-subnetwork broadcasting, which is evolved from intra-subnetwork broadcasting of ETHERNET, will be more important.

It should be pointed out that such facilities will also be essential for intelligent network managements and intelligent programming support environments in future [Tag1].

5. Acknowledgment

The author thank I. Yoshida and Y. Matsumura of OKI Ltd. for their cooperation in the development of INI protocols. The author also express their gratitude to Kazuhiro Fuchi, Director of ICOT Research Center, and Kunio Murakami, Chief of ICOT First Laboratory, for providing the opportunity of this work.

REFERENCES

[Bux1] Bux,W.,"Local Area Subnetworks:A Performance Comparison",IEEE Transactions on Communications Vol.COM-29 No.10,Oct. 1981

[DIX1] DEC,Intel and Xerox,"The Ethernet,A Local Area Network,Data Link Layer and Physical Layer Specification,V 20",1983

[Gfe1] Gfeller,F.,"INFRANET: Infrared Microbroadcasting Network for In-house Data Communication",7th European Conference on Optical Communication,Sept. 1981

[ISO1] "Data Processing-Open Systems Interconection-Basic Reference Model", ISO IS7498,1983

[Kom1] Komachi,Y.,et.al.,"Priority Ethernet Modified with Reassignment Algorithm"(in Japanese),Journal of the Institute of Electronics and Communication Engineers of Japan Vol.J66-D No.1,Jan. 1983

[Kow1] Kowalski,R.A.,"PROLOG AS A LOGIC PROGRAMMING LANGUAGE",AICA, Sept. 1981

[Mat1] Matsumura,Y.,Taguchi,A.,et.al.,"A Study for Interconnection of LANs",(In Japanese),the Institute of Electronics and Communication Engineers of Japan National Convention Record 1984,Mar. 1984

[Mac1] Macdonald,A.,"Visual Programming",Datamation Vol.28 No.11,Oct. 1982

[Mcd1] Mcdermott,J.,"R1: A Rule-Based Configurer of Computer Systems", Artificial Intelligence 19,1982

[Sche1] Scherr,A.L.,"A Perspective on Communications and Computing", IBM Systems Journal Vol.22 No.1,1983

[Schi1] Schindler,S.,"OPEN SYSTEMS MANAGEMENT -A Tutorial Elaboration on the ISO Approach-",Conference on DATA and TELE COMMUNICATIONS INTERNATIONAL-JAPAN'82 ,Jan. 1982

[Tag1] Taguchi,A.,et al,"The INI Internal Network and a Bird's-eye View of New Generation Computer-Networks", ICOT Tech. Rep. TM-0044

[Tag2] Taguchi,A.,et al,"INI: Internal Network in ICOT and its Future", ICCC'84, Oct. 1984

[Ves1]] Vesonder,G.T.,et.al.,"ACE:An Expert System for Telephone Cable Maintenance",Proc. of IJCAI'83,Aug. 1983

[Was1] Warren,D.S.,et.al.,"Executing Distributed PROLOG program on a Broadcast Network",Int. Symp. on Logic Programming,Feb. 1984

[Yam1] Yamazaki,H.and Yoshida,I.,"A Proposal for Broadcast Architecture Network(BANET)",Proc. of ICCC'82,Sep. 1982

[Yok1] Yokoi,T. "A Perspective of the Japanese FGCS Project",ICOT TM-0026, Sept. 1983

```
   .-----.            .-------.
   | PSI |            |       | Terminals/Computers
   |__.__|            |_______|
......|................./.....................
:  .--|--.        .--/--.      .-------.   :
:  | LIA |        | LIA |      | NCLIA |   :
:   --.--          --.--       |___.___|   :
:     |              |             |       :
: =======================/         |       :
: ETHER_cable          /========.========  :
:               \  /            |          :
:              .-\-/-.     < Bridge >      :
:              | LIA |          |          :
:               --.--           |          :
:                 |          \=========.=== :
:            ===============.===\       |  :
:                           |        .--|--. :
:                      < Bridge >    | LIA | :
:                           |         --.--  : .-----.
:  ========.===================/        \___:_|DELTA|
:         .-|--.             /=======       :  -----
:         | LIA|                            :
:          -.--                             :
:........./.\...............................:
    .----/   \.----.
    | PSI|    |PSI |  LIA  :LAN Interface Adaptor
     ----      ----   NCLIA:Network Control LIA
                      PSI  :Personal Sequential
                            Inference machine
                      DELTA:relational database machine
```

