NBSIR 84-2984 (R)

Minutes of the Seventh Workshop for Implementors of ISO Open Systems Interconnection

U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards
Institute for Computer Sciences and Technology
Systems and Network Architecture Division
Gaithersburg, MD 20899

September 5-7, 1984

U.S. DEPARTMENT OF COMMERCE
BUREAU OF STANDARDS

NBSIR 84-2984

MINUTES OF THE SEVENTH WORKSHOP FOR IMPLEMENTORS OF ISO OPEN SYSTEMS INTERCONNECTION

U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards
institute for Computer Sciences and Technology
Systems and Network Architecture Division
Gaithersburg, MD 20899

September 5-7, 1984

U.S. DEPARTMENT OF COMMERCE, Malcolm Baldrige, ***Secretary***
NATIONAL BUREAU OF STANDARDS, Ernest Ambler, ***Director***

MINUTES OF THE SEVENTH NBS WORKSHOP FOR IMPLEMENTORS of ISO OPEN SYSTEMS INTERCONNECTION SEPTEMBER 5-7, 1984

Rob Rosenthal of NBS welcomed attendees to the seventh NBS workshop for implementors of Open Systems Interconnection. He presented a suggested schedule for future workshops (Attachment #1) and pointed out that the dates, which were selected to have minimum conflict with standards meetings, are firm but the hosts are tentative. Any company wishing to host a meeting should contact John Heafner of NBS.

Rob turned the floor over to Maris Graube, the moderator, who requested attendees to introduce themselves to the workshop. A list of participants at the 7th workshop is in Attachment #2.

The agenda for the 7th workshop was modified by the participants as in Attachment #3.

The agenda for the 8th workshop as proposed by the participants is in Attachment #4.

Several lists of vendors interested in implementing protocols were circulated solely to determine the eligibility to vote during the workshop - see Attachment #5 regarding workshop voting privileges. Organizations voting to implement one or more protocols agreed upon in general have listed themselves on Attachment #6. Organizations interested in implementing specific protocols or protocol mechanisms are shown on Attachments 7, 8, and 9, which name the coordinator of each specific interest group.

Jim Moulton of NBS presented the changes made to ISO transport at the June meeting of ISO TC97/SC16 in Copenhagen - see Attachment #10a.

The participants accepted the motion of Laurie Bride of BCS that the workshop Transport be aligned with the ISO Transport - see Attachment #10b.

Jim Moulton then presented a recommendation for enhancing the workshop Transport by adding expedited data and negotiation during connection establishment - see Attachment #11. Participants voted to accept expedited data by a vote of 14 to 0 - see Attachment #12.

Mr. Moulton's detailed proposal for negotiated transport features is in Attachment #13; it was accepted by a vote of 26 to 0.
Dick Swee of Charles River Data suggested that graceful close be implemented in the workshop's Transport. Jim Moulton pointed out that there is a reserved code in the ISO specification for graceful

close; and John Heafner announced that NBS would test graceful close with any vendor who implemented it. A vote was taken resulting in 2 for and 7 against. Dr. Heafner then offered to form a subgroup of vendors to demonstrate and test graceful close and asked interested parties to contact him. Mr. Moulton introduced the participants to the service provided by the Internetwork Protocol in Attachment #14.

Ross Callon of BBN gave an overview of IP architecture as described in the ISO document "Internal Organization of the Network Layer" - see Attachment #15.

Next, Dave Oran of DEC gave an overview of the Internetwork protocol - see Attachment #16.

After lunch, Ross Callon of BBN outlined the addressing schemes of the network layer - see Attachment #17.

Several proposals for a subset of IP functions to be implemented by workshop participants were made.

BCS presented a proposal by Laurie, John Heafner (NBS) and Ross Callon (BBN). It is recorded in two attachments, numbers 18 and 19.

Karl Scholl of GM suggested a subset of IP functions to be implemented as summarized in Attachment #20.

Ben Potter of ICL made a proposal for IP that took into account requirements for speedy implementation as well as constraints. (A copy of Ben's transparencies were not obtained during the workshop and could not be attached to these minutes.)

Francesco Cordera of Olivetti proposed that the workshop version of IP be the non-segmenting subset (ECMA 92) . (Francesco did not use a transparency or handout, so none is in the attachments.)

John Heafner of NBS proposed that companies intending to implement IP use the conformance (for catnets) and inactive subsets (for single subnetworks).

Mr. Cordera objected to implementing segmentation and made the following recommendations:

1. do not implement segmentation in IP but let X.25 segment when necessary;

2. not using segmentation allows the IPDU lifetime to be a simple hop count and thus avoids complex timeout routines; and

3. the IP addressing scheme should be hierarchical with the structure: domain/host/net identifiers.

John Heafner responded that X.25 is not the only wide-area protocol that IP will work over. Others in the audience pointed out that segmentation is needed to talk to other implementations and that originating IP entities can turn off the SP (Segmentation Permitted) flag at will.

A vote on John Heafner's proposal to implement the conformance and inactive subsets of IP resulted in 20 for and 2 against - see Attachment #21. Thus, the proposal was adopted.

The required protocol functions of IP were considered by the group which voted on those on which there was a difference of opinion - see Attachment #22.

Optional protocol functions were considered and voted on with results in Attachment #23 (Note that for Partial Source Routing the vote was to solicit detailed proposals.)

There were two detailed proposals for addressing in Internet: one by NBS-BCS-BBN (see the Addressing section of Attachment #19) and one by Dave Oran of DEC - see Attachment #24. It was decided that participants needed time to study the proposals but that a decision had to be made at the next workshop. (There may be additional proposals, but to be considered they must be fully documented with supporting text and must be mailed to attendees by the end of October 1984.)

At the evening session on Wednesday, September 5, Roy Cadwallader of ICL gave a presentation and slide show on the activities of European vendors participating in the ESPRIT Information Exchange System - see Attachment #25.

On Thursday, September 6, Kevin Mills of NBS provided an overview of session layer services and protocols - see Attachment #26.

Jim Berets of BBN gave a presentation of the ISO FTAM - see Attachment #27.

Pat Amaranth of GM proposed an alignment of ISO FTAM with the NCC '84 version of FTP - see Attachment #28. Paola Bucciarelli of Olivetti made another proposal for implementing ISO FTAM - see Attachments #29(a) and #29(b).

It was noted that the two proposals are basically in agreement but differ in the ability to create and delete files. Roy Cadwallader of ICL also presented an FTAM proposal - see Attachment #30.

ICL's proposal includes file access as an enhancement over the previous demo. and would be implementable for testing by June 1985.

John Heafner of NBS suggested that Olivetti's FTAM proposal be adopted for the next event and that file access be implemented in a later workshop phase.

The FTAM implementors voted 16 for and 1 against to adopt the Olivetti proposal as phase 1, i.e., for the next demo - see Attachment #31. (Note that Olivetti is to provide a document, clearly indicating the chosen subset of ISO FTAM, for the next workshop.

A discussion was held on the features of the Session protocol to implement for the next event. Opinions varied from none (Alan Sciacca of Foxboro), the minimum to support FTAM (Paola Bucciarelli of Olivetti), the minimum to support X.400 (Joe St. Amand of Wang), to the union of the minima to support FTAM and messaging (Kevin Mills of NBS). Dick Swee of CRDS spoke for the need of session, saying that there was no Session layer in the July 1984 demo and that it was necessary to distribute session functions between FTP and Transport. Kevin Mills explained that the union of minima would make it unnecessary to implement features as shown on Attachment #32. Paola recommended the Basic Combined Subset of Session as on Attachment #33 for supporting FTAM. Olivetti's proposal was voted on and accepted by a count of 19 for and none against - see Attachment #33). Kevin Mills's Session recommendation as on Attachment #34 was accepted by a vote of 17 for and 2 against.

A question was raised about which documents contain the standards for Session and FTAM. Kevin Mills stated that DIS 8326 and DIS 8327 were the current ISO Session documents. The consensus of the group was that the Olivetti submission.

Ken Dymond of NBS presented estimates of the cost of implementing Internet, aligning Transport, and of implementing Session and FTP - see Attachments #35(a), #35(b), #35(c), #35(d), and #35(e). Kevin Mills prepared a slide comparing Transport Class 4 and full Session in terms of number of services, states, transitions, etc. - see Attachment #36.

A discussion of the appropriate forum for the next activity was held. Maris Graube proposed that a show distributed in time and space rather than a specific event be selected. He explained that a permanent core location could be chosen where equipment would be more or less permanently running; various remote showrooms could then be connected as opportunity offered to vendor equipment at the core. Thus, there would be a show continuously available on demand. Maris prepared a slide which had as event end points the NCC in July 1985 and the Hanover Trade Fair in April 1986; and which further outlined some commitments - see Attachment #37.

A slide of possible demo events was made and an informal vote taken of those who might participate in each event to gauge the opinion of the group.

The event options considered and the straw vote on each are presented on Attachment #38 and summarized below.

9 for AUTOFAC	November 1985
1 for COMDEX	Fall/Late Summer 1985
4 for INTEC	" "
13 for a Coordinated Media Event	October 1985

14 for a Coordinated Media Event	March 1986
2 for NCC	July 1985
11 for Hanover	April 1986
0 for ACM	October 1985

Maris suggested that John Heafner prepare a form to be circulated after the conclusion of the current workshop. Attendees are to identify on the form which protocols they intend to implement and what systems they would attach long-term to a multi-organization concatenated network - see Attachment #39. The filled out form must be returned to John Heafner by Monday, October 29, 1984.

Maris Graube called for the opinion of those interested in UNIX and ISO layered architecture. A consensus was reached that ISO layers not be implemented in the internals of UNIX, but that the interfaces between ISO layers and UNIX be worked on. Also, the opinion prevailed that UNIX standards efforts be kept separate from networking considerations.

Art Pope of BBN then gave a tutorial on the X.400 message protocol of CCITT and a proposal for features of message handling to be implemented for a demo - see Attachments #40(a) and #40(b). Art noted that his X.400 demo proposal requires the Basic Activity Subset (BAS) of Session.

Fred Burg of AT&T presented a tutorial and recommendation on the Subnetwork Dependent Convergence Protocol (SNDCP) that should be implemented by anyone wishing to run ISO Internetwork Protocol over X.25 - see Attachment #41.

Ross Callon of BBN also proposed a method of using the ISO IP over X.25 - see Attachments #42(a) and #42(b). A motion was accepted to table the X.25 proposals for consideration by a special interest group to be organized and coordinated by Larry Brown of AT&T.

The subject of messaging was then raised again and 3 proposals in addition to Art Pope's (Attachment #41) were made:

Ian Valentine, ICL	- Attachment #43
Joe St. Amand, Wang	- Attachment #44
Dave Oran, DEC	- Attachment #45

Attendees then decided to table the 4 Messaging proposals and to ask Joe St. Amand to chair an interim meeting of companies interested in the X.400 recommendation. The resolutions of the interim meeting will be submitted to the next workshop.

The group unanimously accepted the suggestion of GM and BCS that FTAM be demonstrated in 2 phases (a reconfirmation of the decision noted earlier in these minutes) and that GM/BCS make a detailed proposal thereon - see Attachment #46.

Maris Graube asked John Heafner to compile a separate document from the ongoing minutes that lists the technical decisions adopted by the group. John agreed to do this.

The workshop adjourned at 3:25 p.m. on September 7, 1984.

Att. #1

IMPLEMENTORS OF OSI
WORKSHOP SCHEDULE

The following constitutes the schedule for NBS/OSI Workshops through Sept. 1985. The dates are firm. The hosts are not confirmed.

Nov. 7 - 9	(NBS), Gaithersburg, Md.
Jan. 22 - 24	(HIS), Phoenix, Ariz.
Apr. 16 - 18	(BCS), Seattle, Wash.
June 25 - 27	(NCR), Dayton, Ohio
Sept. 17 - 19	(NBS), Gaithersburg, Md.

ATTENDANCE LIST FOR THE OSI/NBS WORKSHOP

September 5 - 7, 1984

David E. Lawton
ACC
720 Santa Barbara St.
Santa Barbara, CA
93101

Mike Seto
ACC
720 Santa Barbara St.
Santa Barbara, CA
93101

Bob Jones
Allen-Bradley Co.
747 Alpha Dr.
Highland Heights, OH
44139

Laurence Brown
AT&T Bell Labs
190 River Rd.
Summit, NJ
07901

Khiem D. Ho
AT&T Information Systems
307 Middletown-Lincroft Rd.
Lincroft, NJ
07738

Douglas Knisely
AT&T-Bell Laboratories
1100 E. Warrenville Rd.
Naperville, IL
60566

James Berets
BBN
10 Moulton St.
Cambridge, MA
02238

George Chang
Bell Communications
6 Corporate Pl.
Piscataway, NJ
08854

Lean Theisen
Bell Communications Research
331 Newman Springs Rd.
Red Bank, NJ
07701

Kun Park
Bell Communications Research
331 Newman Spring Rd.
Red Bank, NJ
07701

Peter Lin
BNR Inc.
685 A E. Middlefield Rd.
Mountain View, CA
94039

Laurie Bride
Boeing Computer Services
P.O. Box 24346
Seattle, WA
98124

Ross Callon
Bolt, Bernaek & Newman
10 Moulton St.
Cambridge, MA
02278

Arthur Pope
Bolt, Bernaek and Newman
10 Moulton St.
Cambridge, MA
02230

Jon Becker
Burroughs Corp.
P.O. Box 1874
Southeastern, PA
19398

Richard Swee
Charles River Data Systems
983 Concord St.
Framingham, MA
01701

Prentiss Yates
Cincinnati Milacron
Rt. 48 & Mason Rd.
Lebanon, OH
45036

Charles Wade
Codex Corp.
20 Cabot Blvd.
Mansfield, MA
02048

Terence Holt
Compucorp
2211 Michigan Ave.
Santa Monica, CA
90404

Mary Jane Strohl
Concord Data Systems
303 Bear Hill Rd.
Waltham, MA
02154

John Weiss
Data General
4400 Computer Dr.
Westboro, MA
01580

Edward Brady
DCA
1060 Wiehle Ave.
Reston, VA
22090

Philip Selvaggi
DCA
1860 Wiehle Ave.
Reston, VA
22090

David Oran
Digital Equipment
1925 Andove St.
Tewksbury, MA
01876

Jay Jeyabalan
Ford Motor Co.
Scientific Research Lab
Dearborn, MI
48121

Robert Yee
Ford Motor Co.
The American Rd.
Dearborn, MI
48121

K.J. Hawang
General Electric
P.O. Box 8106
Charlottesville, VA
22906

Ronald Smith
General Electric
401 N. Washington St.
Rockville, MD
20850

Richard Friberg
General Electric Co.
P.O. Box 8106
Charlottesville, VA
22906

Barry Dallavalle
General Electric FAPD
P.O. Box 8106
Charlottesville, VA
22906

Gary Workman
General Motors
A/MD-39 12 Mile & Mound Roads
Warren, MI
48084

Karl Schohl
General Motors
30300 Mound Rd. MD/A-39
Warren, MI
48090

David Willcox
Gould
1101 E. University St.
Urban, IL
61801

Cathy Burns
Gould
SVS P.O. Box 3083
Andovor, MA
01810

Andrew Poupart
Gould Computer Systems
6901 W. Sunrise Blvd.
Plantation, FL
33313

Raj Khurana
GTE
1700 Research Blvd.
Rockville, MD
20850

Carl Pfeiffer
GTE Government Systems
1700 Research Dr.
Rockville, MD
20850

Lyle Weiman
Hewlett Packard
19420 Homstead Rd.
Cupertino, CA
95014

George Bankeroff
Honeywell
900 Middlesex Turnpike
Billerica, MA
01821

Roger Thompson
IBM
1501 California Ave.
Palo Alto, CA
94306

A. W. Kleitsch
IBM

RTP, NC
27709

Pat Mulvey
IBM
P.O. Box 1328
Boca Raton, FL
93432

Roy Cadwallader
ICL
Kidsgrove, Staffs
UNITED KINGDOM
ST7 1TL

Ian Rvalentine
ICL
Bracknell Berks
UNITED KINGDOM
RG 12 1XX

Frank Hsu
Intel
3200 Lakeside Dr.
Santa Clara, CA
95051

David Potter
Interlan Inc.
3 Liberty Way
Westford, MD
01886

Ben Potter
International Computer Ltd.
London Street
Reading Berks, LONDON

Sudhi Umerji
ITT Dialcom Inc.
1100 Wayne Ave.
Silver Spring, MD
20910

Ray Denenberg
Library of Congress
Network Development Office
Washington, DC
20540

Jim Clarkson
Motorola
2900 South Diablo Way
Tempe, AZ
85282

Stephen Nightingale
NBS
Bldg. 225, Rm. B218
Gaithersburg, MD
20899

Ken Dymond
NBS
Bldg. 225, Rm. B208
Gaithersburg, MD
20899

Jim Moulton
NBS
Bldg. 225, Rm. B212
Gaithersburg, MD
20899

Bob Blanc
NBS
Bldg. 225, Rm. A231
Gaithersburg, MD
20899

Debra Tang
NBS
Bldg. 225, Rm. B226
Gaithersburg, MD
20899

Mike Wallace
NBS
Bldg. 225, Rm. B226
Gaithersburg, MD
20899

Robert Rosenthal
NBS
Bldg. 225, Rm B226
Gaithersburg, MD
20899

John Heafner
NBS
Bldg. 225, Rm. B218
Gaithersburg, MD
20899

Kevin Mills
NBS
Bldg. 225, Rm. B226
Gaithersburg, MD
20899

Wood Wiles
NCR
1700 S. Patterson Blvd.
Dayton, OH
45479

David Cappell
NCR Comten
2700 Snelling
Roseville, MN
55113

Will McDuffie
Network Solutions Inc.
7700 Leesburg Pike
Falls Church, VA
22043

Paul Masters
Northern Telecom
2305 Mission College Blvd.
Santa Clara, CA
95050

Edward B. Matthews
Northern Telecom Inc.
259 Cumberland Bend
Nashville, TN
37228

M. Sartorio
Olivetti
Ivrea (To) 10015
ITALY

P. Bucciarelli
Olivetti
Ivrea (To) 10015
ITALY

F. Cordera
Olivetti
10015 Ivrea (To)
ITALY

Michale Spina
Prime Computer
500 Old Connecticut Path
Framingham, MA
01701

Ann Jenkins
Prime Computers
Prime Park
Natick, MA
01760

Larry Daldin
Rolm Corp
30150 Telegraph Rd.
Birmingham, MI
48010

Dittmar Janetzky
Siemens AG ESTE 23
7500 Karlsruhe
GERMANY

Edward J. O'Conner
Sperry
Box 500 C2-SE1
Blue Bell, PA
19424

Derham Eginton
Sperry
2276 Highcrest Dr.
Roseville, MN
55113

J. Gansz
Sperry
P.O. Box 500
Blue Bell, PA
19424

Richard K. Shiffer
Sperry
P.O. Box 500
Blue Bell, PA
19424

Daniel J. Ferrara
Sperry
P.O. Box 500
Blue Bell, PA
19424

Brian Vonn
Square D Co.
4041 N. Richards
Milwaukee, WI
53212

Barbara R. Sternick
Systems Development Corp.
7929 Westpark Dr.
McLean, VA
22102

Maria Graube
Tektronix
Box 500
Beaverton, OR
97077

Atul Bhatnagar
Tektronix Inc.
P.O. Box 500
Beaverton, OR
97077

Dan Moon
Texas Instruments
P.O. Drawer 1255
Johnson City, TN
37605-1255

Alan Sciacca
The Foxboro Co.
38 Newpowset Ave.
Foxboro, MA
02035

Joseph St. Amand
Wang
1 Industrial Ave.
Lowell, MA
01851

Joseph Holmes
Wang
2 Cross of Commerce
Rollowing Medows, IL
60008

Jagdee Gahlawat
Wang
One Industrial Ave.
Lowell, MA

Clive Everett
Wang
One Industrial Dr.
Lowell, MA
01851

Herbert S. Falk
Westinghouse
1521 Avis
Madison Heights, MI
48071

Juan Bulnes
Xerox
3450 Hillview Ave.
Palo Alto, CA
93404

Andrew Huang
Ztel
181 Ballarduale St.
Wilmington, MA
01887

Attachment #3

AGENDA

Seventh Workshop for Implementors of Open Systems Interconnection

Wednesday, September 5 Morning

Welcome and Opening Remarks	Rob Rosenthal, NBS, IEEE
Opening Remarks by Moderator	Maris Graube, Chair. 802
Approval of Agenda	Attendees
Transport: NCC/ISO Alignment	Jim Moulton, NBS
Transport: Addition of ISO Features	Jim Moulton, NBS
Internetwork Overview, Services	Jim Moulton, NBS

Wednesday, September 5 Afternoon

Internetwork Overview, Architecture	Ross Callon, BBN
Internetwork Overview, Protocol	Dave Oran, DEC
Internetwork Overview, Addressing	Ross Callon, BBN
Internetwork Proposals	1. Laurie Bridge, BCS Ross Callon, BBN John Heafner, NBS 2. Karl Scholl, GM 3. Ben Potter, ICL 4. Francesco Cordera, Olivetti
Implementation Estimates: layer 3, layer 4	Ken Dymond

Wednesday, September 5 Evening

European Activities:	Ray Cadwallader, ICL

Thursday, September 6 Morning

Session Overview, Services and Protocol	Kevin Mills, NBS
Session Proposal	Kevin Mills, NBS
FTAM Overview	Jim Berrets, BBN

Thursday, September 6 Afternoon

FTAM Alignment ISO/NCC	Pat Amaranth, GM
FTAM: Proposals for Addition of ISO Features	1. Pat Amaranth, GM 2. Paola Bucciarelli, Olivetti 3. Ray Cadwallader, ICL
Implementation Estimates: Session and FTAM	Ken Dymond, NBS

Thursday, September 6 Evening

Determine & Schedule Second OSI Activity	Attendees

Friday, September 7 Morning

X.25 Network Dependent Convergence
Protocol Proposals....................................1. Fred Burg, Bell Labs.
2. Ross Callon, BBN

Friday, September 7 Afternoon

400 Series Overview....................................Art Pope, BBN
400 Series Proposal....................................Art Pope, BBN

Attachment #4

TENTATIVE AGENDA FOR EIGHTH OSI WORKSHOP

Welcome by host organization	
Opening Remarks by Moderator	Maris Graube, Tektronix
Approval of Agenda	Attendees
1. Responses to NBS Questionnaire	John Heafner, NBS
2. ISO Update	NBS
3. Resolution of Internet Proposals	
a. Addressing	
b. Routing	
4. Topologies	
a. Proposals for networks supported by demo	
b. Proposals for addressing support of accepted topologies	
5. FTAM	
a. Presentation of Phase 1 specification	Olivetti
b. Proposal for Phase 2	BCS/GM
c. File naming proposals	(solicited)
d. Data transfer facility	GM
6. X.25	
a. Selection of a proposal	Attendees voting
b. Determination of specific X.25 networks	" "
7. Messaging	
a. Selection of a proposal	" "
b. Selection of a Session Services	" "
8. NBS Schedules	NBS
a. Tests	
b. Cooperative testing	
c. Neutral site facility	

Attachment #5

IMPLEMENTATION INTEREST GROUPS

These interest groups were identified to establish voting rights in the workshops. The organizations so identified have expressed an interest in implementing the protocols or mechanisms indicated. There is no expressed or implied corporate committment at this time to actually provide implementations. The above statements apply to Attachments #6 through 9.

Attachment #6

GENERAL INTEREST PROTOCOLS

The organizations named below have expressed an interest in implementing one or more of the following: IEEE 802.2, IEEE 802.3, IEEE 802.4, NBS/ISO connectionless IP, NBS/ISO transport class 4, ISO session subset, and/or ISO FTAM subset.

Also, indicated is whether the organization would implement an end-system, an internetwork systems or act as a user.

ORGANIZATIONS VOTING TO IMPLEMENT PROTOCOLS

ORGANIZATION	END-SYSTEM	INT.-SYSTEM	USER
ACC	X	X	X
AT&T Information Systems	X	X	
Allen-Bradley	X	X	
AT&T	X	X	
Bell Communications Research	X		X
Boeing Computer Services	X		X
Charles River Data Systems	X	X	X (applications)
Codex Corp.		X	
Concord Data Systems	X	X	
Data General	X	X	
Digital Equipment	X	X	
Foxboro	X	X	
General Electric - FAPD	X	X	
General Motors			ABSTAINED
Gould Computer Systems	X	X	
Gould Prog. Cont. Div.	X		
Hewlitt Packard	X	X	
Honeywell	X	X	
IBM	X		
ICL	X	X	
Motorola	X		
NBS	X	X	X
NCR	X		
NT	X	X	
Olivetti	X		
Square D	X	X	
Tektronix	X		X
Texas Instruments, ISD	X		
Westinghouse	X	X	X

Attachment #7

ORGANIZATIONS INTERESTED IN IMPLEMENTING A NETWORK DEPENDENT CONVERGENCE SUBLAYER PROTOCOL BETWEEN X.25 AND ISO C-LESS IP

A-B

ATT

CONCOD

D.G.

GOULD

ICL

INTEL

NBS

NCR

NT

OLIVETTI

WANG

COORDINATOR: Laurence Brown AT&T
(201)522-6046

Attachment #8

SPECIAL INTEREST GROUP ON ROUTING PRINCIPLES

NAME	COMPANY	PHONE
Bankeroff, George	Honeywell	(617) 671-7476
Bhatnagar, Atul	Tektronix	(503) 627-6833
Cordera, Francesco	Olivetti	39(125) 525 ext. 1339
Dymond, Ken	NBS	(301) 921-2601
Everett, Clive	Wang Labs.	(617) 967-2417
Falk, Herb	Westinghouse	(313) 588-1540
Heafner, John	NBS	(302) 921-3537
Huang, Andrew	Ztel	(617) 657-8730
Jeyabalan, Jay	Ford Motor Co.	(313) 322-3952
Jones, Bob	Allen-Bradley	(216) 449-6700
Knisely, Doug	AT&T	(312) 979-7344
Masters, Paul	NT	(408) 353-3819
Matthews, E.B.	Northern Telecom	(615) 256-5900
O'Connor, Ed	Sperry	(215) 542-5937
Pfeiffer, Carl	GTE	(301) 294-8514
Potter, Ben	ICL	44-734-586244
Vonn, Brian	Square D	(414) 332-2000
Wade, Chuck	Codex Corp.	(617) 364-2000
Weiman, Lyle	HP	(408) 725-8111
Wiles, Wood	NCR	(513) 445-6635
Workman, Gary	GM	(313) 575-0632
Yates, Prentiss	Cincinnati Milacron	(513) 494-5367

*The coordinator is Dr. John Heafner of NBS.

Attachment #9

SPECIAL INTEREST GROUP ON CCITT X.400
DRAFT RECOMMENDATIONS

Ho, Khiem	AT&T Information Systems	(201) 576-6227
Masters, Paul	Northern Telecom	(408) 988-5550
Morgan, John	GE Information Services	(301) 294-5556
*St. Amand, Joseph	Wang Laboratories	(617) 967-5506
Swee, Dick	Charles River Data Systems	(617) 626-1000
Valentine, Ian	ICL	+44 344 424842

*The coordinator is Joseph St. Amand

Att. #10(a)

TRANSPORT PROTOCOL

Alignment Changes

There are two changes necessary to bring the demo version in line with the ISO specification:

1. Octet Ordering - the order of octets in multi octet binary fields has been changed from least significant octet first to most significant octet first.

2. Parameter Code - the value of the flow control parameter has been changed to 1000 1100 make it unique (it had the same value as another parameter).

NBS-J.M.

Att. #10(3)

TRANSPORT PROTOCOL
ALIGNMENT CHANGES

1. OCTET ORDERING
   - FROM least significANT TO MOST significant
   - EOT IS now always MOST SIGNIFICANT BIT

2. PARAMETER CODE
   - FLOW CONTROL PARAMETER CHANGED TO 1000 1100

27/0

AH. #11

TRANSPORT PROTOCOL

Added Features

There were two features of the transport protocol which would enhance the demo version:

1. Expedited Data - The expedited data service and the associated protocol mechanisms were not included in the demo. for the next demo it would be appropriate to include expedited data especially considering the multinetwork approach.

2. Negotiation During Connection Establishment - for the first demo, parameter values were selected to avoid negotiation. In the next demo it would be approppriate to include full parameter negotiation utilizing the connect negotiation rules.

   - Follow ISO rules
   - TPDU size
   - Use of expedited
   - QoS parameters
   - Use of checksum

3. TSDU Size

Att. #12

TRANSPORT PROTOCOL
ADDED FEATURES

1. EXPEDITED DATA
   - ONE "UNACKED" AT A TIME
   - MAY OVERTAKE NORMAL DATA ~~DEFER~~ TO ~~SESSION~~ 14/0

2. NEGOTIATION DURING CONNECTION ESTABLISHMENT
   - FOLLOW ISO rules
   - TPDU size ~~DEFER~~ TO ~~INTERNET~~
   - use of EXPEDITED
   - QOS PARAMETERS
   - USE OF CHECKSUM REVISIT ~~AFTER~~ ~~BREAK~~

3. TSDU SIZE

QOS

thru put
12 octets
MAX values
TARGET →
MIN →
target ←
min. ←
Ave. values

Att. #13

PROPOSAL FOR NEGOTIATION

	ISO	DEMO
16/31 SEQUENCE	OPTIONAL ONLY IF OFFERED	ALL IMPLEMENTATIONS SEND 16/31 in CR TPDU; must be able to accept 4/7 in CR TPDU (more Restrictive)
TPDU SIZE	8K TO 128 ALWAYS NEG. DOWN	ALLOW ANY VALID SIZE IN CR TPDU - FOLLOW ISO rules
SECURITY	OPTIONAL - USER Defined	All implementations should NOT SEND in CR TPDU; if Received ignore (more Restrictive)
CHECKSUM	Both must agree not to use	implementation choice on requesting use; must be able to operate with checksum if requested (ISO rules)
ACK TIME	OPTIONAL	Do not send in CR TPDU; ignore if received (ISO rules)
Throughput Priority Transit Delay	OPTIONAL	Do not send in CR TPDU ignore in CC TPDU -- allowable by ISO rules
User Data in CR TPDU; CC TPDU	OPTIONAL	no implementation should send; all must be prepared to receive (allowed by ISO rules)

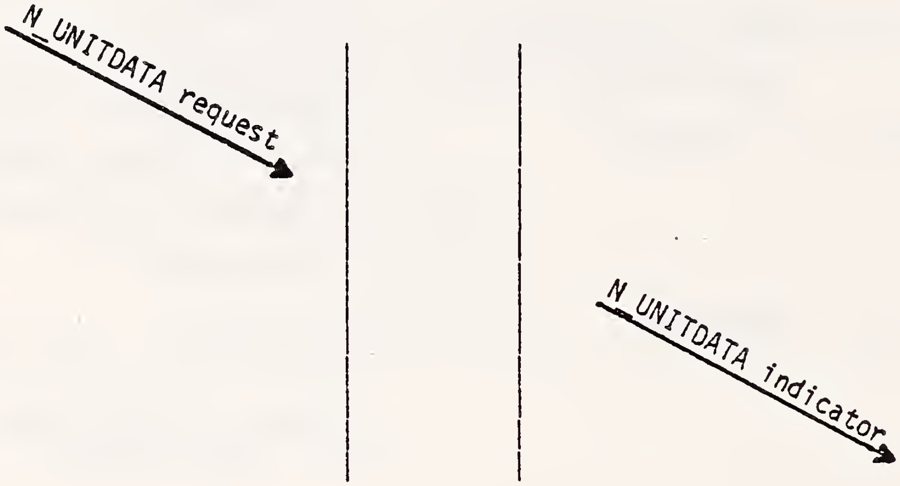
26/0

Internetwork Protocol

Services Provided

The internetwork protocol (IP) provides a connectionless service. That is to say, it does not depend on the establishment of connections or virtual circuits between peer entities. The service provided is on a per request basis. There is no explicit or implicit relationship between service requests. Additionally, there is no confirmation of success or failure to the service requestor.

The service provided by the IP consists of one service interaction:



Associated with the service primitives are Quality of Service Parameters (QOS). Each parameter describes a characteristic that is provided by the service.

1. Transit delay - the time between a request and indication,
2. Protection from Unauthorized Access - the extent to which protection is provided,
3. Cost,
4. Residual Error Probability - the likelihood that an NSDU will be lost, duplicated, or incorrectly delivered,
5. Priority, and
6. Source Routing - a specification of the path an NSDU is to take.

9/5/84 Att. #15
R.Callon on ICNL

INTERNAL ORGANIZATION OF THE NETWORK LAYER

DEFINES ARCHITECTURAL ORGANIZATION OF THE NETWORK LAYER

PROVIDES MAPPING OF ABSTRACT ORGANIZATION TO "REAL WORLD" COMPONENTS

IDENTIFIES AND CATEGORIZES THE FUNCTIONS PERFORMED BY NETWORK LAYER PROTOCOLS

PROVIDES A UNIFORM FRAMEWORK FOR DESCRIPTION OF THE OPERATION OF THE NETWORK LAYER

EXTENDS TERMINOLOGY

DEALS WITH COMPLEX "REAL WORLD"

ROLES OF NETWORK LAYER PROTOCOLS

SUBNETWORK INDEPENDENT CONVERGENCE PROTOCOLS (SNICP)

- OFFERS OSI NETWORK SERVICE ON SUBNETWORK INDEPENDENT BASIS

- MAY BE INTERNETWORK PROTOCOL, SET OF RULES FOR COORDINATING SUBNETWORK SERVICES, OR NULL

SUBNETWORK DEPENDENT CONVERGENCE PROTOCOLS (SNDCP)

- PROVIDES SERVICE REQUIRED BY SNICP, OR PROVIDES OSI NETWORK SERVICE
- OPERATES OVER SNAcP
- MAY BE:
    - EXPLICIT PROTOCOL
    - SET OF RULES (E.G., FOR RUNNING CLIP OVER X.25)
    - NULL

ROLES OF NETWORK LAYER PROTOCOLS (CONT'D)

SUBNETWORK ACCESS PROTOCOLS (SNAcP)

- WHATEVER IS USED TO ACCESS A SPECIFIC SUBNETWORK
- MAY BE:
  - EXISTING SUBNETWORK PROTOCOL (ARPANET 1822, ...)
  - STANDARD PROTOCOL (X.25, ...)
  - NULL (IEEE 802, ...)
  - PRESENT ONLY DURING CONNECTION ESTABLISHMENT AND TERMINATION (X.21)
  - OR ...

GENERALLY

- AT LEAST ONE NETWORK LAYER PROTOCOL MUST BE PRESENT
- RECURSIVE USE OF PROTOCOL ROLES IS POSSIBLE

APPROACHES TO NETWORK LAYER INTERCONNECTION

HOP BY HOP ENHANCEMENT

- SNDCP INDIVIDUALLY ENHANCES EACH SUBNETWORK IN A CHAIN TO OFFER OSI NETWORK SERVICE
- SNICP CONSISTS OF RELAY AND ROUTING RULES FOR CONCATENATING SUBNETWORK SERVICES
- COULD IN PRINCIPLE BE CONNECTIONLESS OR CONNECTION-MODE
- IMPLICIT CONNECTION-MODE (X.25) ORIENTATION

INTERNETWORK PROTOCOL

- OPERATES AS END-TO-END PROTOCOL
- SUBNETWORKS MAY BE DIVERSE
- SNDCP MAY BE REQUIRED IN SOME CASES (E.G., RULES TO MANAGE X.25 CONNECTIONS)
- COULD IN PRINCIPLE BE CONNECTIONLESS OR CONNECTION-MODE
- CURRENTLY IMPLICITLY INTENDED FOR CONNECTIONLESS PROTOCOL (DIS 8473)

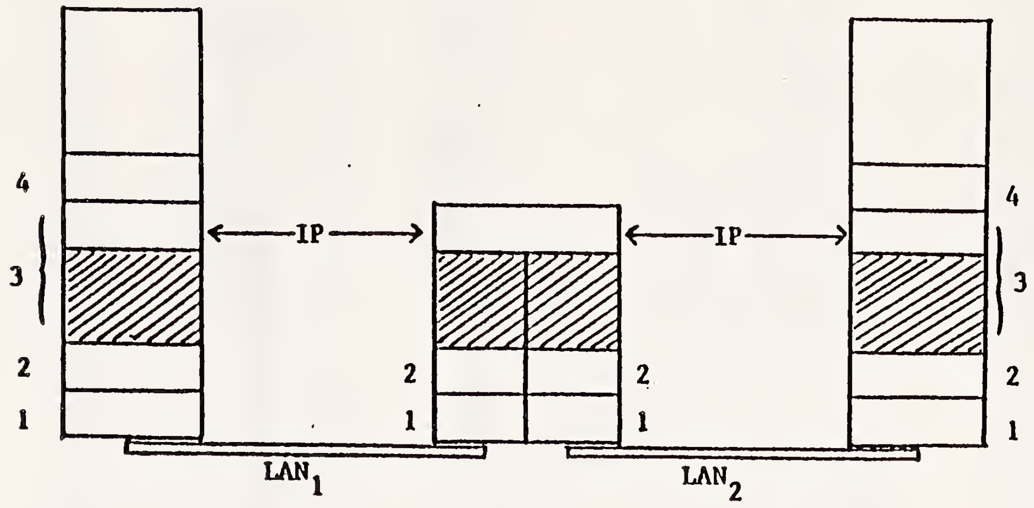
IOOTNL DOCUMENT GIVES EQUAL TREATMENT OF BOTH APPROACHES

EXAMPLE ARCHITECTURE FOR INTERNETWORK PROTOCOL

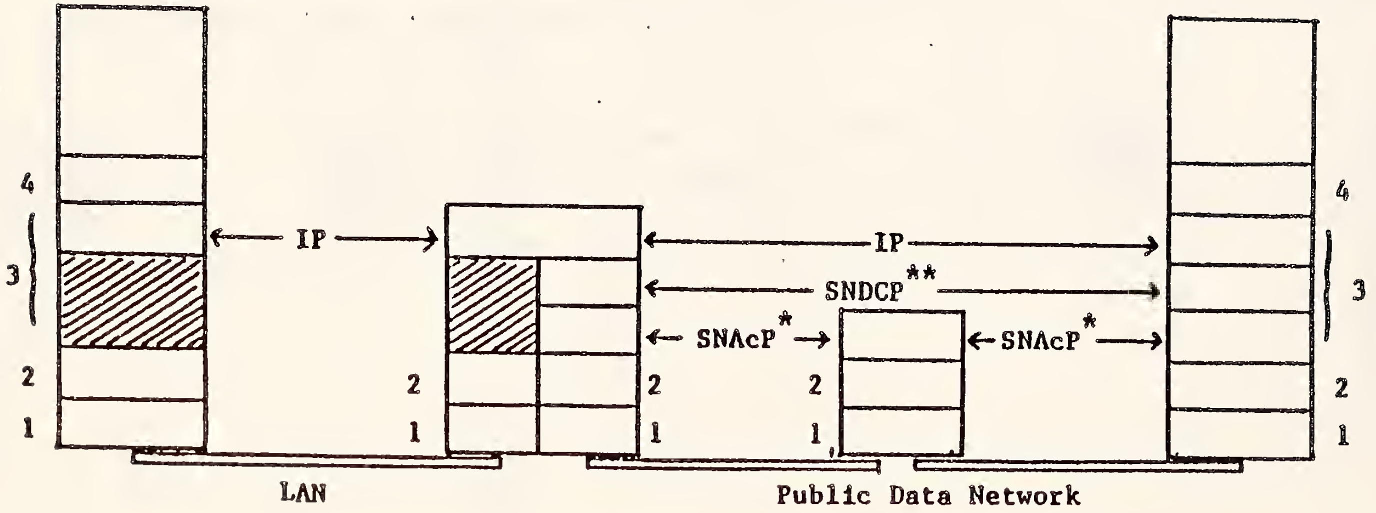
SNICP <==> CONNECTIONLESS INTERNETWORK PROTOCOL (ISO DIS 8473)

SNDCP <==> {
NULL (OVER LANS)
RULES FOR CONNECTION MANAGEMENT, ETC (OVER PDNS)
ETC...

SNAcP <==> {
NULL (OVER LANS)
X.25 PACKET LEVEL (OVER PDNS)
ETC...

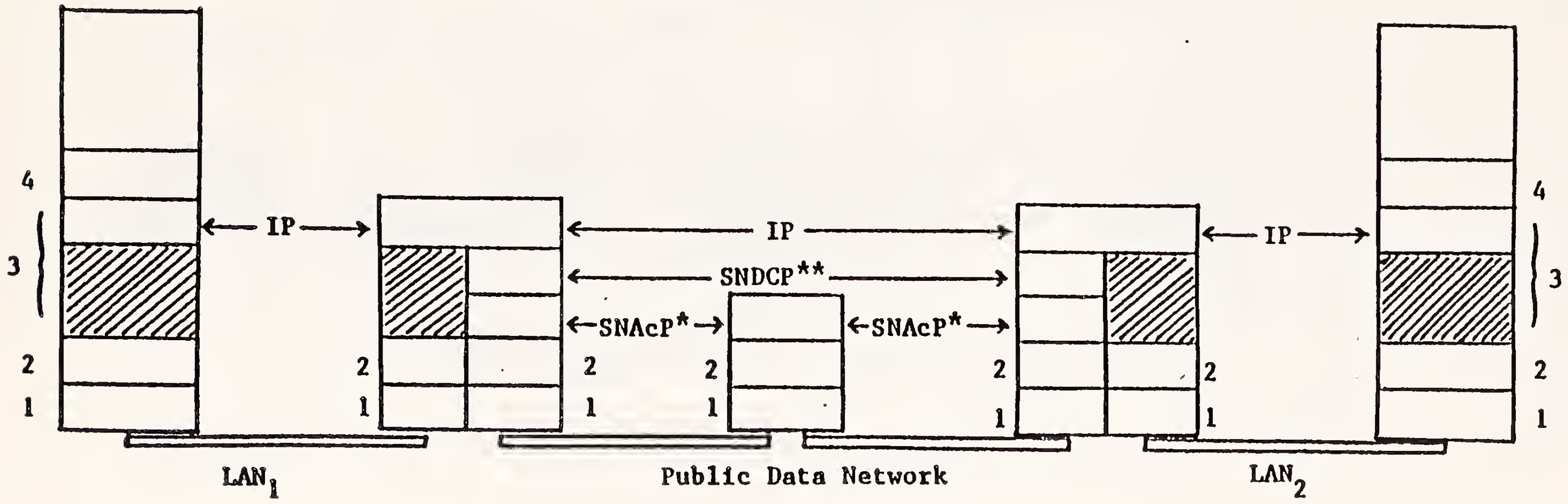


Use of Internetwork Protocol to Interconnect LANs



* X.25 Packet Level
** Rules for Running IP over X.25

Use of Internetwork Protocol for Interconnecting
Local Area Network with Public Data Network



* X.25 Packet Level
** Rules for Running IP over X.25

Use of Internetwork Protocol for Interconnecting
Two Local Area Networks Via a Public Data Network

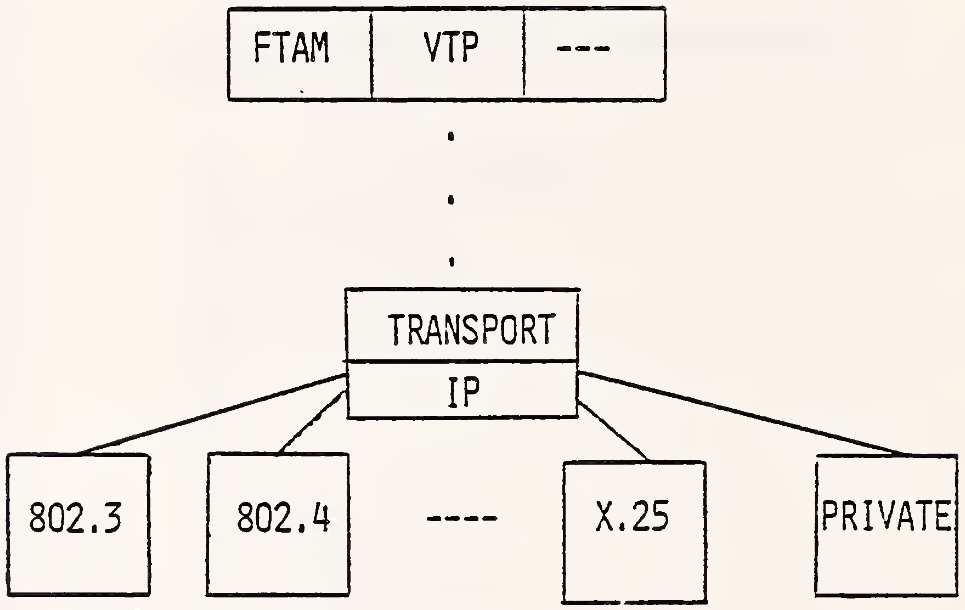
Dave Oran 9/5/84 Att. #16

CONNECTIONLESS INTERNETWORK PROTOCOL

OVERVIEW

PROPERTIES

- DRAFT INTERNATIONAL STANDARD
- CONCATENATION OF DIFFERENT SUBNETWORK TECHNOLOGIES



- SIMPLE, EFFICIENT PROTOCOL

SERVICES

- PROVIDED TO TRANSPORT LAYER
  - UNIT DATA SEND
  - UNIT DATA RECEIVE
- PROVIDED BY THE NETWORK LAYER
  - UNIT DATA SEND
  - UNIT DATA RECEIVE
- PROVIDED BY THE LOCAL ENVIRONMENT
  - TIMER REQUEST
  - TIMER REPONSE
  - TIMER CANCEL

REQUIRED PROTOCOL FUNCTIONS

- PDU COMPOSITION
- PDU DECOMPOSITION
- HEADER ANALYSIS
- LIFE TIME BOUNDING
- ROUTING
- FORWARDING
- SEGMENTING
- REASSEMBLING
- DISCARDING
- ERROR REPORTING
- ERROR DETECTION
- PADDING

OPTIONAL PROTOCOL FUNCTION

- SECURITY
- SOURCE ROUTING
- ROUTE RECORDING
- QUALITY OF SERVICE

PDU STRUCTURE

- FIXED FIELDS
- ADDRESSES
- SEGMENTATION
- OPTIONS
- USER DATA

PDU TYPES

- DATA
- ERROR REPORTS

FORMAL DESCRIPTION

- EXTENDED FINITE STATE MACHINE LANGUAGE
- STATE TRANSITIONS PLUS PASCAL
- MODELS A SINGLE SERVICE REQUEST

CONFORMANCE

- FULL PROTOCOL

CHECKSUMS

- OVER THE HEADER ONLY
- GENERATING CHECKSUMS
- CHECKING CHECKSUMS
- ALTERING CHECKSUMS

Ross Callon 9/5/84

Att. #17.

NETWORK LAYER ADDRESSING

CONCEPTS AND TERMINOLOGY

PRINCIPLES FOR CREATING THE NETWORK LAYER ADDRESSING SCHEME

NETWORK ADDRESS SEMANTIC STRUCTURE

REPRESENTATION AS BINARY AND DECIMAL

RELATIONSHIP BETWEEN SEMANTICS, REPRESENTATION. AND ENCODING

BASIC PRINCIPLES OF THE NETWORK LAYER ADDRESSING SCHEME

HIERARCHICAL STRUCTURE OF NSAP ADDRESSES

- ROUTING
- ADMINISTRATION OF ADDRESS SPACE
- MULTI-LEVEL HIERARCHY
- CONCEPT OF ADDRESS "DOMAINS" AND "SUBDOMAINS"

GLOBAL IDENTIFICATION OF ANY NSAP

ROUTE AND SERVICE TYPE INDEPENDENCE

BINARY AND DECIMAL ADDRESSES ACCOMMODATED

VARIABLE LENGTH ADDRESSES UP TO A DEFINED MAXIMUM SIZE

NETWORK ADDRESS SEMANTIC STRUCTURE

INITIAL DOMAIN PART (IDP)

- AUTHORITY AND FORMAT IDENTIFIER (AFI)
    - CONVEYS FORMAT, LENGTH, AND "ABSTRACT SYNTAX" OF THE REST OF NSAP ADDRESS
    - SPECIFIES AUTHORITY RESPONSIBLE FOR ALLOCATING THE INITIAL DOMAIN IDENTIFIER
- INITIAL DOMAIN IDENTIFIER (IDI)
    - FOLLOWS ONE OF EIGHT FORMATS (SEE NEXT VIEWGRAPH)
    - SPECIFIES THE NETWORK ADDRESSING SUBDOMAIN FROM WHICH VALUES OF THE DSP ARE ALLOCATED
    - SPECIFIES THE AUTHORITY RESPONSIBLE FOR ALLOCATING VALUES OF THE DSP

DOMAIN SPECIFIC PART (DSP)

- SEMANTICS IS (LOCALLY) SIGNIFICANT IN THE CONTEXT SPECIFIED BY THE IDP
- MAY BE BASED ON DECIMAL, BINARY, CHARACTER, OR "NATIONAL CHARACTER"

INITIAL DOMAIN IDENTIFIER FORMATS

X.121-DTE

- IDI IS AN X.121 ADDRESS (UP TO 14 DIGITS)

X.121-DCC

- IDI IS AN X.121 DATA COUNTRY CODE (3 DIGITS)

F.69

- IDI IS A TELEX NUMBER (UP TO 8 DIGITS)

E.163

- IDI IS A TELEPHONE NETWORK (PSTN) NUMBER (UP TO 12 DIGITS)

E.164

- IDI IS AN ISDN NUMBER (UP TO 15 DECIMAL DIGITS)

ISO-6523

- IDI IS ALLOCATED ACCORDING TO ISO 6523. CONSISTING OF A 4 DIGIT INTERNATIONAL CODE DESIGNATOR (ICD), FOLLOWED BY UP TO 28 DIGITS DERIVED FROM AN ORGANIZATION CODE

ISO-6523-ICD

- IDI IS ALLOCATED ACCORDING TO THE ICD FROM ISO 6523

LOCAL

- IDI IS NULL (FOR USE IN A CLOSED COMMUNITY)

RELATIONSHIP BETWEEN SEMANTICS, REPRESENTATION, AND ENCODING

WHAT HAVE WE STANDARDIZED?

- SEMANTICS:
  - AFI (TWO DECIMAL DIGITS)
  - IDI (VARIABLE DEPENDING ON AFI, DECIMAL DIGITS)
  - DSP (VARIABLE, BASED ON DECIMAL, BINARY, CHARACTER, OR NATIONAL CHARACTER)
- PURE DECIMAL REPRESENTATION
- PURE BINARY REPRESENTATION
- ALGORITHMIC TRANSFORMATIONS

WHAT IS ENCODED IN PROTOCOL HEADERS?

- UP TO PROTOCOL DEFINITION (NOT ADDRESS STANDARD)
- MUST CONVEY SEMANTICS OF ADDRESS STRUCTURE
- MAY USE PURE DECIMAL OR BINARY REPRESENTATION
- MAY DEFINE OTHER WAY TO CONVEY SEMANTICS (E.G., SHORTHAND FOR LOCAL ADDRESSES)

ACCOMMODATION OF BINARY AND DECIMAL

BINARY OR DECIMAL ADDRESS ISSUE HAS BEEN CONTROVERSIAL

- IEEE 802 AND MANY PRIVATE NETWORK ADDRESSES BASED ON BINARY

- X.121, PSTN, AND TELEX ADDRESSES BASED ON DECIMAL

DECISION TO ACCOMMODATE BOTH

- ADDRESS IDP (AFI AND IDI) BASED ON DECIMAL

- DSP BASED ON DECIMAL, BINARY, CHARACTER, OR NATIONAL CHARACTER

- <u>EVERY</u> ADDRESS CAN BE FULLY REPRESENTED IN <u>BOTH</u> PURE BINARY AND PURE DECIMAL

- ALGORITHMIC CONVERSION BETWEEN PURE BINARY AND PURE DECIMAL REPRESENTATIONS DEFINED

INTERNETWORK PROTOCOL USES BINARY REPRESENTATION, CARRIES DECIMAL BASED FIELDS AS BCD

- TRANSFORMATIONS NOT REQUIRED IN THIS CASE

L. Bride Wed. 9/5/84

Att. #18

Proposal for a Useful Connectionless Internetwork Protocol

BASIC PROPOSAL

The conformance (full) protocol is proposed according to the ISO D.I.S. for the Data Communications Protocol for Providing the Connectionless-mode Network Service. (See the attachment on Conformance and the Provision of Functions for Conformance, extracted from the D.I.S.)

It is proposed that only type 1 functions be implemented for a timely and useful service. (See the attachment on function types, extracted from the D.I.S.)

SUBNETWORK USER DATA

If source and destination end systems are on the same 802.3 or 802.4 subnetwork, then the same size restrictions as applied to the 1984 NCC demo should prevail. (In this case, the Inactive Network Layer Protocol Subset is being employed.) If the full protocol is being used to concatenate subnetworks then the maximum user data size of 64+K should be permitted, corresponding to the IP specification. Practically, no statement need be made about minimum user data sizes, since it is expected that the transport class 4 header and transport user data will be of non-trivial length.

PDU LIFETIME

For purposes of concatenating LANs, typically an intermediate system might subtract one from the lifetime field. This represents 500 milliseconds for transit between intermediate systems and processing by one intermediate system. Thus, it is recommended that source end systems insert an initial value corresponding to the width of the catnet plus two. This should safely allow the destination end system to process the received PDU.

ROUTING

For purposes of a useful exercise it is recommended that fixed routing tables be used. It is suggested that implementations provide operator control to manipulate routing tables, since the exact topology for any demonstration may be subject to last minute modifications.

SEGMENTATION AND REASSEMBLY

Destination end systems must be able to reassemble. Source end systems must either fit the transport PDU plus IP header into a single subnetwork service data unit or be able to segment. Reassembly by intermediate systems is not recommended, however they must be able to segment. Setting of the segmentation permitted flag should be at the discretion of the source end system, however it is suggested that the flag be set to allow segmenting.

ERROR REPORTING

NOTE: THIS RECOMMENDATION AMENDS AND STRENGTHENS THE BASIC PROPOSAL SECTION REGARDING CONFORMANCE. If the error report flag is on and a PDU of type 3 is discarded because it is not supported, then an error report should be returned even though this is not strictly required for conformance. This is recommended for purposes of debugging. It is further suggested, for debugging, that implementations log all error reports. The error report data field should contain the entire errant PDU, truncated only if necessary.

IDENTIFICATION

The protocol id. of 1000 0001 is used in the catnet situation with a version number of 0000 0001. For the single subnetwork case, the protocol id. is 0000 0000.

CHECKSUM

Although checksum of the header is optional it is recommended that the checksum be used, since emphasis should be on a useful IP rather than simply on a demonstration. It may also be useful for debugging.

ADDRESSING

Binary representation should be used, since the IP header is binary based. (See attached table from the ISO d.p. on addressing.)

TOPOLOGIES

Various topologies are considered below. LAN refers either to 802.3 or 802.4, interchangeably. WAN refers to PDN X.25, 1984. Pri refers to any private (vendor propritary) subnetwork.

Cases considered:

a) LAN
b) LAN-LAN
c) LAN-WAN-LAN
d) WAN-LAN
e) Pri-LAN
f) LAN-Pri-LAN
g) Pri-WAN-LAN

LAN Addressing

The Inactive Network Layer Protocol subset is used with identifier of 0000 0000.

LAN-LAN Addressing

Source and destination addresses in the IP header are of identical construction. The first octet is local binary, hexidecimal 49. The second octet is the subnetwork identifier. Assign 802.3 LANs beginning with 0000 0001 and 802.4 LANs beginning with 1000 0001. Octets three through eight comprise the 48 bit station address. Octet nine is the 8 bit network service access point.

LAN-WAN-LAN Addressing

Source and destination addresses are of identical construction. The first octet is hexidecimal 49, signifying local binary format. The subnetwork identifer octet, interpreted by intermediate systems via a routing table, yields an X.121 DTE address. The subnetwork identifier octet is followed by a station address and NSAP as described in the LAN-LAN case.

WAN-LAN Addressing

Considering the source address on the LAN and the destination address on the WAN, the source address has the format described in the LAN-WAN-LAN case. The first octet of the destination address is hexidecimal 25 (X.121 DTE, binary). Octets two through eight encode the 14 decimal digit X.121 DTE address in BCD. Octet nine is the NSAP.

Where the LAN is the destination and the WAN the source, the above format is reversed.

Pri-LAN Addressing

If source end system is on the private subnetwork and destination end system on the LAN, then the source address is: hexidecimal 49, subnetwork identifier of the LAN, station address of the gateway, followed by the private address of the source end system on the private subnetwork. (The private address on the private subnetwork is interpretable only by the private subnetwork.) Note that the three items preceding the private address specify the intermediate system coupling the private network and LAN. This constitutes routing, not addressing. This may be useful for a demonstration but is not recommended as a general solution. The destination address is: hexidecimal 49, subnetwork identifier of the LAN, station address on the LAN, followed by the one octet NSAP. For PDUs traveling from LAN to private subnetwork the formats are interchanged.

LAN-Pri-LAN Addressing

This structure is the same as the LAN-WAN-LAN addressing.

Pri-WAN-LAN Addressing

Where the source end system is on the private subnetwork, the source address is hexidecimal 25, seven octets of X.121 address of the gateway between the private and PDN subnetworks, followed by the private subnetwork end system address. Here again, the X'25' and X.121 address specifies particular routing information. It may or may not be desirable to do this for a demonstration, but it is not advised as a general solution. The destination address is hexidecimal 49, the LAN identifier, the station address, and the NSAP. PDU flow in the reverse direction formats the source and destination addresses in the opposite format.

ROUTING TABLE LOGIC

X.121-DTE

If the format is hexidecimal 25 then the X.121 address is either this system's or some other system's. If it is this system's, then interpret the information after the X.121 address. If it is some other system's, then send to the PDN.

Local

If the format is hexidecimal 49 then check the subnetwork address. If it is some other subnetwork's then look up in the routing table. If it is this subnetwork's then broadcast it on this LAN.

Attachment

9 CONFORMANCE

For conformance to this International Standard, the ability to originate, manipulate, and receive PDUs in accordance with the full protocol (as opposed to the "non-segmenting" or "Inactive Network Layer Protocol" subsets) is required.

Additionally, the provision of the optional functions described in Section 6.17 and enumerated in Table 9-1 must meet the requirements described therein.

Additionally, conformance to the Standard requires adherence to the formal description of Section 8 and to the structure and encoding of PDUs of Section 7.

If and only if the above requirements are met is there conformance to this International Standard.

9.1 PROVISION OF FUNCTIONS FOR CONFORMANCE

The following table categorizes the functions in Section 6 with respect to the type of system providing the function:

Function	Send	Forward	Receive
PDU Composition	M	-	-
PDU Decomposition	M	-	M
Header Format Analysis	-	M	M
PDU Lifetime Control	-	M	I
Route PDU	-	M	-
Forward PDU	M	M	-
Segment PDU	M	(note 1)	-
Reassemble PDU	-	I	M
Discard PDU	-	M	M
Error Reporting	-	M	M
PDU Header Error Detection	M	M	M
Padding	(note 2)	(note 2)	(note 2)
Security	-	(note 3)	(note 3)
Complete Source Routing	-	(note 3)	-
Partial Source Routing	-	(note 4)	-
Record Route	-	(note 4)	-
QoS Maintenance	-	(note 4)	-

Table 9-1. Categorization of Functions.

Table 6-1 shows how the functions are divided into these three categories:

Function	Type
PDU Composition	1
PDU Decomposition	1
Header Format Analysis	1
PDU Lifetime Control	1
Route PDU	1
Forward PDU	1
Segment PDU	1
Reassemble PDU	1
Discard PDU	1
Error Reporting	1 (note 1)
PDU Header Error Detection	1 (note 1)
Padding	1 (notes 1 & 2)
Security	2
Complete Source Routing	2
Partial Source Routing	3
Priority	3
Record Route	3
Quality of Service Maintenance	3

Table 6-1. Categorization of Protocol Functions

Notes:

1) While the Padding, Error Reporting, and Header Error Detection functions must be provided, they are provided only when selected by the sending Network Service user.

2) The correct treatment of the Padding function involves no processing. Therefore, this could equally be described as a Type 3 function.

3) The rationale for the inclusion of type 3 functions is that in the case of some functions it is more important to forward the PDUs between intermediate systems or deliver them to an end-system than it is to support the functions. Type 3 functions should be used in those cases where they are of an advisory nature and should not be the cause of the discarding of a PDU when not supported.

TABLE 8-1: AFI ALLOCATIONS

00-09	Reserved - will not be allocated
10-19	Reserved for future allocation by joint agreement of ISO and CCITT
20-51	Allocated and assigned to the IDI formats defined in clause 8.2.1.2
52-59	Reserved for future allocation by joint agreement of ISO and CCITT
60-69	Allocated for assignment to new IDI formats by ISO
70-79	Allocated for assignment to new IDI formats by CCITT
80-99	Reserved for future allocation by joint agreement of ISO and CCITT

8.2.1.2 FORMAT AND ALLOCATION OF THE IDI

A specific combination of IDI format and DSP syntax is associated with each allocated AFI value, as summarized in Table 8-2:

TABLE 8-2: AFI Values

DSP syntax IDI format	Decimal	Binary	Character (ISO 646)	National Character
X.121-DCC	20	21	22	23
X.121-DTE	24	25	26	27
F.69	28	29	30	31
E.163	32	33	34	35
E.164	36	37	38	39
ISO 6523	40	41	42	43
ISO 6523-ICD	44	45	46	47
Local	48	49	50	51

NOTE

The need to describe DSP syntaxes involving characters or national characters for these IDI formats has not been established and is for further study

Laurie Bride 9/5/84
wed.

Att. #19

CONNECTIONLESS IP PROPOSAL

- ALL TYPE 1 FUNCTIONS
  - COMPOSITION
  - DECOMPOSITION
  - HEADER ANALYSIS
  - LIFE TIME BOUNDING
  - ROUTING
  - FORWARDING
  - SEGMENTING
  - REASSEMBLING
  - DISCARD
  - ERROR REPORTING
  - ERROR DETECTION
  - PADDING

- OPTIONAL TYPE 2, 3 FUNCTIONS
  - SECURITY
  - COMPLETE SOURCE ROUTING
  - PARTIAL SOURCE ROUTING
  - PRIORITY
  - ROUTE RECORDING
  - QOS

ADDITIONAL RECOMMENDATIONS

- SUBNETWORK USER DATA size
- PDU LIFETIME
- ROUTING
- SEGMENTATION & REASSEMBLY
- ERROR REPORTING
- PROTOCOL AND VERSION ID as in ISO document
- CHECKSUM

ADDRESSING

1 - SINGLE LAN

- INACTIVE NETWORK LAYER PROTOCOL (NO IP HEADER ADDRESSES)
- 48 BIT STATION ADDRESS
- 8 BIT NSAP

2 - LAN-LAN

- X' 49' FORMAT TYPE
- 8 BIT SUBNETWORK IDENTIFIER
- 48 BIT STATION ADDRESS
- 8 BIT NSAP

3 - LAN-WAN-LAN

- X' 49' FORMAT TYPE
- 8 BIT SUBNET ID (TABLE POINTER TO X.121 DTE ADDRESS)
- 48 BIT STATION ADDRESS
- 8 BIT NSAP

ADDRESSING CONT.

4 - WAN-LAN

- LAN-TO-WAN

  SOURCE:
  - X' 49'
  - 8 BIT SUBNET ID
  - 48 BIT STATION ADDRESS
  - 8 BIT NSAP

  DESTINATION:
  - X' 25'
  - 7 OCTET X.121 DTE ADDRESS
  - 8 BIT NSAP

5 - PRIVATE-LAN

- PRI-TO-LAN

  SOURCE:
  - X' 49'
  - 8 BIT SUBNET ID
  - 48 BIT STATION ADDRESS (OF INTERMEDIATE SYSTEM)
  - PRIVATE ADDRESS

  NOTE: THE FIRST THREE ITEMS CONSTITUTE ROUTING AND YOU MAY NOT WANT TO DO THAT.

  DESTINATION:
  - X' 49'
  - 8 BIT SUBNET ID
  - 48 BIT STATION ADDRESS
  - 8 BIT NSAP

6 - LAN-PRI-LAN

- SAME AS LAN-WAN-LAN

7 - PRI-WAN-LAN

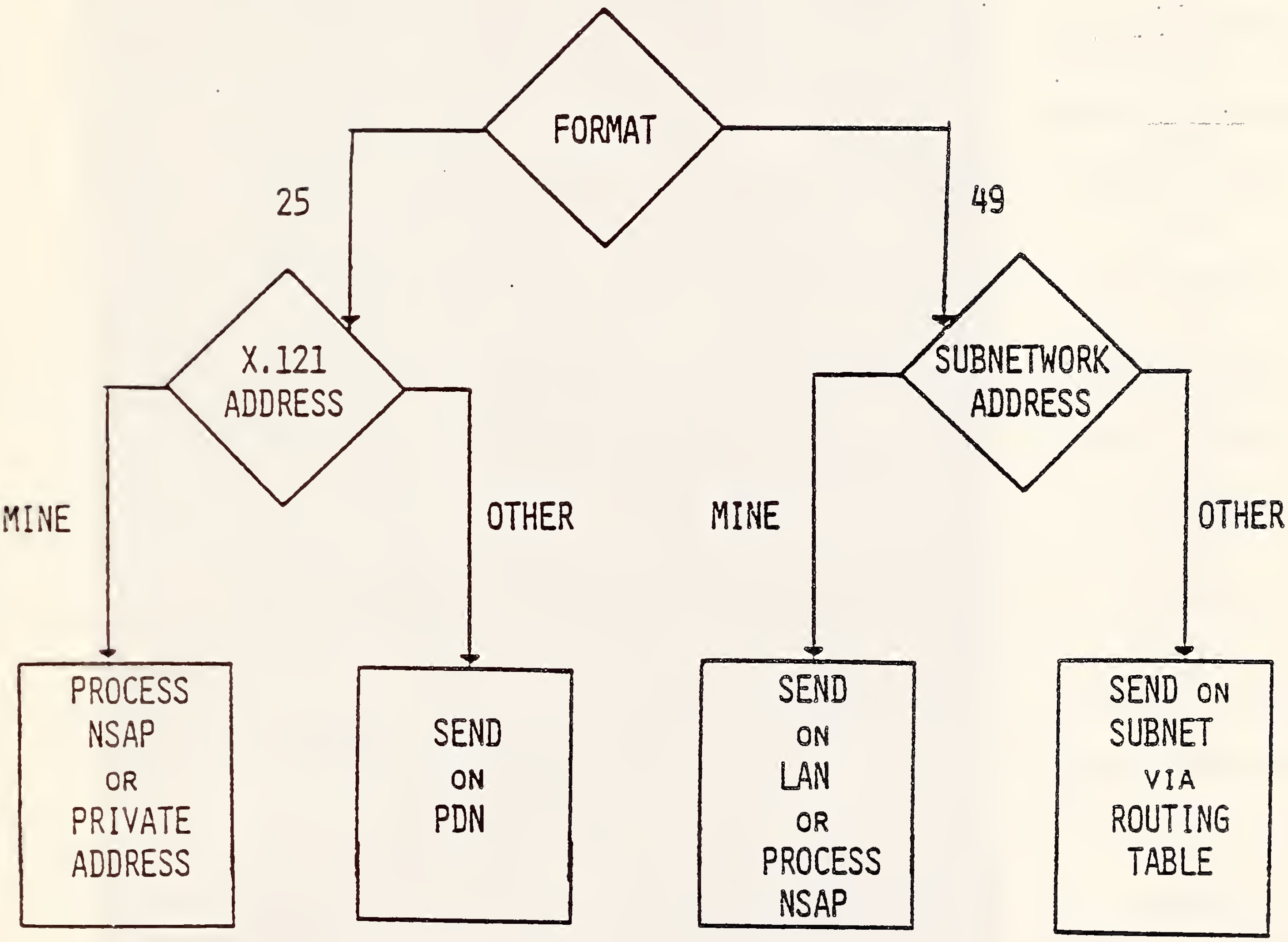
- PRI-TO-LAN

  SOURCE: - X' 25'
  - 7 OCTETS X.121 DTE ADDRESS
  - PRIVATE ADDRESS

NOTE: THE FIRST TWO ITEMS CONSTITUTES ROUTING AND YOU MAY NOT WANT TO DO THAT.

DESTINATION:
- X' 49'
- 48 BIT STATION ADDRESS
- 8 BIT NSAP

ROUTING


FORMAT
25
49
X.121
ADDRESS
SUBNETWORK
ADDRESS
MINE
OTHER
MINE
OTHER
PROCESS
NSAP
OR
PRIVATE
ADDRESS
SEND
ON
PDN
SEND
ON
LAN
OR
PROCESS
NSAP
SEND ON
SUBNET
VIA
ROUTING
TABLE

AH. #20

MATRIX OF THE CLNS PROTOCOL IMPLEMENTATION FEATURES

SOLICITED PROVISIONS BY GENERAL MOTORS

	IMPLEMENT??		IMPLEMENT THEN SUPRESS
PDU COMPOSITION (type 1)	YES		
PDU DECOMPOSITION (type 1)	YES		
HDR FORMAT ANALY (type 1)	YES, VERSION 1		
PDU LIFETIME (type 1)	YES		
ROUTE PDU (type 1)	YES		
FORWARD PDU (type 1)	YES		
SEGMENTATION (type 1)	YES		
REASSEMBLY (type 1)	YES		
DISCARD (type 1)	YES		
ERROR REPORT (type 1)	YES		
HDR ERROR DETECT (type 1)	YES, (00)		YES, other than (00)
SECURITY (type 2)		NO	
COMP. SOURCE RTING (type 2)		NO	YES
PART SOURCE RTING (type 3)		NO	
PRIORITY (type 3)		NO	
RECORD OF ROUTE (type 3)		NO	
QOS (type 3)	YES		
PADDING (type 3)	YES		

KS GM 9/5/89

Att. #21

IP

CONFORMANCE AND INACTIVE SUBSET

AGREE 20

DISAGREE 2

M.C. 9/4/84

Att. #22

REQUIRED PROTOCOL FUNCTIONS

- PDU COMPOSITION
- PDU DECOMPOSITION
- HEADER ANALYSIS
- LIFE TIME BOUNDING (1)
- ROUTING (2)
- FORWARDING
- SEGMENTING
- REASSEMBLING
- DISCARDING (3)
- ERROR REPORTING (3)
- ERROR DETECTION
- PADDING

(1) FOR DEMO: 3 x NO OF INTERMEDIATE SYSTEMS + END SYSTEM 27/0

(2) FOR DEMO: FOR INTERMEDIATE SYS. FIXED TABLES; UPDATED BY OPERATOR. POSSIBLE ENHANCEMENT BY GATEWAY PROVIDERS. 27/0

(3) FOR TESTING LOG DISCARDED PDUS, ERROR PDUS BY SOURCE GENERATING PDU, UNSUPPORTED TYPE 3 OPTIONS. 27/0

M.C. 9/4/84

OPTIONAL PROTOCOL FUNCTION

- SECURITY (1)
- SOURCE ROUTING (2)
- ROUTE RECORDING (3)
- QUALITY OF SERVICE (4)

(1) FOR DEMO: SECURITY NOT USED; (ILL 27/0
DEFINED AT THIS TIME)

FOR DEMO!
(2) a) COMPLETE SOURCE ROUTING NOT WANTED
BY END SYSTEMS, ∴ NOT IN INTERMEDIATE
SYSTEMS.
b. PARTIAL SOURCE ROUTE: DETAILED 27/0
PROPOSALS SOLICITED.

(3) FOR DEMO: SUPPORT 27/0
DESTINATION SHOULD LOG IF
POSSIBLE.

(4) FOLLOW SPECIFICATION 27/0

(5) CHECKSUMS: FOR TESTING, TURN ON;
IN OPERATION, A LOCAL ISSUE. 27/0

R.C. 9/4/84

#24-1

A Modest (Addressing) Proposal

Goals:

1) Comply with letter (and spirit - if possible of ISO Addressing DP

2) Make end-system Routing "easy"

3) Avoid explicit hierarchical dddresses where not needed to do Routing reasonably

4) Be easy to inplenent + administer

5) Be usable in products

6) Be extensible to incorporate ISO Routing protocols as they emerge

7) ACCOMODATE LAN CLUSTER

8) INITIALLY, SUPPORT ONLY ONE ADDRESSING FORMAT

D.O./M.G. 9/6/84

Minimal Routing Functions

- Gateways have static topology map of routes to all Gateways (i.e. no hierarchical Routing) with alternate paths, if desired

- End systems know two pieces of information a priori:

  1) Their own (complete) address
  2) The address of at least 1 Gateway that is "1 hop" away

- End systems Route as follows:

```
If Dest-addr (IDI) = MyAddr (IDI) then
    If NSDU-size < LAN max packet size
       and no options                         Then
        • send using "Inactive header"
    Else
        • send using "Full Header"
Else Send using Full Header to Gateway
```

DO 9/6/84

Att. #24-3

Specific Proposal

- All nodes on a LAN have at least one address of the form:

  AFI = X.25-DTE / Binary DSP
  IDI = X.121 Address
  DSP = IEEE 56 bit LAN address
  + 1 octet demux

- Nodes on a PDN may use above form but if LAN address is not assigned, then

  DSP = Null

- Nodes on private nets, but not on a LAN may be "faked up" to be either of the above

Assignment of Addresses

- PDN nodes are assigned X.121 address by carrier

- LAN nodes are assigned the X.121 part according to TOPOLOGY

DO 9/4/84

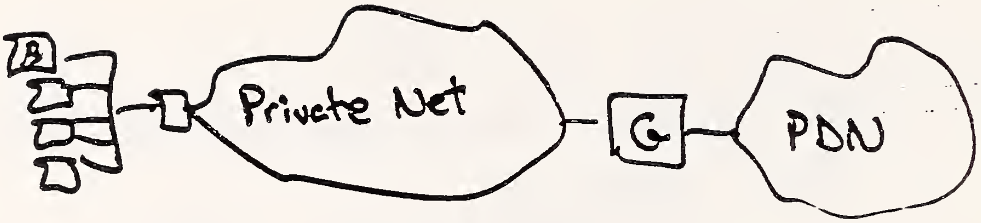
One Plosses

- End Systems learn IDI part of address from Gateway
  - Not possible on PDN
  - Easy on LAN via Gateway multicast
- End Systems learn of Gateways automatically
  - Not possible on PDN
  - Easy on LAN via Gateway multicast

{See R. Perlman - Routing on LAN's - proceedings of 8th Data Comm Symp.)

- Better path selection
  - Redirect (Needs PDU's)
  - Reverse path forwarding (local info)

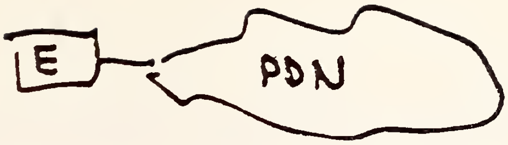
Case 6



Same as 3, 4, or 5 depending on G's and PDN's

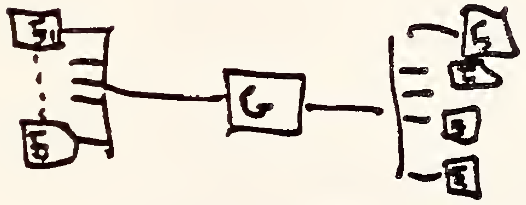
Att. #24-6

Case I



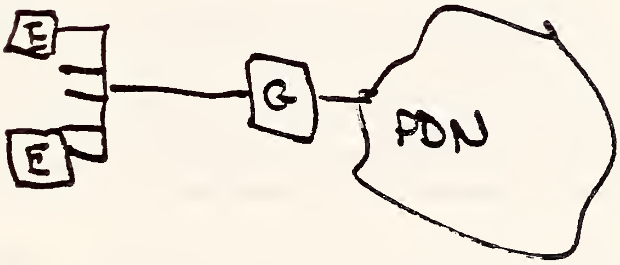
E ← X.121 of PDN

Case 2



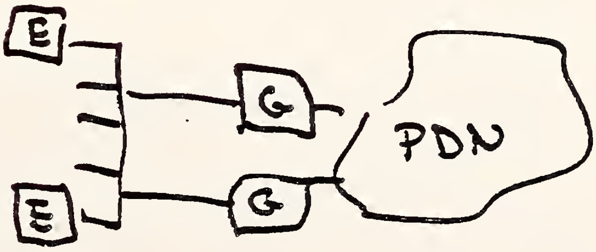
E ← X.121 local escape

Case 3



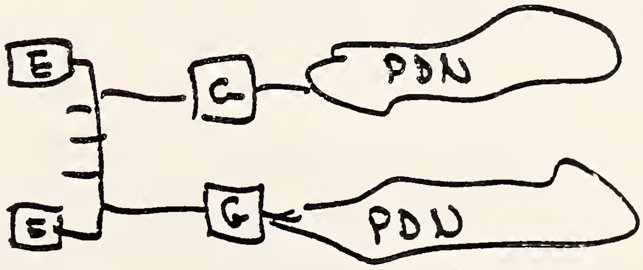
E ← X.121 of G

Case 4



E ← X.121 of other G

Case 5



E ← Both X.121 addresses (multi-homed)

DO 9/6/84

Attachment #25

Implementation of OSI protocols in the ESPRIT Information Exchange System

Authors :

R. BITTLESTON (2), R. CADWALLADER (3),
A. DIEDIW (2), M. ELIE (1), J. LOVELUCK (1),
F. LUNG (4), S. MIESS (6), S. POZZANA (5)

Summary

The Esprit Information Exchange System (EIES) recently renamed ROSE for "Research Open Systems for Europe" is an infrastructure to support collaborative R and D projects in information technology within the European Strategic Program in Information Technology (ESPRIT) launched in 1984 by the European Economic Commission. The work is being carried out by a consortium of 6 industrial partners.

The EIES will provide the electronic mail, teleconferencing, document handling and transfer, file transfer, remote login services which are necessary for cooperative R and D work.

The project aims at a maximum connectivity of potential users through the use of Open Systems Interconnection ISO services and protocols, and starts with an implementation under the UNIX* operating system.

After a short description of the objectives of the project, the paper presents the work presently carried out and planned for the following years. It describes the architecture chosen for the interconnection of local area and wide area networks, the addressing scheme used and some considerations is given to the management aspects of the network. Finally the main choices made for the implementation under UNIX are outlined.

*UNIX is a Trademark of Bell Laboratories.

August 17, 1984

Roy Cadwallader 9/5/84

Keywords:

architecture, Esprit, EIES, implementation, network, standard, UNIX, message handling system, file transfer.

(1) BULL

68, route de Versailles - BP 3
78430 Louveciennes (FRANCE)

(2) GEC

GEC Hirst Research Centre
East lane
Wembley
Middlesex HA9 7PP (ENGLAND)

(3) ICL

West avenue
Kidsgrove
Stoke-on-trent (ENGLAND)

(4) INRIA

Domaine de Voluceau
Rocquencourt - BP 105
78150 Le chesnay (FRANCE)

(5) OLIVETTI

Via Jervis, 77
10015 Ivrea (ITALIE)

(6) SIEMENS

SIEMENS AG ZTI SOF 43
Otto Hahn Ring 6
D-8000 Muenchen 83 (RFA)

August 17, 1984

Keywords:

architecture, Esprit, EIES, implementation, network, standard, UNIX, message handling system, file transfer.

(1) BULL

68, route de Versailles - BP 3
78430 Louveciennes (FRANCE)

(2) GEC

GEC Hirst Research Centre
East lane
Wembley
Middlesex HA9 7PP (ENGLAND)

(3) ICL

West avenue
Kidsgrove
Stoke-on-trent (ENGLAND)

(4) INRIA

Domaine de Voluceau
Rocquencourt - BP 105
78150 Le chesnay (FRANCE)

(5) OLIVETTI

Via Jervis, 77
10015 Ivrea (ITALIE)

(6) SIEMENS

SIEMENS AG ZTI SOF 43
Otto Hahn Ring 6
D-8000 Muenchen 83 (RFA)

August 17, 1984

- In a first step, UNIX is used as a starter operating system on different machines from each contractor, in order to provide a basis for initial portable developments. The SOL operating system is used as a possible alternative to UNIX. SOL is a European UNIX -like system developed by INRIA in France, which offers a UNIX compatible software environment [GIE 83].

1. Introduction

In this section we summarise the objectives of the EIES project, the implementation strategy and the standards to be implemented. Later sections discuss in more detail the EIES architectural and implementation choices. Full details are contained in the EIES Technical Specifications [EIE 84].

1.1. Functional objectives of EIES

1.1.1. Overall aim of EIES

EIES is to provide services to the ESPRIT programme, and will be potentially available to support collaborative R & D projects of other kinds throughout the European Community.

It addresses the requirements and recommendations of the IES Panel Report (IES 82). These can be summarised as providing the following set of services between heterogeneous machines and terminals located within the member states with a maximum use of public networks and services:

message passing
document transfer
file transfer
remote login and job execution
software transfer

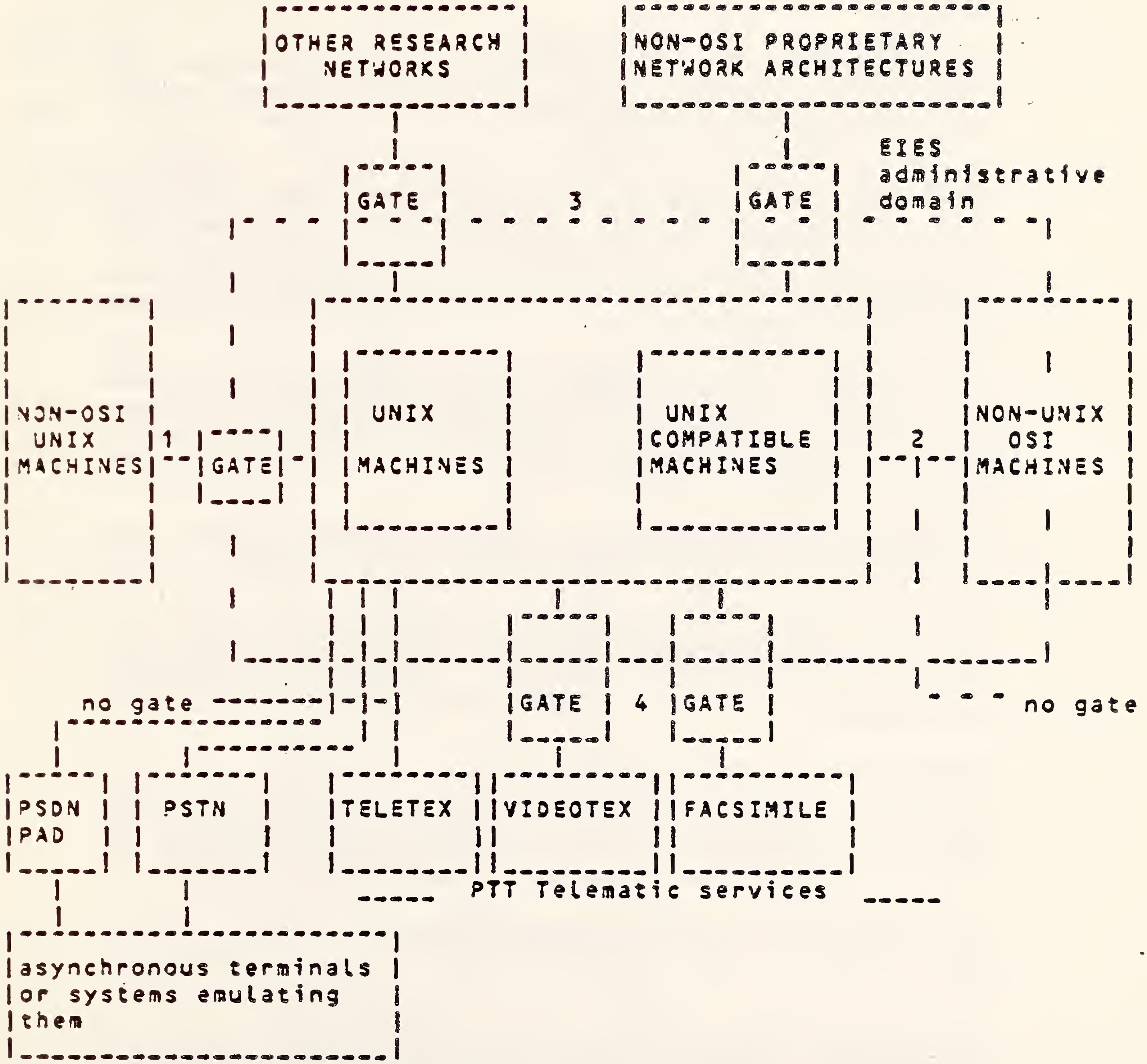
These requirements have been recently enhanced by the IES workplan published by the Commission (IES 84).

The project is performed by a consortium of six partners Bull, GEC, ICL, INRIA, Olivetti and Siemens with subcontracts to University of York, University College Dublin, and the Stichting Mathematical Centrum of Amsterdam, as a pilot project of the ESPRIT programme.

1.1.2. Project overview

The project is essentially a software development and integration project. It implements an operational network providing the services listed in 1.1.1 between the contractors. After a phase of timing and debugging it is intended to be made available to all ESPRIT programme contractors and subcontractors. Its target connectivity is shown in figure 1, and can be described as follows:

Figure 1 - LOGICAL CONNECTIVITY OF EIES



August 17, 1984

Initial connectivity is provided between UNIX or UNIX like systems by UNIX specific facilities (cu/uucp) used over X25 public data networks. This safeguards existing UNIX applications for communication (mail, news ...)

- In parallel, an OSI session service interface is developed, relying on ISO session and transport protocols and on X25, in order to give to UNIX the visibility of an ISO Open System.

- This allows for the implementation over this interface of CCITT/ISO defined applications such as file transfer, message handling system ... and enables communication with non-UNIX systems implementing OSI protocols, without a need for a gateway function, as shown on figure 1, point 2.

- A store and forward service to non EIES UNIX systems using PSTN, is provided through the Stichting Mathematical Centre of Amsterdam, as shown on figure 1, point 1, which already provides this service for the European UNIX Users Group (EUNET).

- Asynchronous terminals, or systems emulating them, can connect through PSTN or the PAD service of PSDN.

- PTT telematic services teletex, videotex and facsimile will be directly connected up to the point where they support OSI protocols. Specific gateway developments will take into account differences, as shown on figure 1, point 4.

- Connection to proprietary network architectures not offering an OSI visibility will be possible through gateways, depending upon the needs of Esprit users, as well as connection to other non-OSI networks of use in R and D projects, as shown on figure 1, point 3.

- New applications will be made available to IES users when needed; conferencing software, distribution service, ...

- The work of other ESPRIT projects dealing with software engineering, such as PCTE, or other subjects, will be integrated into IES.

- New communication media (satellite, broadband communication, ISDN, ...) will be used.

- IES will include tools for administration and maintenance, such as a Control Center, and a distributed name server.

Figure 2 a. EVOLUTION OF THE SERVICES PROVIDED BY EIES (year 0)

```
PREPARATORY                     YEAR 0
-------------><------------------------------------------------>

UNIX ONLY                       UNIX OSI
-------------><------------------------------------------------>

PSTN            X.25            ETHERNET
--------------- --------------- ---------------
|  MAIL, NEWS | |  MAIL, NEWS | |  MAIL, NEWS |
|FILE TRANSFER| |FILE TRANSFER| |FILE TRANSFER|
|-------------| |-------------| |-------------|
|             | |             | |             | ----------------
|             | |             | |             | |     ISO      |
|             | |    UUCP     | |    UUCP     | |SESSION (BCS) |
|             | |=============| |=============| |==============|
|    UUCP     | |     ISO     | |     ISO     | | TR   |  TR   |
|     CU      | |  TRANSPORT  | |  TRANSPORT  | |      |       |
|             | |  CLASS 2/3  | |   CLASS 4   | | CL2/3|  CL 4 |
|             | |-------------| |             | |------|       |
|             | |             | |-------------| |      |-------|
|             | |             | |             | |      |       |
|             | |             | |             | | X25  | LAN   |
|-------------| |     X25     | |     LAN     | |      |CSMA/CD|
|    V21      | |             | |   CSMA/CD   | |      |       |
|-------------| |-------------| |-------------| |------|-------|
```

Figure 2 b. EVOLUTION OF THE SERVICES PROVIDED BY EIES IN YEAR 1.

```
-----------------------------------
|                        |        |
|           M H S        |  FTAM  |
|                        |        |
|------------------------|--------|
|    ISO                          |
|   SESSION  (BAS + BSS)          |
|=================================|
|    TR    |          |           |
|          |    TR    | GATEWAYS  |
| CL 2/3   |          |-----------|
|----------|   CL4    |           |
|          |----------|           |
|          |          |           |
|   X25    |   LAN    |           |
|          | CSMA/CD  |           |
|----------|----------|-----------|
```

1.2. Phased implementation

The implementation plan for ISO standards allows a progressive migration of users and applications to new services.

In year 0 of the project (1984) the users can use existing UNIX communication tools uucp, cu and applications, such as mail, on asynchronous or PAD to PAD connection, to communicate among themselves. At the same time (see figure 2a) the project develops uucp on an ISO transport, using X25 WAN or CSMA-CD LAN, LAN-WAN interconnection, and a basic ISO session service and protocol just used by test programs.

In year 1 of the project (1985), year 0 developments are made available to the users who can then interconnect their UNIX systems through X25 public data networks or to a LAN, and use the same applications they were using over asynchronous lines. During this period (see figure 2b) the project develops new enhanced applications in accordance with the international standardisation, such as File Transfer and a Message Handling Systems. This will allow the distribution of these applications over a set of UNIX and non-UNIX computer systems.

1.3. List of standards applicable to year 0 and 1 of EIES

1.3.1. General

This section gives a list of all the standard protocols which are used in the EIES project during year 0 and year 1 of the project.

Presently the set of standards used for EIES is a subset of the set of OSI standards which a group of 12 European manufacturers proposes to support in their products. They will be the basis of the "IES standard conventions" to be published by the Commission.

1.3.2. X.25

The X.25 protocols used in the EIES project are the protocols defined for physical link and network layer by CCITT in its X 25 recommendation [CCI 80]. The only significant definition for the EIES project is the Network Level standard: for the Lower Layers no definition has been provided ; it is possible to use the data link LAP and/or LAPB and the physical interface X.21 and/or X.21 bis, provided that the system is able to connect to all the national X.25 packet switching data networks that are involved in the project.

The addressing structure used through the Network Layer is the one defined by CCITT in its recommendation X.121. [CCI 80]

1.3.3. Terminal access

Asynchronous terminal access through X25 networks conforms to X3, X28, X29 standards for Packet Assembly Disassembly (PAD).

1.3.4. LAN protocols and LAN-WAN gateway

The LAN protocols used in the EIES project are the protocols described in ECMA 80,81 and 82. Reliable transport over the LAN is provided by using ISO transport class 4 and LAN/WAN interconnection is achieved according to ECMA TR 21 [ECM 21]

1.3.5. Internet

The internet standard defines the protocol to be used in the network layer 3c to interconnect several Local Area Networks. The current stable reference is ECMA92, the corresponding ISO standard being considered to be insufficiently stable.

1.3.6. Transport

The Transport protocol used in the EIES project is the one defined in the ISO DIS 8073 (DIS 8072 for the Transport Service) ; [ISO 72 and 73]. In particular, class 2 and 3 is implemented over the X.25 environment, and class 4 is implemented over the CSMA/CD environment. Special care is taken through implementation rules to provide for a maximum efficiency of the protocol on the LAN while preserving full conformance.

ISO Transport Class 0 will also be implemented in year 1, in order to suoport Teletex.

1.3.7. Session

The Session protocol used in the EIES project is the one defined in ISO DIS 8327 (DIS 8326 for the session service); the Basic Combined Subset (BCS) as defined in the ISO DIS 8327 is implemented [ISO 26 and 27] during year 0.

The BCS includes the following Functional Units:

a) Kernel functional unit ;

b) half-duplex functional unit ;

c) full-duplex functional unit.

The full session protocol and services will be implemented during year 1.

1.3.8. Message handling

1.3.9.

Standards from the X4XX set of CCITT recommendations, which are now stable, will be implemented, namely

X409 Message Handling systems: presentation transfer syntax and notation

X411 Message Handling systems: message transfer layer

X410 Message Handling systems: Remote Operations and Reliable Transfer Server

X420 Message Handling systems: Interpersonal Messaging User Agent Layer.

1.3.10. File Transfer

The File Transfer and Manipulation ISO draft standard DP 8571 will be used for the file transfer implementation under UNIX.

1.3.11. Administration

While administration is restrained in the first year to local statistical information gathering, care is taken that the information contents is in line with current ISO TC97 SC16 WG4 work.

2. EIES Architectural Choices.

The EIES Network Architecture is strictly derived from the ISO-OSI Reference Model ; nevertheless, it was necessary to make some choices in particular areas such as LANs and the Global Network Layer, which are described in this section.

The rationale for these choices has been derived mainly from ECMA TC24 documents; in particular, most of the definitions used throughout this section have been taken from the ECMA TR 21 [ECM 21].

2.1. LANs in the EIES project.

2.1.1. EIES LANs and the OSI Reference Model.

Two alternative solutions exist for LAN integration in an OSI environment :

1) A LAN exists as the information transfer means between the various elements comprising an End System or a number of End Systems.

2) The LAN exists as a subnetwork of the OSI Global network for the purposes of interconnecting complete End Systems.

EIES conforms to the recommendation of ECMA TR/14 [ECM 14] which defines "protocol sets", to avoid incompatibility where internetworkingis required between equipment from different suppliers attached to the same CSMA/CD baseband LAN subnetwork.

The project decided to use the ISO 8072/73 Class 4 protocol on LANs, and to adopt a back-to-back transport class 2-3/class 4 gateway for LAN-WAN interconnection; this corresponds to Case 1, retaining the LAN as an externally invisible component of an End System.

Following this approach, the gateway between the End System LAN and the Global Network incorporates a Transport Layer Relay masking all protocols at and below the Transport Layer in the LAN.

The Transport Layer Relay is not externally visible; it may be regarded, in effect, as an element of a distributed form of End System Transport entity externally visible as any other End System Transport entity.

The protocols operating on the LAN above the Transport Layer become externally visible and must conform to OSI Standards.

2.1.2. End System Architecture.

The combination of regarding the LANs as a single End System and of using the protocols defined in the document [ECM 14] has as a consequence that the EIES view of a LAN in an OSI network conforms with that described in [ECM 21];

some concepts taken from this document are recalled in the following.

In such an environment, the End System is hierarchically divided into two physical components; the division at the layers 5/4 boundary offers a convenient separation of functions, and it is attractive to consider an interface mechanism based on a LAN using layer 4-1 protocols.

The most important component of the Distributed End System is the Gateway (called Distributed System Interworking-Unit (DSI) in the following).

The DSI conforms to the applicable provisions of ECMA TR/20 [ECM 20]. A DSI, together with its associated distributed End System components, is addressed like a normal End System; the relay function in the DSI determines the mapping of the externally known Transport Addresses onto the addressing scheme used internally on the LAN.

In the case of a connection request made from a system on the LAN to an external system, the DSI must be given sufficient information for it to determine the Transport Address of that external system.

2.2. Global EIES Network Layer.

2.2.1. The Network Layer structure.

The Network Layer has the goal of dealing with all the problems related to the different sub-networks which belong to the overall network, providing a homogeneous service to the Transport Layer. The Network Layer is currently divided into three sublayers 3a, 3b and 3c.

An X.25 subnetwork includes a layer 3a with lower layers; it provides more services than those needed to support the standard global Network Service; all these services will not be made available to the global Network Service users in order not to complicate the Transport Protocol, and in order to avoid the need for enhancement of any possible further subnetwork that does not provide these services.

2.2.2. The EIES Network Layer.

The EIES project will use Public Data Networks (PDNs) such as: ITAPAC, EURONET, TRANSPAC, DATEX-P, PSS, etc.; these follow the CCITT X.25 Recommendation and are to be interconnected via relay systems, according to the interface specified in the CCITT X.75 Recommendation (the interface between two PDNs specified in X.75 is quite similar to that in X.25).

Using this addressing facility, the EIES network may see all the different PDNs as a single subnetwork.

As previously defined, the EIES LANs are not seen as subnetworks (or better: are not seen at all) and so it is possible to say that the EIES network has only one subnetwork; therefore, the Global EIES Network Layer consists of just a single 3a layer.

3. Addressing in the EIES.

The EIES network uses only standard protocols and conforms to them exactly; when implementing a real network it is, however, necessary to define a number of details, taking into account the final users requirements, the physical configuration, the kind of service to be provided and so on; in doing this, great imoortance must be given to the definition of the addressing structure.

In this section the addressing structure of the EIES network is described; in particular, the elements (Names and Addresses) which are going to be used through the different layers are specified, and the transformations which will apply to them.

For the first year, the project is concerned with the protocols up to the Session only, therefore only the addressing problems related to the first 5 layers have been taken into consideration; in the following, the users of the Session are generally referred to as "Applications" or "Users".

As a first choice, it has been decided to support only a single type of address at the User interface (the Session Layer interface) to simplify the address management in the Communication Layers.

It has to be noted that a single address type does not impose any real restriction on the Application Layer; the chosen type is general enough to support any reasonable addressing policy, and it is possible to implement Application Level Servers which translate the address type a User wants to use into the EIES one; such translations will not be provided during the first year.

3.1. The address type at the User Interface..

Several address types may be taken into consideration when defining a network implementation :

a) Flat Address: an address with no internal structure which is constructed without reference to any lower layer address.

b) Hierarchic Address: an address which is constructed relative to a lower layer address; it consists of the lower layer address plus a selector component as suffix (it specifies a Service Access Point relative to a lower layer Service Access Point).

c) Partitioned Address: An address built from a set of nested addressing domains which is constructed without reference to any lower layer address.

The EIES project has adopted the hierarchic address type; this means that, at the Session Layer Interface, the three address elements must be specified which are necessary to identify the remote S-SAP, the remote T-SAP and the remote N-SAP.

The reasons for adopting this solution are the following:

a) it is natural, given the network architecture EIES is going to have (in particular, the T-SAP information is necessary to indicate to the Remote Transport Layer which Session Entity on which Distributed End System Component it is required to connect);

b) it has a general meaning, so it is easy to map onto it whichever policy EIES decides to implement at Application Level;

c) it is easier for the communication software to manage such an organization because the translations are limited to the minimum, and they are only on a one-to-one basis (names to addresses).

3.2. Addressing and Naming in the Communication Layers.

The address elements which are passed through the interface with the upper layer are described, then an explanation of their use is given, including the transformations which are applied to them.

A set of definitions is listed below in order to make clear the symbolic addressing names used in the following:

- A = Session User;

- B = Transport Service User in the LAN;
  (host in the LAN + TS user on the host)

- C = Site;

- BC = Transport Service User in the global network;

- D = DTE address in the X.25 subnetwork;

- E = Ethernet address in the LAN;

The corresponding OSI terms are:

- S_SAP = ABC;

- T_SAP = BC;

- N_SAP = D.

On a given host, there may be a number of Transport Service users (Session, UUCP, PAD, ...). These will be referred to collectively as 'Session Entities' in what follows.

3.2.1. Session Layer.

The Session Layer receives from the upper layer three address elements, named "A", "B" and "C".

"A" is used by the Session to identify the remote Application with which a Session Connection must be established; the Session will keep it until an adequate Transport Connection is available and will put it into the Connect SPDU which will be sent through the Transport Connection.

The receiving Session Entity will use it to identify the local Application to which the Connect Indication should be delivered.

"B" is used to identify the remote Session Entity with which a Transport Connection is required; it may assume two values:

* Case 1: "B" = 0 or missing
  this means that the session user has to connect to a user local to the same system "B", without passing through the Transport and the X.25 subnetwork; in this case the session entity gives an S_CONNECT indication to the session user "A".

* Case 2: "B" <> 0
  this means that the session user wants to connect to a user resident on a remote System, in this case the session entity issues a T_CONNECT request to the lower entity (the Relay and/or the Transport Class 4); "B" is

given by the Session to the Transport.

"C" is used to logically identify the remote End System and is given by the Session to the Transport.

The Session Entity does not perform any transformation on the address elements it manages.

3.2.2. Transport Layer (Class 2 and 3).

The Transport Layer (Class 2 and 3) receives from its users (the Session and/or the Relay) two address elements, named "B" and "C".

An internal table is searched for the physical DTE-ID associated with "C", and this is compared with the physical DTE-ID associated with the local system. Two cases may arise:

* Case 1: physical DTE-ID corresponding to "C" = Local Site physical DTE-ID;
  this means that the connection to be made is to a user on the local site; a TC-indication, with the ("B", "C") parameters supplied, is sent by Transport to the Relay Entity.

* Case 2: physical DTE-ID corresponding to "C" <> Local Site physical DTE-ID;
  this means that the connection to be made is to a user on a Remote Site, passing through the X.25 subnetwork. In this case, when an adequate Network Connection is available, the Transport will keep "B" and "C", put them together (to form "BC") which will be included in the Connect TPDU to be sent through the Network Connection.

The receiving Transport Entity will use "B" to identify the Session Entity local to the End System to whic the Connect Indication is delivered.

"C" is transformed by the sending Transport Entity into the Network Address "D". This transformation has been introduced to isolate the upper communication layers, and the high level addressing schemes, from the possible Global Network configuration changes.

"D" is given to the Network Layer to identify the remote Transport Entity with which a Network Connection is required.

3.2.3. Network Layer.

The Network Layer receives from the upper layer one address element, named "D".

"D" is used by the Network to identify the remote Transport Entity with which a Network Connection must be established; the Network will put it into the Connect NPDU which will be sent through the Global Network.

The Global Network will use it to identify the Transport Entity to which the Connect Indication should be delivered.