Fig.2.2-1 Physical Configuration of INI

```
   TERMINAL                                              OSI Model
._______________________________..______________.     ....______.
|<>__________.<>__________.||<>___________.|         |Layer|
||Application||Application||||Application||         |  7  |
||process "A"||process "B"||||process "C"||         |_____|
||___________||___________||||___________||<====>|Layer|
|_____________________________||________Application-|  6  |
                |                  |       protocol  |-----|
  LIA           |                  |              ...|Layer|
._______________|__________________|_______.         |  5  |
|<>_______________.                          Logical- |-----|
||Network-manager|                          protocol |Layer|
||   <>__________|.<>__________.<>__________.|<===>|  4  |
||    |Dummy     ||Dummy      ||Dummy      ||  ...|-----|
||____|process"a" ||Process"b"||process"c"||     |Layer|
| /\  |__________||___________||___________||     |  3  |
| ||                                         |     |-----|
| ||                                         |     |Layer|
| ||                                 Physical-     |  2  |
| =======>                            protocol     |-----|
| Network-management                          |<===>|Layer|
|   protocol                                  |     |  1  |
|___________________.________________________|  ...|_____|
                    |
```

Fig.2.2-2 Logical Structure of LIA

```
.____________________________________________________.
| Node address         | Terminal|   Process  |
|                      |  number |    number  |
|______________________|_________|____________|
.<---LIA-address--->.            .            .
:<------Terminal-address----->:               :
:<--------------Process-address---------->:
```

.The LIA-address corresponds to that of a node of ETHERNET protocol.
.The terminal-address is assigned to each terminal attached to LIA.
.The process-address is assigned to each application-process in terminals.

Fig.2.2-3 INI Addresses

```
  _______________________________________________________
 (          subnetwork/ETHER-cable      <10>   )
  ----.--------------------------------.----
      |                                |
   (B)<11;30>                       (B)<51;70>
  ____________|________           _________|_________
 (  subnetwork  <11>  )          ( subnetwork <51> )
  -.------------.-----           --.---------.----
   |            |                  |         |
 (B)<12;15>  (B)<21;25>         (B)<52;60>   |
 _______|______  ___|__________  _______|_______  |
(subnetwork<12>) (subnetwork<21>) (subnetwork<52>) |
 --------------   --------------   --------------  |
                                         <66;70>(B)
                                   ____________|____
                                  ( subnetwork <70>)
           (B) : bridge           ----------------
           <a> : subnetwork address
           <a;b> : range of subnetwork addresses
```

Fig.2.3-1 INI Topology

Table 2.3-1 LIA Addressing for Intra- and Inter-Subnetwork Communications

Type of LIA address	Destination address field of ETHERNET packet header: Multi-cast bit	LIA-address: Subnetwork address	LIA-address: Node-address-in-subnetwork	Meaning
Individual LIA-address intra subnetwork	off	specified	specified	Designates an LIA in the same network as the sending LIA.
Individual LIA-address inter subnetwork	on	specified	specified	Designates an LIA in a different subnetwork.
Broadcast LIA-address	on	Broadcast-address (Defined in ETHERNET)		Designates all LIAs in INI.
Broadcast LIA-address intra subnetwork	on	specified	not specified	Designates all LIAs within a subnetwork.

Table 2.3-2 Addresses in INI / in ETHERNET

INI	ETHERNET
Individual-LIA-address-intra-subnetwork	Node-address
Individual-LIA-address-inter-subnetwork	Node-address
Broadcast-LIA-address	Broadcast-address
Broadcast-LIA-address-intra-subnetwork	-------------
--------------	Multicast-group-address

Table 2.3-3 Bridge Processing

For packets with Individual-LIA-Address-Intra-Subnetwork: Bridges reject these packets.
For packets with Individual-LIA-Address-Inter-Subnetwork: A bridge discards these packets, if they are received from the subnetwork above (or beneath) the bridge and their subnetwork addresses are out of (or within) the preassigned range of subnetwork addresses. These packets are ineffective. A bridge sends out these packets to the opposite subnetwork after setting their multicast bit off, if their subnetwork addresses are the same as that of the opposite subnetwork linked to the bridge. Otherwise, a bridge allows these packets to pass through unmodified.
For packets with Broadcast-LIA-Address: Bridges allow these packets to pass through unmodified.
For packets with Broadcast-LIA-Address-Intra-Subnetwork: This means broadcasting within one subnetwork specified by the subnetwork-address field. A bridge performs the same processing as packets described above, except when their subnetwork addresses are the same as that of the opposite subnetwork. In this exceptional case, where the opposite subnetwork is the destination, the node-address-intra-subnetworks of these packets are changed to the one reserved for this purpose.