The Network Entity does not perform any transformatio on the address elements.

3.2.4. Transport Layer (Class 4).

The Transport Layer (Class 4) receives from its users (the Session and/or the Relay) two address elements, named "B" and "C".

It tests "C":

* Case 1: "C" = 0 or missing
  this means that the connection to be made is to a user on the Local Site, which can be made without passing through the X.25 subnetwork. In this case, an internal table is searched for the Ethernet address "E" associated with "B", and this is compared with the Ethernet address of the Local System. Two cases arise:

  - Case 1a: Ethernet address corresponding to "B" = Local System Ethernet address;
    this means that the connection to be made is to a user on the Local system. In this case, the Transport Entity sends a TC-indication to the Session Entity, supplying the "B" parameter.

  - Case 1b: Ethernet address corresponding to "B" <> Local System Ethernet address;
    this means that the connection to be made is to a user on a different system on the Local Site, without passing through the X.25 subnetwork. In this case, the Transport Entity sends a CR-TPDU to the Transport Entity, identified by the Remote System Name "B", directly through the Ethernet. The Transport Entity will keep "B" and "C", put them together (to form "BC") and include the result in the Connect TPDU which will be sent through the Ethernet Link.

* CASE 2: "C" <> 0:
  this means that the User has to connect to a User resident on a remote Site, passing through the X.25 subnetwork.

In this case the Transport Entity will keep "B" and "C" and put them together (making "BC"). Afterwards the Transport Entity will send a CR-TPDU to the Transport Entity of the Gateway, using its "well_known" Ethernet address "E", specifying in the Called T_SAP field the "BC" parameter.

3.2.5. Relay Entity

The relay Entity receives inputs from its user, that is the Session layer, and from the lower Entities (Transport Class 3 and/or Class 4). It always receives two address elements, "B" and "C". The Relay behaviour is different depending on whether a TC-request is received from the Session Entity or a TC-indication is received from the Transport Class 3 or from the Transport Class 4. This behaviour is described in detail in [EIE 84].

3.3. Name and Address Formats.

In this section, the formats for all the address elements identified in the previous sections are described.

For each address element, the format conforms to that specified in the ISO or CCITT documents, when available. The length of the Session User Address "A" is 16 bytes maximum [ISO 27]; there is no internal structure.

The System Name (in the Global Network) "BC" length is not defined [ISO 73]; the EIES project uses a length of 16 bytes. "B" is 8 bytes long as is the site name "C". They have no internal structure.

The length of the called/calling DTE international data number is 15 digits maximum (CCITT Document AP V||-No. 11-E). It has an internal structure.

The Ethernet Address "E" length is 6 bytes [ECM 82]; its internal structure is defined in the standard and the EIES project conforms to it.

4. Network Management Software for the first year.

Within the OSI architecture, the needs for managing the addressing structure and for gathering statistics is related to the special problems of initiating, terminating, and monitoring of activities for harmonious operations; these needs are collectively addressed by the network management components of the OSI architecture.

The software which performs all these network management functions may be divided into two different groups:

1) the software which has to be embedded in the communication software to interact with the protocol entities;

2) the software which is outside the communication environment and which manages the different tables in which the management items have to be stored and collected, updating and modifying them; it also provides a Management Service through a User Interface to a local Network Manager.

The Network Management Architecture for the first year is very simple; it just provides a local operator with an interface to access the services provided by two System Management Application Entities (SMAEs), which allows a User to control the different tables.

The different SMAEs are not connected among themselves, neither through the X.25 subnetwork nor through the local LAN.

```
+---------------------------------------+
|                          |            |          |
|                          |   SMAEs    |----------+ Operator
|                          |            |          |
+------------------+       +-+-+--------+
|                  |         | |        |
|  Communication   |      +--|-+---+    |
|                  +------+  |     |    |
|    Software      |    +----+---+ |    |
|                  +----+        | |    |
|                  |    | Tables |-+    |
|                  |    |        |      |
+---------------------------------------+
```

5. Non Standard Usage of the Network.

EIES users will have access to services other than those based on the Session Layer, and such facilities will be supported by the Transport Layer. In particular, for the year 0 EIES software, UUCP is implemented over the Transport Layer, and we describe below how this is done. In addition, in year 1 of EIES, PAD connections over transport will be supported, using the X.29 protocol, and this will be accomplished in a similar manner to that used for UUCP.

5.1. UUCP Sessions.

UUCP is used during the first year of the project as the only file transfer available in the EIES network; from the point of view of layered communication protocols it is seen by the Transport layer as another Session Entity; this means that any instance of UUCP, both on a Single End System or on a DE Component must be associated with a T-SAP.

It is clear that these T-SAPs are completely separated from the others used by the Session Entities, and it will never be possible to establish a Transport Connection between two T-SAPs belonging to the two groups.

The address management tools available for the EIES project are able to manage this second T-SAP group too, but no tool is provided to avoid an attempt at interworking between the two groups.

6. EIES Implementation Overview.

6.1. General.

In this section the major choices for the implementation of EIES services under Unix are described; perhaps the most important choice which has had to be made was to decide which of the services should be implemented in the Unix kernel and which should not.

The choice must balance the increased performance of a kernel implementation against the danger of reducing the overall efficiency of the operating system by overburdening the kernel.

Because of their intimate links to the hardware and the need for high performance, the X.25 and Ethernet drivers must be implemented in the kernel; there are also good reasons for preferring a kernel implementation of the transport service: again on the ground of efficiency and also because the required multiplexing and de-multiplexing are easier to implement.

A flexible approach has been adopted, which includes the following features:

a) Two implementations of the ISO transport service class 4 are proposed, one in the kernel and one in the user environment: this will cater for small kernel Unix machines on a LAN and will also allow a comparison of performance of the two implementations.

b) The architecture for the User level services is such that modules can be moved from kernel to user level in a relatively straightforward way.

c) A homogeneous approach to the user level services has been adopted, based on an Inter-Process-Communication mechanism; again this results in great flexibility.

d) Crucial features of the communications between different user level processes will be implemented as kernel devices; these drivers will be used by all user level services, so that a small investment in kernel code is effectively used.

Finally, a phased implementation strategy has been adopted, and in fact, features c) and d) are not generally implemented in the first phase, although a prototype implementation is being carried out.

A further important criterion concerns the portability of the Unix implementation; it is relatively straightforward to add new drivers to the Unix kernel; in particular, this can be done without having access to the source code of the existing Unix kernel, consequentely kernel modifications have been limited to the addition of drivers, which can then be readily installed on other machines.

The implementation proposals are based on Version 7 Unix; however, as a further aid to portability, upward compatibility with System III and System V Unix has been aimed at; when features were required which are not part of standard Version 7 Unix, these later versions of Unix have been used as a model.

As a further aid to portability, and to facilitate eventual changes in implementation details, procedural interfaces to each service have been defined, which can be readily mapped onto either Unix system calls, if the service is implemented in the kernel, or onto a set of primitives for communication between user processes.

In the first phase of the implementation, a standard UNIX system call interface to the kernel communications services will be provided. However, Version 7 Unix (and, to a lesser extent, System III and System V Unix) lack certain features which would facilitate the implementation of communications software. This will affect especially later phases of the project, when a number of user-level processes will be present. In order to make up for these deficiencies, it is planned, during a second phase, and as a prototype during the first phase, to implement two Unix pseudo-device drivers to permit Inter-Process Communication and Memory Management. These are described below.

6.2. EIES Pseudo Device.

In our software architecture the task implementing the OSI n-entity is related to the (n+1)-entity by a server / user relationship, and again the top level entity is the server of the final user.

In order to allow for a common process interaction scheme with such a relationship, an Inter Process Communication mechanism has been introduced. IPC is one of the two functions of the "EIES driver", the second one is to provide an eventually non-blocking interaction mechanism between a task at user level and a server in the kernel. The IPC mechanism introduces a multiple wait scheme, so that a server is never blocked waiting for a particular event, but eventually only on all events it must serve.

In our environment, each entity which implements a protocol is provided with a special port with a "port name" known by any other entity that has to interact with it; this special port is used by the entity to receive special messages, that is, newly created port names or system messages. Each entity is always able to listen on such a port; that is, a message on that port is always delivered to the entity.

An entity at user level (in the Unix sense) is allowed to interact with its server only through the EIES driver; in particular, if the server is in the kernel, the EIES driver interface will mask an access to the specific Protocol Handler.

6.3. Kernel Memory Management Pseudo Device.

The purpose of this pseudo device is to avoid multiple copying of data; data is copied into a chain of buffers once only, and thereafter it is manipulated using a chain identifier: the transfer of information between user processes is accomplished by passing the appropriate chain identifier.

A number of primitive functions for the creation, destruction, assembling and fragmentation of chains of data will be provided by the memory management pseudo device.

On transmission, an application process copies data to the memory management pseudo device; this copy creates a chain of data buffers and a chain identifier is returned.

The communication services use this chain identifier to add headers, fragment data, etc. before the data is passed eventually to communications hardware.

On reception, the memory management pseudo device copies data into a chain of buffers and provides a chain identifier to the lowest layer of the communication protocol services.

The different protocol layers remove their headers and possibly concatenate data before passing a chain identifier to the next higher protocol layer; eventually the user data is copied into user space by the application process; thus, only one data copy is required for both transmission and reception.

7. Conclusions.

A survey of the main implementation choices made for the EIES has been made. It is to be noted that these choices are fully in line with the set of OSI standards recommended by the group of twelve European manufacturers.

It should be noted that only in the next years of the project will OSI applications become available, thus allowing distributed applications with non UNIX systems implementing OSI protocols.

8. References.

CCI 10 CCITT AP VII-No. 10-E "Recommendation X.75"

CCI 11 CCITT AP VII-No. 11-E "Recommendation X.121. International numbering plan for Public Data Networks"

CCI 80 CCITT VIII.2 Rec. X.25 "Interface Between DTE and DCE for ..."

CCI 84 CCITT Draft Recommendations X.4XX: "Message Handling Systems"

ECM 80 Local Area Networks (CSMA/CD baseband)

Basic cable system

ECM 81 Local Area Networks (CSMA/CD baseband)

Physical layer

ECM 82 Local Area Networks (CSMA/CD baseband)

Link layer

ECM 92 Connectionless Internetwork Protocol

ECM 13 Network layer principles

ECM 14 Local Area Networks Layer 1 to 4 architecture and protocols

ECM 20 Layer 4 to 1 addressing

ECM 21 Local Area Networks Distributed system Interworking Units June 1983

EIE 84 EIES Technical Specifications (version 2)

GIE 83 GIEN M.

SOL: A UNIX environment in PASCAL Jan.83

IES 84a Panel 6 ESPRIT Infrastructure Preliminary Draft Report

IES 84b IES Workplan (ESPRIT area 6.1)

IES 82 JEPE-IT ESPRIT-IES Panel final report

ISO 26 ISO DIS 8326

Basic connection oriented session service definition

ISO 27 ISO DIS 8327 Basic connection oriented session protocol specification

ISO 72 ISO DIS 8072

Transport service

ISO 73 ISO DIS 8073

Transport protocol

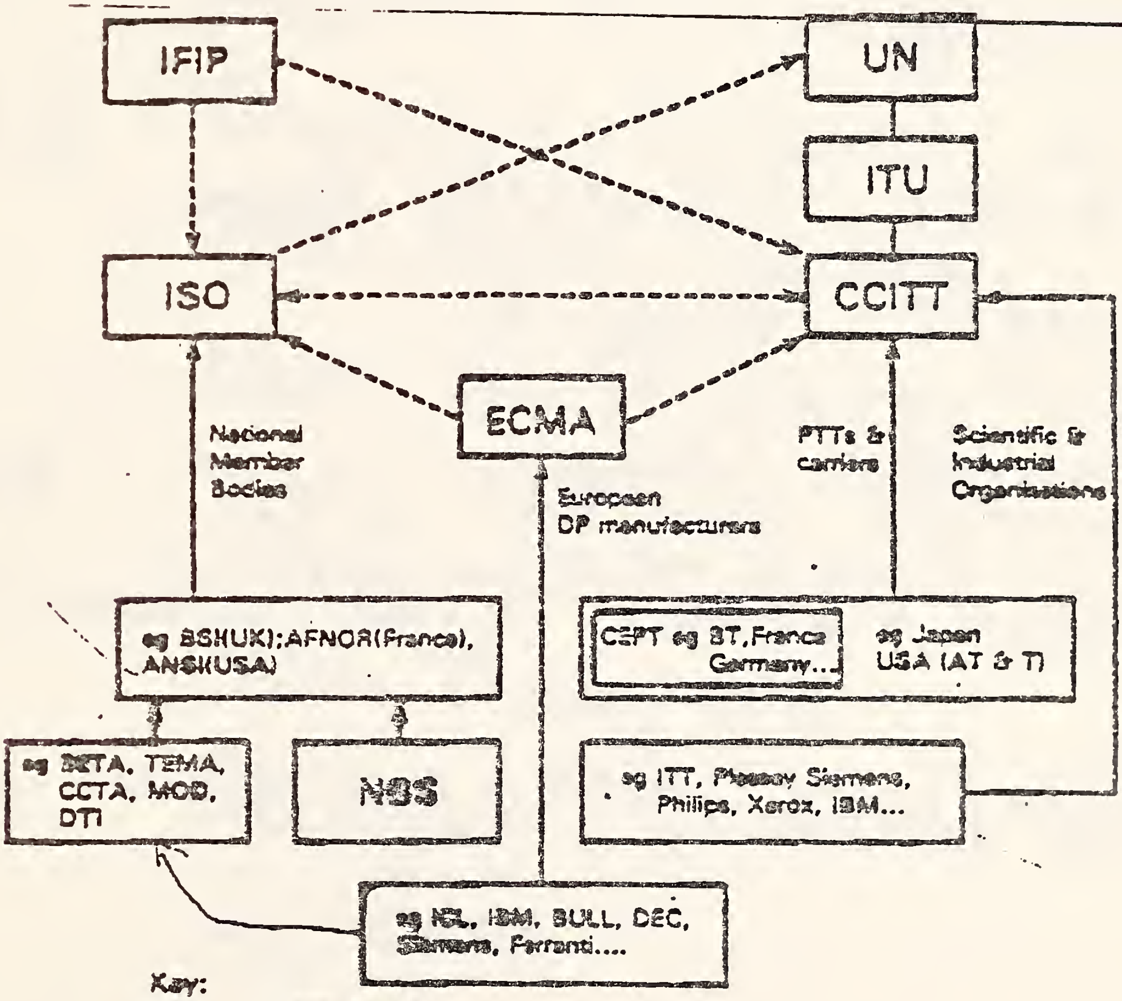
ISO 71 ISO DP 8671

File Transfer and Manipulation

ISO 98 ISO DP 74 98

Reference model

UBI 83 UBIES Final Report by CII-HB, GEC, ICL, SIEMENS


IFIP
UN
ITU
ISO
CCITT
ECMA
National Member Bodies
European DP manufacturers
PTTs & carriers
Scientific & Industrial Organisations
eg BSI(UK);AFNOR(France), ANSI(USA)
CEPT eg BT, France Germany...
eg Japan USA (AT & T)
eg BETA, TEMA, CCTA, MOD, DTI
NBS
eg ITT, Plessey Siemens, Philips, Xerox, IBM...
eg ICL, IBM, BULL, DEC, Siemens, Ferranti....

Key:

UN	United Nations
ITU	International Telegraph Union
CCITT	International Telegraph & Telephone Consultative Committee
ISO	International Standards Organisation
CSPT	Conference of European PTT's
ECMA	European Computer Manufacturers Association
IFIP	International Federation for Information Processing
NBS	National Bureau of Standards (US)
BSI	British Standards Institution
AFNOR	Association francaise de Normalisation
ANSI	American National Standards Institute
BETA	Business Equipment Trade Association
TEMA	Telecommunication Equipment Manufacturers Association
CCTA	Central Computer and Telecommunications Agency
MOD	Ministry of Defence
DTI	Department of Trade and Industry

Fig 1.1 Relationship of Relevant Organisations

ITSU - UK.

ECMA

"Ad-Hoc" standards Group.

ESPRIT Panel 6.

'Zander' Group

ESPRIT/ALVEY Steering Committees.

Standards Harmonisation.

ECMA Technical Committees

TC23 Layers 5,6, VTP, Mgt.

~~Guerrino de Luca, Olivetti~~

TC24 Transport, Internet, LAN
PABX, HDLC link level.

Pieter van Studaity, Phillips

TC25 Public Network Services
ISDN, CEPT/CCITT liaison.

Alan Thoms, ICL

TC29 Messaging, telematics services
eg. teletex.

Zeckendorff, Phillips

TC30 SCSI.

CERN	GENEVA	
SERC	UK	SERCNET/ JANET
INRIA	FRANCE	CNET
	UK	ALVEYNET
	W.GERMANY	DFN
	ITALY	OSIRIDE

ESPRIT

European Strategic Program for Research into Information Technology

ALVEY - U.K.

VLSI

IKBS

ECRC

European Computer Research Centre

- Bull, ICL, Siemens.

ESPRIT 'Round Table'

ICL	UK
GEC	
Plessey	
Phillips	Holland
Siemens	W. Germany
AEG-Telefunken	
Nixdorf	
Olivetti	Italy
CSELT	
Bull	France.
Thomson-CSF	
SIP	

Teletex

Committee
Support
System

X400

Portable
Common
Tool
Environment

FTAM

UNIX
+
OSI
Comms.

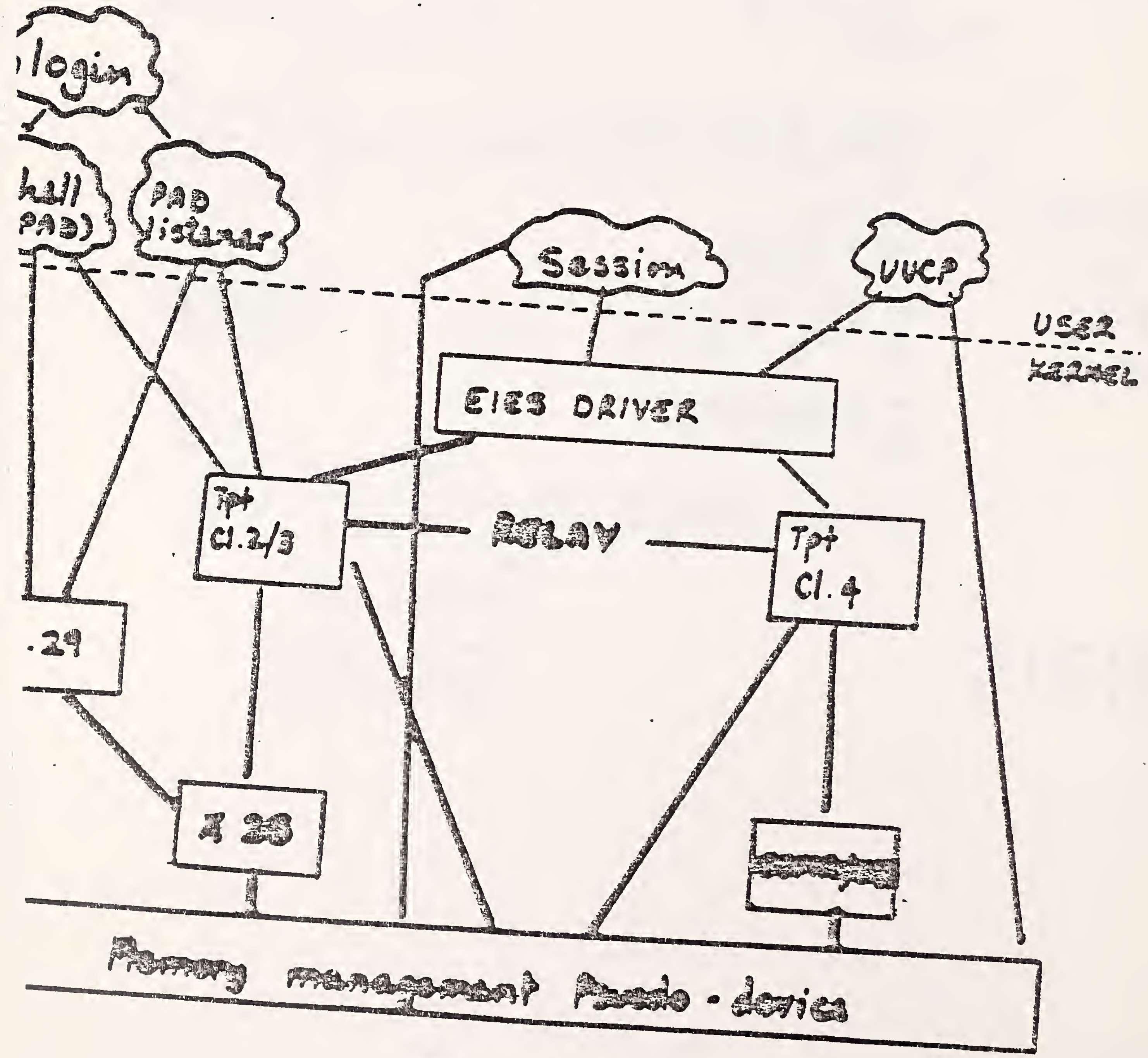
Research
Open
System for
Europe
(alias UBIES, EROS, EIES)

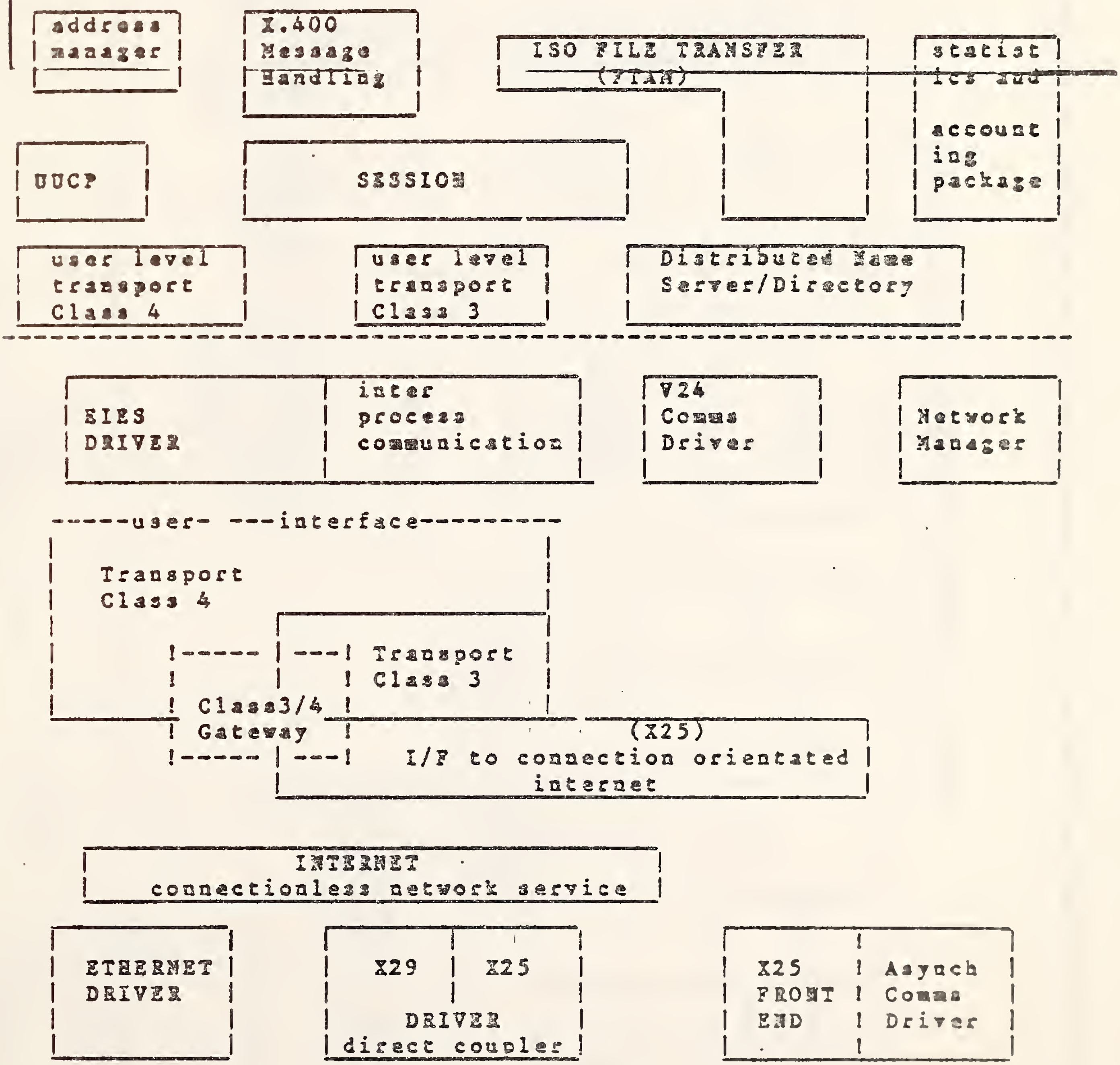
X25

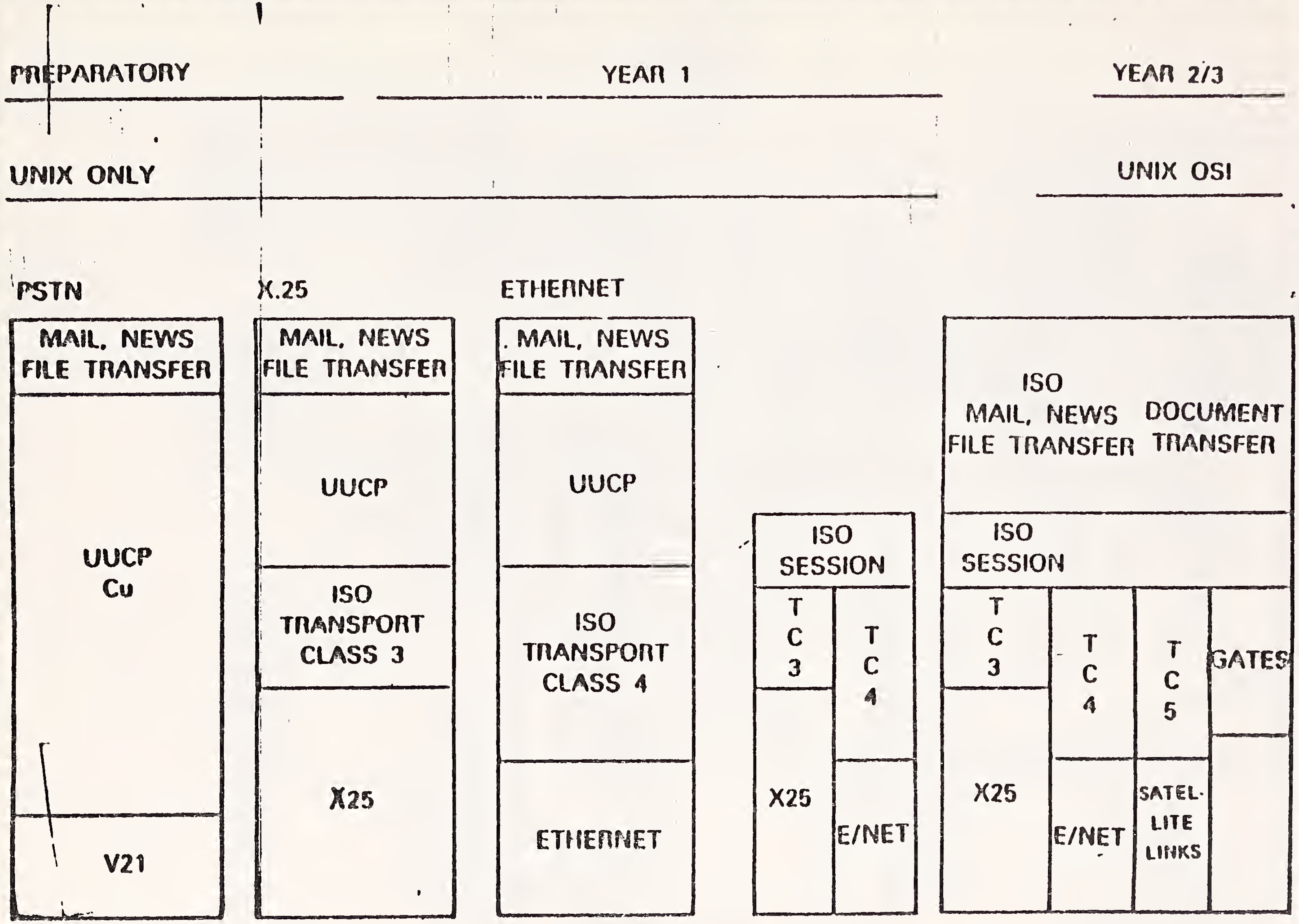
Internet

INSIS

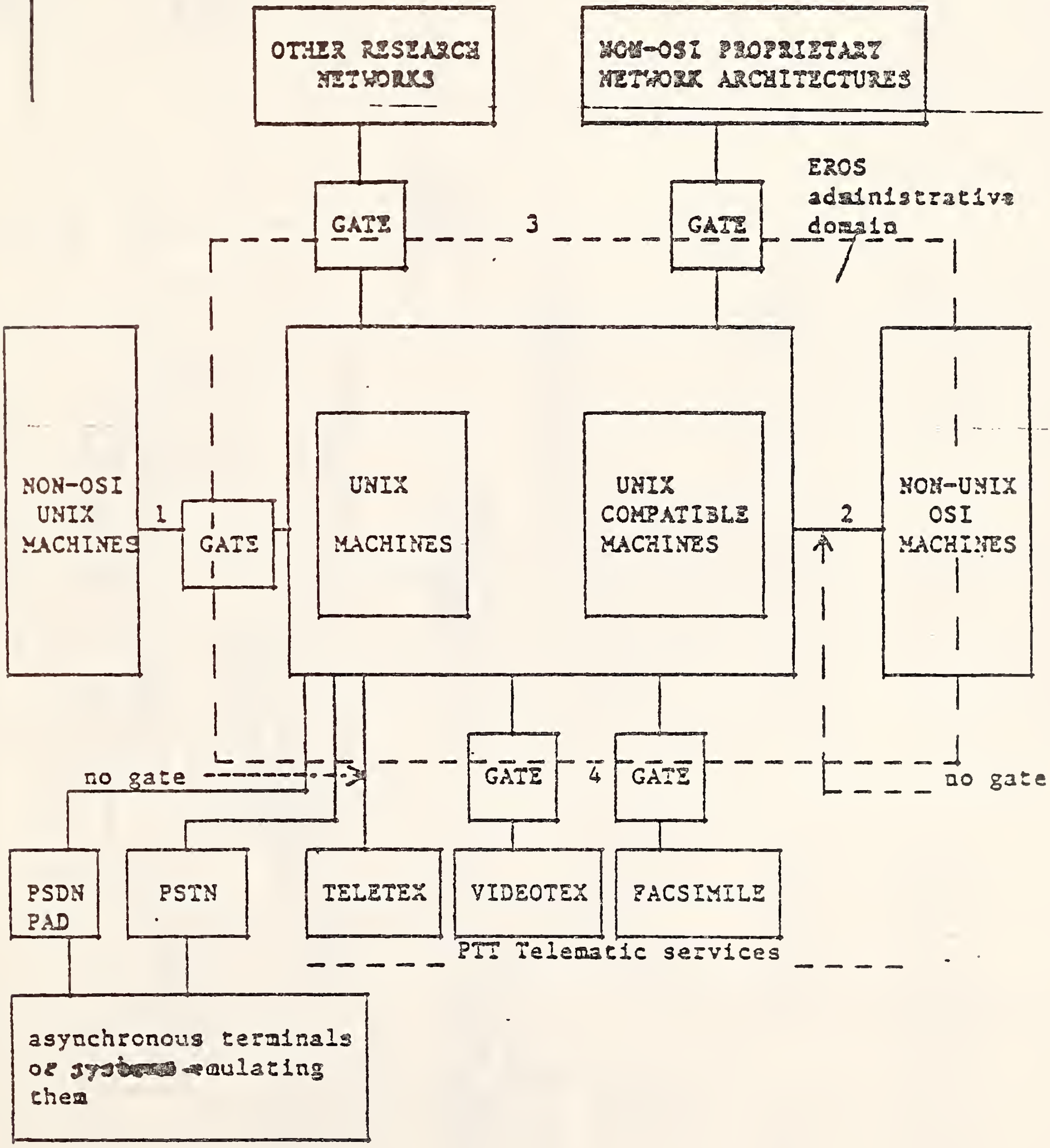
ELAN

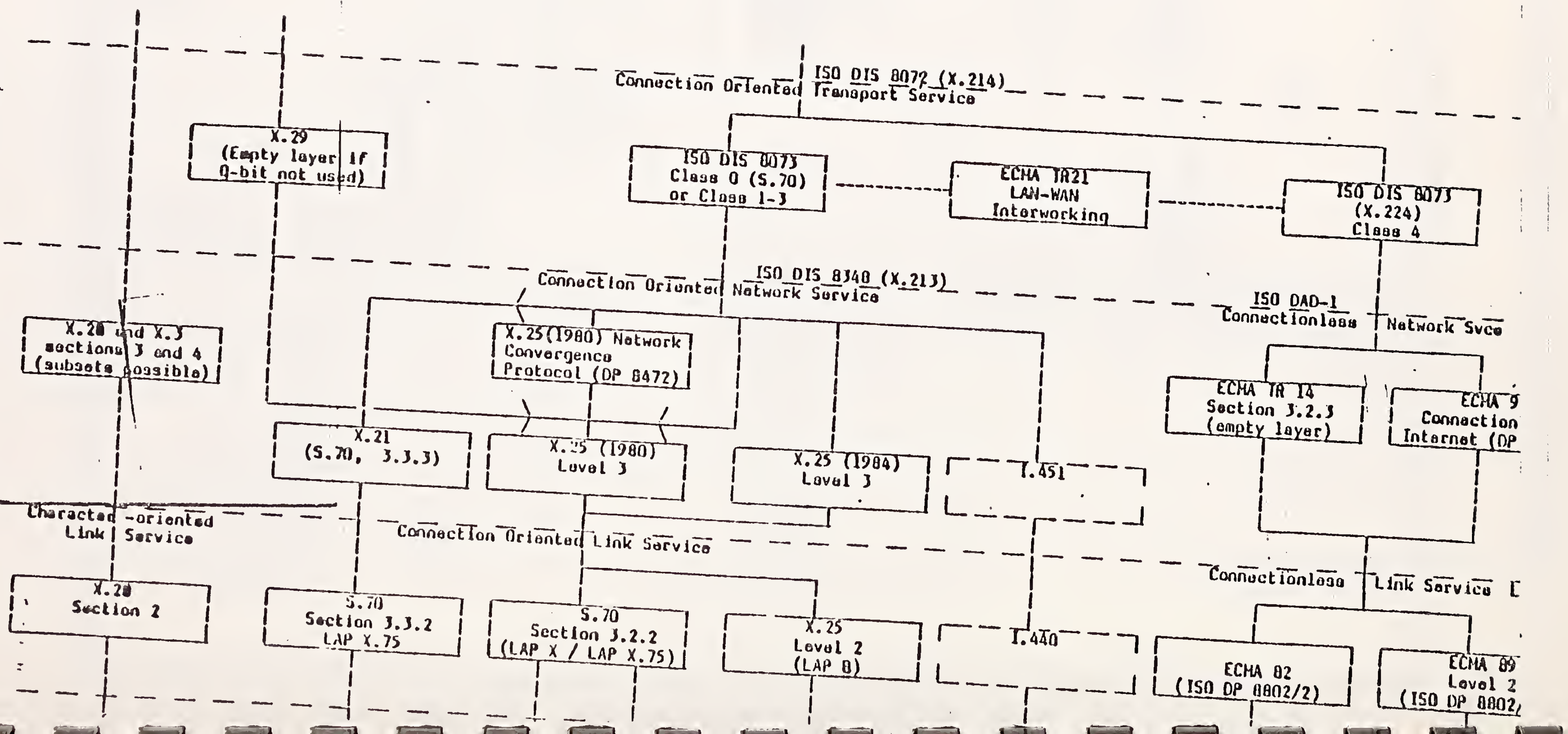

login
hell
PAD)
PAD
listener
Session
UUCP
USER
KERNEL
EIES DRIVER
Tpt
Cl.2/3
RELAY
Tpt
Cl.4
.29
X 25
Memory management Pseudo - devices

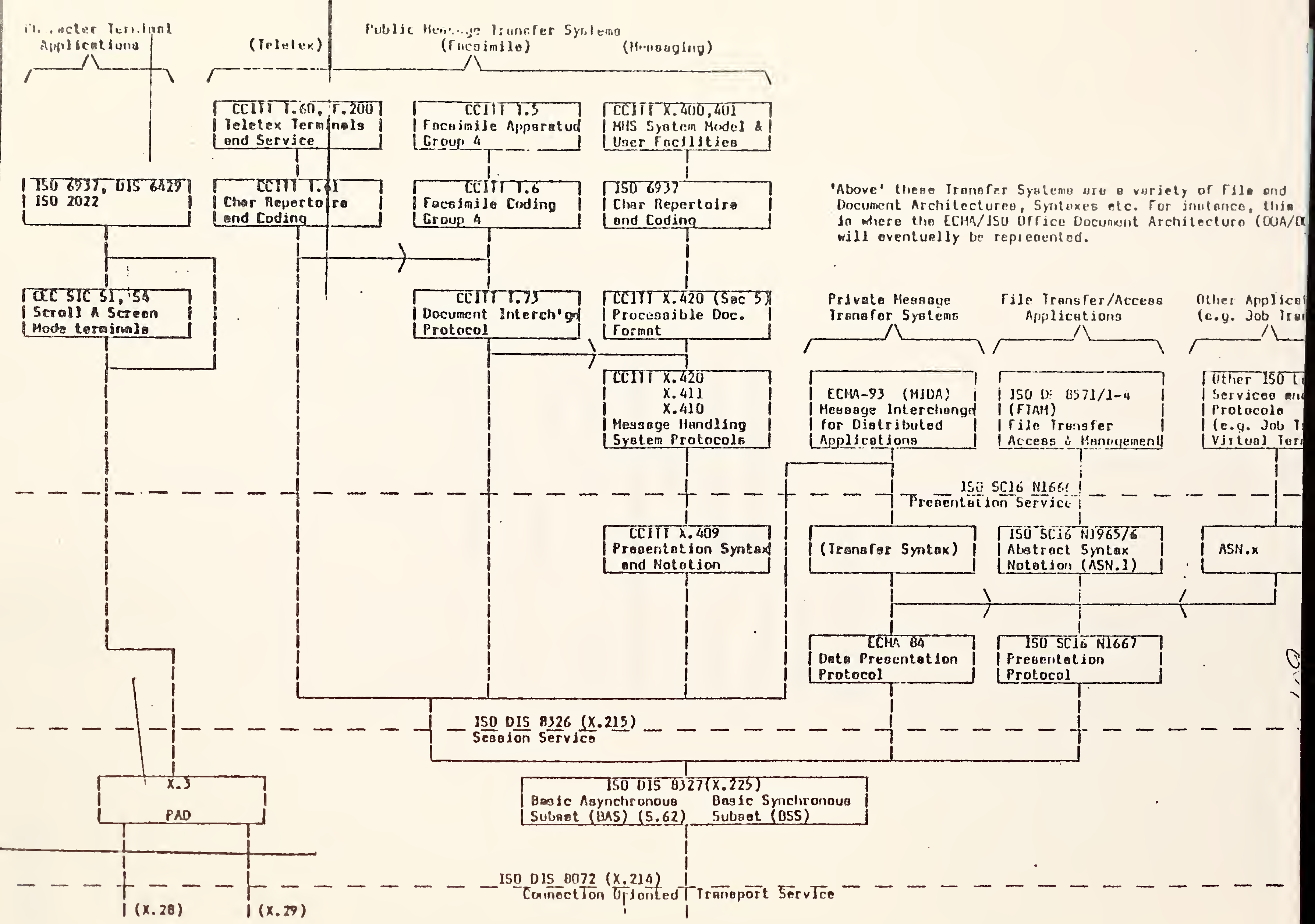

address manager
X.400 Message Handling
ISO FILE TRANSFER (FTAM)
statistics and accounting package
UUCP
SESSION
user level transport Class 4
user level transport Class 3
Distributed Name Server/Directory
EIES DRIVER
inter process communication
V24 Comms Driver
Network Manager
-----user- ---interface---------
Transport Class 4
Transport Class 3
Class3/4 Gateway
(X25) I/F to connection orientated internet
INTERNET connectionless network service
ETHERNET DRIVER
X29
X25
DRIVER direct coupler
X25 FRONT END
Asynch Comms Driver



EVOLUTION OF THE SERVICES PROVIDED BY UBIES


OTHER RESEARCH NETWORKS
NON-OSI PROPRIETARY NETWORK ARCHITECTURES
EROS administrative domain
GATE
3
GATE
NON-OSI UNIX MACHINES
1
GATE
UNIX MACHINES
UNIX COMPATIBLE MACHINES
2
NON-UNIX OSI MACHINES
no gate
GATE
4
GATE
no gate
PSDN PAD
PSTN
TELETEX
VIDEOTEX
FACSIMILE
PTT Telematic services
asynchronous terminals or systems emulating them


ISO DIS 8072 (X.214)
Connection Oriented Transport Service
X.29 (Empty layer if Q-bit not used)
ISO DIS 8073 Class 0 (S.70) or Class 1-3
ECMA TR21 LAN-WAN Interworking
ISO DIS 8073 (X.224) Class 4
ISO DIS 8348 (X.213)
Connection Oriented Network Service
ISO DAD-1
Connectionless Network Svce
X.28 and X.3 sections 3 and 4 (subsets possible)
X.25(1980) Network Convergence Protocol (DP 8472)
ECMA TR 14 Section 3.2.3 (empty layer)
ECMA 9 Connection Internet (DP
X.21 (S.70, 3.3.3)
X.25 (1980) Level 3
X.25 (1984) Level 3
I.451
Character-oriented Link Service
Connection Oriented Link Service
Connectionless Link Service
X.28 Section 2
S.70 Section 3.3.2 LAP X.75
S.70 Section 3.2.2 (LAP X / LAP X.75)
X.25 Level 2 (LAP B)
I.440
ECMA 82 (ISO DP 8802/2)
ECMA 89 Level 2 (ISO DP 8802/


Character Terminal Applications
(Teletex)
Public Message Transfer Systems
(Facsimile)
(Messaging)
CCITT T.60, F.200 Teletex Terminals and Service
CCITT T.5 Facsimile Apparatud Group 4
CCITT X.400,401 MHS System Model & User Facilities
ISO 6937, DIS 6429 ISO 2022
CCITT T.61 Char Repertoire and Coding
CCITT T.6 Facsimile Coding Group 4
ISO 6937 Char Repertoire and Coding
CEC SIC 51, 54 Scroll A Screen Mode terminals
CCITT T.73 Document Interch'ge Protocol
CCITT X.420 (Sec 5) Processible Doc. Format
Private Message Transfer Systems
File Transfer/Access Applications
Other Applica
(e.g. Job Tra
CCITT X.420 X.411 X.410 Message Handling System Protocols
ECMA-93 (MIDA) Message Interchange for Distributed Applications
ISO DP 8571/1-4 (FTAM) File Transfer Access & Management
Other ISO L
Services an
Protocols
(e.g. Job T
Virtual Ter
ISO SC16 N1666 Presentation Service
CCITT X.409 Presentation Syntax and Notation
(Transfer Syntax)
ISO SC16 N1965/6 Abstract Syntax Notation (ASN.1)
ASN.x
ECMA 84 Data Presentation Protocol
ISO SC16 N1667 Presentation Protocol
ISO DIS 8326 (X.215) Session Service
X.3
PAD
ISO DIS 8327(X.225) Basic Asynchronous Subset (BAS) (S.62) Basic Synchronous Subset (BSS)
ISO DIS 8072 (X.214) Connection Oriented Transport Service
(X.28)
(X.29)

'Above' these Transfer Systems are a variety of File and Document Architectures, Syntaxes etc. For instance, this is where the ECMA/ISO Office Document Architecture (ODA/O will eventually be represented.

PROTOCOL LAYERS 5 TO 7

Att. #26

OSI Session Service

- CONCEPTS
- GENERAL RULES
- CONNECTION ESTABLISHMENT
- DATA TRANSFER
- CONNECTION RELEASE
- COLLISION RESOLUTION

KM 9/6/84

CONCEPTS

- TOKENS
- SYNCHRONIZATION & DIALOGUE UNITS
- ACTIVITIES
- RESYNCHRONIZATION
- FUNCTIONAL UNITS & SUBSETS
- NEGOTIATION
- DATA CATEGORIES

TOKENS

- PERMIT ALTERNATING CONTROL OF SERVICES
- FOUR SESSION TOKENTS
  - DATA
  - RELEASE
  - SYNCH-MINOR
  - MAJOR-ACTIVITY
- TOKEN STATES
  - AVAILABLE
  - NOT AVAILABLE
- AVAILABLE SUB-STATES
  - ASSIGNED
  - NOT ASSIGNED

SYNCHRONIZATION & DIALOGUE UNITS

- SYNCHRONIZATION POINT TYPES
  - MINOR
  - MAJOR

- MAJOR SYNCH POINTS DELIMIT DIALOGUE UNITS

- MINOR SYNCH POINTS DELIMIT SUB-UNITS

- CONFIRMATION
  - EXPLICIT FOR MAJOR SYNCH
  - MAY BE EXPLICIT FOR MINOR SYNCH

- NO SEMANTICS ASSOCIATED WITH SYNCH POINTS

ACTIVITIES

- DISTINGUISH DIFFERENT PIECES OF LOGICAL WORK
- CONSIST OF ONE OR MORE DIALOGUE UNITS
- ONE ACTIVITY ON A CONNECTION AT ONE TIME
- SEVERAL ACTIVITIES MAY USE A CONNECTION SEQUENTIALLY
- AN ACTIVITY MAY SPAN MORE THAN ONE SESSION CONNECTION
- ACTIVITIES MAY BE INTERRUPTED AND RESUMED
- USERS MAY SEND DATA OUTSIDE AN ACTIVITY
- ACTIVITY END = MAJOR SYNCH POINT

RESYNCHRONIZATION

- SETS SESSION CONNECTION TO A DEFINED STATE
  - TOKENS
  - SYNCH POINT SERIAL NUMBER
- PURGES ALL UNDELIVERED DATA
- THREE OPTIONS
  - ABANDON
  - RESTART
  - SET
- SEMANTICS ARE USER DEFINED

FUNCTIONAL UNITS & SUBSETS

- FUNCTIONAL UNITS ARE LOGICAL GROUPINGS OF RELATED SERVICES
- FUNCTIONAL UNITS ARE INDIVIDUALLY NEGOTIABLE AT CONNECTION ESTABLISHMENT
- CERTAIN FUNCTIONAL UNITS IMPLY TOKEN AVAILABILITY
- TOKEN MANAGEMENT SERVICES ARE REQUIRED WITH TOKEN AVAILABILITY
- SUBSETS ARE COMBINATIONS OF THE KERNEL FUNCTIONAL UNIT TOGETHER WITH ANY OTHER SET OF FUNCTIONAL UNITS
- SUBSETS HAVE NO MEANING IN THE SESSION PROTOCOL

FUNCTIONAL UNITS

- KERNEL
- DUPLEX
- HALF-DUPLEX
- NEGOTIATED RELEASE
- EXPEDITED DATA
- TYPED DATA
- CAPABILITY DATA
- MINOR SYNCH
- MAJOR SYNCH
- RESYNCH
- EXCEPTIONS
- ACTIVITY MANAGEMENT

PREDEFINED SUBSETS

- BASIC COMBINED SUBSET
  - KERNEL
  - DUPLEX OR HALF-DUPLEX
- BASIC SYNCHRONIZED SUBSET
  - KERNEL
  - NEGOTIATED RELEASE
  - HALF-DUPLEX
  - TYPED DATA
  - MINOR & MAJOR SYNCH
  - RESYNCH
- BASIC ACTIVITY SUBSET
  - KERNEL
  - HALF-DUPLEX
  - TYPED DATA & CAPABILITY DATA
  - MINOR SYNCH
  - EXCEPTIONS
  - ACTIVITY MANAGEMENT

NEGOTIATION

- OCCURS DURING CONNECTION ESTABLISHMENT
- FUNCTIONAL UNITS ON CONNECTION
- INITIAL TOKEN SETTINGS
- INITIAL SYNCH POINT SERIAL NUMBER

DATA CATEGORIES

- NORMAL
- EXPEDITED
- TYPED
- CAPABILITY

GENERAL RULES

- TOKEN RESTRICTIONS
- NEGOTIATION RULES
- PRIMITIVE SEQUENCE
- SYNCH POINT SÉRIAL NUMBER MANAGEMENT

TOKEN RESTRICTIONS

- INITIATE RELEASE
  - ALL AVAILABLE TOKENS ASSIGNED
- SEND DATA
  - DATA TOKEN ASSIGNED (HALF-DUPLEX)
  - DATA TOKEN UNAVAILABLE (DUPLEX)
- GIVE TOKEN REQUEST
  - TOKEN ASSIGNED
- PLEASE TOKEN REQUEST
  - TOKEN UNASSIGNED
- ACTIVITY START, RESUME, END
  - DATA & SYNCH MINOR TOKEN UNAVAILABLE OR ASSIGNED
  - MAJOR-ACTIVITY TOKEN ASSIGNED

TOKEN RESTRICTIONS (CONTINUED)

- ACTIVITY INTERRRUPT, DISCARD
  - MAJOR-ACTIVITY TOKEN ASSIGNED
- MINOR SYNCH REQUEST
  - DATA TOKEN UNAVAILABLE OR ASSIGNED
  - MINOR SYNCH TOKEN ASSIGNED
- MAJOR SYNCH REQUEST
  - DATA & SYNCH MINOR TOKENS ASSIGNED OR UNAVAILABLE
  - MAJOR-ACTIVITY TOKEN ASSIGNED
- EXCEPTION REPORT REQUEST
  - DATA TOKEN UNASSIGNED
- CAPABILITY DATA
  - DATA & SYNCH MINOR TOKENS ASSIGNED OR UNAVAILABLE
  - MAJOR-ACTIVITY TOKEN ASSIGNED

FUNCTIONAL UNIT NEGOTIATION

- REQUESTOR PROPOSES A SET OF FUNCTIONAL UNITS
- ACCEPTOR ALSO PROPOSES A SET OF FUNCTIONAL UNITS
- HALF-DUPLEX & DUPLEX MAY NOT BOTH BE PROPOSED BY ACCEPTOR
- CAPABILITY DATA MAY BE PROPOSED ONLY IF ACTIVITY MANAGEMENT IS PROPOSED
- EXCEPTION REPORTING MAY BE PROPOSED ONLY IF HALF-DUPLEX IS PROPOSED
- SELECTED FUNCTIONAL UNITS ARE THE INTERSECTION OF THE REQUESTOR AND ACCEPTOR PROPOSALS

INITIAL SYNCH POINT SERIAL NUMBER

- PROPOSED WITH MINOR SYNCH, MAJOR SYNCH, OR RESYNCH FUNCTIONAL UNITS WHEN ACTIVITY MANAGEMENT IS NOT PROPOSED

- ACCEPTOR SELECTING ANY OF THE PROPOSED FUNCTIONAL UNITS RETURNS A VALUE THAT WILL BE INITIAL SYNCH POINT FOR CONNECTION

- ACTIVITY MANAGEMENT FUNCTIONAL UNIT IMPLIES INITIAL SYNCH POINT SERIAL NUMBER OF ONE

INITIAL TOKEN ASSIGNMENTS

- WHEN A FUNCTIONAL UNIT REQUIRING TOKEN IS PROPOSED AN INITIAL TOKEN LOCATION IS ALSO PROPOSED
- POSSIBILITIES: CALLING SIDE, CALLED SIDE, CALLED CHOICE
- WHEN FUNCTIONAL UNIT IS SELECTED TOKEN REVERTS TO SIDE PROPOSED BY CALLER EXCEPT -
- IF CALLER SAID CALLED CHUICE, TOKEN REVERTS TU SIDE PROPOSED BY CALLED USER

PRIMITIVE SEQUENCING

- USER REQUESTS & RESPONSES ARE DELIVERED BY PROVIDER IN THE ORDER SUBMITTED EXCEPT -
- SEVERAL REQUESTS WHICH MAY BE DELIVERED EARLIER THAN NORMAL
- EXCEPTIONS INCLUDE:

S-EXPEDITED-DATA

S-RESYNCHRONIZE

S-ACTIVITY-INTERRUPT

S-ACTIVITY-DISCARD

S-U-ABORT

SYNCH POINT SERIAL NUMBER MANAGEMENT

- DEFINED AS OPERATIONS ON FOUR ABSTRACT VARIABLES - V(M), V(A), V(R), Vsc
- V(A) - LOWEST SERIAL NUMBER TO WHICH SYNCH POINT CONFIRMATION IS EXPECTED
- V(M) - NEXT SERIAL NUMBER TO BE USED
- V(R) - LOWEST SERIAL NUMBER TO WHICH RESYNCH RESTART IS PERMITTED
- Vsc - CONTROLS RIGHT OF USER TO ISSUE MINOR SYNCH POINT CONFIRMATIONS
- SYNCH & RESYNCH REQUESTS, RESPONSES, INDICATIONS, & CONFIRMATIONS EXAMINE THESE VARIABLES AND CAUSE OPERATIONS TO BE PERFORMED ON THEM

SESSION CONNECT PARAMETERS

- CONNECTION INDENTIFIER
- CALLING/CALLED SSAP
- RESULT
- QOS
- SESSION FUNCTIONAL UNIT REQUIREMENTS
- INITIAL SYNCH POINT SERIAL NUMBER
- INITIAL TOKEN ASSIGNMENTS
- USER DATA (TO 512 OCTETS)

SESSION DATA TRANSFER

- UNLIMITED NORMAL DATA PER SDU
- EXPEDITED SDU 1 TO 14 OCTETS
- UNLIMITED TYPED DATA PER SDU
- CAPABILITY DATA SDU 1 to 512 OCTETS
- NORMAL DATA SUBJECT TO TOKEN RESTRICTIONS
- CAPABILITY DATA SUBJECT TO TOKEN RESTRICTIONS AND ACTIVITY CONTEXT
- TYPED & EXPEDITED DATA ARE FULL-DUPLEX

TOKEN MANAGEMENT

- PLEASE TOKENS
  - LIST OF REQUESTED TOKENS
  - UP TO 512 OCTETS USER DATA
- GIVE TOKENS
  - LIST OF SURRENDERED TOKENS
- GIVE CONTROL
  - SURRENDERS ALL AVAILABLE TOKENS
  - ONLY PERMITTED WHEN ACTIVITY MANAGEMENT IS SELECTED AND NO ACTIVITY IS IN PROGRESS

SYNCH POINTS

- MINOR SYNCH POINT
  - EXPLICIT OR OPTIONAL CONFIRMATION
  - SERIAL NUMBER
  - UP TO 512 OCTETS USER DATA
- MAJOR SYNCH POINT
  - SERIAL NUMBER
  - UP TO 512 OCTETS USER DATA
- RESYNCH
  - TYPE: ABANDON, RESTART, SET
  - SERIAL NUMBER
  - TOKENS & LOCATIONS
  - UP TO 512 OCTETS USER DATA

EXCEPTION REPORTING

- PROVIDER EXCEPTION
  - REASON
  - NO DATA TRANSFER OR SYNCH POINTS UNTIL ERROR IS CLEARED
  - CLEARED BY RESYNCH, ABORT, INTERRUPT, DISCARD, OR GIVING DATA TOKEN
- USER EXCEPTION
  - REASON
  - UP TO 512 OCTETS USER DATA
  - WORKS ONLY IN HALF-DUPLEX MODE
  - NO DATA TRANSFER OR SYNCH POINTS UNTIL ERROR IS CLEARED
  - SAME CLEARING PROCEDURES AS FOR PROVIDER EXCEPTION

ACTIVITY MANAGEMENT

- START
  - ACTIVITY IDENTIFIER
  - UP TO 512 OCTETS USER DATA
- END
  - SERIAL NUMBER
  - UP TO 512 OCTETS USER DATA
  - EQUIVALENT TO MAJOR SYNCH POINT
- DISCARD
  - REASON
  - DATA WILL BE LOST
- INTERRUPT
  - REASON
  - UNDELIVERED DATA WILL BE LOST
- RESUME
  - NEW & OLD ACTIVITY IDENTIFIERS
  - SERIAL NUMBER
  - OLD SESSION CONNECTION IDENTIFIER
  - UP TO 512 OCTETS USER DATA

CONNECTION RELEASE

o ORDERLY RELEASE

  - RESPONSE RESULT (IF NEGOTIATED)
  - UP TO 512 OCTETS USER DATA

o USER ABORT

  - UP TO 9 OCTETS USER DATA

o PROVIDER ABORT

  - REASON

COLLISION RESOLUTION

- HIERARCHY OF REQUESTS
  - ABORT
  - DISCARD
  - INTERRUPT
  - RESYNCH (ABANDON)
  - RESYNCH (SET)
  - RESYNCH (RESTART)
  - USER EXCEPTION
- RESYNCH (ABANDON) COLLISIONS RESOLVED IN FAVOR OF CALLING USER
- RESYNCH (RESTART) COLLISIONS RESOLVED IN FAVOR OF LOWEST SERIAL NUMBER OR CALLING USER FOR EQUAL SERIAL NUMBERS
- RESYNCH (SET) COLLISIONS RESOLVED IN FAVOR OF CALLING USER

Att. #27

ISO File Transfer, Access, and Management: Model, Services, and Protocol

James C. Berets
Bolt Beranek and Newman Inc.

9/10/86

STATUS OF FTAM

Work internationally progressing in ISO/TC97/SC21/WG5 (recently moved from SC16).

Work in U.S. progressing in ANSI/X3T5.5.

FTAM recently balloted as ISO Draft Proposal 8571.

Second DP ballot probable in early 1985.

Mapping to Session pass-through services still under discussion.

THE VIRTUAL FILESTORE

Descriptive model to uniformly represent the properties of filestores and the files contained in those filestores.

Allows differences in filestore implementation to be absorbed into a local mapping.

Virtual filestore representation not limited to real filestores.

Virtual filestore defines: file access structure, file and activity attributes, actions on files.

File access structure

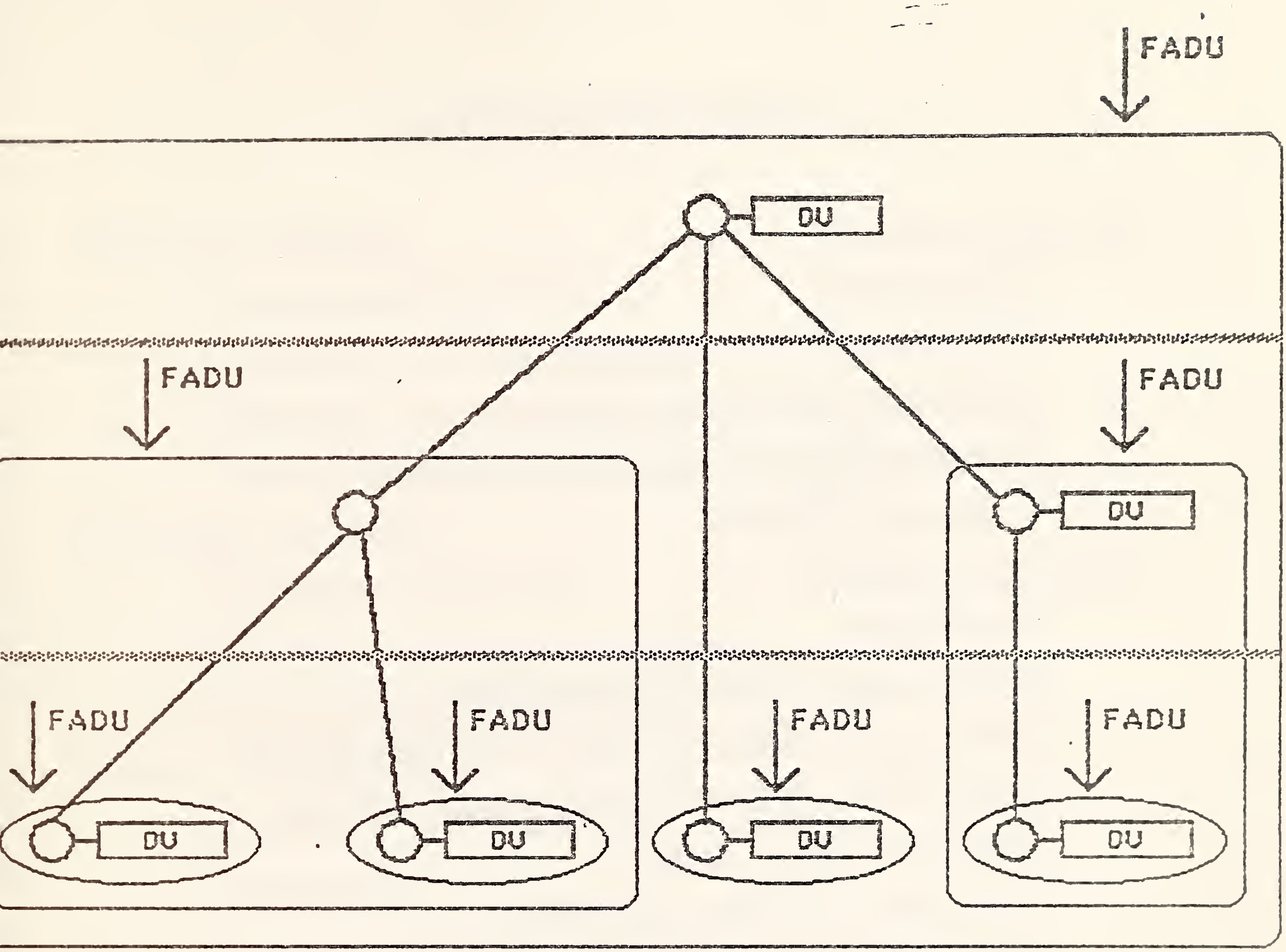
Files contain one or more *Data Units* possibly related in some fashion (e.g., sequential, network, relational, or hierarchical).

Virtual filestore provides a tree structure (called the *access structure* to represent the relation between Data Units.

Subtree of the access structure is known as a File Access Data Unit (FADU).

Essentially a hierarchical model.

Two special cases of access structure: unstructured files and flat files.



Access Structure
Using Tree Notation

File Attributes

- Kernel subset
    - Filename
    - Presentation context
    - Access structure type
    - Presentation structure name
    - Current filesize
- Storage subset
    - Account
    - Date and time of creation
    - Date and time of last modification
    - Date and time of last read access
    - Identity of creator
    - Identity of last modifier
    - Identity of last reader
    - File availability
    - Possible access type
    - Future filesize
- Security subset
    - Access control
    - Encryption name
    - Legal qualifications

Activity Attributes

- Kernel subset
  - Requested access
  - Location of initiator
  - Current access structure type
  - Current presentation context
- Storage subset
  - Current account
  - Current access context
  - Concurrency control
- Security subset
  - Identity of initiator
  - Password

THE FTAM SERVICE

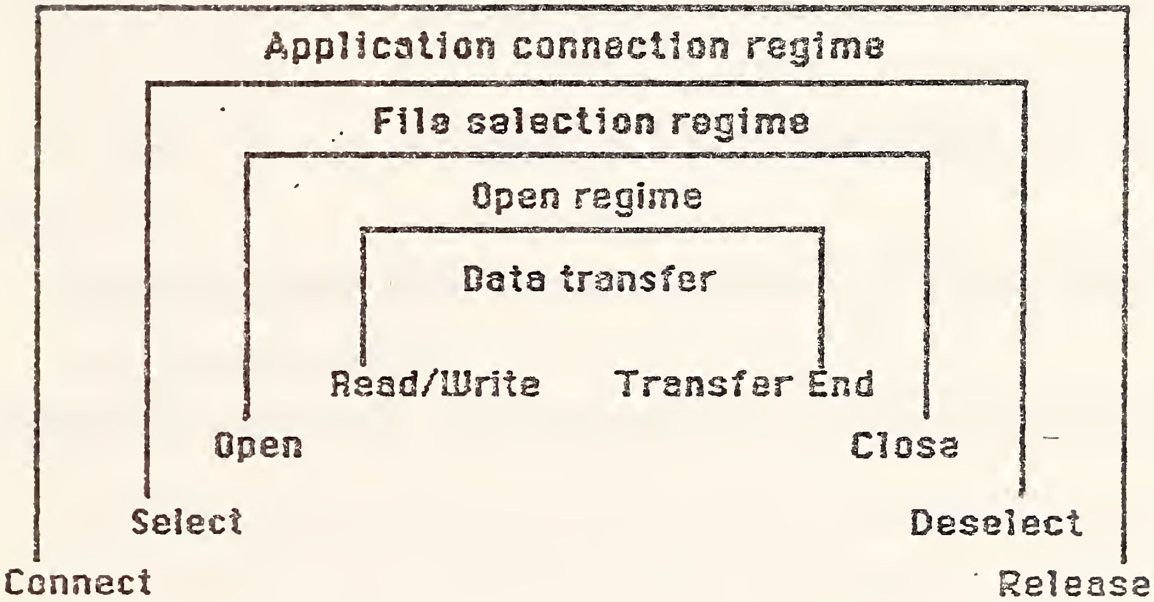
Based on establishing and disestablishing a series of *regimes*.

Entering regimes builds up the *operational context* of the entities step-by-step.

Asymmetric model

> Initiating and responding entities during application connection, file selection, and open regimes.
>
> Sending and receiving entities during data transfer regime.

Reliable and user-correctable services.



FTAM REGIME NESTING

Service Subsets

File Transfer Service Subset

File Access Service Subset

Limited File Management Service Subset

Enhanced File Management Service Subset

File Transfer Service Subset

CONNECT

Establish an application association with the specified FTAM entity.

SELECT

Select the file on which actions are to be performed.

OPEN

Open the selected file and negotiate the context in which its contents will be interpreted.

READ / WRITE

Establish the direction of the data transfer.

DATA

Transfer the data.

DATA_END

All data has been sent.

TRANSFER_END

Data transfer is complete.

CLOSE

Close the open file.

DESELECT

Release the selected file.

RELEASE

Release the application association.

CANCEL

Cancel the data transfer in progress.

ABORT

Release the application association unconditionally, abandoning any activity in progress.

BEGIN_GROUP

Indicate the start of a set of concatenated requests.

END_GROUP

Indicate the end of a set of concatenated requests.

File Access Service Subset

LOCATE

Locate the specified FADU.

ERASE

Erase the specified FADU.

Limited File Management Service Subset

CREATE

Create a new file with the specified attributes and select that file.

DELETE

Delete and deselect the selected file.

READ_ATTRIB

Obtain information about the selected file.

Enhanced File Management Service Subset

CHANGE_ATTRIB
Modify the attributes of the selected file.

Error Recovery Service Subset

RECOVER

Recreate the open regime following a failure.

CHECK

Mark / acknowledge transferred data.

RESTART

Interrupt data transfer and negotiate restart point.

THE FTAM PROTOCOL

Relies on underlying services of OSI Presentation Layer.

Uses OSI Session Layer pass-through services to provide checkpointing and recovery.

Currently provided are *basic* and *error recovery* protocol.

Basic protocol provides for the establishment and disestablishment of regimes and the movement of data.

Error recovery protocol provides a standard set of error recovery procedures.

Non-standard error recovery procedures may be implemented by using the user-correctable service.

FTAM Session Layer Requirements (First DP)

- Kernel functional unit
  - S_CONNECT (req, ind, resp, conf)
  - S_DATA (req, ind)
  - S_RELEASE (req, ind, resp, conf)
  - S_U_ABORT (req, ind)
  - S_P_ABORT (ind)
- Duplex functional unit
- Minor synchronize functional unit
  - S_SYNC_MINOR (req, ind, resp, conf)
  - S_TOKEN_GIVE (req, ind)
  - S_TOKEN_PLEASE (req, ind)
- Resynchronize functional unit
  - S_RESYNCHRONIZE (req, ind, resp, conf)

FTAM Protocol Data Units

FTAM protocol data units (PDUs) specified in notation based on that used in CCITT X.409.

More complex structures defined in terms of a set of primitive and constructor types (e.g, BOOLEAN, INTEGER, OCTET STRING, SET).

Rules for encoding the specified *abstract syntax* are independent of the abstract syntax itself.

At least one set of encoding rules will be specified by ISO.

```
FABORTrequest ::= SET {
    originator [0] INTEGER {
        fileServiceUserInitiated (0),
        fileServiceProvidedInitiated (1)},
    diagnostic Diagnostic}

Diagnostic ::= [APPLICATION 2] IMPLICIT SET {
    errorTypeIdentifier [0] ErrorTypeIdentifier,
    errorIdentifier [1] ErrorIdentifier,
    suggestedDelay [2] INTEGER OPTIONAL,
    furtherDetails CHOICE {
        humanReadable [3] ACharString,
        machineReadable [4] OCTET STRING} OPTIONAL}

ErrorIdentifier ::= INTEGER {
    noReasonProvided (0),
    mandatoryParameter (2),
    illegalParameterValue (3),
    unsupportedParameterValue (4)}

ErrorTypeIdentifier ::= INTEGER {
    success (0),
    warning (1),
    recoverableError (2),
    unrecoverableError (3)}
```

AREAS FOR FUTURE EXPANSION

Filestore and file management.

File access.

Manipulation of groups of files simultaneously.

FURTHER TUTORIAL MATERIAL

D. Lewan, and H.G. Long, "The OSI File Service," Proceedings of the IEEE, Volume 71, Number 12, pp. 1414-1419, December 1983.

P.F. Linington, "The Virtual Filestore Concept," Computer Networks, Volume 8, Number 1, pp. 13-16, February 1984.

Att. #28

GM FILE TRANSFER PROPOSAL
SEPTEMBER 6, 1984

1. UPGRADE NCC'84 FTP TO ISO FILE TRANSFER SERVICE SUBSET FOR INTERNET DEMO

   - REQ'D CHANGES FOR ISO COMPATIBILITY
   - SUPPORT FOR F_READ AND F_WRITE
   - BINARY AND TEXT FILES

2. COMPLETE ISO FTAM IMPLEMENTATON FOR LONGER RANGE TIME FRAME

   - FILE ACCESS SERVICE SUBSET
   - FILE MANAGEMENT SERVICE SUBSETS (LIMITED AND ENHANCED)
   - ERROR RECOVERY SERVICE SUBSET
   - VIRTUAL FILESTORE STORAGE SUBSET

PA 9/6/84

ISO FTP SUBSET FOR INTERNET DEMO

1.0 SERVICE PRIMITIVES

1. F_CONNECT, F_RELEASE, F_ABORT
2. F_SELECT, F_DESELECT
3. F_OPEN, F_CLOSE
4. F_READ AND F_WRITE *
5. F_DATA, F_DATA_END, F_TRANSFER_END
6. F_CANCEL
7. F_BEGIN_GROUP *, F_END_GROUP *

* INDICATES ADDITION

2.0 CHANGES REQUIRED FOR ISO COMPATIBILITY

1. X.409 ENCODING (ASN1) FOR FTP PDUs

   - NCC'84 FORMAT WAS INTERIM CHOICE TO BE COMPATIBLE WITH LOWER LAYERS

2. ADDITION OF CONCATENATION CONTROL

   - BEGIN, END GROUP PRIMITIVES AND SUPPORTING PCI

   - READ OR WRITE ACTIVITY INITIATED AS A SEQUENCE:

F_BEGIN_GROUP F_SELECT F_OPEN F_READ F_END_GROUP
F_BEGIN_GROUP F_SELECT F_OPEN F_WRITE F_END_GROUP

   - FILE RELEASED AS A SEQUENCE:

F_BEGIN_GROUP, F_CLOSE, F_DESELECT, F_END_GROUP.

3.0 ISO FTP SUBSET RESTRICTIONS

1. A FILE SELECTION REGIME MAY HAVE AT MOST ONE OPEN AND ONE READ OR WRITE ACTIVITY

2. ONLY COMPLETE FILES MAY BE TRANSFERRED

ISO FTAM IMPLEMENTATION DEMO

4.0 ADDITIONAL SERVICE PRIMITIVES

1. F_LOCATE, F_ERASE
2. F_CREATE, F_DELETE, F_READ_ATTRIBUTE
3. F_CHANGE_ATTRIBUTE

5.0 ERROR RECOVERY SERVICE PRIMITIVES

1. F_RECOVER
2. F_CHECK
3. F_RESTART
4. F_CANCEL

6.0 IMPLEMENTATION OF VIRTUAL FILESTORE

- DEFINITION OF FILE STRUCTURE
- ALLOWS "RECORD LEVEL" FILE ACCESS

Attachment #29(a)

OLIVETTI PROPOSAL FOR AN ISO FTAM DP IMPLEMENTATION

contact person: P. Bucciarelli
OLIVETTI
Via Jervis 77 10015 IVREA (TO) - ITALIA
tel +39-125-522566
tlx : 210030

INTRODUCTION

This document proposes a user Interface of a File Transfer and Management Service to be demonstrate at the ISO-NCC '85 and shows which subsets and options of the ISO FTAM DP must be selected in order to provide such a user visible service.
In order to avoid confusion between the realization and the standard itself, in the following we will use the term "FTF" for the facility and "FTAM DP" for the standard.
This proposal derives from a preliminary study and may contain some incompleteness. It mainly aims at minimizing the implementation efforts without penalizing a rich and remarkable user visible FTF definition.
The document is organized as follows:

sect.1 : USER INTERFACE OF FTF

sect.2 : SELECTION OF SUBSETS AND OPTIONS OF ISO FTAM DP

1. USER INTERFACE OF FTF

The FTF provides a set of services. The user Interface consists of a set of commands with related parameters. Each command is used to request a service and will be invoked from a terminal or from a program.
The following table shows the list of commands with the associated input parameters and the parameters returned by the FTF. A short description of each command and parameters is also given.

command	parameters	returns
SEND	local file name host name remote file name effect	transfer identifier return code
RECEIVE	local file name host name remote file name effect	transfer identifier return code
CREATE	as above (no "effect")	return code
CANCEL	transfer identifier	return code
DELETE	host name remote file name password(s)	return code
WAITA	transfer identifier time interval.	return code
STATUS	transfer identifier	return code

table 1 : FTT user interface

Commands meaning

SEND : to export a file from the filesystem where the user is, to a remote filesystem.

RECEIVE : to import a file from a remote file system to the file system which is local to the user

CREATE : to create an empty file on a remote file system

DELETE : to delete a file on a remote file system

CANCEL : to abort a SEND or RECEIVE operation which is in progress.

STATUS : to ask about the progress of a previously activated SEND or RECEIVE operation

Parameters meaning

The meaning of the parameters is explaned in the following list:

local file name — identifier of the local file, both in SEND and RECEIVE operation.

host name — name of the remote host involved in the operation.

remote file name — identifier of the remote file, both in SEND and RECEIVE operation.

effect   indication of the operation type which the user wants to execute on accessed file. The admitted destination effects are:

- . NEW — if the destination file does not exist, it will be created, with the specified file name, the contents and attributes of the source file, as a copy of the source. If the file exists, an error will be returned.
- . REPLACE — if the file exists, the entire file is rebuilt, according to source file organization, attributes and contents. If the file does not exist, an error is returned.

password(s) — key word for accessing the remote file

transfer identifier — identifier of the transfer request. it is assigned by the FTF to the SEND or RECEIVE user request. No algorithms are described for the assignement of the identifier, the only restriction being that it must be unique. it must be issued by the user for any other request concerning the same transfer (e.g. CANCEL, STATUS , etc.)

return code — result of the user request. The list of possible values of return codes is provided in

time interval — maximum time a user wants to be suspended waiting for a file transfer termination.

1.1. SERVICE LIMITATIONS

The above listed services will be provided to the user with the following limitations:

- The files are text files containing ASCII characters or binary files
- The files are unstructured.
- No recovery function is provided. If a crash occure during the transfer, the transfer will restart from the beginning and under the explicit user request. A recovery function which could be easily provided is the capability to re-establish the initial condition in case of a failure. That means that a file transfer is successfully terminated or does not leave any dangling situation. This proposal currently does not offer this capability ("ROLLBACK" parameter = NO ROLLBACK).
- A security mechanism is used only in case of "DELETE" a remote file. This is based on passwords.

2. DEFINITION OF SUBSETS AND OPTIONS OF ISO FTAM DP

In the following are described the subsets and the options of the ISO FTAM DP Services and Virtual File Store (parts III and II of the DP, respectively) which are necessary to support the FTF services proposed in Sect.1.
The FTAM DP Protocol subsets (part IV of the DP) will be directly determined by the choices we make on Services and Virtual File Store.
The session services required to support the selected subset of FTAM DP will be listed.
Note that in the following, the knowledge of the ISO FTAM DP is assumed and its terms are referred with no further explanation.

2.1. [illegible]

Service Type: "User correctable service"

Service subsets: "File Transfer subset" plus "Limited File Management".

These include the following service primitives:

F-CONNECT, F-RELEASE, F-ABORT
F-SELECT, F-DESELECT
F-OPEN, F-CLOSE,
F-READ, F-WRITE,
F-DATA, F-DATA-END, F-TRANSFER-END
* F-CANCEL, with diagnostic indicating non recoverable errors,
F-BEGIN-GROUP, F-END-GROUP.

F-CREATE
F-DELETE
F-READ-ATTRIB.

* Implementation of F-CANCEL can be avoided and replaced by F-ABORT. This conforms to the standard DP and does not imply limitations in the ~~entemot~~ services provided to final user.
The difference is that the F-CANCEL preserves the application connection, ~~and~~ while, after F-ABORT it has to be re-established.

4.1.1. [illegible]

Giving the FTP services we want to provide with the constraints listed in 1.1, the parameter of each ISO FTAM UP service primitives can be chosen and set as follows. The choices are reported directly on a copy of the ISO document and attached here (see ATTACHMENT 1)

4.2. [illegible]

Attribute subset: "kernel" subset

The attributes included and their values which can be established "a priori" are the following:

__file attributes__

filename
ISO 646 presentation context " 7-bit coded character set "
Unstructured access structure type
ISO646 presentation structure name
current filesize

__activity attributes__

requested access
location of initiator
current access structure type unstructured
current presentation context ISO 646

ATTACHMENT 1

(to OLIVETTI proposal for FTAM impl.)

The following pages are got from the ISO FTAM DP. They contain the service primitives of the two subsets which have been proposed, namely the "File Transfer subset" and the "Limited Management Subset".

Each service primitive has been updated by crossing the optional parameters which are not necessary in order to provide ~~at~~ the final user visible service described in Sect.1 of the OLIVETTI contribution.

F_CONNECT:

Parameter	F-CONNECT request	F-CONNECT indication	F-CONNECT response	F-CONNECT confirm	
Called Address	Mandatory	Mandatory			
Calling Address	Mandatory	Mandatory			
Responding Address			Mandatory	Mandatory	=
Service Type	Mandatory	Mandatory	Mandatory	Mandatory	= User correctable file service
Service Subsets	Mandatory	Mandatory	Mandatory	Mandatory	= Limited file mgt
~~Communication Quality of Service~~	~~Optional~~	~~Optional~~	~~Optional~~	~~Optional~~	
Rollback Availability	Optional	Optional	Optional	Optional	= MO rollback
Presentation Context	Optional	Optional	Optional	Optional	= ISO646

cont.

NOTE: the "optional" can be assumed as

~~Identity~~ of Initiator	~~Optional~~	~~Optional~~			
~~Current~~ Account	~~Optional~~	~~Optional~~			
Diagnostic			Optional	Optional	
~~Additional Parameters in the User Correctable Service~~					
~~Checkpoint Window~~	~~Optional~~	~~Optional~~	~~Optional~~	~~Optional~~	

9.1.2.1 Called Address The called address is the address used by the calling service user to identify the filestore to which the connection is to be established. The value is an address.

9.1.2.2 Calling Address The calling address is the address from which the connection is established. The value of this parameter is assigned to the "location of initiator" activity attribute associated with the connection. The value is an address.

9.1.2.3 Responding Address The responding address is the address which should be used in re-establishing the connection after failure. It is not necessarily textually identical to the Called Address. It may differ, for example, if generic addressing or redirection are in use. The value is an address.

9.1.2.4 Service Type The service type parameter takes the value "reliable service" or "user correctable service", depending on the service offered.

9.1.2.5 Service Subsets The service subsets parameter conveys the file service subset to be used (see clause 7). The value consists of two parts. The first indicates the set of optional service features which are required on this connection. The values may be Limited File Management, Enhanced File Management, Access or, for the user correctable service, Error Recovery. The second indicates the set of optional virtual filestore features which are required from the filestore provider on this connection. The values may be Storage or Security (see part II, clause 10). Either set may be empty, in which case the kernel subset is used.

~~9.1.2.6 Communication Quality of Service The communication quality of service parameter conveys the quality of service associated with the connection. On the request and indication primitives, it indicates the quality of service requested, and on the response and confirm primitives it indicates the quality of service achieved. The values taken by the communication Quality of Service parameter are defined in the Presentation Service definition.~~

9.1.2.7 Rollback Availability The rollback availability parameter indicates whether or not the file service user can support transaction rollback after failure. The values of the parameter are "no rollback" and "rollback available".

9.2 Application connection termination (orderly)

9.2.1 Function

An application connection may be terminated by an exchange of F-RELEASE primitives. This exchange is initiated by the initiating user. Such an exchange will not be completed until any actions previously requested have been completed.

9.2.2 Types of primitives and parameters

F-RELEASE

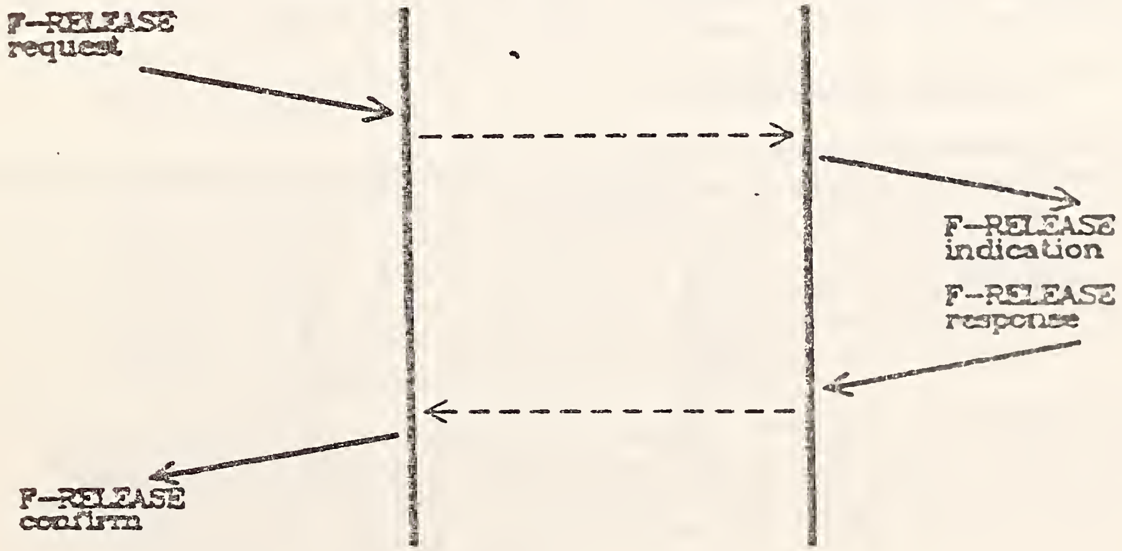
The following table indicates the types of primitive and the parameters needed for orderly connection termination.

Parameter	F-RELEASE request	F-RELEASE indication	F-RELEASE response	F-RELEASE confirm
~~Charging~~			~~Optional~~	~~Optional~~

~~9.2.2.1 Charging The charging parameter conveys information on the costs incurred during the connection. The value of the parameter is a list of triples; the elements of each triple are: a character string resource identifier, a character string charging unit and an integer charge value.~~

9.2.3 Sequence of primitives

The F-RELEASE request primitive can be issued by the file transfer initiator (the issuer of the F-CONNECT request) at any time after the receipt of an F-CONNECT confirmation primitive, providing no file is selected. The issue of an F-RELEASE request does not imply the success of any previous activity. Indications of success or failure are given on the completion of each activity. The sequence of events in a successfu orderly connection termination is as follows.



9.3 Application connection termination (abrupt)

9.3.1 Function

Either file service user may issue an F-ABORT request primitive at any time after an F-CONNECT request primitive has been issued, or an F-CONNECT indication primitive has been received. The F-ABORT primitives terminate the connection unconditionally, abandoning any file activity that was in progress and leaving the selected file in an undefined state. The users of the file service may agree that the file activity is to be rolled back in this circumstance. If error recovery is to be performed, the responsibility for initiating the recovery lies with the initiator. Once an F-ABORT request primitive has been issued, the connection will be terminated; the request cannot be rejected.

The filestore provider performs the close file and deselect file actions on receipt of an F-ABORT indication if the file is open, and the deselect file action if the file is closed but selected. However, no commitment semantics should be associated with such an automatic close.

9.3.2 Types of primitives and parameters

F. ABORT

The following table indicates the types of primitives and parameters needed for destructive connection termination.

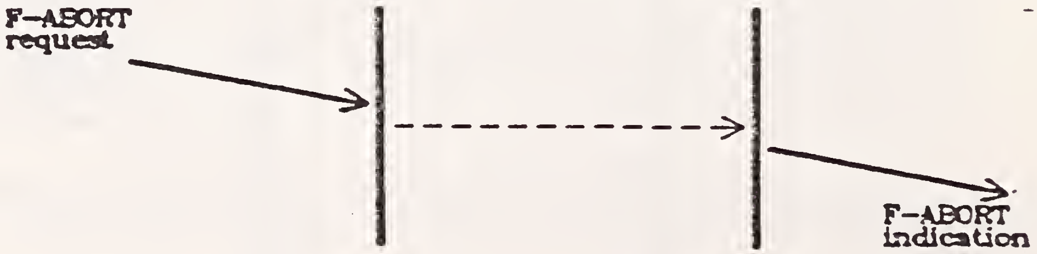
Parameter	F-ABORT request	F-ABORT indication
Originator		Mandatory
Diagnostic	Mandatory	Mandatory

9.3.2.1 Originator The originator parameter indicates the source of the termination. Its value indicates either File Service User or File Service Provider initiated termination.

9.3.2.2 Diagnostic The diagnostic parameter conveys the reason for the breakdown of the connection. The possible values for the diagnostic parameter are given in Annex A.

9.3.3 Sequence of primitives

The sequence of events in a user initiated abrupt termination is defined in the following time sequence diagram.



Parameter	F-SELECT request	F-SELECT indication	F-SELECT response	F-SELECT confirm
Filename	Mandatory	Mandatory	~~Optional~~	~~Optional~~
~~Attributes~~	~~Optional~~	~~Optional~~	~~Optional~~	~~Optional~~
~~Concurrency Control~~	~~Optional~~	~~Optional~~		
~~Access Control~~	~~Optional~~	~~Optional~~		
~~Access Passwords~~	~~Optional~~	~~Optional~~		
~~Access Structure Name~~	~~Optional~~	~~Optional~~		
~~Accounting~~	~~Optional~~	~~Optional~~		
Diagnostic			Optional	Optional

10.1.2.1 Filename The filename parameter identifies the file selected. In the request and indication primitives it indicates the file required, and in the response and confirm primitives it indicates the file actually selected. If, for example, the filename requested gave a generic name or a generation name, the name selected might differ from that requested.

~~10.1.2.2 Attributes The attributes parameter provides attribute values for use in identifying the file to be selected. The attributes which may be referenced and the range of values they may take are defined in the virtual filestore definition (Part II of this standard).~~

~~10.1.2.3 Concurrency Control The concurrency control parameter indicates the relation of this selection to other activity on the same file. The value is a vector whose elements indicate, for each of the classes of access control (see 10.1.2.4) whether the access is shared or exclusive.~~

~~10.1.2.4 Access Control The access control parameter indicates the basis on which the file is being selected. The value gives as a vector the actions to be performed during the selection. The elements of the vector correspond to the read, insert, replace, erase, extend, read attribute, change attribute and delete file actions, and each element indicates whether the action is required or not. The value of the parameter determines the value of the requested access activity attribute associated with the connection.~~

~~10.1.2.5 Access Passwords The access passwords parameter provides passwords associated with the actions specified in the access control parameter. The value of the parameter determines the value of the password activity attribute, and the range of values the parameter may take is equal to that defined for the attribute.~~

10.2.2 Types of primitives and parameters F-DESELECT

The following table indicates the types of primitives and the parameters needed for file selection.

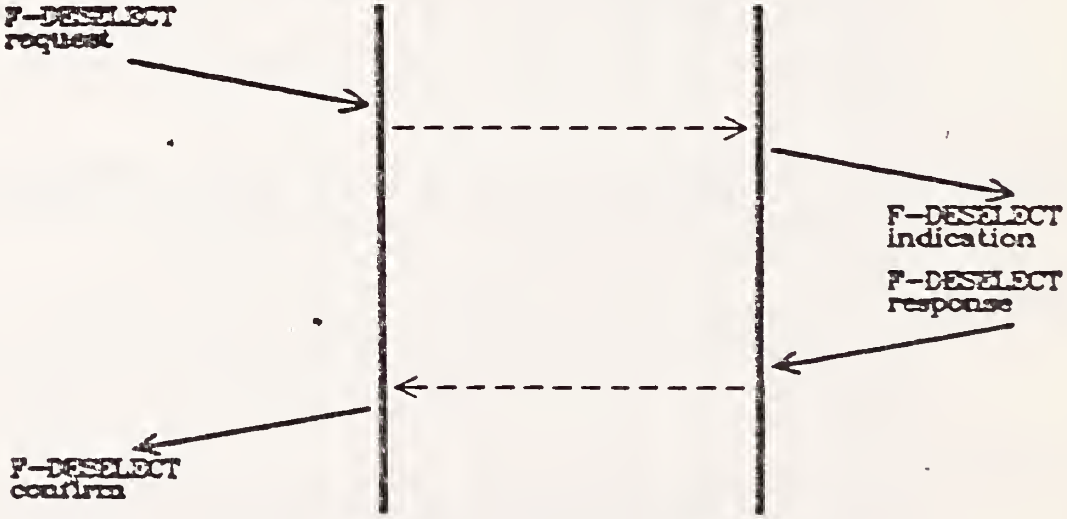
Parameter	F-DESELECT request	F-DESELECT indication	F-DESELECT response	F-DESELECT confirm
~~Charging~~			~~Optional~~	~~Optional~~
Diagnostic			Optional	Optional

10.2.2.1 Charging The charging parameter conveys information on the costs incurred during this file selection. The form of the parameter value is specified in clause 9.2.2

10.2.2.2 Diagnostic The diagnostic parameter indicates the reason for any failure. The possible values for the diagnostic parameter are given in Annex A. The selection regime is terminated whatever the value of the diagnostic parameter.

10.2.3 Sequence of primitives

The sequence of events in a successful deselection is defined in the following time sequence diagram.



10.3 File creation

10.3.1 Function

The F-CREATE primitives cause a file to be created and establish a selection of the newly created file. They may only be used if there is no currently selected file.

The filestore provider performs the create file action after receiving the F-CREATE indication primitive, and before issuing the F-CREATE response primitive with a diagnostic parameter indicating success.

10.3.2 Types of primitives and parameters

F-CREATE

The following table indicates the types of primitives and the parameters needed for file creation.

see File attr. choice

Parameter	F-CREATE request	F-CREATE indication	F-CREATE response	F-CREATE confirm
Filename	Mandatory	Mandatory	~~Optional~~	~~Optional~~
Attributes	Optional	Optional	Optional	Optional
~~Concurrency~~ Control	~~Optional~~	~~Optional~~		
~~Access~~ Control	~~Optional~~	~~Optional~~		
~~Access~~ Passwords	~~Optional~~	~~Optional~~		
~~Accounting~~	~~Optional~~	~~Optional~~		
Diagnostic			Optional	Optional

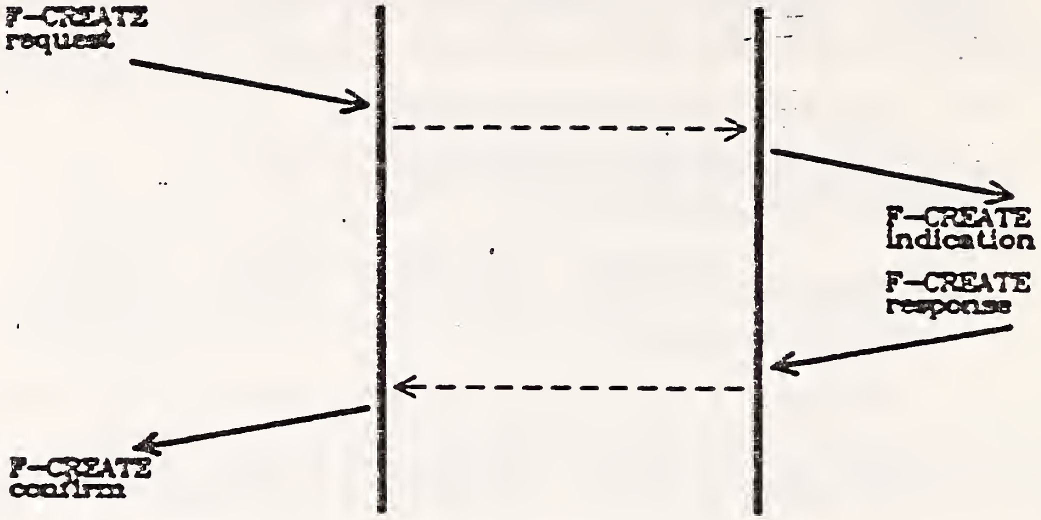
10.3.2.1 Filename, Concurrency, Control, Accounting and Diagnostic The filename, concurrency control, accounting and diagnostic parameters are defined in clause 10.1.2.

10.3.2.2 Attributes The attributes parameter gives a list of attribute names and values to be associated with the newly created file. The attributes are defined in the virtual filestore definition (Part II of this standard). The attributes which may be set by these primitives are listed in Annex B to this Part.

~~10.3.2.3 Access Control and Access Password The access control attribute is defined in clause 10.1.2. The access password attribute has elements matching those of the access control attribute, plus an additional element which may be required by the filestore provider to permit the specified file to be created.~~

10.3.3 Sequence of primitives

The sequence of events in a successful creation is defined in the following time sequence diagram. An F-CREATE indication is rejected by using an F-CREATE response with a diagnostic parameter with an error type more severe than warning.



10.4 File deletion

10.4.1 Function

The F-DELETE primitives release an existing selection in such a way that the selected file ceases to exist, and is not available for reselection. The primitives may only be issued while a file is selected. Even if the deletion fails, the file is deselected.

The filestore provider performs the delete file action after receiving the F-DELETE indication, and before issuing the F-DELETE response primitive. The delete file action can be performed only if the initiating entity has the "delete file" access control permission (see clause 10.1.2.4).

10.4.2 Types of primitives and parameters F-DELETE

The following table indicates the types of primitives and the parameters needed for file deletion.

Parameter	F-DELETE request	F-DELETE indication	F-DELETE response	F-DELETE confirm
~~Access~~ Control	~~Optional~~	~~Optional~~		
Access Password	Optional	Optional		
~~Charging~~			~~Optional~~	~~Optional~~
Diagnostic			Optional	Optional

10.4.2.1 Access Control, Access Password The access control and access password parameters are a subset of the parameters defined in clause 10.1.2, having only the element controlling the "delete file" action.

Parameter	F-READ-ATTRIB request	F-READ-ATTRIB indication	F-READ-ATTRIB response	F-READ-ATTRIB confirm
Attribute Names	Mandatory	Mandatory		
Attributes			Mandatory	Mandatory
Diagnostic			Optional	Optional

See Attr.

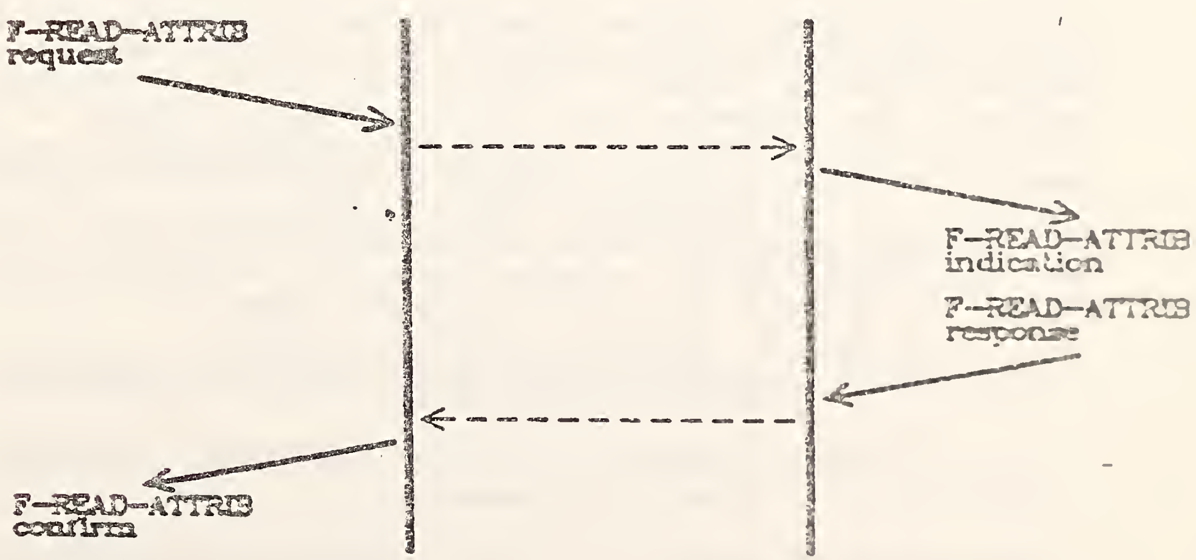
11.1.2.1 Attribute Names The attribute names parameter indicates which attributes from the set given in the virtual filestore definition are to be read. The parameter may indicate "all attributes" or it may be a list, each element of which names an attribute defined in part II of this standard.

11.1.2.2 Attributes The attributes parameter returns a list of the names and values of the requested attributes. The values may either be a value defined in part II of this standard or an indication that no value is available.

11.1.2.3 Diagnostic The diagnostic parameter indicates the success or failure of the operation, and the reason for any failure. The possible values for the diagnostic parameter are given in Annex A.

11.1.3 Sequence of primitives

The sequence of events in a successful reading of attributes is defined in the following time sequence diagram.



Parameter	F-OPEN request	F-OPEN indication	F-OPEN response	F-OPEN confirm	
Processing Mode	Mandatory	Mandatory			= read | replace
Access Context	Optional	Optional			
Presentation Context	Optional	Optional	Optional	Optional	?
~~Concurrency Control~~	~~Optional~~	~~Optional~~	~~Optional~~	~~Optional~~	
~~Commitment Control~~	~~Optional~~	~~Optional~~			
Diagnostic			Optional	Optional	

<u>Additional parameters in the user correctable service</u>

~~Activity~~ Identifier	~~Optional~~	~~Optional~~		
~~Recovery~~ Mode	~~Optional~~	~~Optional~~	~~Optional~~	~~Optional~~

12.1.2.1 <u>Processing Mode</u> The processing mode parameter indicates the possible operations to be performed as a result of data transfer requests; this determines the filestore actions which the filestore entity can perform. The parameter value indicates whether F-READ or F-WRITE primitives are to be permitted, and, for the F-WRITE primitive, whether the data unit operations "replace", "insert" and "extend" are to be permitted.

12.1.2.2 <u>Access context</u> The access context parameter specifies a view of the file content access structure which is to be used during this open regime. The parameter value is one of the four views:

- access context 1: access to all DUs within each FADU, together with all structuring information.

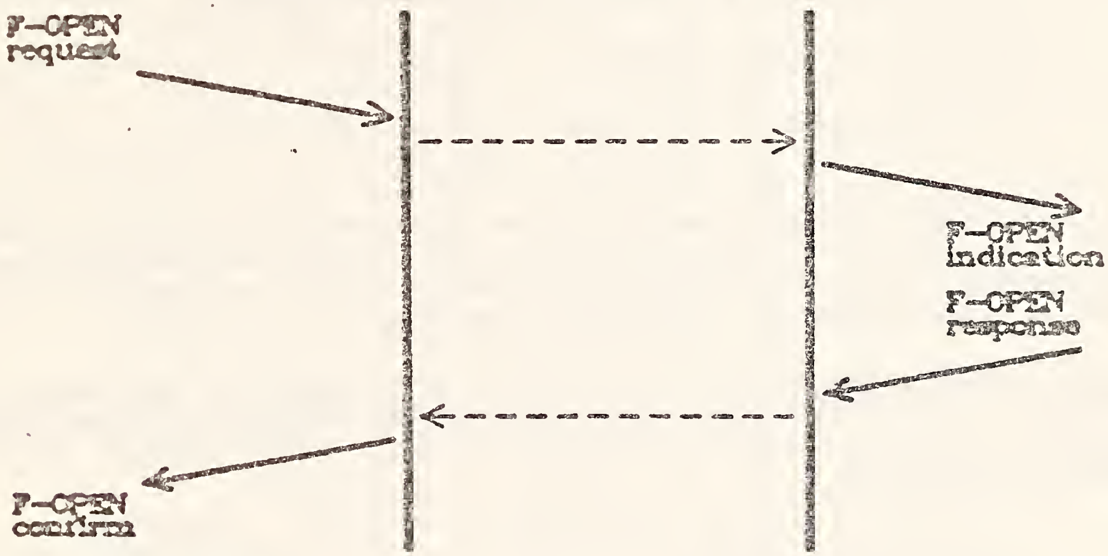
- access context 2: access to all DUs within each FADU, but without any structuring information.

- access context 3: access only to the DU associated with the root of each FADU

- access context 4: access to all the DUs in a given level of the addressed FADU, but without any structuring information.

12.1.3 Sequence of primitives

The sequence of events in a successful open is defined in the following time sequence diagram. An F-OPEN indication is rejected by use of an F-OPEN response with a diagnostic parameter indicating failure.



12.2 File close

12.2.1 Function

The F-CLOSE primitives release an existing file open regime. Once a close procedure has been initiated, the file will be closed; the request cannot be rejected.

The filestore provider performs the close file action after receiving the F-CLOSE indication primitive, and before issuing the F-CLOSE response primitive.

12.2.2 Types of primitives and parameters F-CLOSE

The following table indicates the types of primitives and the parameters needed for file closing.

Parameter	F-CLOSE request	F-CLOSE indication	F-CLOSE response	F-CLOSE confirm
Diagnostic	Optional	Optional	Optional	Optional

12.2.2.1 Diagnostic The diagnostic parameter indicates the reason for any failure. The possible values for the diagnostic parameter are given in Annex A. The close regime is terminated notwithstanding the value of the diagnostic parameter. Use of the diagnostic parameter on the request allows the file service initiator to cause rollback of the activity.

13.1.2 Types of primitives and parameters

F-DATA

The following table indicates the types of primitives and the parameters needed for data item transfer.

Parameters	F-DATA request	F-DATA indication
F-Data Item Type	Mandatory	Mandatory (=)
F-Data Item Value	Mandatory	Mandatory (=)

13.1.2.1 F-Data Item Type The F-Data Item Type parameter indicates, for the data item transferred, the data item type and thus the set of values which are possible. The F-data item type must be within the set implied by the value of the presentation context file attribute.

13.1.2.2 F-Data Item Value The F-Data Item Value parameter indicates, for the data item transferred, the value taken by the data item in this instance. The value must be within the set of permitted values for the specified data type.

13.2 End of data transfer

13.2.1 Function

The completion of the data transfer is indicated by the F-DATA-END primitives. The sender issues an F-DATA-END request primitive when it has sent all the necessary data; receipt of the F-TRANSFER-END indication or confirm as appropriate informs the sender that no further error recovery actions will be requested.

13.2.2 Types of primitives and parameters

F-DATA-END

The following table indicates the types of primitives and the parameters needed to end a data transfer.

Parameter	F-DATA-END request	F-DATA-END indication
Diagnostic	Optional	Optional

13.2.2.1 Diagnostic The diagnostic parameter indicates the success or failure of the data transfer, and the reason for any failure. It also indicates whether a failure is amenable to recovery or not. The possible values of the diagnostic parameter are listed in Annex A.

13.4 Write file access data unit

13.4.1 Function

The F-WRITE group of primitives specifies a data transfer from the file service initiator to the filestore provider. A file must previously have been opened, and only one F-WRITE procedure may be in progress at any time. The file store provider performs the locate file access data unit action before issuing the F-WRITE response, and subsequently performs the insert, replace or extend actions as data is received depending on the data unit operation specified. The direction of data flow established continues until the exchange of F-TRANSFER-END primitives. An F-WRITE indication can be rejected by including a diagnostic parameter with error type more severe than a warning in the response. If the transfer is rejected, no data transfer takes place and the file is not changed.

13.4.2 Types of primitives and parameters F. WRITE

The following table indicates the types of primitives and the parameters needed for a write interaction.

Parameter	F-WRITE request	F-WRITE indication	F-WRITE response	F-WRITE confirm
File Access Data Unit Operation	Mandatory	Mandatory		
File Access Data Unit Identity	Mandatory	Mandatory		
~~Concurrency Control~~	~~Optional~~	~~Optional~~		
Diagnostic			Optional	Optional
Additional parameters of the user correctable service				
~~Recovery~~ Point			~~Optional~~	~~Optional~~

replace/insert

13.4.2.1 File Access Data Unit Operation The file access data unit operation parameter indicates the action to be taken by the filestore provider on receipt of the data transferred. Possible values are "replace", "insert" or "extend".

13.4.2.2 File Access Data Unit Identity The file access data unit identity parameter gives the identity of the file access data unit to (or from) which the transferred data is to be associated. The value of the parameter is either:

a) "first" or "last", in terms of the preferred traversal sequence for the structure,

13.5.2 Types of primitives and parameters F-READ

The following table indicates the types of primitives and the parameters needed for a read file access data unit interaction.

Parameter	F-READ request	F-READ indication	F-READ response	F-READ confirm
File Access Data Unit Identity	Mandatory	Mandatory		
~~Concurrency Control~~	~~Optional~~	~~Optional~~		
Diagnostic			Optional	Optional
Additional parameters of the user correctable service				
~~Recovery~~ Point	~~Optional~~	~~Optional~~		

13.5.2.1 Data Unit Identity, Concurrency Control, Diagnostic, Recovery Point The data unit identity, concurrency control and recovery point parameters are as defined in clause 13.4, except that the data is being read from the filestore, and not written into it.

13.6 Erase file access data unit

13.6.1 Function

The F-ERASE group of primitives specifies the identity of a file access data unit which is to be erased by the filestore provider.

The filestore provider performs the erase action after receiving the F-ERASE indication, and before issuing the F-ERASE response primitive.

13.6.2 Types of primitives and parameters F. ERASE

The following table indicates the types of primitives and the parameters needed for an erase interaction.

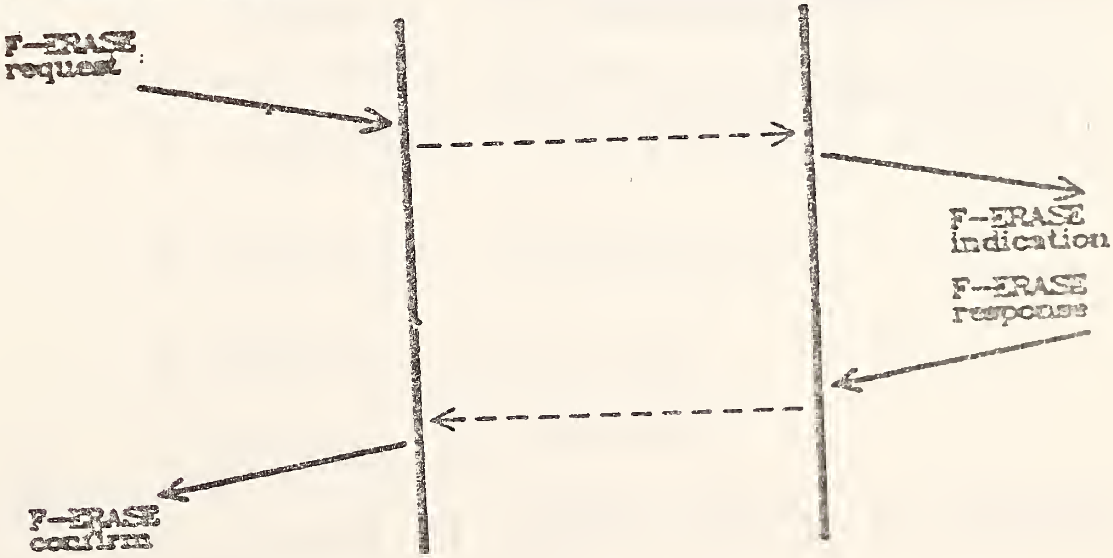
Parameter	F-ERASE request	F-ERASE indication	F-ERASE response	F-ERASE confirm
File Access Data Unit Identity	Mandatory	Mandatory (s)		
~~Concurrency~~ Control	~~Optional~~	~~Optional~~		
Diagnostic			Optional	Optional

13.6.2.1 File Access Data Unit Identity, Concurrency Control, Diagnostic

The file access data unit identity, concurrency control and diagnostic parameters are defined in clause 13.4.

13.6.3 Sequence of primitives

The sequence of events in a successful erase is defined by the following time sequence diagram.



13.7 End of transfer

13.7.1 Function

The completion of a transfer is indicated by an exchange of F-TRANSFER-END primitives. This exchange is initiated by the intiator after having issued or received an F-DATA-END primitive. After issuing or receiving the F-DATA-END primitive the F-TRANSFER-END request primitive with a diagnostic parameter more severe than a warning must be used if the transfer is to be rejected.

13.7.2 Types of primitives and parameters

F.TRANSFER.END

The following table indicates the types of primitives and the parameters needed for data transfer ending.

Parameter	F-TRANSFER -END request	F-TRANSFER -END indication	F-TRANSFER -END response	F-TRANSFER -END confirm
~~Commitment~~ Control			~~Optional~~	~~Optional~~
Diagnostic	Optional	Optional	Optional	Optional

13.7.2.1 Commitment Control ~~The commitment control parameter allows the users of the service to signal information relating to any commitment structure of which the files activity may form a part. The value of the parameter is a string (see Annex C).~~

13.7.2.2 Diagnostic The diagnostic parameter allows qualifications, such as readiness to commit, to be signalled at the end of the data transfer. It also indicates the reason for any failure. The possible values for the diagnostic parameter are given in Annex A.

13.7.3 Sequence of primitives

The sequence of events in a successful transfer end is defined by the time sequence diagrams in clauses 13.8 and 13.9.

CHOICE OF
ISO FTAM <u>SERVICES</u>

TYPE : "USER CORRECTABLE"

SUBSETS : "FILE TRASTER"

+

"LIMITED FILE MANAG."

i.e. : F.CONNECT , F.RELEASE, F.ABORT

F.SELECT , F.DESELECT,

F.OPEN, F.CLOSE,

F.READ, F. WRITE,

$ (Scie) $ F.DATA, F.DATA END, F-TRANSF. END,

F-CANCEL*, F.BEGIN & END GROUP,

MGT { F.CREATE
F-DELETE
F-READ-ATTRIBUTES

13.10 Cancelling data transfer

13.10.1 Function

Either of the service users may cancel a data transfer activity by issuing an F-CANCEL request primitive. The F-CANCEL primitive may be issued at any time after the issue or receipt of an F-READ or F-WRITE response or confirm and before the issue or receipt of an F-DATA-END request or indication. After an F-CANCEL procedure the two users may have different views of the state of the activity. The F-CANCEL primitives interrupt any activity in progress (including an F-RESTART sequence) and any undelivered indications or confirms may be discarded. The file remains open after a sequence of F-CANCEL primitives, although the result of interrupted operations is not defined. Further F-READ or F-WRITE operations, not related to any previous read or write attempt, may be attempted after the completion of the sequence of F-CANCEL primitives has disposed of any previous activity.

13.10.2 Types of primitives and parameters

F. CANCEL

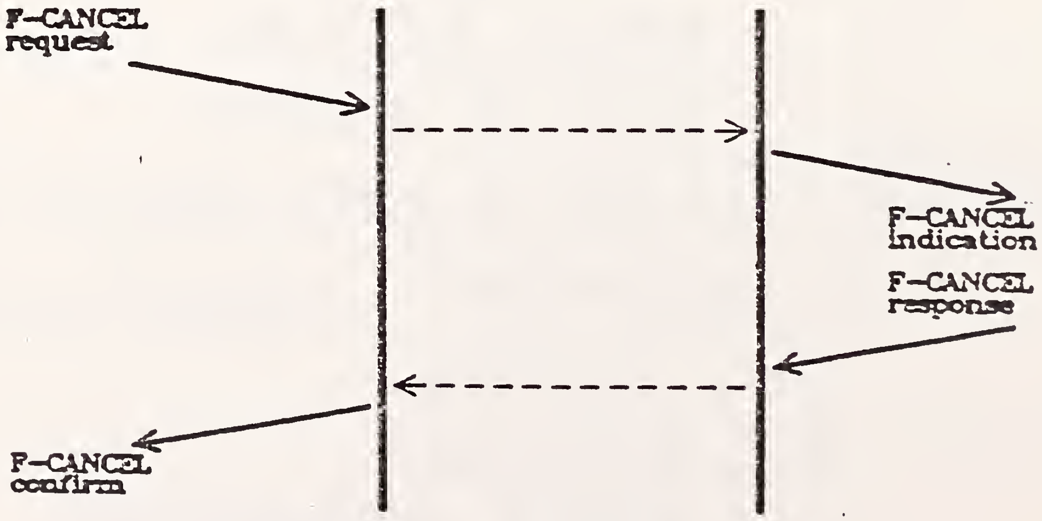
The following table indicates the types of primitives and the parameters needed to cancel data transfer.

Parameter	F-CANCEL request	F-CANCEL indication	F-CANCEL response	F-CANCEL confirm
Diagnostic	Mandatory	Mandatory	Mandatory	Mandatory

13.10.2.1 Diagnostic The diagnostic parameter indicates the reason for the cancellation. In the user correctable service, it also indicates whether the transfer may be recovered or not. The possible values for the diagnostic parameter are listed in Annex A.

13.10.3 Sequence of primitives

The sequence of events for a successful cancel procedure is defined in the following time sequence diagram.



Att. #29(5)-2

CHOICE OF VIRTUAL FILE

SUBSET : "KERNEL"

i.e. :

file attributes : . FILENAME

.

: see DP

activity attributes : .

: see DP

PROTOCOL : allign to ISO encoding (X.409)

pB 9.b.Ay

Att. #29(b)-3

LIMITATIONS

- TEXT or BINARY FILES
- UNSTRUCTURED
- NO RECOVERY
  (maybe rollback?)
- NO FILE ACCESS

PB 9/6/89

Att. #29(b)-4

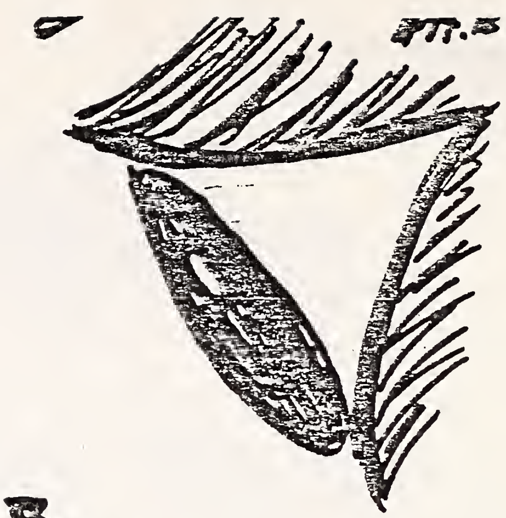
proposal:

1. AGREE ON USER VISIBLE SERVICES

↓

2. SELECT THE SUBSETS OF FTAM D.P. NECESSARY TO PROVIDE THEM

PB 9/2/89



- SEND
- RECEIVE
- CREATE
- DELETE
- CANCEL
- STATUS

(• WAITA)

← user visibility

PB 9/3/79

Att. #30-1

MAJOR COMPONENTS

- VIRTUAL FILESTORE MAPPING
- MAPPING TO UNDERLYING SERVICE(SESSION, DUMMY PRESENTATION)
- FILE SERVICE INTERFACE
- FILE PROTOCOL ENGINE

SUBSETTING

- FILE TRANSFER SERVICE AND PROTOCOL
- FILE ACCESS SERVICE AND PROTOCOL
- LIMITED FILE MANAGEMENT SERVICE AND PROTOCOL
- ENHANCED FILE MANAGEMENT AND PROTOCOL
- ERROR RECOVERY SUBSET OF USER CORRECTABLE FILE SERVICE

Roy Cadwallader 9/2/84

AH. #30-2

ISO FILE TRANSFER, ACCESS AND MANAGEMENT(FTAM)

PROVIDES

- o FILE TRANSFER
- o FILE CREATION
- o FILE DELETION
- o MANIPULATION OF DATA WITHIN FILES
- o MANAGEMENT OF INDIVIDUAL FILES

CURRENT FTAM SPECIFICATION

- o SCHEDULED FOR REVISION IN OCTOBER, BALLOT IN JANUARY

Rov Cadwallader 9/6/84

Att. #31

FTAM

OLIVETTI PROPOSAL 16/1

TO BE REVIEWED IN NOV. FOR
COMPLIANCE TO ISO.

Att. #32

UNION OMITS

NEGOTIATED RELEASE
EXPEDITED DATA
TYPED DATA
CAPABILITY DATA
MAJOR SYNCHRONIZE

ALSO OMIT
EXTENDED CONCATENATION

KM 9/7/84

USE OF SESSION

S_CONNECT

S_DATA (FULL DUPLEX)

S_RELEASE 19/0

S_U_ABORT

S_P_ABORT

NO SYNC

RESYNC

TOKEN MGT

= basic combined subset

P.B 9/6/84

Att. #34

Session Protocol

Proposal

Y Rec: Use of Transport Expedited

N • Segmenting For Normal & Typed Data

N • Reuse of TC

N • Concatenation of SPDUs Into TSDUs

n/2

KM 9/7/84

Att.# 35(a)

LAYER 3 - INTERNET: TIME, RESOURCES ESTIMATES

o WHAT

- IP: DEMO SUBSET OF CONFORMANCE SUBSET (NO TYPE 2 OR TYPE 3 FUNCTIONS I.E., NO SECURITY, SOURCE ROUTING, PRIORITY, ROUTE RECORDING, QOS, OR PRIORITY FUNCTIONS)

COMPLETE: 2/1/85 (INSTALLED AND RUNNING, READY FOR PROTOCOL TESTING BY NBS)

- IP TEST BED

ARCHITECTURES

TEST CASES

- TEST BED ARCHITECTURES

  ENCODER/DECODER

  REFERENCE IMPLEMENTATION (RI) WITH TEST HARNESS

- ENCODER/DECODER

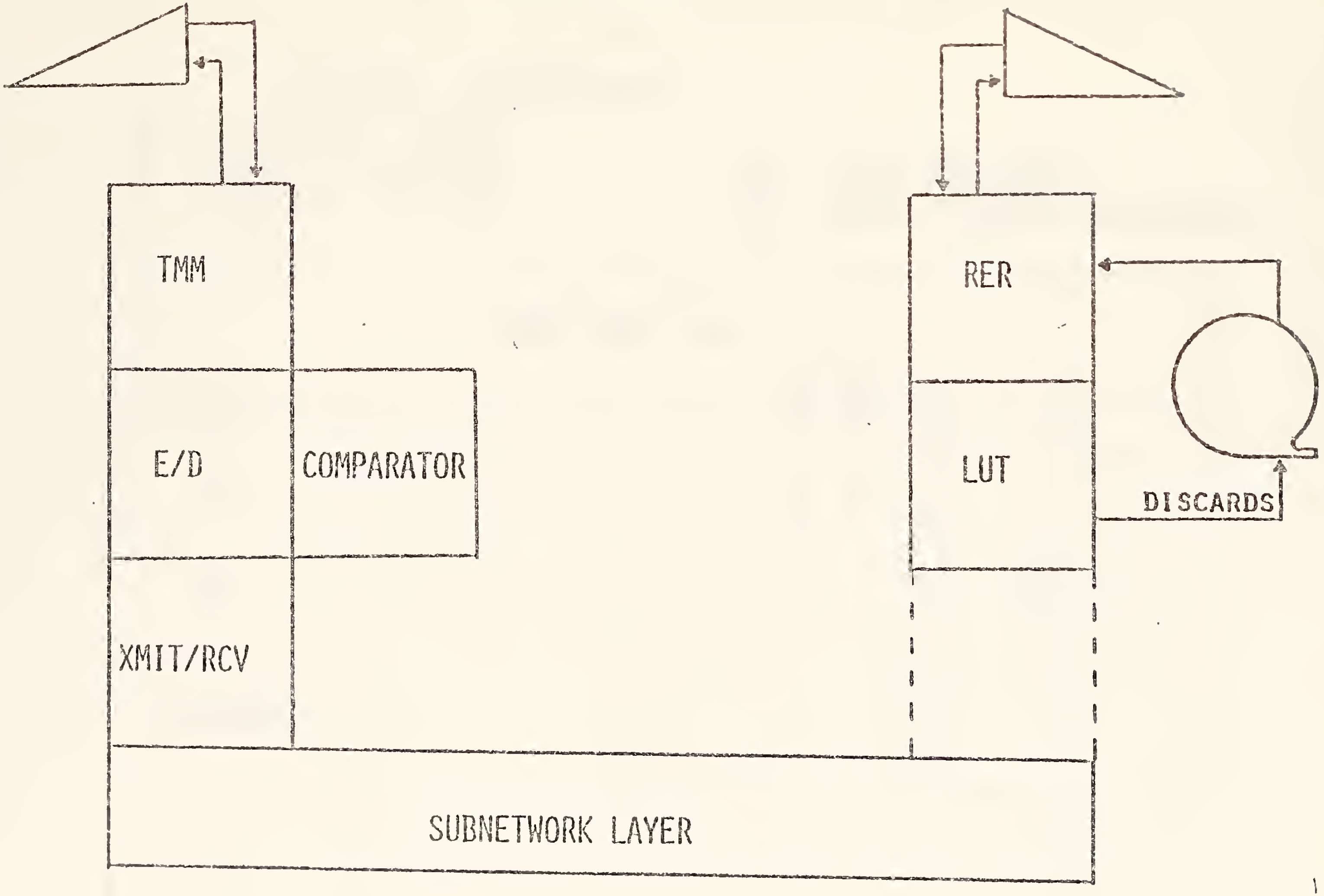
  CAN CONSTRUCT, RECOGNIZE ALL TYPE OF VALID IPDUs

  CAN CONSTRUCT, RECOGNIZE ERRONEOUS IPDUs

- RI WITH TEST HARNESS (SCENARIO INTERPRETER AND TRANSPORT OVER IP)

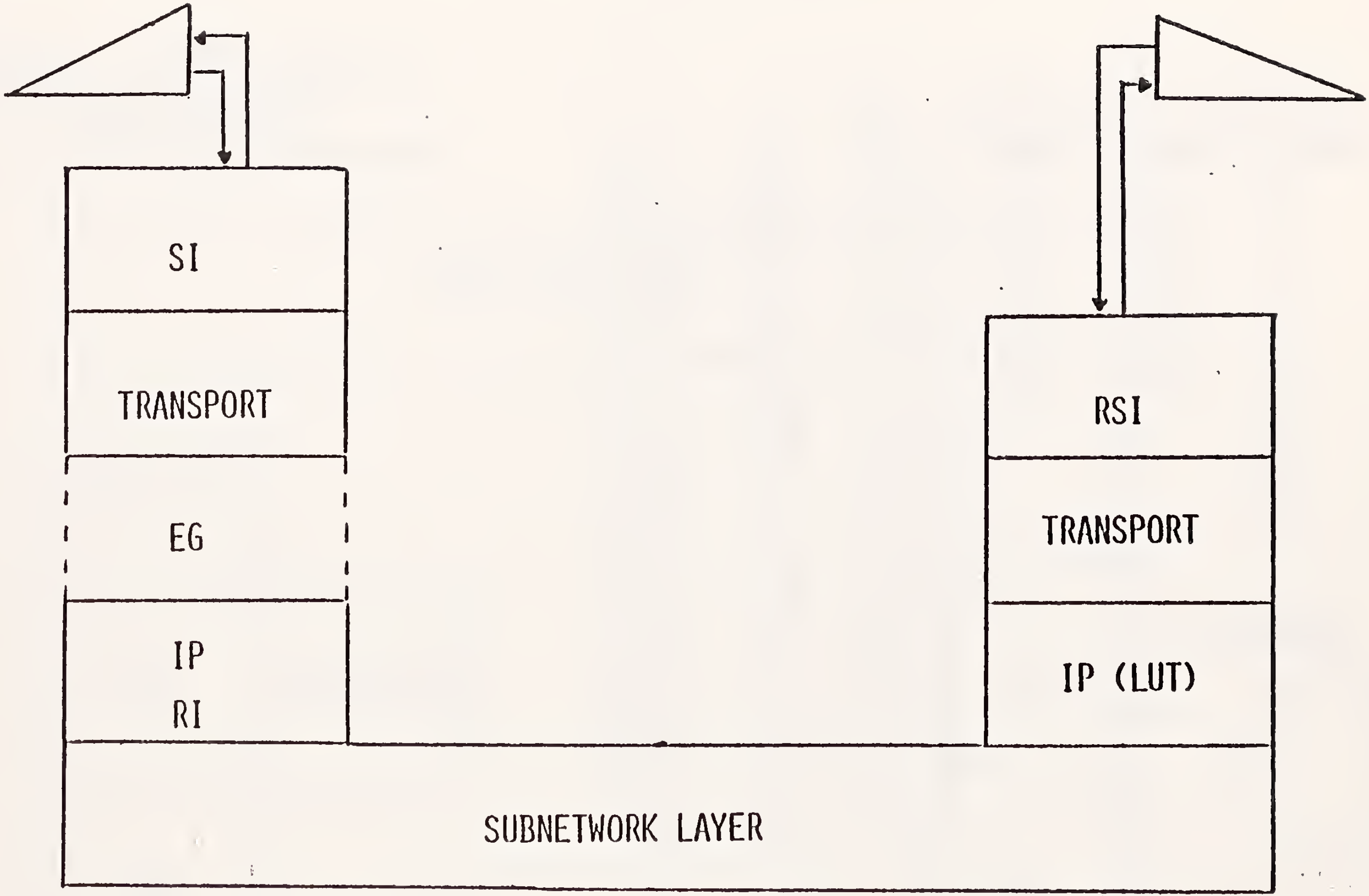
  CAN CONSTRUCT A SUBSET OF VALID PDUs (ONLY THOSE INDUCIBLE VIA THE IP SERVICE INTERFACE)

ENCODER/DECODER



TMM - TEST MANAGEMENT MODULE
E/D - ENCODER/DECODER
XMIT/RCV - SENDER/RECEIVER

RER - REMOTE ECHOER/RESPONDER
LUT - LAYER UNDER TEST (IP)



SI - SCENARIO INTERPRETER
EG - EXCEPTION GENERATOR
IP RI - INTERNET PROTOCOL
REFERENCE IMPLEMENTATION

RSI - REMOTE SCENARIO INTERPRETER
LUT - LAYER UNDER TEST

LAYER 3 TIME, RESOURCE ESTIMATES

- ACCURACY: PROBLEMATIC
- 3 SECTIONS:
  - TASK AND CRITICAL PATH TIME
  - ASSUMPTIONS
  - LIKELY SLIPPAGE

RI WITH TEST HARNESS

	TIME-TO-COMPLETE (PM)			
ITEM	DESIGN, SPECIFY	CODE, UNIT TEST	CRITICAL PATH TIME	CRITICAL PATH COMPLETE DATE
MODIFY INTERFACES FOR IP	1	1	2	10/31/84
MODIFY PDU LOG ANALYSIS TOOLS	1	1	2	
TEST CASES	1		1	10/31/84
PORT TEST HARNESS TO 32-BIT ENVIRONS			2	12/31/84
INTEGRATION TESTING			1	2/28/85
INTERNET PROTOCOL TESTING			1	3/31/85
INSTALLATION TESTING, DISTRIBUTION TO TEST CENTERS, FIELD TESTING			1	4/30/85

RI WITH TEST HARNESS

o ASSUMPTIONS:

1) MAXIMUM LEVEL OF STAFFING (AT LEAST 1 STAFF PERITEM)

2) MAXIMUM CONCURRENCY

3) START DATE 9/4/84

4) AVAILABILITY OF NBS VAX DEVELOPMENT SYSTEM BY START DATE (EARLIEST DELIVERY DATE: OCTOBER '84, EARLIEST AVAILABLE DATE: NOVEMBER '84)

5) SEPARATE DEVELOPMENT PATH FOR INTERNET WITH IP INSTALLED AND RUNNING (READY FOR PROTOCOL TESTING) BY 2/1/85

6) SEPARATE DEVELOPMENT PATHS FOR EXTENDING, UPDATING TRANSPORT AND FOR FTP; SEPARATE TESTING OF MODIFIED TRANSPORT AND OF FTP.

o LIKELY SLIPPAGE:

3 MONTHS (FOR ASSUMPTION #4) TO 7/31/85.

ENCODER/DECODER

	TIME-TO-COMPLETE (PM)			
	DESIGN, SPECIFY	CODE, UNIT TEST	CRITICAL PATH TIME	CRITICAL PATH COMPLETE DATE
TMM	2	2	4	12/31/84
E/D	2	2	4	
COMPARATOR	2	2	4	
XMIT/RCV	2	2	4	
LOG ANALYZERS	2	2	4	
RER	2	2	4	
USER COMMAND LANGUAGE	1			
TEST CASES	1			
INTEGRATION TESTING			1	1/31/85
IP TESTING			1	2/28/85
INSTALLATION TESTING, DISTRIBUTION TO TEST CENTERS, FIELD TESTING			1	3/31/85

ENCODER/DECODER

- ASSUMPTIONS:

  1) MAXIMUM LEVEL OF STAFFING (AT LEAST 1 STAFF PER ITEM).

  2) MAXIMUM DEGREE OF CONCURRENCY

  3) START DATE: 9/4/84

  4) AVAILABILITY OF NBS VAX DEVELOPMENT SYSTEM BY START DATE (EARLIEST DELIVERY DATES OCTOBER '84, EARLIEST AVAILABLE DATE: NOVEMBER '84)

  5) SEPARATE DEVELOPMENT PATH FOR INTERNET WITH IP INSTALLED AND RUNNING (READY FOR PROTOCOL TESTING) BY 2/1/85.

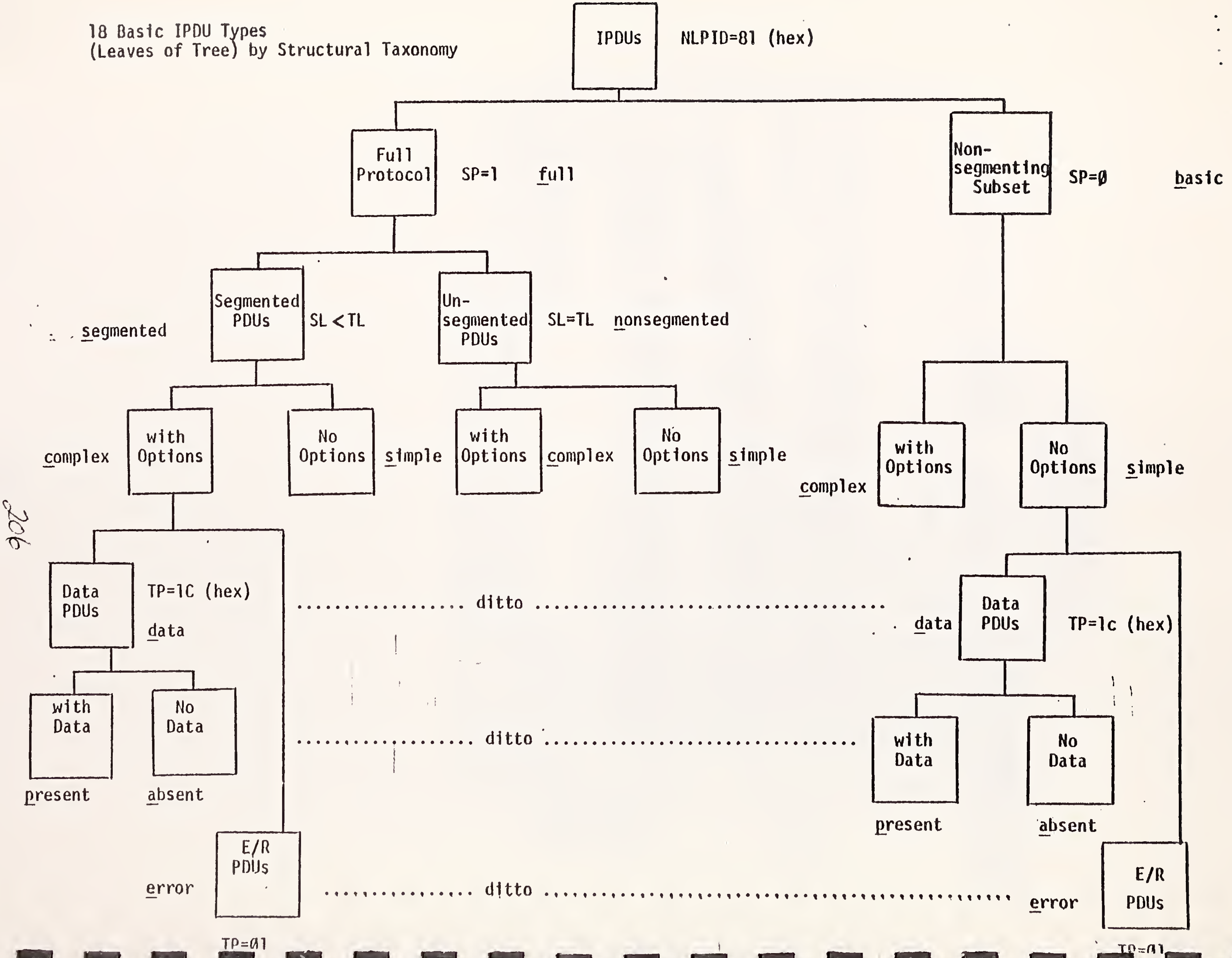
  6) SEPARATE DEVELOPMENT PATHS FOR EXTENDING, UPDATING TRANSPORT AND FOR FTP: SEPARATE TESTING OF MODIFIED TRANSPORT AND OF FTP.

- LIKELY SLIPPAGE:

  3 MONTHS (FOR ASSUMPTION #4) TO 6/30/85

  +4 MONTHS (FOR ASSUMPTIONS #1, #2) TO 10/31/85

18 Basic IPDU Types
(Leaves of Tree) by Structural Taxonomy



Att. #25(b)1

Transport Implementation / Testing

- What we have
- What we would like to have
- Time estimates to update

Att. # 35(b)-2

- What we have:

1. Initiating / Checking Service-Primitives
   - Connection Establishment/Disestablishment
   - Data Transfer
   - Expedited Transfer
   - Multiple Connections
   - Editing PDUs
2. Semi-manual / Connection Oriented
   - Up to 3 concurrent connections may be initiated manually
   - Human analysis of several logs to determine outcome
3. 370 scenarios

What we would like to have HT.#35(b)-3

- More automation – multiple consecutive tests without human intervention
- More automated results analysis
- Automated handling of multiple concurrent tests
- Enforced negotiation testing.
  - Edit PDU and let transport know the new value
- PDU blocking/deblocking/size negotiation
- Use of expedited
- Quality of service
- Checksum negotiation
- Change octet ordering
- Change flow control/parameter code
- Review of scenarios-more data transfer require

NBS/ICST 9/5-86

Time estimates to update and test

AH.#35(1)-4

- Transport enhancements and testing

  2~4 months

- Test system automation

  2-3 months

- "Proper" negotiation testing

  - 2~3 months (manual)

  - 12~18 months (automated)

JSW/BBS 9/5/84

A.H. #35(c)

Layer 5: Session Time, Resource Estimates

Session functional units for ANSI FTAM (BCS*)	≤ 12 PM
Session BAS* subset for X.400	18 PM

Assumptions:

Experienced protocol implementor

No machine-generated code

* BCS = Basic Combined Subset
BAS = Basic Activity Subset

AH. #35(d)

Layer 7: FTAM Time, Resources Estimates

FTAM: ~~ISO~~ ANSI features, X.409 encoding	12-14 PM
Planning, design	1 PM
Map real to virtual file store	2 PM
Learn PDU encoding rules	1 PM
PDU encoder-decoder software	2 PM
Implement protocol machines	4-5 PM
Testing, debugging	2-3 PM

Assumptions:

Experienced protocol implementors

Manual (as opposed to machine) generation of code

KMD 9/6/84

Att #35(e)

Layer 7: Messaging: Time, Resources Estimates

Messaging (CCITT X.400 Message Transfer and Interpersonal Message Services)	35 PM
Planning, design	5 PM
PDU encoder-decoder software	3 PM
Reliable Transfer Server	2 PM
Message Dispatcher & Association Manager	2 PM
Minimal User Agent	10 PM
Component test	8 PM
Integration & system test	5 PM

Assumptions:

Experienced protocol implementors

Manual (as opposed to machine) generation of code.

Omissions:

- optional conversion services of Message Transfer
- support for P3 protocol defined in X.411
- support for interworking with teletex (as def in X.430)

Att. #36

CL4 Transport Protocol

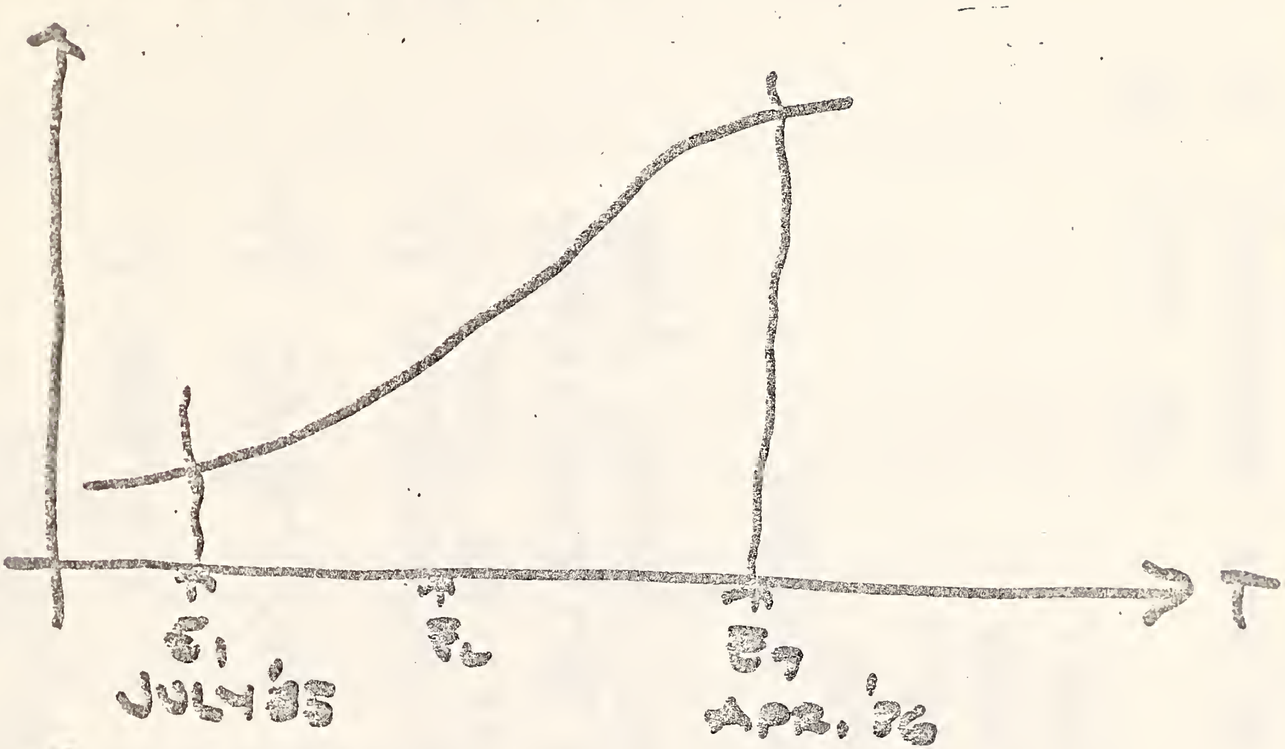
5	SERVICES
9	PDUS
8	STATES
21	EVENTS (INCOMING)
11	PREDICATES

FULL SESSION SERVICE

		TP
21	SERVICES	1
32	PDUS	2
29	STATES	3
75	EVENTS	3
72	PREDICATES	?
623	TRANSITIONS	8

AH. #37

OF VENDORS MEETING SOME CRITERIA



COMMITMENTS

- IMPLEMENT PROTOCOLS
- PARTICIPATE IN GIVEN TRADE SHOW
- NBS COMMITMENT
- EQUIPMENT ON ONGOING BASIS

Att #38

EVENT OPTIONS

1) AUTOFAC - ~~OCTOBER~~ 1985
NOV 5-7

2) COMDEX/OR INTEC - FALL/LATE SUMMER 1985

3) COORDINATED MEDIA EVENT
OCT '85 - 13 MAR '86 14

802.3 OFFICE, CAD, GEN AIR. DEMO - BOEING
802.4 FACTORY DEMO - GM
} INTERNET

HOSTED COCKTAIL/BUFFET
PRESS
LAN (MAP USER GROUP)
NETWORK USERS ASSOCIATION
VENDORS

2 NCC 85
11 HANOVER APR '86
0 ACM ~~AUG~~ OCT '85

Att. #39

OSI WORKSHOP: Request for Information

Please provide the information requested below by Monday, October 29, 1984 to:

John Heafner
National Bureau of Standards
B218 Technology
Gaithersburg, MD 20899

This information is needed for planning and promoting the development of OSI implementations. It will be made public. A list of the organizations submitting the information will be made public but organizations will not be individually identified with the information they provide.

1. My organization will make prototype implementations available for multi-organization testing by the following dates.

   - Subnetworks (identify which, e.g., 802.3, X.25)
     (Please provide dates.)

   - The conformance subset of the ISO connectionless internetwork protocol.
     (Please provide date.)

   - The ISO transport class 4.
     (Please provide date.)

   - The ISO session protocol.
     (Please provide date.)

   - The ISO FTAM subset chosen by the workshop participants.
     (Please provide date.)

   - The CCITT 400 series draft recommendations for messaging.
     (Please provide date.)

2. My organization is willing to participate in one or more of the following exhibitions. (Please indicate which protocols will be available for each exhibit.)

NCC:	July, 1985
Intec:	August, 1985
Autofac:	~~October,~~ 1985 Nov. 5-7
ACM:	October, 1985
Hanover Fair:	April, 1986

3. The organization participating in the NBS OSI implementation workshops plan to construct a multi-organization concatenated network and leave their equipment attached for perhaps 24 to 36 months for the purposes of: a) developing and testing OSI protocols, and b) demonstrating OSI. My organization will attach a system(s) on the following date.
   (Please give date.)

The protocols available for multi-organization testing will be available on this system on the dates given in item 1 above.

Att. #40

CCITT Message Handling Systems: Model, Services and Protocol

September 7, 1984

Arthur R. Pope
Bolt Beranek and Newman Inc.

CCITT X.400 Series

Eight Draft Recommendations form the X.400 series on Message Handling Systems:

X.400 - System Model-- Service Elements

X.401 - Basic Service Elements and Optional User Facilities

X.408 - Encoded Information Type Conversion Rules

X.409 - Presentation Transfer Syntax and Notation

X.410 - Remote Operations and Reliable Transfer Server

X.411 - Message Transfer Layer

X.420 - Interpersonal Messaging User Agent Layer

X.430 - Access Protocol for Teletex Terminals

CCITT X.400 Tutorial -- Outline

1. What is a Message Handling System?

2. What services are provided by a Message Handling System?

3. How are the users of Message Handling Systems named?

4. Where does a Message Handling System belong in the Basic Reference Model for Open Systems Interconnection?

5. What is the X.409 presentation transfer syntax?

6. What are the protocols used in a Message Handling System?

7. Special topics:

   a) How Teletex users may access Message Handling Systems.

   b) How Message Handling Systems facilitate communication between different kinds of devices.

   c) The Reliable Transfer Server and its use of the Session Service.

   d) Areas for future consideration.

What is a Message Handling System?

Electronic mail: Users exchange messages--analogous to postal service.

Message Handling System: A collection of computer systems, interconnected so as to provide an electronic mail service.

A Message Handling System provides fast, efficient, message-oriented communication, within a building or around the world.

Functional Model of a Message Handling System

The functional components of a Message Handling System are:

User Agents (UAs), which help users prepare and receive messages.

Message Transfer Agents (MTAs), which transport messages across the network.

The Message Transfer Agents are interconnected to form a Message Transfer System (MTS).

The Message Transfer System provides a general, application-independent, store-and-forward message transfer service.

An originator sends a message through a process called submission. A recipient receives it through a process called delivery.

A message consists of a message content and a message envelope.

The message envelope contains the information necessary to route and deliver the message to its recipients. The message content is generally transferred without change by the Message Transfer System.

The Interpersonal Messaging System

The Interpersonal Messaging System (IPMS) builds on the Message Transfer System, supplying services of particular use to individuals.

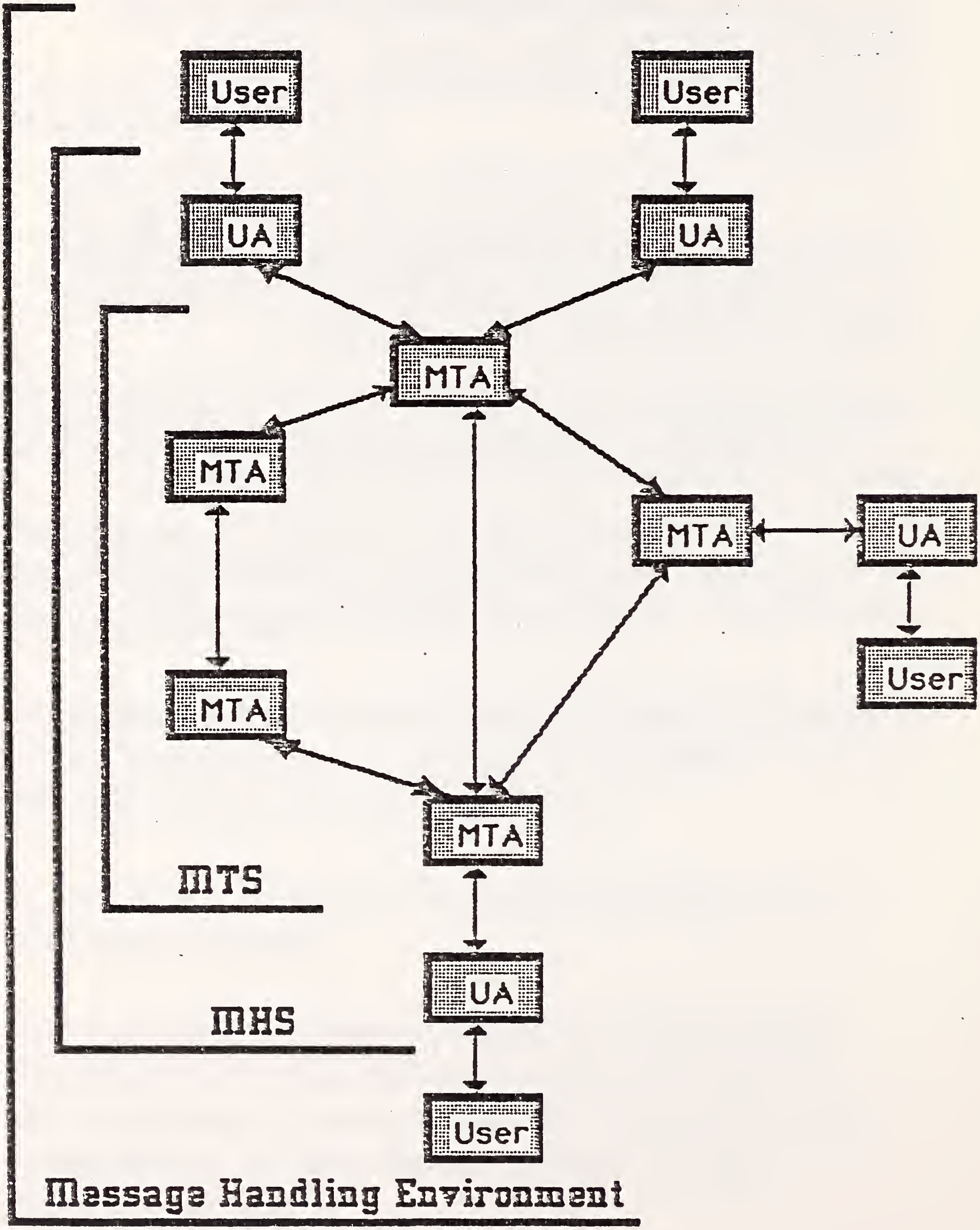
The Interpersonal Messaging System comprises the Message Transfer System and a specific class of cooperating User Agents.

The messages exchanged by users of the IPMS are of a certain form, called IP-messages.

IP-messages are analogous to typical office memoranda. Each has a heading and a body.

The heading includes such information as "to:", "from:", "subject:", "cc:".

Functional Model of an MHS



Physical Mapping

A Message Transfer Agent resides in a computer system.

A User Agent may reside in a computer system, an intelligent terminal, or be distributed among both.

A User Agent may reside in the same computer system as a Message Transfer Agent, or in a physically separate computer system.

Many User Agents may be co-resident with a single Message Transfer Agent.

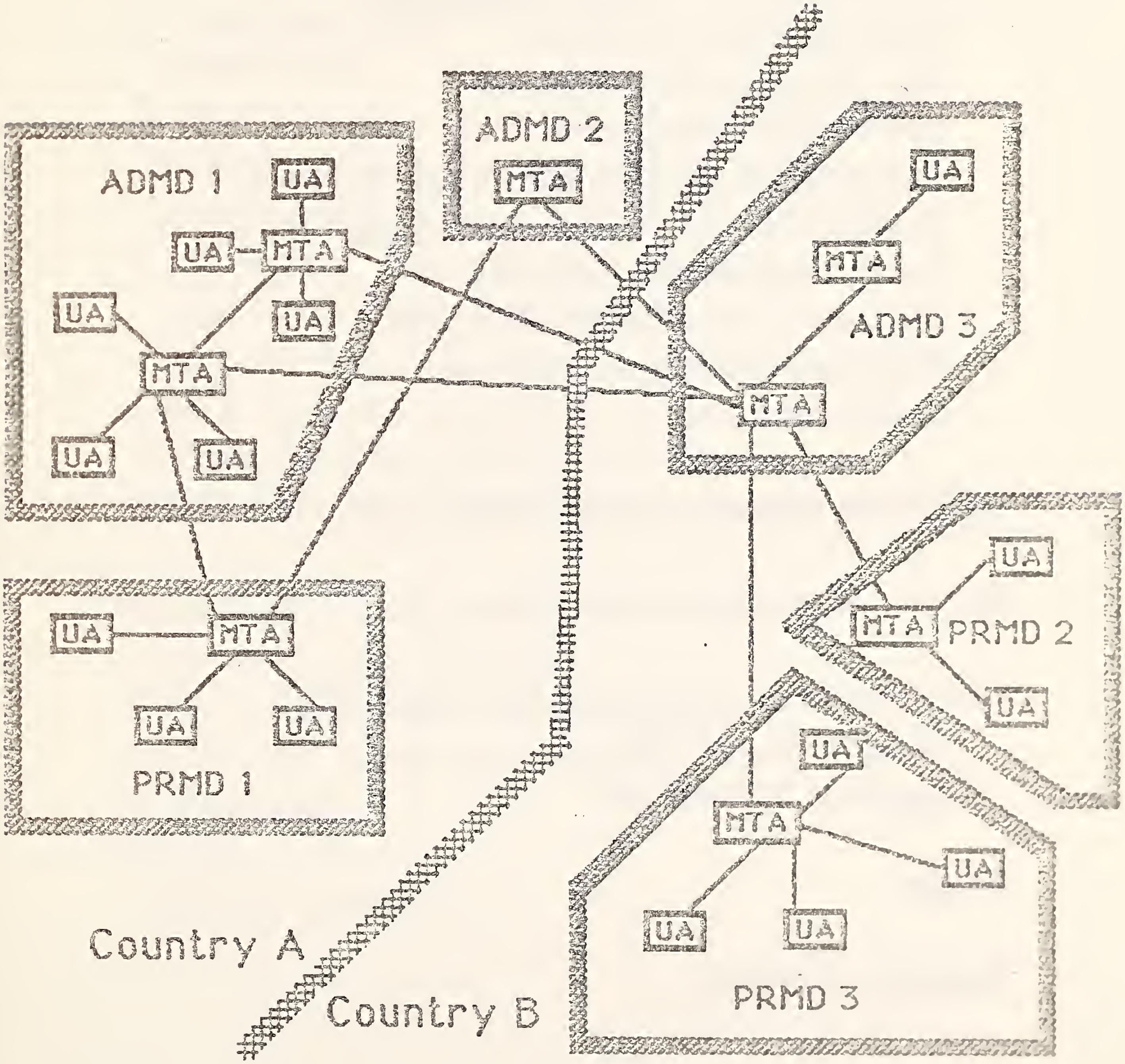
Organizational Mapping

A collection of at least one Message Transfer Agent and zero or more User Agents owned by an organization constitutes a Management Domain.

Administration Mangement Domains (ADMDs) are operated by PTTs and RPOAs.

Private Management Domains (PRMDs) are operated by other organizations.

Management Domains



Message Transfer Service

Basic message transfer, including:
- Message identification.
- Non-delivery notification.
- Delivery notification.
- Submission and delivery time stamps.
- Urgent, non-urgent or normal delivery.
- Multi-destination delivery.
- Disclosure of other recipients.
- Alternate recipient allowed.
- Deferred delivery.

Deferred delivery cancellation.

Return of message contents.

Conversion of message body, including:
- Explicit conversion.
- Implicit conversion.

Probe.

Hold for delivery.

Interpersonal Messaging Service

Basic interpersonal messaging, including:

- IP-message identification.
- Typed body.
- Primary, copy, and blind copy recipients.
- Receipt and non-receipt notification.
- Cross-referencing among IP-messages.
- Expiry date, importance, subject and sensitivity indications.
- Body encription.
- Reply request indication.
- Forwarding and autoforwarding.
- Multi-part body.

Incorporates Message Transfer Service:

- Basic message transfer.
- Deferred delivery cancellation.
- Return of message contents.
- Conversion.
- Probe.
- Hold for delivery.

What's in a name?

Users of the Message Handling System are identified by originator/recipient names (O/R names).

An O/R name is a set of attributes, where an attribute is either assigned by some naming authority or chosen to be descriptive of the thing being named.

Personal attributes: personal name.
Geographical attributes: street, town, country.
Organizational attributes: org. name, position.
Architectural attributes: X.121, MD name.

Each O/R name must include, at minimum, attributes which identify the Management Domain to which the user subscribes. This is the base attribute set.

Initially, two forms of base attribute set are to be supported by every Management Domain:

Country name and MD name.
X.121 address.

Additional attributes, such as organization and personal name, identify the user within the Management Domain.

Basic, Essential and Additional Services

Services are classified as...

Basic services: inherent in the Message Handling System

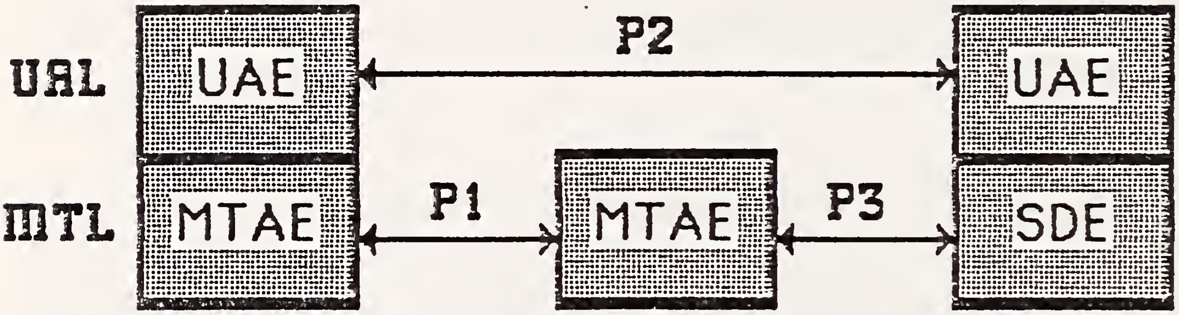
Essential optional user facilities: may be selected by the user on a per-message basis or for a period of time. Must be supported by all Administration Management Domains.

Additional optional user facilities: may be selected by the user on a per-message basis or for a period of time. Need not be supported by all Administration Management Domains.

For example, explicit conversion, hold for delivery and return of contents are additional optional user facilities.

Many IPM services are such that they must be supported for receiving by User Agents, but need not be supported for originating. E.g., expiry date, importance and sensitivity indications.

Layered Description of the IPMS



Both MHS layers lie within the Application Layer of the OSI Reference Model.

Definitions

UAE -- User Agent Entity
MTAE -- Message Transfer Agent Entity
SDE -- Submission and Delivery Entity

P1 -- Message Transfer Protocol
P2 -- Interpersonal Messaging Protocol
P3 -- Submission and Delivery Protocol

Presentation Transfer Syntax

CCITT Draft Recommendation X.409 defines an encoding scheme for protocol data units, and a notation for describing protocol data units.

Simple primitive types: BOOLEAN, INTEGER, BIT STRING, OCTET STRING.

Combined using constructor types: CHOICE, SET, SEQUENCE.

Using this convention, one can describe the form of arbitrarily complex protocol data units.

Each type is encoded as:

- a particular identifier of one or more octets.
- the length of the encoded type, in one or more octets.
- the contents of the type, encoded in a form according to the type.

For example, the value TRUE of type BOOLEAN is encoded as:

identifier = 01_{16}
length = 01_{16}
content = FF_{16}

The Message Transfer Protocol

Denoted P1. Its protocol data units are MPDUs, specified in notation defined in CCITT Draft Recommendation X.409.

Relies on underlying Basic Activity Subset of Session Service (CCITT Draft Recommendation X.215).

Carries messages, probes and delivery reports from one Message Transfer Agent to another.

```
MPDU ::= CHOICE {UserMPDU,
                 DeliveryReportMPDU,
                 ProbeMPDU}

UserMPDU ::= SEQUENCE {Envelope, Content}

Envelope ::= SET {
   MPDUIdentifier,
   originator ORName,
   priority INTEGER
      {normal (0), nonUrgent (1), Urgent (2)}
   recipients SEQUENCE OF ORName}

ORName ::= SEQUENCE OF Attribute

Content ::= OCTET STRING
```

Interpersonal Messaging Protocol

Denoted P2. Its protocol data units are UAPDUs, specified in notation defined in CCITT Draft Recommendation X.409.

Relies on the Message Transfer Service to transfer UAPDUs between User Agents.

UAPDUs are IP-messages and status reports (receipt and non-receipt notifications).

```
UAPDU ::= CHOICE {IP-Message, StatusReport}

IP-Message ::= SEQUENCE {Heading, Body}

Heading ::= SET {
   IP-Message-Identifier,
   originator ORName OPTIONAL,
   primaryRecipients SEQUENCE OF ORName,
   copyRecipients SEQUENCE OF ORName,
   blindCopyRecipients SEQUENCE OF ORName,
   subject OCTET STRING OPTIONAL,
   crossReferences SEQUENCE OF
        IP-Message-Identifier}

Body ::= SEQUENCE OF BodyPart

BodyPart ::= OCTET STRING
```

Submission and Delivery Protocol

Denoted P3. It is in the style of a Remote Procedure Call protocol, where the operations (remote procedure calls) are specified in the notation defined in CCITT Draft Recommendations X.409 and X.410.

Relies on underlying Basic Activity Subset of Session Service (CCITT Draft Recommendation X.215).

Has operations for:

Submitting a message or probe
Cancelling a previously-submitted message
Delivering a message
Notifying of message (non-)delivery
Changing subscription parameters

There is also a mechanism for access control, based on passwords.

Teletex Access to a MHS

Teletex users may access a Message Handling System through the Teletex system.

CCITT Draft Recommendation X.430.

A Teletex Access Unit (TTXAU) is a gateway between a Teletex system and an MHS.

Using the Teletex system, the TTXAU exchanges documents with a Teletex user. The form of these documents and the manner in which they are exchanged are called the Teletex Access Protocol (P5).

P5 is based upon S.62.

The TTXAU is co-resident with an MTAE, from which it obtains the Message Transfer Service. It uses this service, and the P2 protocol, to exchange messages with User Agents, on behalf of the Teletex user.

The TTXAU may provide document storage for the Teletex user.

Different Kinds of Devices

User Agents may differ in their capabilities for rendering different information representations: text, facsimile, videotex, voice, structured documents.

These different representations are called encoded information types.

The body of an IP-message may be represented using one or a mixture of different encoded information types.

Each User Agent may register with the Message Transfer Service the encoded information types it can accept.

The Message Transfer Service can convert among encoded information types.

Conversion may be:
invoked explicitly by the user,
performed by the Message Transfer System
if it is found to be necessary, or,
prohibited by the user.

Examples of conversions described in X.408:
text to facsimile
structured document to text

The Reliable Transfer Server

The Message Transfer Protocol and the Submission and Delivery Protocol both use a mechanism called the Reliable Transfer Server (RTS), described in CCITT Draft Recommendation X.410.

The Reliable Transfer Server is a functional entity that resides in a computer system. It uses the Session Service to reliably move application layer protocol data units from one system to another.

The Reliable Transfer Server takes care of:
establishing and releasing sessions
negotiating the use of checkpoints
inserting checkpoints in the data stream it sends
recovering lost sessions
resuming data transfer at a recent checkpoint

A small amount of protocol is exchanged between the Reliable Transfer Server and its peer in accomplishing this.

Areas for Future Consideration

User-friendly O/R names.

Directory services to support user-friendly names.

Compatibility with ISO Message-Oriented Text Interchange System standards.

Att. # 40(i)

MHS Demonstration Proposal

Message Transfer Service and Interpersonal Messaging Service.

Each participant supplies one or more Management Domains.

Each Management Domain contains:
  one or more Message Transfer Agents; and
  zero or more User Agents.

P1 implemented by all participants.
P2 implemented by some participants.
P3 and P5 need not be implemented.

All essential optional user facilities implemented. Additional optional user facilities need not be implemented.

O/R name contains Management Domain name and, possibly, personal name.

User Agents support IA5Text encoded information type.

Participants are free to construct User Agent user interfaces as they wish.

Participants must implement Basic Activity Subset of Session Service.

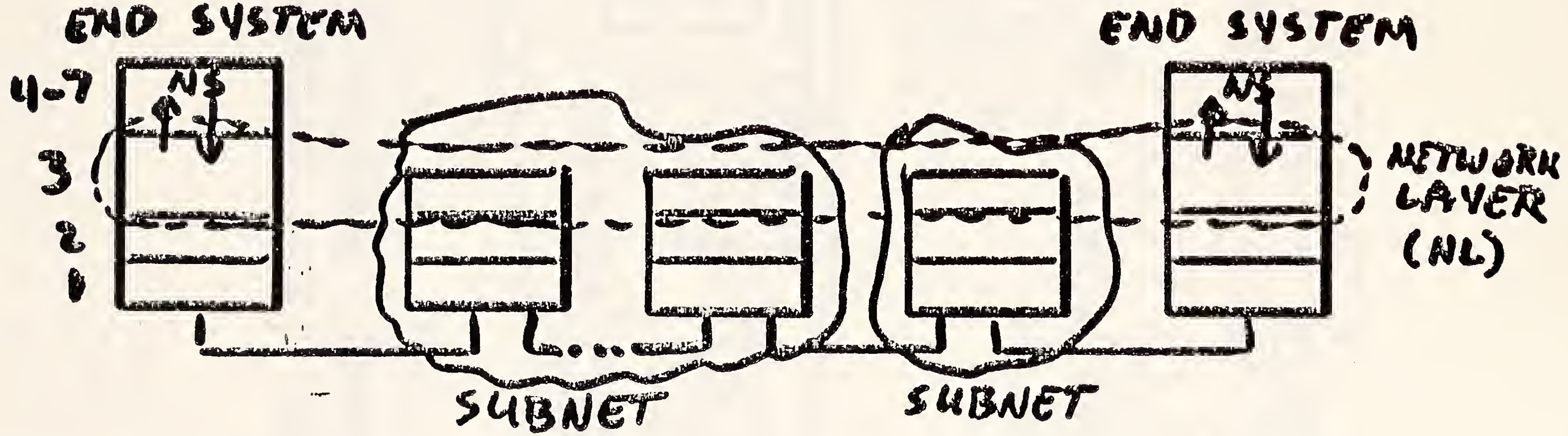
SNDCP FOR X.25 TO PROVIDE THE CLNS

F. BURG

AT&T-IS

FMB-1

INTERNAL ORGANIZATION OF THE NETWORK LAYER TERMINOLOGY



- END SYSTEM
- RELAY SYSTEM
- SUBNETWORK (SN)

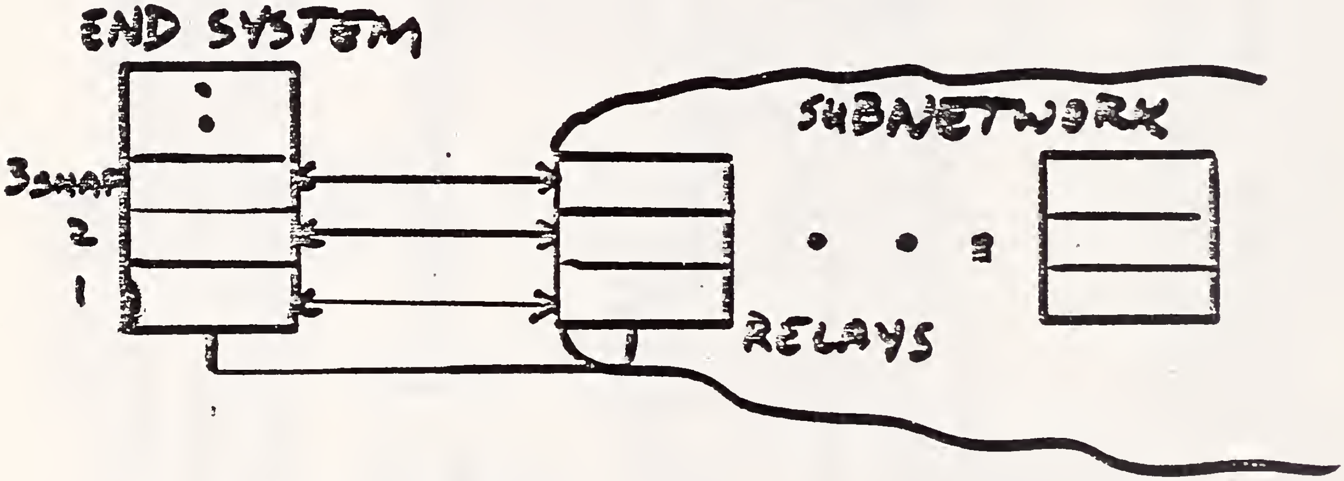
} INTERMEDIATE SYSTEM

- SN ACCESS "PROTOCOL" (SNAcP)
- SN DEPENDENT CONVERGENCE "PROTOCOL" (SNDCP)
- SN INDEPENDENT CONVERGENCE "PROTOCOL" (SNICP)
- "PROTOCOL" ROLES

FAB-2

WHAT ARE SNAcPs?

SNAcP = SUBNETWORK ACCESS PROTOCOL



SNAcP OPERATES UNDER CONSTRAINTS CHARACTERISTIC OF A PARTICULAR SUBNETWORK

FMB-3

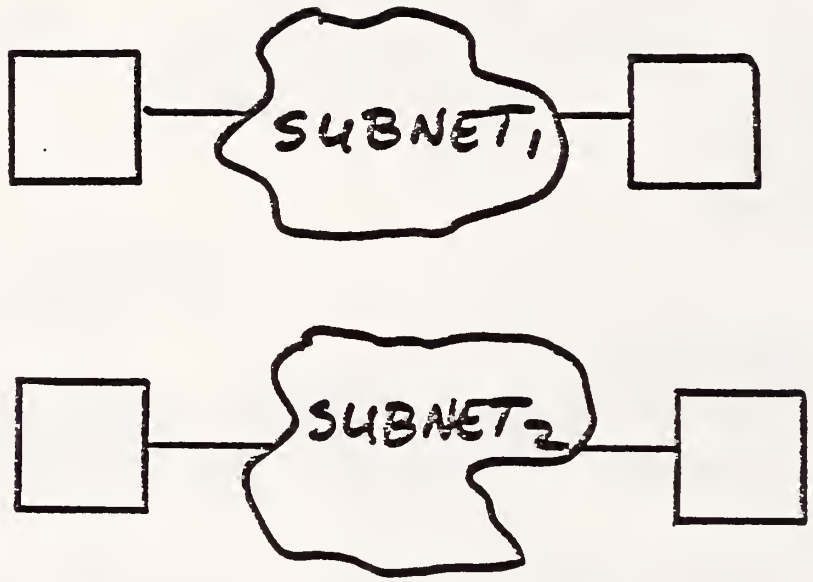
ROLES OF A NETWORK LAYER PROTOCOL

- SNAcP
- SNDcP
- SNICP

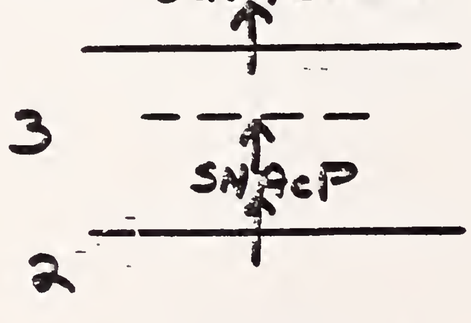
- A NL PROTOCOL MAY FULFILL ONE OF THESE ROLES IN A PARTICULAR SITUATION
- THE SAME PROTOCOL MAY FULFILL DIFFERENT ROLES IN A DIFFERENT SITUATION
- A SINGE NL PROTOCOL MAY PROVIDE ALL THE FUNCTIONS OF PROTOCOLS OPERATING IN THE OTHER ROLES

FMB-4

WHY SNDCPs?



— SNAcPs MAY PROVIDE DIFFERENT CAPABILITIES WITH RESPECT TO THE OSI NETWORK SERVICE (CONS OR CLNS)

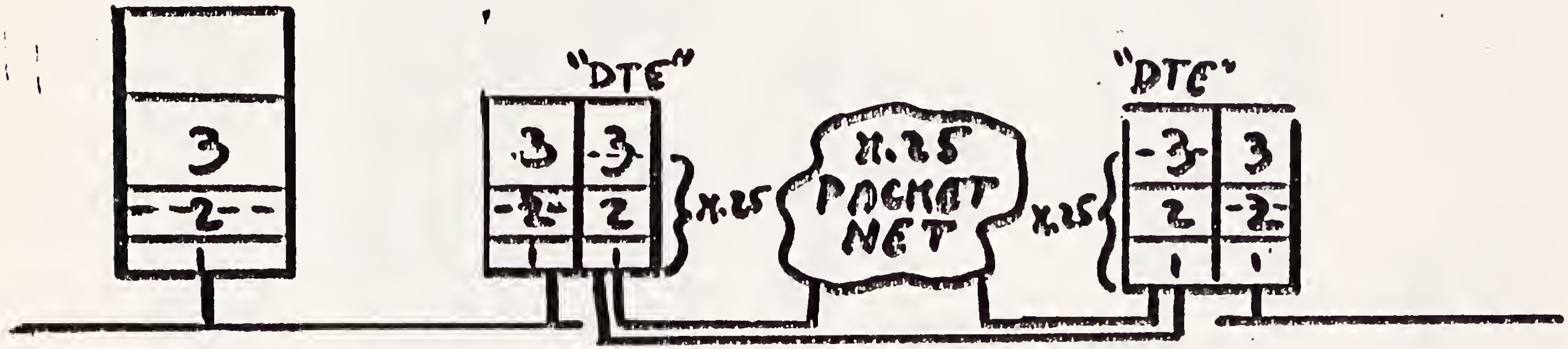


FMB-5

WHAT ABOUT SNICPs?

- ASSUMPTIONS MADE ABOUT UNDERLYING SERVICES ARE MINIMALLY RESTRICTED: THEY NEED NOT BE BASED ON THE CHARACTERISTICS OF ANY PARTICULAR SUBNETWORK
- EXAMPLES:
  - PROTOCOL TO PROVIDE THE CLNS (DIS 8473)

FMB-6

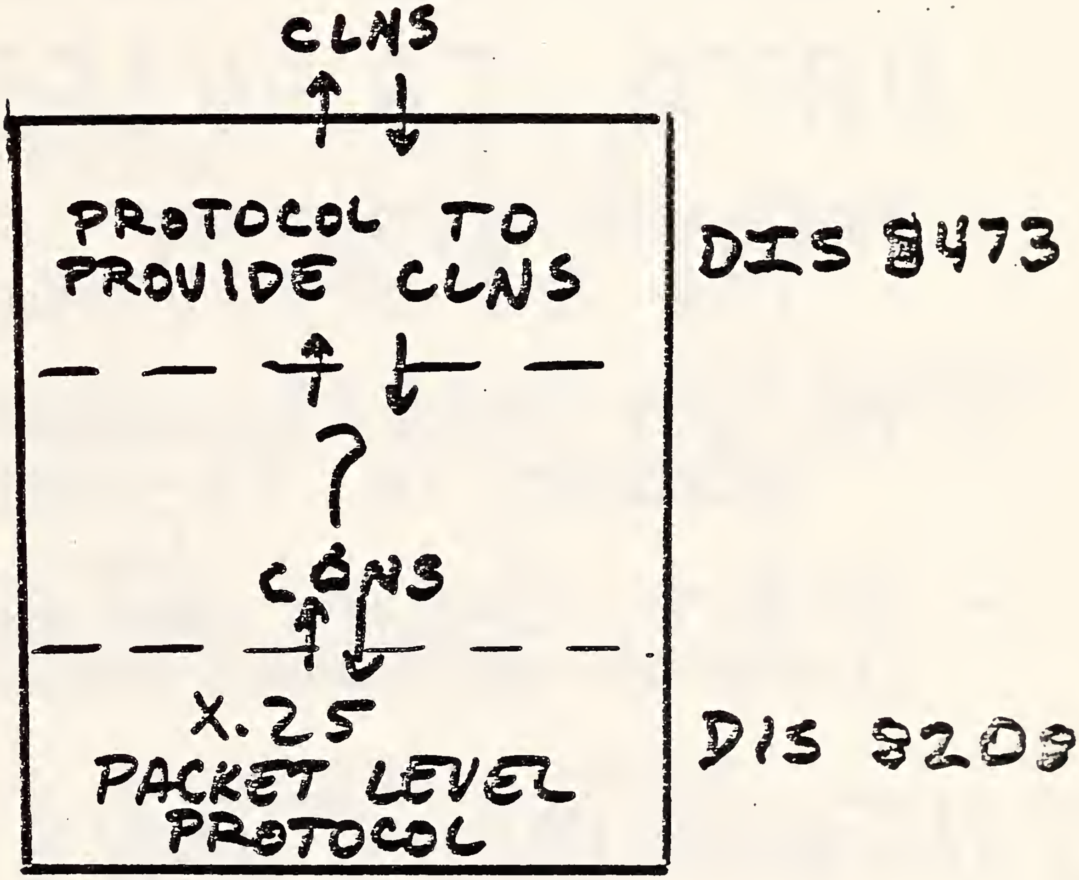


X.25 CAPABILITIES

- RESTART
- VIRTUAL CIRCUIT "SETUP/CLEARING"
  - VIRTUAL CALL (SVC)
  - PERMANENT VIRTUAL CIRCUIT (PVC)
- DATA TRANSFER
- INTERRUPT TRANSFER
- FLOW CONTROL
- RESET
- OPTIONAL USER FACILITIES/PARMS
- CAUSE/DIAG INFO

FM

FMD-7



CLNS

PROTOCOL TO PROVIDE CLNS

DIS 8473

?

CONS

X.25 PACKET LEVEL PROTOCOL

DIS 8208

? = SET OF SNDC FUNCTIONS; NO PROTOCOL INVOLVED

DATA TRANSFER

- PACKET SIZE (P)
  128 OCTETS; BIGGER?
- M-BIT FOR SEGMENTING
  NEEDED IF CL-PDU > 128
- D-BIT, Q-BIT: NOT NEEDED
  (DISCARD DATA_ACK FOR D-BIT)

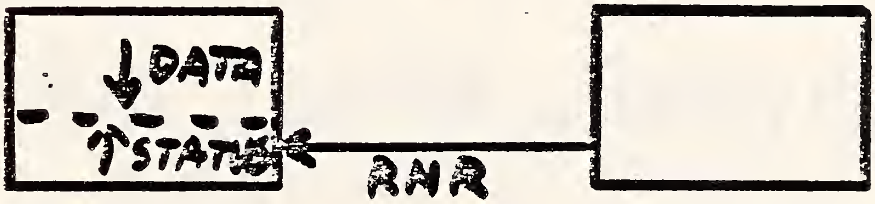
INTERRUPT TRANSFER

- NOT NEEDED
- DISCARD INTERRUPT DATA

FMD-9

FLOW CONTROL

- WINDOW SIZE (W)
  2, OR ANYTHING ELSE AVAILABLE
- PEER RECEIVE-NOT-READY



- DISCARD DATA
- NEW VIRTUAL CIRCUIT
- QUEUE DATA

NEED X.25 TO INDICATE BUSY/NON-BUSY STATUS

FMB-10

RESET

- DISCARD PARTIALLY TRANSMITTED/RECEIVED M-BIT SEQUENCE (DONE BY X.25 PACKET LEVEL)
- DISCARD RESET INDICATION

FMB-11

OPTIONAL USER FACILITIES

- NONSTANDARD DEFAULT W, P SIZES (PER ABOVE)

- 1-WAY INCOMING/OUTGOING LOGICAL CHANNELS
- INCOMING/OUTGOING CALLS BARRED
- CLOSED USER GROUP
- DEFAULT THROUGHPUT CLASSES ASSIGNMENT
- FAST SELECT/ACCEPTANCE
- REVERSE CHARGING/ACCEPTANCE

- HUNT GROUP

- "DIAL-UP"

FMB-12

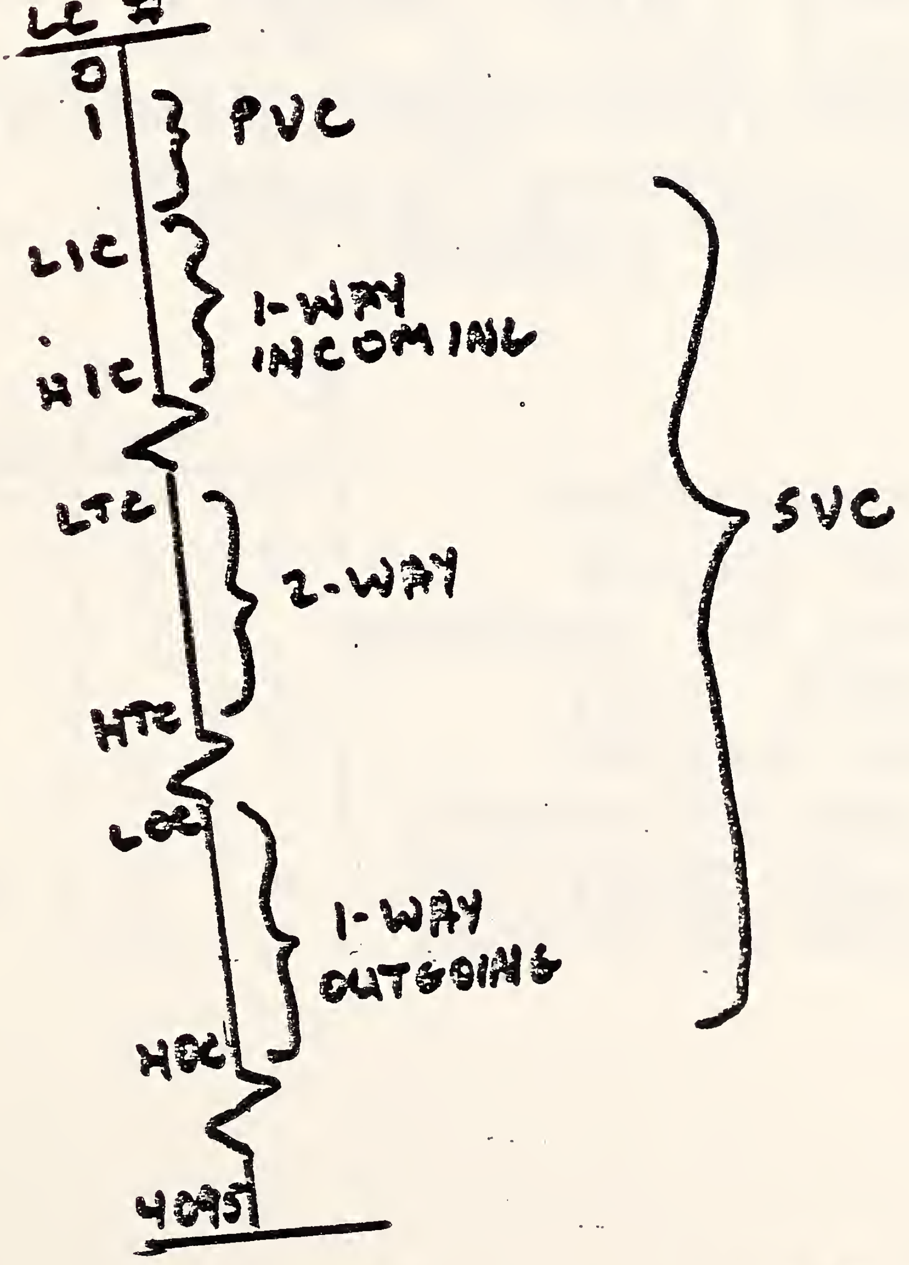
PACKET LEVEL PARAMETERS

- T20/R20 (RESTART)
- T21 (CALL SETUP)
- T23/R23 (CALL CLEARING)
- T22/R22 (RESET)
- T24 (WINDOW STATUS)
- T25/R25 (WINDOW ROTATION)
  R25=0
- T26 (INTERRUPT)
  NOT USED
- T27/R27 (REJECT)
  NOT USED
- T28/R28 (REGISTRATION)
  NOT USED

- LOGICAL CHANNEL BOUNDARIES
  (NEXT VG)

FMB-13

LOGICAL CHANNELS



CHOOSE VALUES FOR
LIC/HIC, LTC/HTC, LOC/HOC

FMB-14

CAUSE/DIAG INFO

- IN CLEAR (C), RESET (Re), RESTART (Rr) PACKETS
- GENERAL STRATEGY: DISCARD
- MAY WANT TO MONITOR FOLLOWING CAUSES:

	C	Re		Rr
		PVC	SVC	
- DTE ORIGINATED	✓	✓	✓	
- OUT OF ORDER	✓	✓		
- REMOTE DTE OPERATIONAL		✓		
- NUMBER BUSY	✓			
- ACCESS BARRED	✓			
- NETWORK OPERATIONAL		✓		✓
- NETWORK CONGESTION	✓	✓	✓	✓
- NETWORK OUT OF ORDER		✓		

X.25 CALL SETUP

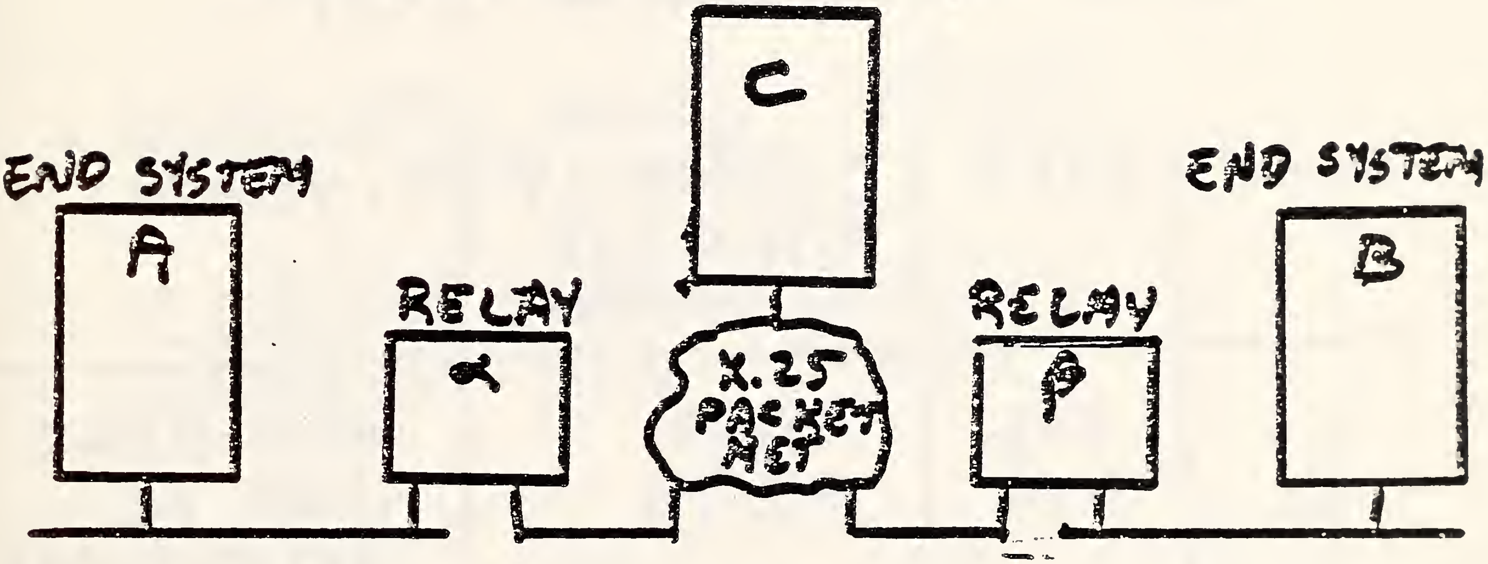
- NO NL "CONNECT" STIMULUS
- SNDC FUNCTION
  - ARRIVAL OF DATA (NO AVAILABLE LC TO DEST: NONE OPEN, NONE NOT BUSY)
  - OTHER (MGMT - TIME OF DAY, ETC.)

IAB-16

X.25 CALL CLEARING

- NO NL "DISCONNECT" STIMULI
- SNDC FUNCTION
  - TRANSMISSION OF DATA
  - NO DATA FOR "X" SECS.
  - NEED ANOTHER LC + NONE AVAILABLE
  - ADDITIONAL LC TO SAME DEST
  - OTHER (MGMT-TIME OF DAY, ETC)

ADDRESSING



X.121 ADDRESS CAN IDENTIFY

- RELAYS: α, β
- END SYSTEMS: A, B, C

  XMTR: NSAP → X.121
  RCVR: X.121 → MAC/NSAP ADDR

LC TABLE

LC#	TYPE	CURRENTLY ATTACHED ADDRESSES	STATUS
1, 2, • • 4095	PVC, SVC	AAAA, ZZZZ	AVAILABLE (PER LC SELECTION), OPEN/CLOSE, BUSY/NOT BUSY, USABLE (PER CAUSE CODES)

RESTART

- EXECUTE WHEN BRING X.25 INTF UP
- CLEARS ALL SVCs, RESETS ALL PVCs

X.25 LEVELS 1 + 2

LEVEL 2 (DIS 7776):

- LAPB
- MODULO 8 NUMBERING
- PARMS
  - T1: ACK TIMER (T1 > T2)
  - T2: RESPONSE TIMER
  - T3: IDLE CHANNEL TIMER (T3 > T1)
    NOT USED
  - T4: LINK ASSURANCE
    NOT USED
  - N2: TRANSMISSION ATTEMPTS
  - N1: MAX I FRAME (BITS)
    DTE-TRANSMIT: ≥ 135 OCTETS
    DTE-RECEIVE: ≥ 263 OCTETS
    (FOR FAST SELECT)
  - k: MAX # OUTSTANDING I FRAMES
    k = 7
- MULTILINK

LEVEL 1 (RS-232C):

- SPEED

Att. # 42 (a)

Proposal for Implementors of OSI for Use of the Connectionless Internetwork Protocol (ISO DIS 8473) Over X.25 Subnetworks

August 1984

1. General

This paper briefly describes the operation of a subnetwork dependent convergence function to provide a connectionless subnetwork service (as required by the Protocol for Providing the Connectionless-Mode Network Service, DIS 8473) over an X.25 subnetwork. The method described here may be generalized for use over other connection-oriented subnetworks. This paper is specifically oriented towards use in a proposed internetwork demonstration making use of a variety of subnetworks interconnected using the protocol defined in DIS 8473.

To avoid confusion, in this paper the Protocol for Providing the Connectionless-Mode Network Service (as described in DIS 8473) will be referred to as the Internetwork Protocol or "IP".

2. Management of Logical Channels

The most difficult issue to be considered is when to open and close X.25 logical channels (subnetwork connections).

Opening of logical channels can be initiated by three events: (1) The arrival of a data unit (or data request from a user in an end system) to be transmitted over the X.25 subnetwork where there is no appropriate logical channel available; (2) Additional logical channels may be opened when the local queue exceeds a certain threshold size, (3) Logical Channels may be opened by intervention of the network management system (possibly by explicit human interaction).

Closing of logical channels can similarly be initiated by three events: (1) The expiration of a timeout period following transmission of one or more PDUs; (2) The need to use a specific interface to open an alternate logical channel from the local Network Layer entity to a different remote Network Layer entity. (3) explicit intervention by the network management system (possibly by explicit human intervention).

The timeout period for closing logical channels may, in general, be chosen by economic and implementation specific means. For example, if there is no duration charge for leaving a logical channel open, and if there is a large charge or time delay in opening logical channels, then the timeout period may be very large (or even infinite). The timeout period may vary with the time of day, traffic load (averaged over the recent past), or other factors. The timeout period for additional logical channels (opened when there is an excessive queue of data units waiting for the first channel) may be shorter than the timeout period for the first channel (for example, some implementations may choose to close all additional logical channels if the queue reaches zero). In the simplest implementation, the timeout period may be a fixed period of time.

For demonstration purposes, it is proposed that logical channels be classified into two categories: "Type A" logical channels are left open "semi-permanently" (for the duration of the demonstration), and can be closed only by human action. "Type B" logical channels are closed when either end system determines that there has been no traffic on that channel in either direction for a duration of five minutes.

3. Addressing

In general, the IP does not constrain the addressing scheme used by underlying subnetworks. Thus the addresses used in subnetwork service requests will be precisely the addresses to be used in the X.25 call request packet, and are separate from the NSAP addresses used in the IP header. Routing tables are used to determine the subnetwork on which to transmit the data unit, and the subnetwork address to be used on that subnetwork.

More specifically, at each source and intermediate system along the path followed by a particular data unit, there are two possibilities: (1) the next IP entity to handle the data unit will be located within a gateway on a local subnetwork, or (2) the next IP entity to handle the data unit will be located within the destination of the data unit. In case (1), the subnetwork to be used and the subnetwork address of the gateway are both determined from routing tables. In case (2), The subnetwork and destination address on that subnetwork to use are both identified by the destination NSAP address.

4. Data Transfer

Data Transfer over logical channels occurs in a full duplex (two-way) manner.

The service requires that the subnetwork be capable of carrying a data unit of up to 255 octets in length. It will be desirable in many cases to carry data units larger than this, if necessary by use of the M bit facility.

5. Quality of Service Maintenance

It is proposed that for the purposes of the demonstration, no specific quality of service maintenance functions are necessary. This implies that the "QOS Maintenance" and "Security" optional fields of the IP header are not used.

6. Detailed Specification of SNDCF Operation

This section provides a detailed specification of the operation of the subnetwork dependent convergence function.

6.1 Service Provided by the SNDCF

The subnetwork service provided by the SNDCF is precisely the service required by the IP, and is abstractly summarized by the following table of service primitives:

Primitives	Parameters
SN_UNITDATA_request	SN_Destination_Address
SN_UNITDATA_indication	SN_Source_Address
	SN_Quality_of_Service
	SN_Userdata

The SN_Source_Address and SN_Destination_Address are subnetwork dependent. Thus, in this case, these are the X.121 addresses used by the X.25 subnetwork. The SN_Userdata parameter carries user data up to a specified maximum size.

6.2 SNDCF Operations

The protocol actions described here are all performed within the SNDCF, and are therefore logically separate from actions of the IP entities. Thus, for example, the destination address parameter in the SN_UNITDATA_request is the immediate address on the local subnetwork to which the pdu is to be sent. No additional routing table within the SNDCF is necessary except to determine which logical channel to use for a particular destination address. The subnetwork itself is likely to perform routing internally, but this again is independent of the SNDCF.

The operations initiated by an SN_UNITDATA_request are complex enough that they will be described using a sort of pseudo-code. The operations initiated by the management function, or by timeout, are simple enough that they can be informally described in English text.

6.2.1 Operations Initiated by an SN_UNITDATA_request

```
begin
  check destination X.121 address to logical channel mapping

  if (existing logical channel is available) then

    begin
      add outgoing pdu to queue for existing logical channel;
      reset the timer for the logical channel.
      if (queue length exceeds threshold) then
          open an additional logical channel.
    end;

  else if (new logical channel can be opened) then
    begin
      open new channel (note: this may require closing
                              an existing channel),
      start the timer for the new logical channel.
      add pdu to queue for the new channel.
    end.

  else
      discard data unit:
end.
```

6.2.2 Operations Initiated by The Management Function

When requested by the management function, the SNDCF may open a new logical channel, clear an existing logical channel.. reset an existing logical channel, restart all logical channels over a given interface, or provide status information.

6.2.3 Operations Initiated by Timeout

When a timeout occurs on a given logical channel, the channel is closed.

Att. #42(b)-1

IP OVER X.25 FOR DEMO

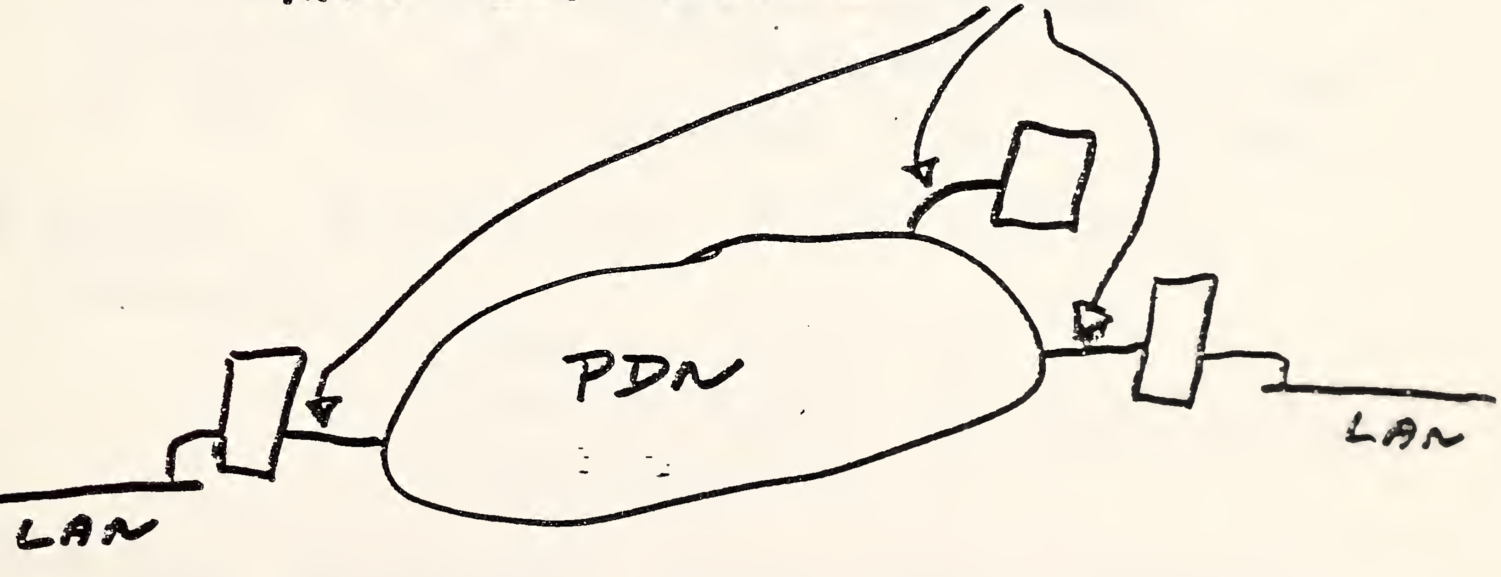
OPENING/CLOSING LOGICAL CHANNELS

- "PERMANENT" (FOR DEMO)
- TEMPORARY
  - OPEN WHEN DATA ARRIVES
  - CLOSE " IDLE 5 MIN.
- ADDITIONAL
  - OPEN WHEN Q > LIMIT
  - CLOSE " Q = 0 + IDLE

?X DOES ONLY THE ORIGINATOR CLOSE L.C.
?N OPEN ADDITIONAL L.C. BY OTHER CRITERIA
(EG; Q 0 FOR LONGTIME;

ADDRESSES

- X.25 CR CONTAIN "LOCAL" ADDR.



Att. #42(b)-2

DATA TRANSFER

- FULL DUPLEX
- USE MBIT WHERE NEEDED
- NO FAST SELECT

QOS, CLOSED USER GROUP

- NONE

Att. # 43 (2)

6 points to achieve an early interworking Demo of X400.

1) Cut back MTL services and P1 protocol

2) Cut back UAL services and P2 protocol

(both 1 & 2 should be done on a Symetric basis, ie reception capabilities should be the same as creation capabilities)

3) The RTS to remap onto the BCS session. This requires only the definition of a 'TRANSFER OPERATION' in the Remote Operation Service of X410

4) Adopt ECMA 93 Name & Address philosophy.

- Domain Name
- MTA Name } printable strings
- User Agent ID

5) Adopt ECMA 93 Association Establishment conventions

6) Agree on X409 usage.

- single byte 'I' octet
- Use of definite and indefinite length.

ICL-IV 9/3/84

Att. #43 (b)

User Agent Sublayer Services

NOT supported for DEMO

- Authorising User
- Blind Copy
- In Reply to
- Obsoleting
- Cross referencing
- expiry date
- reply by
- Reply to users
- Importance
- Sensitivity
- Auto Forwarding
- Body types other than IA5 text.

Att. #43 (4)

Message Transfer Sublayer Services

NOT supported for Demo.

- PROBE
- CONVERSION
- PRIORITY
- DEFERRED DELIVERY
- Per Domain Bilateral Information
- Return of contents
- Billing Information
- Supplementary Information

ICL-IV 9/7/84

Att. #44

~~Specific Proposal~~

~~Definitions: LAN Cluster – A collection of LAN's, end-systems and Gateways connected to a PDN~~

X.400 Proposal (Wang)

- full comformance to CCITT X.400 series
- arbitrary number of MTA's per authority
- BAS of Session Service
- Inclusion of P5

J. St. Amand
9/7/84

ATT. #45

Proposal for Mapping P1 Protocol of X.40x onto BCS :

- Each message is sent between MTA's as a Sequence of SSDU's on a Session connection. Messages are delimited by the establishment & release of Session Connections

- As a performance optimization, TC's used for MTA ⟺ MTA Sessions are allowed to survive the Release of a Session Connection

AH. #46

AGREE THAT THERE WILL BE A PHASE 2 EXPANSION UPON THE CURRENTLY APPROVED PHASE 1 SUBSET OF FTAM. SPECIFICS OF THIS EXPANSION WILL BE DETERMINED AT A FUTURE ~~SESSION~~ WORKSHOP (NOV) - GM/BOEING WILL PROVIDE A DETAILED PROPOSAL FOR THIS EXPANSION.

FOR ——　　　AGAINST ——

Unanamous

Session being organized on
"Usage of OSI Standards"
for the next ICC in
Chicago (June 23-26, 1985)
by Day & Zimmerman

Contact: John Day
Codex Corp
20 Cabot Blvd
Mansfield, MA 02048
(617) 364-2000

TRANSPORT PROTOCOL

Alignment Changes

There are two changes necessary to bring the demo version in line with the ISO specification:

1. Octet Ordering - the order of octets in multi octet binary fields has been changed from least significant octet first to most significant octet first.

2. Parameter Code - the value of the flow control parameter has been changed to 1000 1100 make it unique (it had the same value as another parameter).

Proposal for Demonstration of Message Handling System Standards

Overview

A *Message Handling System* is a collection of computer systems. interconnected so as to provide a service whereby users may exchange electronic mail messages. The CCITT have produced a series of Draft Recommendations the X.400 series, that define standard ways in which Message Handling Systems may be interconnected. Herein we propose a cooperative demonstration of a practical Message Handling System based on the CCITT X.400 series of Draft Recommendations.

Background

The CCITT X.400 series specifies the service elements and protocols for Message Handling Systems. There are two message handling services. the *Message Transfer Service* and the *Interpersonal Messaging Service*. The Message Transfer Service is sufficiently general that any application can use it to transfer data in a store-and-forward manner. The protocols used to provide the Message Transfer Service are specified in Draft Recommendation X.411.

While any application can use the Message Transfer Service, the CCITT recognized that normally the Message Transfer Service would be used by individuals to send messages to each other Accordingly the CCITT developed an Interpersonal Messaging Service to accomodate this kind of communication. The Interpersonal Messaging Service is implemented using the facilities of the Message Transfer Service. The protocols used to provide this service are specified in Draft Recommendation X.420

The CCITT views a Message Handling System as comprising *Message Transfer Agents* responsible for routing and transferring messages. and *User Agents* responsible for supporting users in their roles as message originators and recipients. A collection of Message Transfer Agents and User Agents provided by a single organization is called a *Management Domain*. The CCITT concerns itself with standardizing the communication between Management Domains (such as communication between Message Transfer Agents in different Management Domains) but the standards it develops may be used within Management Domains as well.

The Message Transfer Agents in one Management Domain communicate with Message Transfer Agents in other Management Domains by means of the standard protocol *P1* defined in Draft Recommendation X.411. User Agents communicate by means of the standard protocol *P2*, defined in Draft Recommendation X.420. Finally, a User Agent may communicate with a Message Transfer Agent by means of the standard protocol *P3* defined in Draft Recommendation X.411 P1 and P3 make use of a *Reliable Transfer Service*, defined in Draft Recommendation X.410. that reliably transfers data using the Session Service.

Users of the Message Handling System are assigned names so that they may be identified as the originators and recipients of messages. These are called *O/R names* CCITT Draft Recommendation X.400 prescribes two forms of O/R name: one containing the name of the Management Domain which serves the user, and the other containing an X.121 address identifying the user's point of attachment to a public data network. This latter form is intended primarily for naming users who access the Message Handling Service via Teletex terminals. They make use of the Message Handling System indirectly, by means of the Teletex service and a standard protocol *P5*. defined in CCITT Draft Recommendation X.430.

Draft Recommendation X.401 lists the service elements of the Message Transfer Service and the Interpersonal Messaging Service. The basic service elements of both services are defined in Sections 2.1 and 3.1 of that document. In addition, Draft Recommendation X.401 defines service elements that are optional for the user to use but essential for the service provider to implement. These are called *essential optional user facilities*. There are also service elements that may or may not be implemented. These are called *additional optional user facilities*

The service elements of the Interpersonal Messaging Service are further categorized as essential or additional for each of the sending and receiving User Agents. There are certain service elements that a system need not provide for its users who originate messages but that it must recognize in messages it receives Tables 1/X.401 through 4/X.401 list the essential and optional user facilities for the Message Transfer Service and the Interpersonal Messaging Service There are also local functions of the Interpersonal Messaging Service, such as a message editing capability or a user alert when a new message arrives, that are not subject to standardization but will be provided in most implementations.

The Message Transfer Service makes extensive use of the Basic Activity Subset of the Session Service described in CCITT Draft Recommendation X.215 The following functional units of the Session Service are used: kernal functional unit, half-duplex functional unit, minor synchronize functional unit, exceptions functional unit and activity management functional unit. The use of the Session Service is described in Section 4 of Draft Recommendation X.410.

Proposal

We propose a demonstration of a practical Message Handling System based on the CCITT X.400 series of Draft Recommendations. The demonstration would involve a number of participants each contributing some portion of the overall Message Handling System. It would serve to demonstrate both the utility of the Recommendations and the ability of each participant to implement the Recommendations.

The scope of the proposed demonstration is described below It defines the minimum degree of functionality which would be supported by each participant in the demonstration, individual participants would not be constrained from exceeding this minimum, for example by supporting additional optional services or protocols.

1. The demonstration would illustrate the Message Transfer Service and the Interpersonal Messaging Service.

2. Each participant would supply one or more Management Domains

3. Each Management Domain would comprise one or more Message Transfer Agents and zero or more User Agents. Each participant would therefore implement P1 If the Management Domain(s) contributed by a participant included User Agents, that participant would also implement P2. Participants need not implement P3 or P5.

4. Participants would implement all essential optional user facilities. They may implement additional optional user facilities at their own discretion.

5. O/R names would be restricted to those forms that contain a Management Domain name, and, perhaps, a personal name.

6. Each User Agent would support the "IA5Text" encoded information type (i.e., simple text messages) Participants may implement other encoded information types, such as facsimile or voice, at their own discretion

7. Access to the Message Transfer Service by means of the Teletex service, using the P5 protocol and procedures defined in Draft Recommendation X.430, would not be included in the demonstration.

8. Participants would be free to design and build their User Agents user interfaces as they wish.

9. Each participant would implement the Basic Activity Subset of the Session Service, including the functional units listed in the previous section

<u>References</u>

[1] *Draft Recommendation X.215. Session Service Definition*
CCITT, 1984.

[2] *Draft Recommendation X 400 Message Handling Systems: System Model-- Service Elements*
CCITT, 1984.

[3] *Draft Recommendation X 401: Message Handling Systems. Basic Service Elements and Optional User Facilities*
CCITT, 1984

[4] *Draft Recommendation X 410. Message Handling Systems. Remote Operations and Reliable Transfer Server*
CCITT, 1984.

[5] *Draft Recommendation X.411. Message Handling Systems. Message Transfer Layer*
CCITT, 1984

[6] *Draft Recommendation X.420. Message Handling Systems. Interpersonal Messaging User Agent Layer*
CCITT, 1984.

[7] *Draft Recommendation X 430. Message Handling Systems. Access Protocol for Teletex Terminals*
CCITT, 1984

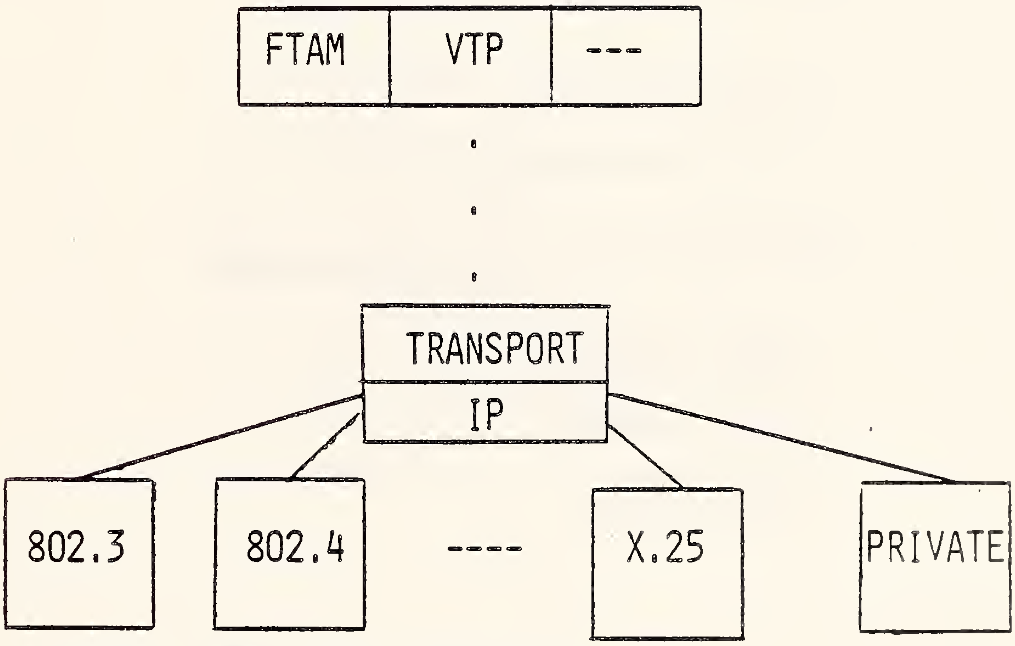
CONNECTIONLESS INTERNETWORK PROTOCOL

OVERVIEW

PROPERTIES

- DRAFT INTERNATIONAL STANDARD

- CONCATENATION OF DIFFERENT SUBNETWORK TECHNOLOGIES



- SIMPLE, EFFICIENT PROTOCOL

SERVICES

- PROVIDED TO TRANSPORT LAYER
  - UNIT DATA SEND
  - UNIT DATA RECEIVE
- PROVIDED BY THE NETWORK LAYER
  - UNIT DATA SEND
  - UNIT DATA RECEIVE
- PROVIDED BY THE LOCAL ENVIRONMENT
  - TIMER REQUEST
  - TIMER REPONSE
  - TIMER CANCEL

REQUIRED PROTOCOL FUNCTIONS

- PDU COMPOSITION
- PDU DECOMPOSITION
- HEADER ANALYSIS
- LIFE TIME BOUNDING
- ROUTING
- FORWARDING
- SEGMENTING
- REASSEMBLING
- DISCARDING
- ERROR REPORTING
- ERROR DETECTION
- PADDING

OPTIONAL PROTOCOL FUNCTION

- SECURITY
- SOURCE ROUTING
- ROUTE RECORDING
- QUALITY OF SERVICE

PDU STRUCTURE

- FIXED FIELDS
- ADDRESSES
- SEGMENTATION
- OPTIONS
- USER DATA

PDU TYPES

- DATA
- ERROR REPORTS

FORMAL DESCRIPTION

- EXTENDED FINITE STATE MACHINE LANGUAGE
- STATE TRANSITIONS PLUS PASCAL
- MODELS A SINGLE SERVICE REQUEST

CONFORMANCE

- FULL PROTOCOL

CHECKSUMS

- OVER THE HEADER ONLY
- GENERATING CHECKSUMS
- CHECKING CHECKSUMS
- ALTERING CHECKSUMS

MINUTES OF THE SIXTH WORKSHOP FOR IMPLEMENTORS OF OSI

U.S. Department of Commerce
National Bureau of Standards
Institute for Computer Sciences and Technology
Systems and Network Architecture Division
Gaithersburg, MD 20899

August 8 - 9, 1984

* Minutes of the first workshops were entitled "Proceedings of the ____ LAN/Transport Workshop Series."

CONTENTS

ABSTRTACT

The National Bureau of Standards, Institute for Computer Sciences and Technology (ICST) has sponsored a series of five LAN-Transport workshops for local area network implementors of International Organization for Standardization's (ISO) Class 4 Transport Protocol, ISO's File Transfer and Access Manipulation, and the Institute of Electrical and Electronics Engineers (IEEE) 802 compatible local area networks. The workshops focused on implementation techniques and strategies so that a multi-vendor demonstration of these protocols could occur at the 1984 National Computer Conference. As a follow-up to the LAN-Transport Workshops, ICST sponsored the Sixth OSI Implementor's Workshop to discuss the continued development of OSI computer network protocols.

Keywords: communication protocols; computer networks; local area networks; open systems interconnection;

SUMMARY

This report documents the Sixth OSI Implementors Workshop, one of a series of workshops for implementors of international standard protocols. The workshop reviewed the recent multi-vendor demonstration at the 1984 NCC, and began work on objectives for a second activity based upon additional international standard protocols. Work also began on a plan for a second workshop series to further develop Open Systems Interconnection.

1. WELCOME

John Heafner, NBS, welcomed the attendees to the workshop and introduced Maris Graube, Chairman of IEEE 802, as the workshop moderator. Maris gave a brief summary of the history of the past LAN-Transport Workshops.

An attendance list is included in Attachment 1.

2. APPROVAL OF AGENDA

The agenda was reviewed and the following additions were incorporated:

- European activities was added to item 5.
- Performance aspects of protocols was added to item 6.

The final agenda is included in Attachment 2.

3. RECAPITULATION OF 1984 NCC DEMO

Ron Yara, Intel, presented the following objectives developed for the 802.3 booth.

- Establish the ISO protocol suite as a "must" suite.
- Establish awareness of commitment to transport in particular and the OSI in general.
- Demonstrate that de jure standards work.
- Educate editors, OEMs and end users.
- Demonstrate that multiple vendors can work together.

Laurie Bride, BCS, stated that BCS is seeking a migratory path to the use of standards in their networks Their goal is to use fully integrated networks. They were interested in feedback from the NCC demonstration. They were satisfied with the outcome of the demonstration and are fully committed to continuing the work begun in previous workshops.

Ron Floyd presented GM's objectives for the demonstration:

- o Support of Standards: IEEE 802.4 Token Bus and NBS/ISO Class IV Transport,

- o To make other computer equipment suppliers and other industrial users aware of MAP,

- o Qualify a group of equipment suppliers for consideratin for GM MAP pilots, and

- o Provide a focal point/deadline for this work.

GM was pleased with the results of the demonstration.

John Heafner presented the NBS objectives for the NCC Demonstration as follows:

- o Those objectives for the 802.3 booth listed previously by Ron Yara;

- o Transfer test methodology: architecture, support software, tests; and

- o Bring users and suppliers together to determine the importance/priority of OSI protocols.

NBS is pleased with the results of the demonstration and is looking forward to future activities.

ICL has been trying to develop support in Europe for the activities of the workshop. They feel that the deadline of the demonstration helped to provide a focus for their efforts. The demonstration received some publicity in Europe and is expected to generate participation by European companies.

Charles River Data Systems sought to extend their use of standards. They were pleased with the progress made, but were aware that there is still much work to do.

DEC was interested in establishing OSI as de facto standards not just de jure standards, and in demonstrating DEC's commitment to OSI. DEC also believes that selecting the proper options for the various protocols is important. DEC feels that the momentum generated by the demonstration should be maintained.

IBM sought to advance their understanding of standard protocols. IBM wanted to promote the use of standards and were pleased with the progress made through the demonstration.

NCR wanted to increase their understanding of standards. NCR felt that the demonstration was successful in helping to achieve this goal, however it was only the beginning. NCR looks forward to future workshops to reinforce the work being done in the standards bodies.

Concord Data Systems sought publicity of standards. They felt that this goal was achieved and that the effort had benefitted their company.

Allen-Bradley wanted to show support for MAP and establish their reputation as a communications vendor. They felt the their goals were achieved in the demonstration.

Honeywell wanted to demonstrate their commitment to standards and to develop momentum for the standards process. They felt that their goals were accomplished and want to maintain the momentum for future work.

Motorola wanted to develop in-house expertise in standard protocols. They were pleased with the progress made toward this goal.

Intel wanted to show their support for standards in their products. They felt that this was achieved through the demonstration.

Resources Invested

Bob Blanc, NBS, stated that between $160,000 and $170,000 were spent for the booth and the brochure for the 802.3 demonstration.

ICL estimated that they invested 157,000 British Pounds to participate in the demonstration. This did not include much of the work already done on the development of an ISO Transport implementation or participation in standards meetings. A considerable portion of their expenses was in travel.

Intel invested 50 man-months in the development of products and 9 man-months specific to the demonstration.

Charles River Data Systems invested 24 man-months in the demonstration.

GM stated that its costs for the demonstration were very difficult to identify because the costs were integrated into corporate plans.

Leads Generated

GM reported that it had received 500 requests for MAP specifications and 2000 requests for literature.

NBS reported that they received about 600 requests for information. Many of the requests included requests for protocol specifications.

Ron Yara, Intel, said that the mailing labels were being made and would be distributed to the vendors as soon as possible.

Press Coverage

The signing ceremony for the cooperative agreement generated more press coverage than anticipated.

ICL requested a summary of press activity concerning workshop and demonstration activities. NBS said that it would provide this summary. Copies of the press kit and the press activity summary were distributed. The press summary is included in Attachment 12.

Residual Benefits

A residual benefit of the demonstration expressed was that the demonstration "got the ball rolling" for standards implementation. It was also noted that the demonstration caught the attention of PBX and other vendors, as displayed by their attendance at this meeting.

Criticisms

The criticisms of the two booths centered around the need for continuing work and were as follows:

- o The most recurrent criticism was that the protocols demonstrated were not products.
- o The booths were not interconnected.
- o The FTP demonstrated was not rich enough in capabilities.
- o The model demonstrated in the 802.3 booth was not clear.
- o The transport implemented was a subset of the full protocol.

Gary Workman, GM, made a presentation on GM's position on future participation in the workshops. (See attachment 5).

Performance Data

NBS collected data from all four days of the demonstration in the 802.3 booth. Rob Rosenthal, NBS, said that NBS had not analyzed the data from the demonstration. He offered copies of the data to those who were interested in performing their own analyses.

4. THE TARGET MARKETS

Sheldon Blauman, BCS, expressed interest in interconnecting the CSMA/CD booth and the token bus booth. The CSMA/CD booth would model the engineering/office environment with a connection to the token bus booth modelling the factory environment. BCS has many LANs with a wide geographic separation of facilities. The ISO internet protocol is vital to their use of networks.

5. GROUP OBJECTIVES

BCS suggested that development of the ISO internet and enhanced FTP should be the focus of future workshop activities. DEC expressed a desire that the primary emphasis of the workshop be protocol product development, with a secondary emphasis on demonstrations.

Joe St. Amand, Wang, presented a proposal for future workshop activity (see Attachment 3) as a strawman for discussion. Several changes and additions were made, as shown in the attachment.

A straw ballot was taken for interest in participation in a demonstration sometime in 1985. Twenty-one vendors expressed intrest in a show. Sixteen would participate if no changes were made to the old show with six of these sixteen being new vendors.

Ken Dymond, NBS, and John Heafner described the plans for Internet Protocol (IP) implementation and testing at NBS. The current estimate for having IP testing available is August 1985. NBS was interested in being an ISO intermediate system coupling IEEE 802.3 and IEEE 802.4 with measurement capabilities at the next demonstration.

NBS noted that there have recently been changes to the ISO Transport Standard. NBS will distribute the update information for Transport.

James Isaak, Charles River Data Systems, noted that the IEEE is forming a committee to standardize UNIX. Maris Graube, Tektronix, suggested incorporating ISO protocols into this standardization effort. (See Attachment 9, September 7 of agenda.)

Using the Wang proposal as a basis and after considerable discussion, a priority list for future workshop activity was produced (Attachment 5). The list incorporated updates and enhancements to the protocols already developed and various special interest subnets and applications.

John Heafner offered to provide contacts for protocol specifications. A list of addresses for contacts for the various standards documentations was distributed (see Attachment 6).

6. STRATEGIES/TACTICS TO ACHIEVE OBJECTIVES

A proposed set of workshop objectives and goals are included in Attachments 7 and 8. Decision on a final set of objectives was deferred to the next meeting.

7. SCHEDULE OF WORKSHOPS

A desire for tutorial sessions to help bring new workshop participants up to date was expressed. The number of attendees interested in participating in such sessions did not seem to justify the effort required. As an alternative, John Heafner said he will distribute a recent paper on experiences gained from transport and FTP testing for the 1984 NCC demonstrations.

A proposed agenda for the next workshop was developed (see Attachment 9). The next meeting was scheduled for September 5-7, 1984 in Gaithersburg, Maryland.

The goals for the next meeting are included in Attachment 10. The tutorial will be defered to a later meeting, however the paper John Heafner will distribute will cover many of the issues.

Those interested in attending the the Special Intrest Group Meetings should fill out and return the Special Interest Registration Form, Attachment 11, to allow planning of required meeting space.

ATTACHMENT 1.

REVISED
August 8, 1984

OSI/NBS Workshop

AC&C

Mike Seto

ALLEN BRADLEY

Bob Jones

AT&T

Lawrence Brown
Fred Burg
Doug Knisely
Steve Milton
Charles Young

BELL COMMUNICATIONS RESEARCH

George Chang
A. T. Galli

BOEING

Sheldon Blauman
Laurie Bride

BURROUGHS

Jon Becker

CHARLES RIVER DATA SYSTEMS

James Isaak
Richard Swee

CODEX

Paula Belair
Charles Wade

COMSAT

Mark Neibert

CONCORD DATA SYSTEMS

Mike Champa

DATA GENERAL

Lyman Chapin

DIGITAL EQUIPMENT

Tony Lauck
Gail Poulter

FORD MOTOR CO.

Jay Jeyabalan

FLORIDA STATE LEGISLATURE

Ed Levine
Glenn Mayne

GENERAL MOTORS

Ron Floyd
Mike Kaminski
Gary Workman

GOULD

Allen Brown

GOULD CSD

John Capurro

HONEYWELL

Bruce Carlson
Lea Quackenboss
S. Wales

ICL

Roy Cadwallder

IBM

A. W. Kleitsch
Jim Miller
Bob Yingling

LIBRARY OF CONGRESS

Ray Denenberg

MACOM

David Roos

MOTOROLA

Jim Clarkson

NCR

Gerald Brinda
Wood Wiles

NBS

Paul Amer
Bob Blanc
Ken Dymond
John Heafner
Dan Rorrer
Rob Rosenthal
Robert Toense
Mike Wallace
Evette Meni

NORTHERN TELECOM

Paul W. Masters
Edward Matthews

OLIVETTI

Francesco Cordera

SPERRY

W. P. Engstrom
Daniel Farrara
J. G. Nemanich

SYSTEMS DEVELOPMENT CORP.

Barbara Sternick

TEKTRONICS

Maris Graube

WANG

Joseph Holmes
Joseph St. Amand
Gerard White

XEROX

Juan Bulnes

ATTACHMENT 2

August 8 & 9, 1984

AGENDA

Implementors of ISO/NBS Open Systems Interconnection

Wednesday a.m.

1. Welcome (by John Heafner) and introduction of Moderator, Mr. Maris Graube

2. Approval of agenda

3. Recapitulation of 1984 NCC demo

   o Stated objectives (OSI, transport)

   o Resources invested

   o Leads generated (not marketing) press responses, feedback

   o Press coverage

   o Residual benefits

   o Were objectives achieved ?

   o Criticisms (Lack of products, limited FTP)

Wednesday p.m.

4. The target markets

   o What is the office systems model ?

   o What is the factory model ?

Wednesday p.m. & Thursday a.m.

5. Group Objectives

   o OSI and ISO/NBS protocols

   o Development of products, visibility of standards efforts, information/educational activities

   o Assuring multi-vendor compatibility

   o Follow-on activity

   o European activities

Thursday p.m.

6. Strategies/tactics to achieve objectives

   - o Workshop structure
   - o Laboratory resources
   - o Other resources

7. Schedule for workshops

ATTACHMENT 3

Follow-on Activity (Wang)

- topology
  - "802" LAN ↔ $\overset{X.25}{\text{WAN}}$ ↔ "802" LAN

    ↓

    (cross Atlantic, Pacific)

- comm. protocols
  - ISO 8802.3 +.4 LAN (10M b/s)
  - ISO 8802.2 (LLC I)

(15) 1. + ISO connectionless ISO CONFORM. SUBSET IP

- ISO/NBS COMPLETE + UPDATE Transport class IV

+ ISO SESSION SUBSET FOR LAYER 7

X.25

- application protocols

(13) 2. – ISO FTAM UPDATE AND EXTEND PRESENT PROTOCOL

(5) – CCITT X.400

P1 SUBSET

– Z39 INFO. RETRIEVAL

- 18 months lead time
- PRODUCTS, NOT JUST PROTOTYPES

ATTACHMENT 4

NBS WORKSHOP FOR OSI IMPLEMENTORS

AUGUST 8TH AND 9TH, 1984

GM PRESENTATION

GM INTENDS TO PARTICIPATE IN THE NBS INTERNET WORKSHOPS AND THE MULTI-VENDOR DEMONSTRATION BECAUSE OF:

1. GM'S INTEREST IN THE INTERNETWORKING OF LANS AND WANS.

2. GM'S INTEREST IN THE DEMONSTRATION OF CONTINUED MAP EVOLUTION AND DEVELOPMENTS.

3. GM'S INTEREST IN EXPEDITING IDENTICAL VENDOR IMPLEMENTATIONS OF OSI NETWORKING SOFTWARE.

GM BELIEVES THE INTERNET WORKSHOPS SHOULD BEGIN AS SOON AS POSSIBLE.

GM BELIEVES THE 1985 NCC WOULD BE THE IDEAL FORUM FOR AN INTERNETWORK PROTOCOL DEMONSTRATION. (CLNS PROTOCOL)

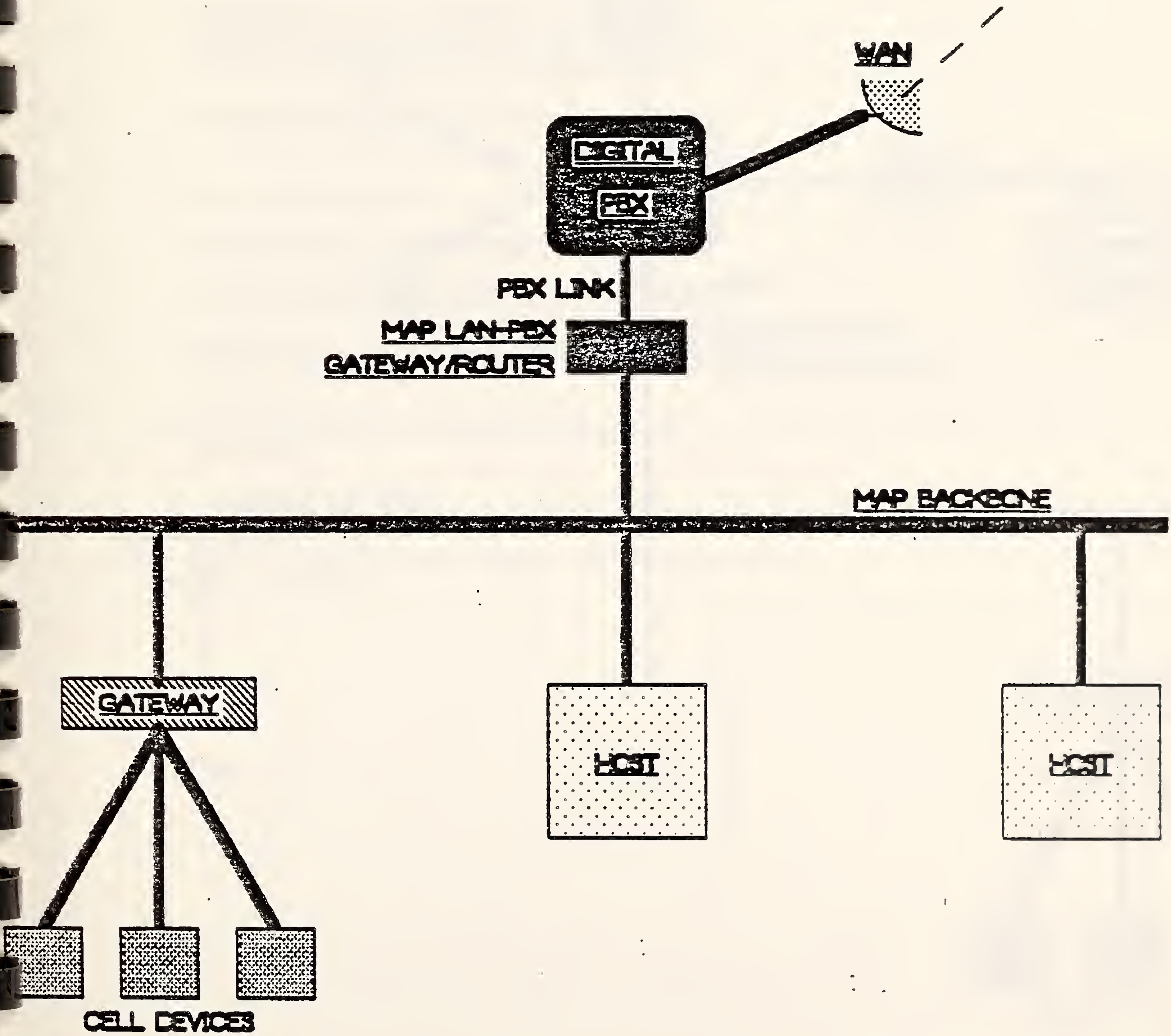
GM PROPOSES TO:

1. OBTAIN BOOTH SPACE AT THE 1985 NCC. (ALREADY APPLIED FOR)

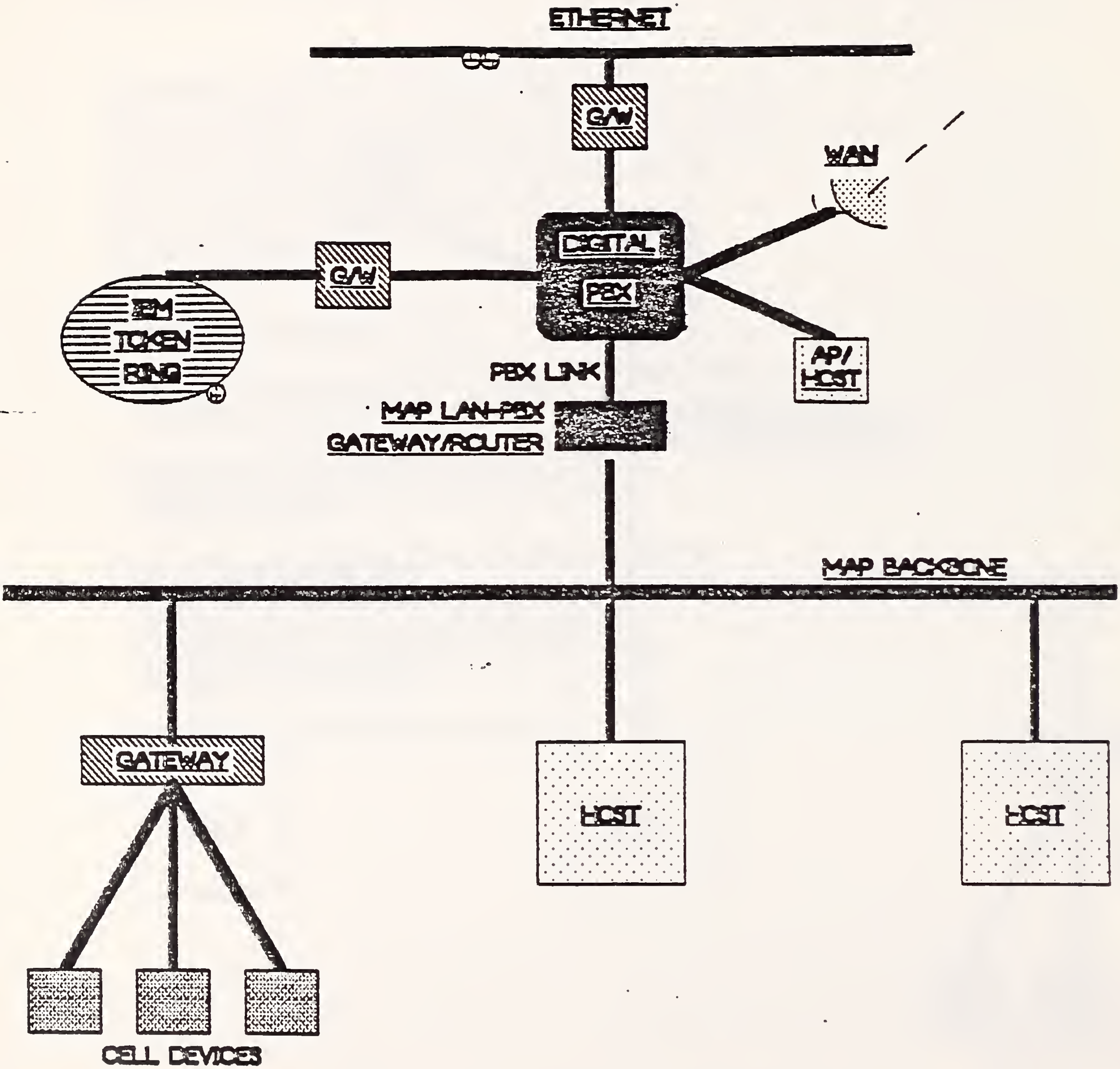
2. PERFORM REFERENCE TESTING FOR THE MAP (TOKEN-BUS LAN) PARTICIPANTS IN THE DEMONSTRATION.

3. INVOLVE PBX VENDORS TO IMPLEMENT INTERNET ROUTER CAPABILITIES.
(DISCUSSIONS ALREADY HELD WITH FOUR PBX MANUFACTURERS WHO HAVE EXPRESSED INTEREST IN COOPERATING WITH GM IN THIS DEVELOPMENT EFFORT.)

4. COORDINATE NCC BOOTH ACTIVITIES.



GATEWAY/ROUTER CONNECTION PBX-MAP LAN



GATEWAY/ROUTER CONNECTION PBX-MAP LAN

GM'S PROPOSAL FOR 1985 NCC DEMONSTRATION

-FOR ALL PARTICIPANTS

1. CLNS PROTOCOL

2. ENHANCED FILE TRANSFER CAPABILITIES (WRITE/CREATE, TRANSFER OF BIT STREAM FILES, CCITT MHSX409 PDU ENCODING)

3. "COMPLETE" TRANSPORT IMPLEMENTATION

-FOR MAP PARTICIPANTS

1. SIMPLE DIRECTORY INQUIRY CAPABILITY

2. TRIVIAL NETWORK MANAGEMENT CAPABILITY

3. UPGRADE OF TIM (TOKEN INTERFACE MODULE) HDLC INTERFACE

* 4. MODIFICATIONS TO THE MAP MESSAGING PROTOCOL

* Non MAP participants may want to implement the MAP messaging capability.

FUTURE CONCERNS

- PERFORMANCE ISSUES
- INTERNETWORK MANAGEMENT
- INTERNETWORK DIRECTORY SERVICES
- FILE TRANSFER UPGRADES
- VIRTUAL TERMINAL CAPABILITIES
- SESSION AND PRESENTATION PROTOCOLS

ATTACHMENT 5

General Interest

1- Encourage Product
2- Update Transport
3- internet
~~[illegible] file transfer~~
4- extend transport
5- session subset
6- extend file transfer

Special Interest Subnet

- 802.2
- 802.3
- 802.4
- 802 5 ?
- X.25 for WAN

Special Interest Layer ?

- 400 series msg subset
- GM MAP
- UNIX

ATTACHMENT 6

ANSI
ATTN: Ms. Fran Schrotter
ISO TC97/SC6 SECRETARIAT
1430 Broadway
New York, NY 10018

(212) 354-3343

ISO Internet Documents	Status	ISO/TC97/SC6
Network Service Definition	DIS 8348	N2990
Addendum to NSD Covering Connectionless Data Transmission	DIS 8348 DAD1	N3152
Addendum to NSD Covering Network Layer Addressing	DP8348 DAD2	N3134
Internal Organization of Network Layer	WD	N3141
Protocol for Providing the Connectionless Network Service	DIS 8473	N3154

*DAD = Draft Addendum

To Get a Copy of:

X, 400 X, 410
X, 401 X, 411
X, 408 X, 420
X, 409 X, 430

Call Rick Wildanger
(415) 496-6052

(Before Sept 1)

---

or (After Sept 1)
Stan Suk
(415) 494-4787

To get a
Corrigendum –
Call Jeanette Rusin
(201) 576-6217

Special Questions – Call Kim Ho
at same number
(201) 576-6217

ATTACHMENT 7

OBJECTIVES FOR NBS/OSI WORKSHOP

I. TO DEMONSTRATE INCREASING VALIDITY + POWER OF ISO STANDARDS BY:

A. ESTABLISHING TECHNICAL BASIS FOR VENDORS to IMPLEMENT:

1) INTERNET TRANSFER
2) FILE TRANSFER
3) SPECIAL INTERESTS

B) ESTABLISHING PROCESS FOR COMMUNICATING TO USERS + VENDORS TECHNICAL FINDINGS

II. DEVELOP PRODUCTS

III. MAINTAIN MOMENTUM OF '84 NCC DEMO + PROCESS

III. INCREASE LINKAGES TO OTHER USERS + VENDORS

ATTACHMENT 8

GOALS FOR NBS/OSI WORKSHOP PROCESS

I. DEMONSTRATE ABILITY OF ISO STANDARDS TO ACHIEU MULTI-VENDOR COMPATABILITY

II. DEVELOP + IMPLEMENT TESTING PROCEDURES

III. ASSESS ADEQUACY OF EXISTIN + EMERGING STANDARDS

IV. EDUCATE USERS + VENDORS RE: STANDARDS DEVELOPMEN PROCESS, STANDARDS, BENEFIT + IMPLEMENTATION

V. PROVIDE FORUM FOR SPECIAL INTEREST (MAP) PROT. WITHIN ISO ENVIRON MENT

ATTACHMENT 9

AGENDA:

PRE MEETING
- GET DOCUMENTS
- READ + UNDERSTAND
- ~~PRIOR WORKSHOP RESULTS TUTORIAL~~
- PREPARE STRAW~~MAN~~PERSON PROPOSALS

SEPT. 5
1ST DAY TRANSPORT + INTERNET

- TRANSPORT UPGRADE
- INTERNET TUTORIAL + DOCUMENT REVIE
- LIST OF OPTIONS + SELECT
- ESTIMATES OF TIME, EFFORT, CONSRAIN

EVENING: UPDATE ON ECMA, [illegible].

SEPT. 6
2ND DAY SESSION + FTP

- FTP TUTORIAL + DOCUMENT REVIEW
- INCREMENTAL FTP CAPABILITY DEFINITI
- SESSION SERVICE DEFINITION
- ESTIMATE OF TIME, EFFORT, CONSTRAINTS

EVENING: ESTIMATE SCHEDULES, DETERMIN
SHOWS

SEPT. 7

3rd DAY SPECIAL INTEREST

• MESSAGE	10
• X.25	15
• UNIX	9
• MAP	11

GOALS + OBJECTIVES

ATTACHMENT 10

TUTORIAL

- MISSION STATEMENT
- PREVIOUS AGREEMENTS
- TESTING
- WHAT WAS LEARNED
- WHAT TO DO NEXT

GOALS FOR NEXT MEETING

- TECHNICALLY REVIEW POINTS 2-6
- DETERMINE EFFORT, TIME, CONSTRAINTS
- REACH AGREEMENT ON STANDARD OPTIONS
- REVIEW SPECIAL INTEREST TOPICS
- REVIEW TEST PLAN

SPECIAL INTEREST GROUP REGISTRATION

Name: ____

Organization: ________________________________

Please check the Special Interest Group meetings you plan to attend.

__Subnet Special Interest Groups__

____ IEEE 802.2

____ IEEE 802.3

____ IEEE 802.4

____ X.25 for WAN

__Layer 7 Special Interest Groups__

____ X.400 Series Message Subset

____ GM MAP

____ UNIX

Please return as soon as possible to:

OSI WORKSHOP SERIES
Attn: Mary Lou Fahey
or Joan Wyrwa
National Bureau of Standards
Bldg. 225, Rm B226
Gaithersburg, MD 20899

Press coverage of NBS/industry program on networking standards

28 reporters attended the April 24 press briefing, including Business Week, Computerworld, New Scientist, UPI, WRC-TV, Business Times, and National Public Radio.

Over 50 reporters have made follow-up inquiries.

Following is a list of some of the general media that have carried stories:

Wall Street Journal (circulation 780,000)
Wall Street Journal (Wash., D.C. edition, 162,000)
Wall Street Journal (Chicago edition, 523,000)
Wall Street Journal (Cleveland edition, 523,000)
Wall Street Journal (European edition, circulation not available)
Business Week (900,000)
Discover (monthly, 506,300)
New York Times (873,255)
Financial Times (daily, London, England)
UPI business and finance wire (the UPI wire story was carried in newspapers across the country such as Sacramento, Ca. Union, 111,650, and Raleigh, N.C. News and Observer, 130,000)
San Francisco Examiner (150,000)
Dallas Times Herald (269,410)
Toronto, Ont. Globe & Mail (330,000)
Business Times, Entertainment and Sports Programming Network (ESPN is the largest cable network in the country reaching 30 million homes daily. Daily, over 300,000 business executives watch Business Times.)
National Public Radio
Washington Post (750,000)
Dow Jones Wire
WRC-TV, NBC in Washington, D.C.
KGW-TV, NBC in Portland, Ore.

Following is a list of some of the trade and technical press that have carried stories:

Computerworld (102,100)
Electronics (in two separate issues, 94,600)
New Scientist (London, England)
MIS Week (120,000)
Mini Micro Systems (95,400)
Electronic News (70,000)
Computer Decisions (129,400)
Computer Design (67,000)
Modern Materials Handling (109,450)
Electronic Design (101,276)
Information Systems News (100,500)
Purchasing (95,000)
Systems & Software (50,000)
Digital Design (56,000)

SUMMARY OF ARTICLES PUBLISHED:

Trade & Technical	- 46
Business	- 15
General	- 26
Total Articles	87

GM FILE TRANSFER PROPOSAL
SEPTEMBER 6, 1984

1. UPGRADE NCC'84 FTP TO ISO FILE TRANSFER SERVICE SUBSET FOR INTERNET DEMO

   - REQ'D CHANGES FOR ISO COMPATIBILITY
   - SUPPORT FOR F_READ AND F_WRITE
   - BINARY AND TEXT FILES

2. COMPLETE ISO FTAM IMPLEMENTATON FOR LONGER RANGE TIME FRAME

   - FILE ACCESS SERVICE SUBSET
   - FILE MANAGEMENT SERVICE SUBSETS (LIMITED AND ENHANCED)
   - ERROR RECOVERY SERVICE SUBSET
   - VIRTUAL FILESTORE STORAGE SUBSET

ISO FTP SUBSET FOR INTERNET DEMO

1.0 SERVICE PRIMITIVES

1. F_CONNECT, F_RELEASE, F_ABORT
2. F_SELECT, F_DESELECT
3. F_OPEN, F_CLOSE
4. F_READ AND F_WRITE *
5. F_DATA, F_DATA_END, F_TRANSFER_END
6. F_CANCEL
7. F_BEGIN_GROUP *, F_END_GROUP *

* INDICATES ADDITION

2.0 CHANGES REQUIRED FOR ISO COMPATIBILITY

1. X.409 ENCODING (ASN1) FOR FTP PDUs

   - NCC'84 FORMAT WAS INTERIM CHOICE TO BE COMPATIBLE WITH LOWER LAYERS

2. ADDITION OF CONCATENATION CONTROL

   - BEGIN, END GROUP PRIMITIVES AND SUPPORTING PCI

   - READ OR WRITE ACTIVITY INITIATED AS A SEQUENCE:

F_BEGIN_GROUP F_SELECT F_OPEN F_READ F_END_GROUP

F_BEGIN_GROUP F_SELECT F_OPEN F_WRITE F_END_GROUP

   - FILE RELEASED AS A SEQUENCE:

F_BEGIN_GROUP, F_CLOSE, F_DESELECT, F_END_GROUP

3.0 ISO FTP SUBSET RESTRICTIONS

1. A FILE SELECTION REGIME MAY HAVE AT MOST ONE OPEN AND ONE READ OR WRITE ACTIVITY

2. ONLY COMPLETE FILES MAY BE TRANSFERRED

ISO FTAM IMPLEMENTATION DEMO

4.0 ADDITIONAL SERVICE PRIMITIVES

1. F_LOCATE, F_ERASE
2. F_CREATE, F_DELETE, F_READ_ATTRIBUTE
3. F_CHANGE_ATTRIBUTE

5.0 ERROR RECOVERY SERVICE PRIMITIVES

1. F_RECOVER
2. F_CHECK
3. F_RESTART
4. F_CANCEL

6.0 IMPLEMENTATION OF VIRTUAL FILESTORE

- DEFINITION OF FILE STRUCTURE
- ALLOWS "RECORD LEVEL" FILE ACCESS

INTERNAL ORGANIZATION OF THE NETWORK LAYER

DEFINES ARCHITECTURAL ORGANIZATION OF THE NETWORK LAYER

PROVIDES MAPPING OF ABSTRACT ORGANIZATION TO "REAL WORLD" COMPONENTS

IDENTIFIES AND CATEGORIZES THE FUNCTIONS PERFORMED BY NETWORK LAYER PROTOCOLS

PROVIDES A UNIFORM FRAMEWORK FOR DESCRIPTION OF THE OPERATION OF THE NETWORK LAYER

EXTENDS TERMINOLOGY

DEALS WITH COMPLEX "REAL WORLD"

ROLES OF NETWORK LAYER PROTOCOLS

SUBNETWORK INDEPENDENT CONVERGENCE PROTOCOLS (SNICP)

- OFFERS OSI NETWORK SERVICE ON SUBNETWORK INDEPENDENT BASIS
- MAY BE INTERNETWORK PROTOCOL, SET OF RULES FOR COORDINATING SUBNETWORK SERVICES, OR NULL

SUBNETWORK DEPENDENT CONVERGENCE PROTOCOLS (SNDCP)

- PROVIDES SERVICE REQUIRED BY SNICP, OR PROVIDES OSI NETWORK SERVICE
- OPERATES OVER SNAcP
- MAY BE:
    - EXPLICIT PROTOCOL
    - SET OF RULES (E.G., FOR RUNNING CLIP OVER X.25)
    - NULL

ROLES OF NETWORK LAYER PROTOCOLS (CONT'D)

SUBNETWORK ACCESS PROTOCOLS (SNAcP)

- WHATEVER IS USED TO ACCESS A SPECIFIC SUBNETWORK
- MAY BE:
    - EXISTING SUBNETWORK PROTOCOL (ARPANET 1822, ...)
    - STANDARD PROTOCOL (X.25, ...)
    - NULL (IEEE 802, ...)
    - PRESENT ONLY DURING CONNECTION ESTABLISHMENT AND TERMINATION (X.21)
    - OR ...

GENERALLY

- AT LEAST ONE NETWORK LAYER PROTOCOL MUST BE PRESENT
- RECURSIVE USE OF PROTOCOL ROLES IS POSSIBLE

APPROACHES TO NETWORK LAYER INTERCONNECTION

HOP BY HOP ENHANCEMENT

- SNDCP INDIVIDUALLY ENHANCES EACH SUBNETWORK IN A CHAIN TO OFFER OSI NETWORK SERVICE
- SNICP CONSISTS OF RELAY AND ROUTING RULES FOR CONCATENATING SUBNETWORK SERVICES
- COULD IN PRINCIPLE BE CONNECTIONLESS OR CONNECTION-MODE
- IMPLICIT CONNECTION-MODE (X.25) ORIENTATION

INTERNETWORK PROTOCOL

- OPERATES AS END-TO-END PROTOCOL
- SUBNETWORKS MAY BE DIVERSE
- SNDCP MAY BE REQUIRED IN SOME CASES (E.G., RULES TO MANAGE X.25 CONNECTIONS)
- COULD IN PRINCIPLE BE CONNECTIONLESS OR CONNECTION-MODE
- CURRENTLY IMPLICITLY INTENDED FOR CONNECTIONLESS PROTOCOL (DIS 8473)

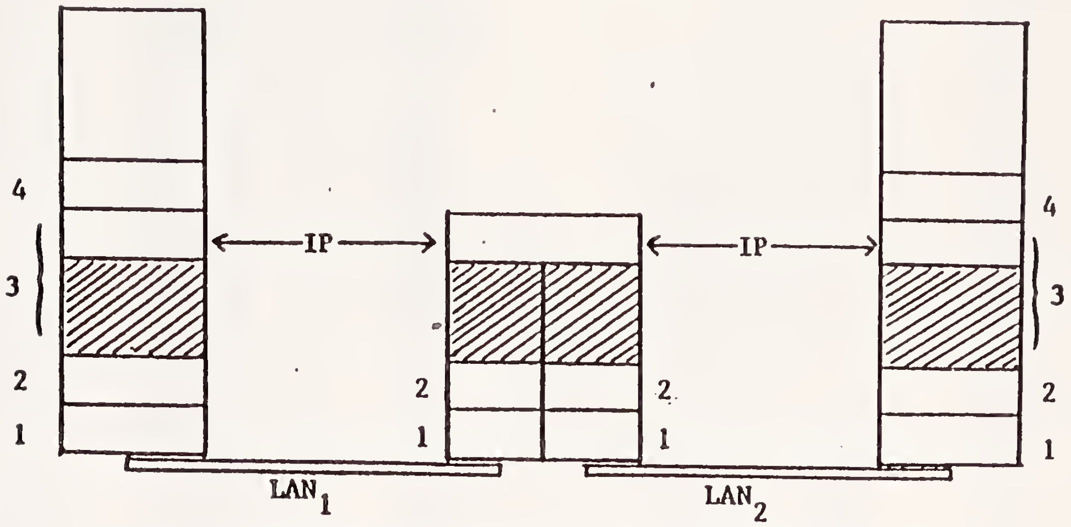
IOOTNL DOCUMENT GIVES EQUAL TREATMENT OF BOTH APPROACHES

EXAMPLE ARCHITECTURE FOR INTERNETWORK PROTOCOL

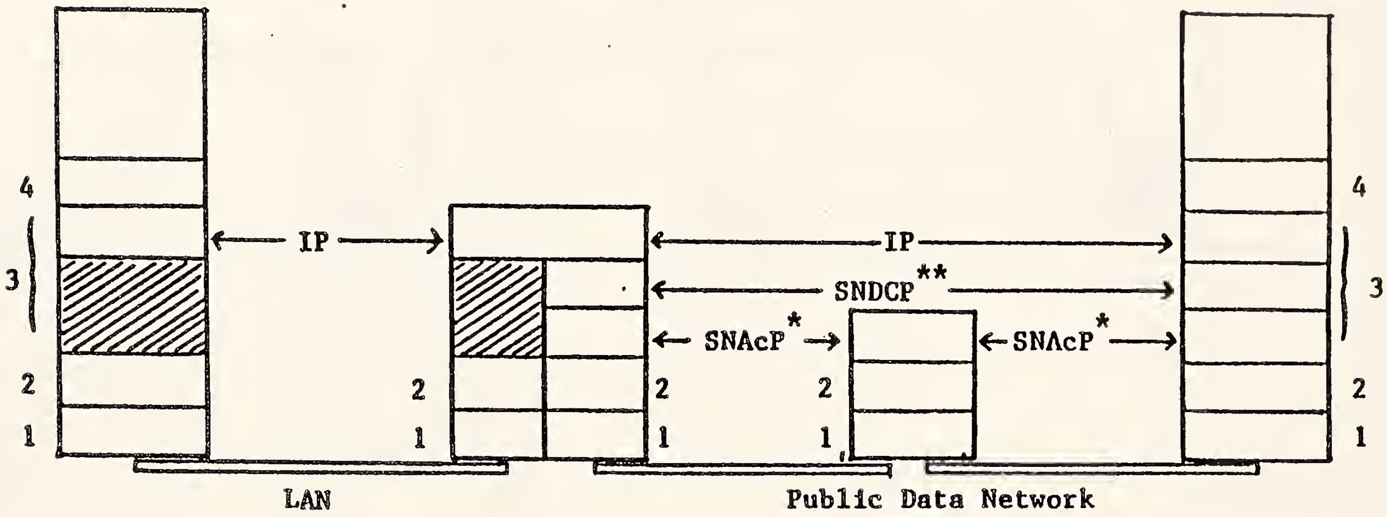
SNICP <==> CONNECTIONLESS INTERNETWORK PROTOCOL (ISO DIS 8473)

SNDCP <==>
{ NULL (OVER LANS)
RULES FOR CONNECTION MANAGEMENT, ETC (OVER PDNS)
ETC...

SNAcP <==>
{ NULL (OVER LANS)
X.25 PACKET LEVEL (OVER PDNS)
ETC...

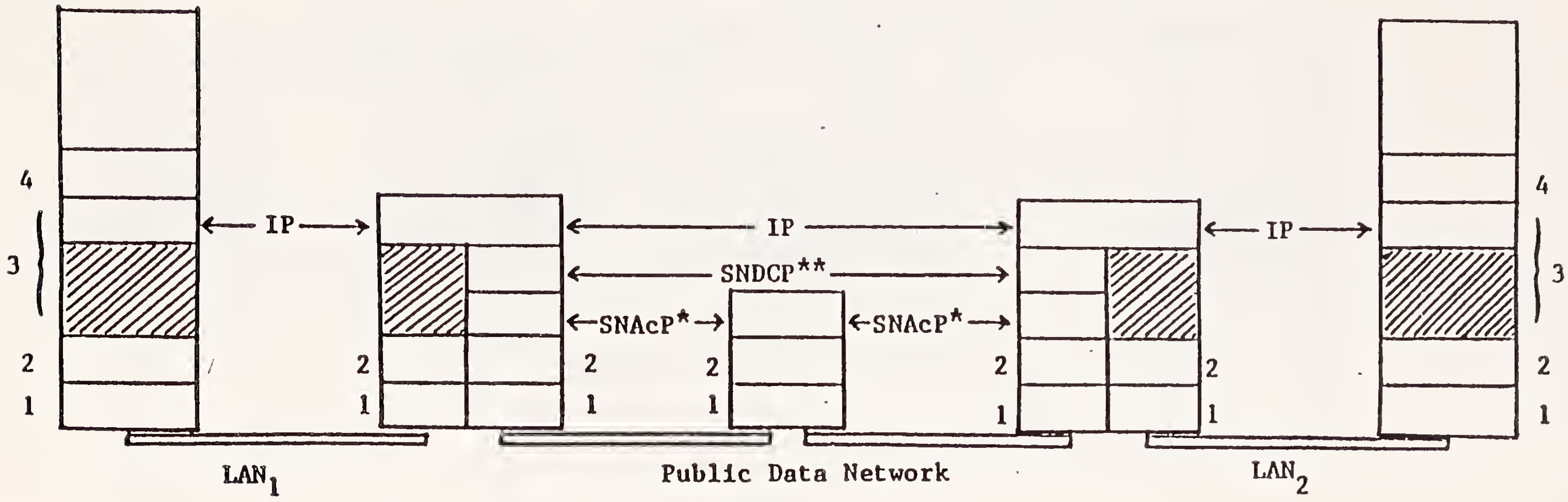


Use of Internetwork Protocol to Interconnect LANs



* X.25 Packet Level
** Rules for Running IP over X.25

Use of Internetwork Protocol for Interconnecting
Local Area Network with Public Data Network



* X.25 Packet Level
** Rules for Running IP over X.25

Use of Internetwork Protocol for Interconnecting
Two Local Area Networks Via a Public Data Network

TRANSPORT PROTOCOL

Added Features

There were two features of the transport protocol which would enhance the demo version:

1. Expedited Data - The expedited data service and the associated protocol mechanisms were not included in the demo. for the next demo it would be appropriate to include expedited data especially considering the multinetwork approach.

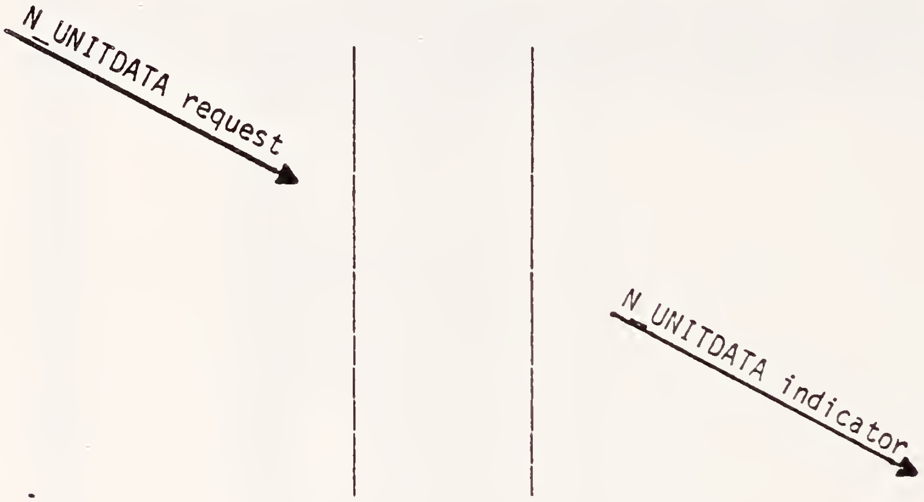
2. Negotiation During Connection Establishment - for the first demo, parameter values were selected to avoid negotiation. In the next demo it would be approppriate to include full parameter negotiation utilizing the connect negotiation rules.

Internetwork Protocol

Services Provided

The internetwork protocol (IP) provides a connectionless service. That is to say, it does not depend on the establishment of connections or virtual circuits between peer entities. The service provided is on a per request basis. There is no explicit or implicit relationship between service requests. Additionally, there is no confirmation of success or failure to the service requestor.

The service provided by the IP consists of one service interaction:



Associated with the service primitives are Quality of Service Parameters (QOS). Each parameter describes a characteristic that is provided by the service.

1. Transit delay - the time between a request and indication,

2. Protection from Unauthorized Access - the extent to which protection is provided,

3. Cost,

4. Residual Error Probability - the likelihood that an NSDU will be lost, duplicated, or incorrectly delivered,

5. Priority, and

6. Source Routing - a specification of the path an NSDU is to take.

OSI Session Service

- CONCEPTS
- GENERAL RULES
- CONNECTION ESTABLISHMENT
- DATA TRANSFER
- CONNECTION RELEASE
- COLLISION RESOLUTION

CONCEPTS

- TOKENS
- SYNCHRONIZATION & DIALOGUE UNITS
- ACTIVITIES
- RESYNCHRONIZATION
- FUNCTIONAL UNITS & SUBSETS
- NEGOTIATION
- DATA CATEGORIES

TOKENS

- PERMIT ALTERNATING CONTROL OF SERVICES
- FOUR SESSION TOKENTS
  - DATA
  - RELEASE
  - SYNCH-MINOR
  - MAJOR-ACTIVITY
- TOKEN STATES
  - AVAILABLE
  - NOT AVAILABLE
- AVAILABLE SUB-STATES
  - ASSIGNED
  - NOT ASSIGNED

SYNCHRONIZATION & DIALOGUE UNITS

- SYNCHRONIZATION POINT TYPES
  - MINOR
  - MAJOR
- MAJOR SYNCH POINTS DELIMIT DIALOGUE UNITS
- MINOR SYNCH POINTS DELIMIT SUB-UNITS
- CONFIRMATION
  - EXPLICIT FOR MAJOR SYNCH
  - MAY BE EXPLICIT FOR MINOR SYNCH
- NO SEMANTICS ASSOCIATED WITH SYNCH POINTS

ACTIVITIES

- DISTINGUISH DIFFERENT PIECES OF LOGICAL WORK
- CONSIST OF ONE OR MORE DIALOGUE UNITS
- ONE ACTIVITY ON A CONNECTION AT ONE TIME
- SEVERAL ACTIVITIES MAY USE A CONNECTION SEQUENTIALLY
- AN ACTIVITY MAY SPAN MORE THAN ONE SESSION CONNECTION
- ACTIVITIES MAY BE INTERRUPTED AND RESUMED
- USERS MAY SEND DATA OUTSIDE AN ACTIVITY
- ACTIVITY END = MAJOR SYNCH POINT

RESYNCHRONIZATION

- SETS SESSION CONNECTION TO A DEFINED STATE
  - TOKENS
  - SYNCH POINT SERIAL NUMBER
- PURGES ALL UNDELIVERED DATA
- THREE OPTIONS
  - ABANDON
  - RESTART
  - SET
- SEMANTICS ARE USER DEFINED

FUNCTIONAL UNITS & SUBSETS

- FUNCTIONAL UNITS ARE LOGICAL GROUPINGS OF RELATED SERVICES
- FUNCTIONAL UNITS ARE INDIVIDUALLY NEGOTIABLE AT CONNECTION ESTABLISHMENT
- CERTAIN FUNCTIONAL UNITS IMPLY TOKEN AVAILABILITY
- TOKEN MANAGEMENT SERVICES ARE REQUIRED WITH TOKEN AVAILABILITY
- SUBSETS ARE COMBINATIONS OF THE KERNEL FUNCTIONAL UNIT TOGETHER WITH ANY OTHER SET OF FUNCTIONAL UNITS
- SUBSETS HAVE NO MEANING IN THE SESSION PROTOCOL

FUNCTIONAL UNITS

- KERNEL
- DUPLEX
- HALF-DUPLEX
- NEGOTIATED RELEASE
- EXPEDITED DATA
- TYPED DATA
- CAPABILITY DATA
- MINOR SYNCH
- MAJOR SYNCH
- RESYNCH
- EXCEPTIONS
- ACTIVITY MANAGEMENT

PREDEFINED SUBSETS

- BASIC COMBINED SUBSET
    - KERNEL
    - DUPLEX OR HALF-DUPLEX
- BASIC SYNCHRONIZED SUBSET
    - KERNEL
    - NEGOTIATED RELEASE
    - HALF-DUPLEX
    - TYPED DATA
    - MINOR & MAJOR SYNCH
    - RESYNCH
- BASIC ACTIVITY SUBSET
    - KERNEL
    - HALF-DUPLEX
    - TYPED DATA & CAPABILITY DATA
    - MINOR SYNCH
    - EXCEPTIONS
    - ACTIVITY MANAGEMENT

NEGOTIATION

- OCCURS DURING CONNECTION ESTABLISHMENT
- FUNCTIONAL UNITS ON CONNECTION
- INITIAL TOKEN SETTINGS
- INITIAL SYNCH POINT SERIAL NUMBER

DATA CATEGORIES

- NORMAL
- EXPEDITED
- TYPED
- CAPABILITY

GENERAL RULES

- TOKEN RESTRICTIONS
- NEGOTIATION RULES
- PRIMITIVE SEQUENCE
- SYNCH POINT SERIAL NUMBER MANAGEMENT

TOKEN RESTRICTIONS

- INITIATE RELEASE
  - ALL AVAILABLE TOKENS ASSIGNED
- SEND DATA
  - DATA TOKEN ASSIGNED (HALF-DUPLEX)
  - DATA TOKEN UNAVAILABLE (DUPLEX)
- GIVE TOKEN REQUEST
  - TOKEN ASSIGNED
- PLEASE TOKEN REQUEST
  - TOKEN UNASSIGNED
- ACTIVITY START, RESUME, END
  - DATA & SYNCH MINOR TOKEN UNAVAILABLE OR ASSIGNED
  - MAJOR-ACTIVITY TOKEN ASSIGNED

TOKEN RESTRICTIONS (CONTINUED)

- ACTIVITY INTERRRUPT, DISCARD
  - MAJOR-ACTIVITY TOKEN ASSIGNED
- MINOR SYNCH REQUEST
  - DATA TOKEN UNAVAILABLE OR ASSIGNED
  - MINOR SYNCH TOKEN ASSIGNED
- MAJOR SYNCH REQUEST
  - DATA & SYNCH MINOR TOKENS ASSIGNED OR UNAVAILABLE
  - MAJOR-ACTIVITY TOKEN ASSIGNED
- EXCEPTION REPORT REQUEST
  - DATA TOKEN UNASSIGNED
- CAPABILITY DATA
  - DATA & SYNCH MINOR TOKENS ASSIGNED OR UNAVAILABLE
  - MAJOR-ACTIVITY TOKEN ASSIGNED

FUNCTIONAL UNIT NEGOTIATION

- REQUESTOR PROPOSES A SET OF FUNCTIONAL UNITS
- ACCEPTOR ALSO PROPOSES A SET OF FUNCTIONAL UNITS
- HALF-DUPLEX & DUPLEX MAY NOT BOTH BE PROPOSED BY ACCEPTOR
- CAPABILITY DATA MAY BE PROPOSED ONLY IF ACTIVITY MANAGEMENT IS PROPOSED
- EXCEPTION REPORTING MAY BE PROPOSED ONLY IF HALF-DUPLEX IS PROPOSED
- SELECTED FUNCTIONAL UNITS ARE THE INTERSECTION OF THE REQUESTOR AND ACCEPTOR PROPOSALS

INITIAL SYNCH POINT SERIAL NUMBER

- PROPOSED WITH MINOR SYNCH, MAJOR SYNCH, OR RESYNCH FUNCTIONAL UNITS WHEN ACTIVITY MANAGEMENT IS NOT PROPOSED

- ACCEPTOR SELECTING ANY OF THE PROPOSED FUNCTIONAL UNITS RETURNS A VALUE THAT WILL BE INITIAL SYNCH POINT FOR CONNECTION

- ACTIVITY MANAGEMENT FUNCTIONAL UNIT IMPLIES INITIAL SYNCH POINT SERIAL NUMBER OF ONE

INITIAL TOKEN ASSIGNMENTS

- WHEN A FUNCTIONAL UNIT REQUIRING TOKEN IS PROPOSED AN INITIAL TOKEN LOCATION IS ALSO PROPOSED
- POSSIBILITIES: CALLING SIDE, CALLED SIDE, CALLED CHOICE
- WHEN FUNCTIONAL UNIT IS SELECTED TOKEN REVERTS TO SIDE PROPOSED BY CALLER EXCEPT -
- IF CALLER SAID CALLED CHOICE, TOKEN REVERTS TO SIDE PROPOSED BY CALLED USER

PRIMITIVE SEQUENCING

- USER REQUESTS & RESPONSES ARE DELIVERED BY PROVIDER IN THE ORDER SUBMITTED EXCEPT -

- SEVERAL REQUESTS WHICH MAY BE DELIVERED EARLIER THAN NORMAL

- EXCEPTIONS INCLUDE:

  S-EXPEDITED-DATA

  S-RESYNCHRONIZE

  S-ACTIVITY-INTERRUPT

  S-ACTIVITY-DISCARD

  S-U-ABORT

SYNCH POINT SERIAL NUMBER MANAGEMENT

- DEFINED AS OPERATIONS ON FOUR ABSTRACT VARIABLES - V(M), V(A), V(R), Vsc

- V(A) - LOWEST SERIAL NUMBER TO WHICH SYNCH POINT CONFIRMATION IS EXPECTED

- V(M) - NEXT SERIAL NUMBER TO BE USED

- V(R) - LOWEST SERIAL NUMBER TO WHICH RESYNCH RESTART IS PERMITTED

- Vsc - CONTROLS RIGHT OF USER TO ISSUE MINOR SYNCH POINT CONFIRMATIONS

- SYNCH & RESYNCH REQUESTS, RESPONSES, INDICATIONS, & CONFIRMATIONS EXAMINE THESE VARIABLES AND CAUSE OPERATIONS TO BE PERFORMED ON THEM

SESSION CONNECT PARAMETERS

- CONNECTION INDENTIFIER
- CALLING/CALLED SSAP
- RESULT
- QOS
- SESSION FUNCTIONAL UNIT REQUIREMENTS
- INITIAL SYNCH POINT SERIAL NUMBER
- INITIAL TOKEN ASSIGNMENTS
- USER DATA (TO 512 OCTETS)

SESSION DATA TRANSFER

- UNLIMITED NORMAL DATA PER SDU
- EXPEDITED SDU 1 TO 14 OCTETS
- UNLIMITED TYPED DATA PER SDU
- CAPABILITY DATA SDU 1 to 512 OCTETS
- NORMAL DATA SUBJECT TO TOKEN RESTRICTIONS
- CAPABILITY DATA SUBJECT TO TOKEN RESTRICTIONS AND ACTIVITY CONTEXT
- TYPED & EXPEDITED DATA ARE FULL-DUPLEX

TOKEN MANAGEMENT

- PLEASE TOKENS
  - LIST OF REQUESTED TOKENS
  - UP TO 512 OCTETS USER DATA
- GIVE TOKENS
  - LIST OF SURRENDERED TOKENS
- GIVE CONTROL
  - SURRENDERS ALL AVAILABLE TOKENS
  - ONLY PERMITTED WHEN ACTIVITY MANAGEMENT IS SELECTED AND NO ACTIVITY IS IN PROGRESS

SYNCH POINTS

- MINOR SYNCH POINT
  - EXPLICIT OR OPTIONAL CONFIRMATION
  - SERIAL NUMBER
  - UP TO 512 OCTETS USER DATA
- MAJOR SYNCH POINT
  - SERIAL NUMBER
  - UP TO 512 OCTETS USER DATA
- RESYNCH
  - TYPE: ABANDON, RESTART, SET
  - SERIAL NUMBER
  - TOKENS & LOCATIONS
  - UP TO 512 OCTETS USER DATA

EXCEPTION REPORTING

- PROVIDER EXCEPTION
  - REASON
  - NO DATA TRANSFER OR SYNCH POINTS UNTIL ERROR IS CLEARED
  - CLEARED BY RESYNCH, ABORT, INTERRUPT, DISCARD, OR GIVING DATA TOKEN

- USER EXCEPTION
  - REASON
  - UP TO 512 OCTETS USER DATA
  - WORKS ONLY IN HALF-DUPLEX MODE
  - NO DATA TRANSFER OR SYNCH POINTS UNTIL ERROR IS CLEARED
  - SAME CLEARING PROCEDURES AS FOR PROVIDER EXCEPTION

ACTIVITY MANAGEMENT

- START
  - ACTIVITY IDENTIFIER
  - UP TO 512 OCTETS USER DATA
- END
  - SERIAL NUMBER
  - UP TO 512 OCTETS USER DATA
  - EQUIVALENT TO MAJOR SYNCH POINT
- DISCARD
  - REASON
  - DATA WILL BE LOST
- INTERRUPT
  - REASON
  - UNDELIVERED DATA WILL BE LOST
- RESUME
  - NEW & OLD ACTIVITY IDENTIFIERS
  - SERIAL NUMBER
  - OLD SESSION CONNECTION IDENTIFIER
  - UP TO 512 OCTETS USER DATA

CONNECTION RELEASE

- o ORDERLY RELEASE
    - RESPONSE RESULT (IF NEGOTIATED)
    - UP TO 512 OCTETS USER DATA
- o USER ABORT
    - UP TO 9 OCTETS USER DATA
- o PROVIDER ABORT
    - REASON

COLLISION RESOLUTION

- HIERARCHY OF REQUESTS
    - ABORT
    - DISCARD
    - INTERRUPT
    - RESYNCH (ABANDON)
    - RESYNCH (SET)
    - RESYNCH (RESTART)
    - USER EXCEPTION

- RESYNCH (ABANDON) COLLISIONS RESOLVED IN FAVOR OF CALLING USER

- RESYNCH (RESTART) COLLISIONS RESOLVED IN FAVOR OF LOWEST SERIAL NUMBER OR CALLING USER FOR EQUAL SERIAL NUMBERS

- RESYNCH (SET) COLLISIONS RESOLVED IN FAVOR OF CALLING USER

Proposal for Sessions

Protocol

We are not currently working on a Session implementation. This section gives our "ballpark" estimates of how long a reasonably experienced implementor would take for a hand coded version.

FTAM's usage of the Session layer is in a state of confusion. However, if we take as a given that at the workshop we should support the ANSI position, then FTAM will require the following Session "functional units":

- Kernel functional unit
  - S_CONNECT (req, ind, resp, conf)
  - S_DATA (req, ind)
  - S_RELEASE (req, ind, resp, conf)
  - S_U_ABORT (req, ind)
  - S_P_ABORT (ind)
- Duplex functional unit
- Minor synchronize functional unit
  - S_SYNC_MINOR (req, ind, resp, conf)
  - S_TOKEN_GIVE (req, ind)
  - S_TOKEN_PLEASE (req, ind)
- Resynchronize functional unit
  - S_RESYNCHRONIZE (req, ind, resp, conf)

These requirements are documented on page 3 of ISO DP 8571/4.

The X.400 series requires the much more complex BAS subset of session. Specifically the following features are required:

- Kernel functional unit
  - S_CONNECT (req, ind, resp, conf)
  - S_DATA (req, ind)
  - S_RELEASE (req, ind, resp, conf)
  - S_U_ABORT (req, ind)
  - S_P_ABORT (ind)
- Half-duplex functional unit
  - S_TOKEN_GIVE (req, ind)
  - S_TOKEN_PLEASE (req, ind)
- Minor synchronize functional unit
  - S_SYNC_MINOR (req, ind, resp, conf)
  - S_TOKEN_GIVE (req, ind)
  - S_TOKEN_PLEASE (req, ind)
- Exceptions functional unit
  - S_P_EXCEPTION_REPORT (ind)
  - S_U_EXCEPTION_REPORT (req, ind)

```
Activity management functional unit
        S_ACTIVITY_START (req, ind)
        S_ACTIVITY_RESUME (req, ind)
        S_ACTIVITY_INTERRUPT (req, ind, resp, conf)
        S_ACTIVITY_DISCARD (req, ind, resp, conf)
        S_ACTIVITY_END (req, ind, resp, conf)
        S_TOKEN_GIVE (req, ind)
        S_TOKEN_PLEASE (req, ind)
        S_CONTROL_GIVE (req, ind)
```

These requirements are documented on page 18 of CCITT DR X.410.

Implementation of the union of these two subsets of session omits the negotiated release, expedited data, typed data, capability data exchange and major synchronize functional units. In addition, we assume that the implementation will not include the extended concatenation capability (which is an optional, negotiated feature of a session).

GM PRESENTATION FOR THE NATIONAL BUREAU OF STANDARDS SPONSORED MULTI-VENDOR WORKSHOP ON OSI STANDARDS IMPLEMENTATION

SEPTEMBER 5, 1984

GAITHERSBURG, MD

CLNS PROTOCOL:

DOCUMENTS REVIEWED:

1. OCTOBER, 1983 ISSUE OF ISO CONNECTIONLESS IP
2. MAY, 1984 ISSUE OF ISO DIS 8473 "THE DATA COMMUNICATIONS PROTOCOL FOR PROVIDING THE CONNECTIONLESS-MODE NETWORK SERVICE"
3. JULY 13, 1984 ANSI X3S3.3 84-169 "DRAFT US COMMENTS ON ISO DIS 8473"

TYPES OF CLASSIFICATIONS OF FUNCTIONS

TYPE 1: MUST BE SUPPORTED (INTRINSIC)

TYPE 2: IF NOT SUPPORTED, RETURN ERROR PDU AND DISCARD PDU SENT.

TYPE 3: PDU NOT DISCARDED IF FUNCTION NOT SUPPORTED.

NOTE: ALL OPTIONS IN THE DOCUMENT ARE ONE OF THE ABOVE TYPES.

THE INTERNETWORKING PROTOCOL (IP) TITLE HAS BEEN CHANGED TO:

"THE DATA COMMUNICATIONS PROTOCOL FOR PROVIDING THE CONNECTIONLESS-MODE NETWORK SERVICE"

TYPES OF CLASSIFICATIONS OF FUNCTIONS (CONT'D)

IN REGARD TO THE TYPE 3 FUNCTIONS:

THE RETURN OF AN ERROR_PDU WHEN THESE FUNCTIONS ARE NOT SUPPORTED IS A VIOLATION OF THE SPECIFICATION. IT WOULD BE BENEFICIAL, HOWEVER, IF ALL END-SYSTEMS WOULD IMPLEMENT THE IDENTICAL SET OF THESE TYPE 3 FUNCTIONS, WHICH WILL BE OUTLINED LATER. IN ADDITION, THE USEFUL ERROR_PDU LIST HAS BEEN EXPANDED TO INCLUDE THESE TYPE 3 ERROR_TYPES FOR FUTURE USE. BECAUSE OF THIS, TYPE 3 FUNCTIONS WHICH ARE NOT SUPPORTED BY EITHER AN END OR INTERMEDIATE SYSTEM SHOULD RETURN AN ERROR_PDU FOR THIS DEVELOPMENT WORK, WHICH MUST BE SUPRESSED AT THE COMPLETION OF SUCH WORK IN ORDER TO MEET ALL STANDARDS WITHIN THE SPECIFICATION.

SPECIFICATION OF THE PROTOCOL
TYPE 1 FUNCTIONS
SOLICITED PROVISIONS BY GENERAL MOTORS

1. PDU Composition:
   - concerns given at ANSI X3S3.3
   - format is acceptable

2. PDU Decomposition:
   - concerns given at ANSI X3S3.3
   - format is acceptable

3. Header Format Analysis:
   - indicate a standard (first) version of the protocol via the network layer protocol ID.
   - format is acceptable

4. PDU Lifetime Control Function:
   - lifetime agreed upon by all originating network-entities (500 ms units)
   - based on the topology of the network as well as routing algorithms to be applied
   - based on worst case performance criteria, numbers of nodes and/or hops
   - without it, we may parse a packet forever

5. Route PDU Function:
   - need for this function is inherent in any network layer implementation
   - without this function, we have no functional protocol

6. Forward PDU Function:

- need for this function is inherent in any network layer implementation
- requires segmentation/reassembly functions to be implemented

7. Segmentation Function:

- needed for forward PDU Function unless a maximum Data PDU size which is acceptable by all participating subnets is enforced.
- unrealistic to perform the above, so implement this function

8. Reassembly Function:

- implied as segmentation is used, we must re-form the Initial PDU, whether Data or Error type
- requires the correct selection of PDU lifetime such that all transmitted PDU's that were segmented will be reassembled within this "reasonable" timeframe
- also implies a need for error reporting

9. Discard PDU Function:

- required if any errors will occur, this mechanism must be provided to enable a standard recovery procedure.
- format acceptable

10. Error Reporting Function:

- will be extremely useful for diagnostic purposes
- the source network entity must set the error report flag to "one"
- the error_types in section 8.3.1, along with those in ANSI X3S3.3 84-169 should suffice for explaination of these types, but we may wish to add additional non-standard error_types for diagnostic purposes, which may be deleted after a functional network is provided for.

11. PDU Header Error Detection

- error rate of LAN is low already
- the end-end checksum is not required at transport, not here either
- the LANs have checksums in effect at the Data Link Layer already, and this is adequate for our needs.
- obviously, we will accept and process all incoming checksums accordingly
- no checksum provided for Data PDU (because not in xport implementation)
- if header is being altered inadveratantly in the course of processing, the mechanism provided for checksum is good, especially page 84, section C5.
- useful for debugging and system development
- therefore, set the two octets to "zero" for this Function, but we are open to utilizing the function for this system development work, providing its overhead as well as its function can be supressed after this work is completed.

12. Padding Function:

- is a type 3 function, as outlined in note 2, p. 25

SPECIFICATION OF THE PROTOCCL

TYPE 2 FUNCTIONS

SOLICITED PROVISIONS BY GENERAL MOTORS

1. Security Function:

- we have no requirement to implement at the present time
- we would like to discard incoming PDUs, and therby return an error PDU "unsupported_security_option"

2. Complete Source Routing Function:

- we have no need to implement at the present time
- the function is a good intermediate system diagnostic tool, and we are open its utilization for this purpose, as long as the parameter can be supressed after initial development work.

SPECIFICATION OF THE PROTOCOL

TYPE 3 FUNCTIONS

SOLICITED PROVISIONS BY GENERAL MOTORS

1. Padding Function:
   - useful for ease of implementation, diagnostic tool
   - useful for screening or commenting-out any unapplicable data fields
   - implement this function

2. Partial Source Routing Function:
   - as Complete source routing, it is useful as a diagnostic tool
   - not as useful for the above as complete source routing
   - do not implement

3. Priority Function:
   - this is important in the LAN – WAN scenarios
   - there will be messages which are of greater importance than others
   - we need a mechanism to support this requirement, so that they will be processed first
   - the format of 16 priority levels is acceptable
   - we would like the highest prioity to be processed first, FIFO within any given priority level
   - implement this function

4. Recording of Route Function:
   - not required at this time
   - useful for diagnostics, especially in conjunction with complete source routing
   - do not implement, unless a great need for additional diagnostic tools is forseen. Supress after development work is completed.

5. QOS Maintenance Function:

- useful for continued qos on a LAN-WAN connection
- important for continued qos within intermediate systems
- assignment of another qos to a requested qos is acceptable
- implement this function

MATRIX OF THE CLNS PROTOCOL IMPLEMENTATION FEATURES

SOLICITED PROVISIONS BY GENERAL MOTORS

	IMPLEMENT??		IMPLEMENT THEN SUPRESS
PDU COMPOSITION	YES		
PDU DECOMPOSITION	YES		
HDR FORMAT ANALY	YES, VERSION 1		
PDU LIFETIME	YES		
ROUTE PDU	YES		
FORWARD PDU	YES		
SEGMENTATION	YES		
REASSEMBLY	YES		
DISCARD	YES		
ERROR REPORT	YES		
HDR ERROR DETECT		NO	YES
SECURITY		NO	
COMP. SOURCE RTING		NO	YES
PART SOURCE RTING		NO	
PRIORITY	YES		
RECORD OF ROUTE		NO	
QOS	YES		
PADDING	YES		

ISO File Transfer, Access, and Management: Model, Services, and Protocol

James C. Berets
Bolt Beranek and Newman Inc.

STATUS OF FTAM

Work internationally progressing in ISO/TC97/SC21/WG5 (recently moved from SC16).

Work in U.S. progressing in ANSI/X3T5.5.

FTAM recently balloted as ISO Draft Proposal 8571.

Second DP ballot probable in early 1985.

Mapping to Session pass-through services still under discussion.

THE VIRTUAL FILESTORE

Descriptive model to uniformly represent the properties of filestores and the files contained in those filestores.

Allows differences in filestore implementation to be absorbed into a local mapping.

Virtual filestore representation not limited to real filestores.

Virtual filestore defines: file access structure, file and activity attributes, actions on files.

File access structure

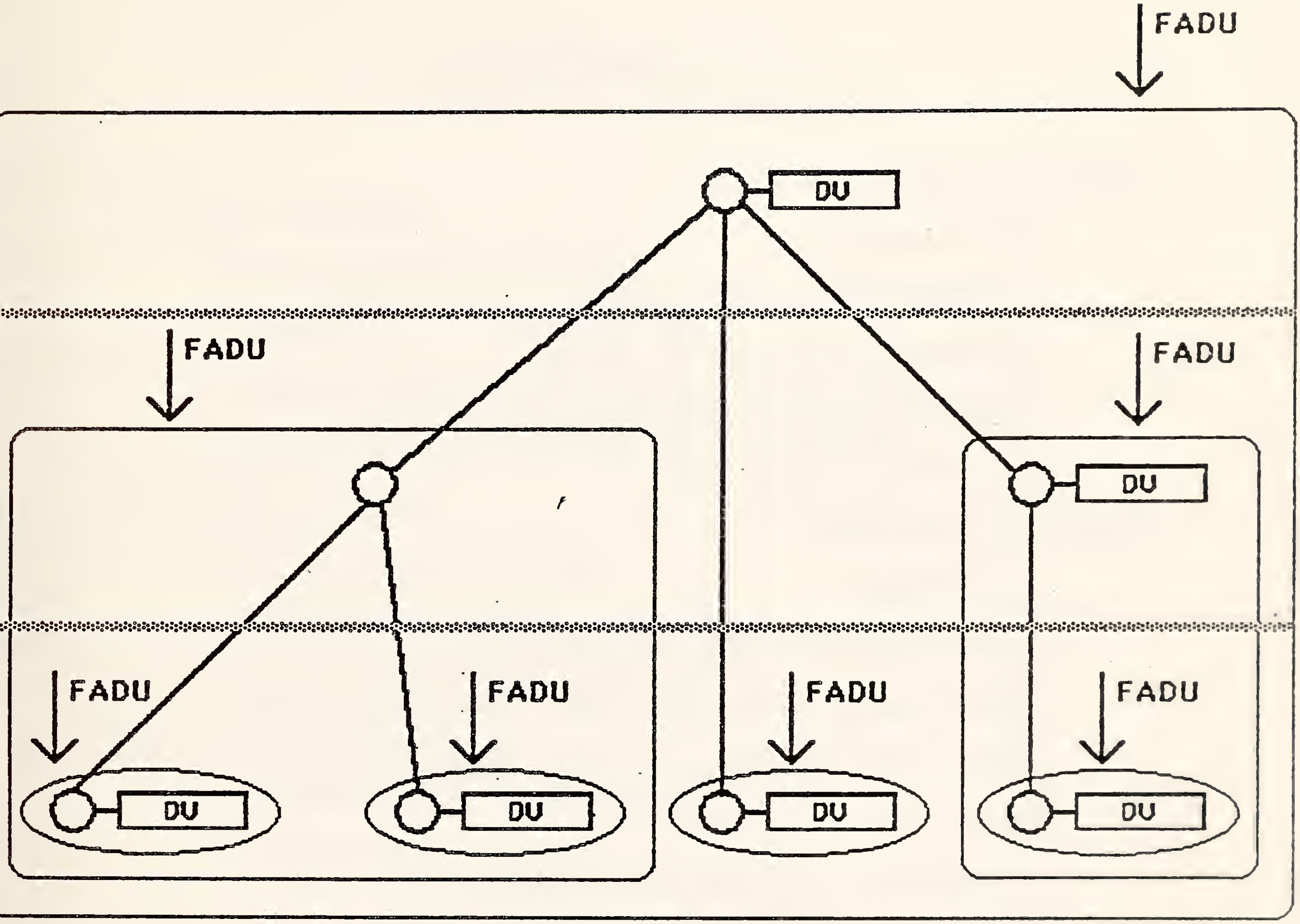
Files contain one or more *Data Units* possibly related in some fashion (e.g., sequential, network, relational, or hierarchical).

Virtual filestore provides a tree structure (called the *access structure* to represent the relation between Data Units.

Subtree of the access structure is known as a File Access Data Unit (FADU).

Essentially a hierarchical model.

Two special cases of access structure: unstructured files and flat files.



Access Structure Using Tree Notation

File Attributes

- Kernel subset
    - Filename
    - Presentation context
    - Access structure type
    - Presentation structure name
    - Current filesize
- Storage subset
    - Account
    - Date and time of creation
    - Date and time of last modification
    - Date and time of last read access
    - Identity of creator
    - Identity of last modifier
    - Identity of last reader
    - File availability
    - Possible access type
    - Future filesize
- Security subset
    - Access control
    - Encryption name
    - Legal qualifications

Activity Attributes

- Kernel subset
    - Requested access
    - Location of initiator
    - Current access structure type
    - Current presentation context
- Storage subset
    - Current account
    - Current access context
    - Concurrency control
- Security subset
    - Identity of initiator
    - Password

THE FTAM SERVICE

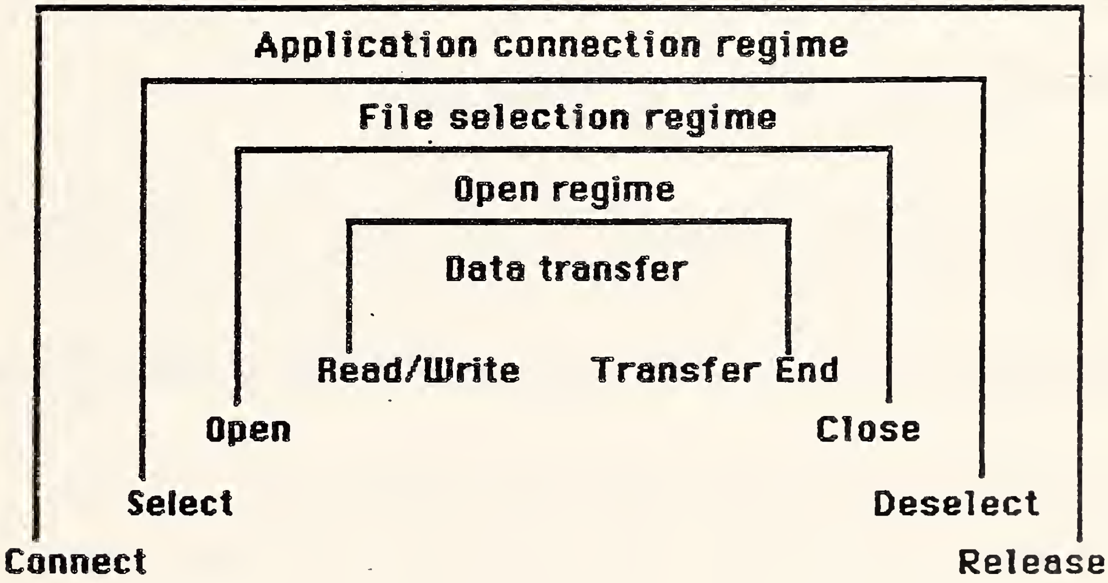
Based on establishing and disestablishing a series of *regimes*.

Entering regimes builds up the *operational context* of the entities step-by-step.

Asymmetric model

> Initiating and responding entities during application connection, file selection, and open regimes.
>
> Sending and receiving entities during data transfer regime.

Reliable and user-correctable services.



FTAM REGIME NESTING

Service Subsets

File Transfer Service Subset

File Access Service Subset

Limited File Management Service Subset

Enhanced File Management Service Subset

File Transfer Service Subset

CONNECT

Establish an application association with the specified FTAM entity.

SELECT

Select the file on which actions are to be performed.

OPEN

Open the selected file and negotiate the context in which its contents will be interpreted.

READ / WRITE

Establish the direction of the data transfer.

DATA

Transfer the data.

DATA_END

All data has been sent.

TRANSFER_END

Data transfer is complete.

CLOSE

Close the open file.

DESELECT

Release the selected file.

RELEASE

Release the application association.

CANCEL

Cancel the data transfer in progress.

ABORT

Release the application association unconditionally, abandoning any activity in progress.

BEGIN_GROUP

Indicate the start of a set of concatenated requests.

END_GROUP

Indicate the end of a set of concatenated requests.

File Access Service Subset

LOCATE

Locate the specified FADU.

ERASE

Erase the specified FADU.

Limited File Management Service Subset

CREATE

Create a new file with the specified attributes and select that file.

DELETE

Delete and deselect the selected file.

READ_ATTRIB

Obtain information about the selected file.

Enhanced File Management Service Subset

CHANGE_ATTRIB
    Modify the attributes of the selected file.

Error Recovery Service Subset

RECOVER

Recreate the open regime following a failure.

CHECK

Mark / acknowledge transferred data.

RESTART

Interrupt data transfer and negotiate restart point.

THE FTAM PROTOCOL

Relies on underlying services of OSI Presentation Layer.

Uses OSI Session Layer pass-through services to provide checkpointing and recovery.

Currently provided are *basic* and *error recovery* protocol.

Basic protocol provides for the establishment and disestablishment of regimes and the movement of data.

Error recovery protocol provides a standard set of error recovery procedures.

Non-standard error recovery procedures may be implemented by using the user-correctable service.

FTAM Session Layer Requirements (First DP)

Kernel functional unit

S_CONNECT (req, ind, resp, conf)

S_DATA (req, ind)

S_RELEASE (req, ind, resp, conf)

S_U_ABORT (req, ind)

S_P_ABORT (ind)

Duplex functional unit

Minor synchronize functional unit

S_SYNC_MINOR (req, ind, resp, conf)

S_TOKEN_GIVE (req, ind)

S_TOKEN_PLEASE (req, ind)

Resynchronize functional unit

S_RESYNCHRONIZE (req, ind, resp, conf)

FTAM Protocol Data Units

FTAM protocol data units (PDUs) specified in notation based on that used in CCITT X.409.

More complex structures defined in terms of a set of primitive and constructor types (e.g, BOOLEAN, INTEGER, OCTET STRING, SET).

Rules for encoding the specified *abstract syntax* are independent of the abstract syntax itself.

At least one set of encoding rules will be specified by ISO.

```
FABORTrequest ::= SET {
    originator [0] INTEGER {
        fileServiceUserInitiated (0),
        fileServiceProvidedInitiated (1)},
    diagnostic Diagnostic}

Diagnostic ::= [APPLICATION 2] IMPLICIT SET {
    errorTypeIdentifier [0] ErrorTypeIdentifier,
    errorIdentifier [1] ErrorIdentifier,
    suggestedDelay [2] INTEGER OPTIONAL,
    furtherDetails CHOICE {
        humanReadable [3] ACharString,
        machineReadable [4] OCTET STRING} OPTIONAL}

ErrorIdentifier ::= INTEGER {
    noReasonProvided (0),
    mandatoryParameter (2),
    illegalParameterValue (3),
    unsupportedParameterValue (4)}

ErrorTypeIdentifier ::= INTEGER {
    success (0),
    warning (1),
    recoverableError (2),
    unrecoverableError (3)}
```

AREAS FOR FUTURE EXPANSION

Filestore and file management.

File access.

Manipulation of groups of files simultaneously.

FURTHER TUTORIAL MATERIAL

D. Lewan, and H.G. Long, "The OSI File Service," Proceedings of the IEEE, Volume 71, Number 12, pp. 1414-1419, December 1983.

P.F. Linington, "The Virtual Filestore Concept," Computer Networks, Volume 8, Number 1, pp. 13-16, February 1984.

Proposal for a Useful Connectionless Internetwork Protocol

BASIC PROPOSAL

The conformance (full) protocol is proposed according to the ISO D.I.S. for the Data Communications Protocol for Providing the Connectionless-mode Network Service. (See the attachment on Conformance and the Provision of Functions for Conformance, extracted from the D.I.S.)

It is proposed that only type 1 functions be implemented for a timely and useful service. (See the attachment on function types, extracted from the D.I.S.)

SUBNETWORK USER DATA

If source and destination end systems are on the same 802.3 or 802.4 subnetwork, then the same size restrictions as applied to the 1984 NCC demo should prevail. (In this case, the Inactive Network Layer Protocol Subset is being employed.) If the full protocol is being used to concatenate subnetworks then the maximum user data size of 64+K should be permitted, corresponding to the IP specification. Practically, no statement need be made about minimum user data sizes, since it is expected that the transport class 4 header and transport user data will be of non-trivial length.

PDU LIFETIME

For purposes of concatenating LANs, typically an intermediate system might subtract one from the lifetime field. This represents 500 milliseconds for transit between intermediate systems and processing by one intermediate system. Thus, it is recommended that source end systems insert an initial value corresponding to the width of the catnet plus two. This should safely allow the destination end system to process the received PDU.

ROUTING

For purposes of a useful exercise it is recommended that fixed routing tables be used. It is suggested that implementations provide operator control to manipulate routing tables, since the exact topology for any demonstration may be subject to last minute modifications.

SEGMENTATION AND REASSEMBLY

Destination end systems must be able to reassemble. Source end systems must either fit the transport PDU plus IP header into a single subnetwork service data unit or be able to segment. Reassembly by intermediate systems is not recommended, however they must be able to segment. Setting of the segmentation permitted flag should be at the discretion of the source end system, however it is suggested that the flag be set to allow segmenting.

ERROR REPORTING

NOTE: THIS RECOMMENDATION AMENDS AND STRENGTHENS THE BASIC PROPOSAL SECTION REGARDING CONFORMANCE. If the error report flag is on and a PDU of type 3 is discarded because it is not supported, then an error report should be returned even though this is not strictly required for conformance. This is recommended for purposes of debugging. It is further suggested, for debugging, that implementations log all error reports. The error report data field should contain the entire errant PDU, truncated only if necessary.

IDENTIFICATION

The protocol id. of 1000 0001 is used in the catnet situation with a version number of 0000 0001. For the single subnetwork case, the protocol id. is 0000 0000.

CHECKSUM

Although checksum of the header is optional it is recommended that the checksum be used, since emphasis should be on a useful IP rather than simply on a demonstration. It may also be useful for debugging.

ADDRESSING

Binary representation should be used, since the IP header is binary based. (See attached table from the ISO d.p. on addressing.)

TOPOLOGIES

Various topologies are considered below. LAN refers either to 802.3 or 802.4, interchangeably. WAN refers to PDN X.25, 1984. Pri refers to any private (vendor propritary) subnetwork.

Cases considered:

a) LAN
b) LAN-LAN
c) LAN-WAN-LAN
d) WAN-LAN
e) Pri-LAN
f) LAN-Pri-LAN
g) Pri-WAN-LAN

LAN Addressing

The Inactive Network Layer Protocol subset is used with identifier of 0000 0000.

LAN-LAN Addressing

Source and destination addresses in the IP header are of identical construction. The first octet is local binary, hexidecimal 49. The second octet is the subnetwork identifier. Assign 802.3 LANs beginning with 0000 0001 and 802.4 LANs beginning with 1000 0001. Octets three through eight comprise the 48 bit station address. Octet nine is the 8 bit network service access point.

LAN-WAN-LAN Addressing

Source and destination addresses are of identical construction. The first octet is hexidecimal 49, signifying local binary format. The subnetwork identifer octet, interpreted by intermediate systems via a routing table, yields an X.121 DTE address. The subnetwork identifier octet is followed by a station address and NSAP as described in the LAN-LAN case.

WAN-LAN Addressing

Considering the source address on the LAN and the destination address on the WAN, the source address has the format described in the LAN-WAN-LAN case. The first octet of the destination address is hexidecimal 25 (X.121 DTE, binary). Octets two through eight encode the 14 decimal digit X.121 DTE address in BCD. Octet nine is the NSAP.

Where the LAN is the destination and the WAN the source, the above format is reversed.

Pri-LAN Addressing

If source end system is on the private subnetwork and destination end system on the LAN, then the source address is: hexidecimal 49, subnetwork identifier of the LAN, station address of the gateway, followed by the private address of the source end system on the private subnetwork. (The private address on the private subnetwork is interpretable only by the private subnetwork.) Note that the three items preceding the private address specify the intermediate system coupling the private network and LAN. This constitutes routing, not addressing. This may be useful for a demonstration but is not recommended as a general solution. The destination address is: hexidecimal 49, subnetwork identifier of the LAN, station address on the LAN, followed by the one octet NSAP. For PDUs traveling from LAN to private subnetwork the formats are interchanged.

LAN-Pri-LAN Addressing

This structure is the same as the LAN-WAN-LAN addressing.

Pri-WAN-LAN Addressing

Where the source end system is on the private subnetwork, the source address is hexidecimal 25, seven octets of X.121 address of the gateway between the private and PDN subnetworks, followed by the private subnetwork end system address. Here again, the X'25' and X.121 address specifies particular routing information. It may or may not be desirable to do this for a demonstration, but it is not advised as a general solution. The destination address is hexidecimal 49, the LAN identifier, the station address, and the NSAP. PDU flow in the reverse direction formats the source and destination addresses in the opposite format.

ROUTING TABLE LOGIC

If the format is hexidecimal 25 then the X.121 address is either this system's or some other system's. If it is this system's, then interpret the information after the X.121 address. If it is some other system's, then send to the PDN.

If the format is hexidecimal 49 then check the subnetwork address. If it is some other subnetwork's then look up in the routing table. If it is this subnetwork's then broadcast it on this LAN.

Attachment

CONFORMANCE

For conformance to this International Standard, the ability to originate, manipulate, and receive PDUs in accordance with the full protocol (as opposed to the "non-segmenting" or "Inactive Network Layer Protocol" subsets) is required.

Additionally, the provision of the optional functions described in Section 6.17 and enumerated in Table 9-1 must meet the requirements described therein.

Additionally, conformance to the Standard requires adherence to the formal description of Section 8 and to the structure and encoding of PDUs of Section 7.

If and only if the above requirements are met is there conformance to this International Standard.

PROVISION OF FUNCTIONS FOR CONFORMANCE

The following table categorizes the functions in Section 6 with respect to the type of system providing the function:

Function	Send	Forward	Receive
PDU Composition	M	-	-
PDU Decomposition	M	-	M
Header Format Analysis	-	M	M
PDU Lifetime Control	-	M	I
Route PDU	-	M	-
Forward PDU	M	M	-
Segment PDU	M	(note 1)	-
Reassemble PDU	-	I	M
Discard PDU	-	M	M
Error Reporting	-	M	M
PDU Header Error Detection	M	M	M
Padding	(note 2)	(note 2)	(note 2)
Security	-	(note 3)	(note 3)
Complete Source Routing	-	(note 3)	-
Partial Source Routing	-	(note 4)	-
Record Route	-	(note 4)	-
QoS Maintenance	-	(note 4)	-

Table 9-1. Categorization of Functions.

Table 6-1 shows how the functions are divided into these three categories:

Function	Type
PDU Composition	1
PDU Decomposition	1
Header Format Analysis	1
PDU Lifetime Control	1
Route PDU	1
Forward PDU	1
Segment PDU	1
Reassemble PDU	1
Discard PDU	1
Error Reporting	1 (note 1)
PDU Header Error Detection	1 (note 1)
Padding	1 (notes 1 & 2)
Security	2
Complete Source Routing	2
Partial Source Routing	3
Priority	3
Record Route	3
Quality of Service Maintenance	3

Table 6-1. Categorization of Protocol Functions

<u>Notes:</u>

1) While the Padding, Error Reporting, and Header Error Detection functions must be provided, they are provided only when selected by the sending Network Service user.

2) The correct treatment of the Padding function involves no processing. Therefore, this could equally be described as a Type 3 function.

3) The rationale for the inclusion of type 3 functions is that in the case of some functions it is more important to forward the PDUs between intermediate systems or deliver them to an end-system than it is to support the functions. Type 3 functions should be used in those cases where they are of an advisory nature and should not be the cause of the discarding of a PDU when not supported.

TABLE 8-1: AFI ALLOCATIONS

00-09	Reserved - will not be allocated
10-19	Reserved for future allocation by joint agreement of ISO and CCITT
20-51	Allocated and assigned to the IDI formats defined in clause 8.2.1.2
52-59	Reserved for future allocation by joint agreement of ISO and CCITT
60-69	Allocated for assignment to new IDI formats by ISO
70-79	Allocated for assignment to new IDI formats by CCITT
80-99	Reserved for future allocation by joint agreement of ISO and CCITT

8.2.1.2 FORMAT AND ALLOCATION OF THE IDI

A specific combination of IDI format and DSP syntax is associated with each allocated AFI value, as summarized in Table 8-2:

TABLE 8-2: AFI Values

DSP syntax IDI format	Decimal	Binary	Character (ISO 646)	National Character
X.121-DCC	20	21	22	23
X.121-DTE	24	25	26	27
F.69	28	29	30	31
E.163	32	33	34	35
E.164	36	37	38	39
ISO 6523	40	41	42	43
ISO 6523-ICD	44	45	46	47
Local	48	49	50	51

NOTE

The need to describe DSP syntaxes involving characters or national characters for these IDI formats has not been established and is for further study

IMPLEMENTORS OF OSI
WORKSHOP SCHEDULE

Nov. 7 - 9	(INTEL)
Jan. 22 - 24	(HIS)
Apr. 16 - 18	(NBS)
June 25 - 27	(NBS)
Sept. 17 - 19	(NBS)

CONNECTIONLESS IP
PROPOSAL

- ALL TYPE 1 FUNCTIONS
  - COMPOSITION
  - DECOMPOSITION
  - HEADER ANALYSIS
  - LIFE TIME BOUNDING
  - ROUTING
  - FORWARDING
  - SEGMENTING
  - REASSEMBLING
  - DISCARD
  - ERROR REPORTING
  - ERROR DETECTION
  - PADDING

- OPTIONAL TYPE 2, 3 FUNCTIONS

  - SECURITY
  - COMPLETE SOURCE ROUTING
  - PARTIAL SOURCE ROUTING
  - PRIORITY
  - ROUTE RECORDING
  - QOS

ADDITIONAL RECOMMENDATIONS

- SUBNETWORK USER DATA
- PDU LIFETIME
- ROUTING
- SEGMENTATION & REASSEMBLY
- ERROR REPORTING
- PROTOCOL AND VERSION ID
- CHECKSUM

ADDRESSING

1 - SINGLE LAN

- INACTIVE NETWORK LAYER PROTOCOL (NO IP HEADER ADDRESSES)
- 48 BIT STATION ADDRESS
- 8 BIT NSAP

2 - LAN-LAN

- X' 49' FORMAT TYPE
- 8 BIT SUBNETWORK IDENTIFIER
- 48 BIT STATION ADDRESS
- 8 BIT NSAP

3 - LAN-WAN-LAN

- X' 49' FORMAT TYPE
- 8 BIT SUBNET ID (TABLE POINTER TO X.121 DTE ADDRESS)
- 48 BIT STATION ADDRESS
- 8 BIT NSAP

ADDRESSING CONT.

4 - WAN-LAN

- LAN-TO-WAN

  SOURCE: - X' 49'
  - 8 BIT SUBNET ID
  - 48 BIT STATION ADDRESS
  - 8 BIT NSAP

  DESTINATION:
  - X' 25'
  - 7 OCTET X.121 DTE ADDRESS
  - 8 BIT NSAP

5 - PRIVATE-LAN

- PRI-TO-LAN

  SOURCE: - X' 49'
  - 8 BIT SUBNET ID
  - 48 BIT STATION ADDRESS (OF INTERMEDIATE SYSTEM)
  - PRIVATE ADDRESS

  NOTE: THE FIRST THREE ITEMS CONSTITUTE ROUTING AND YOU MAY NOT WANT TO DO THAT.

  DESTINATION:
  - X' 49'
  - 8 BIT SUBNET ID
  - 48 BIT STATION ADDRESS
  - 8 BIT NSAP

6 - LAN-PRI-LAN

- SAME AS LAN-WAN-LAN

7 - PRI-WAN-LAN

- PRI-TO-LAN

  SOURCE: - X' 25'

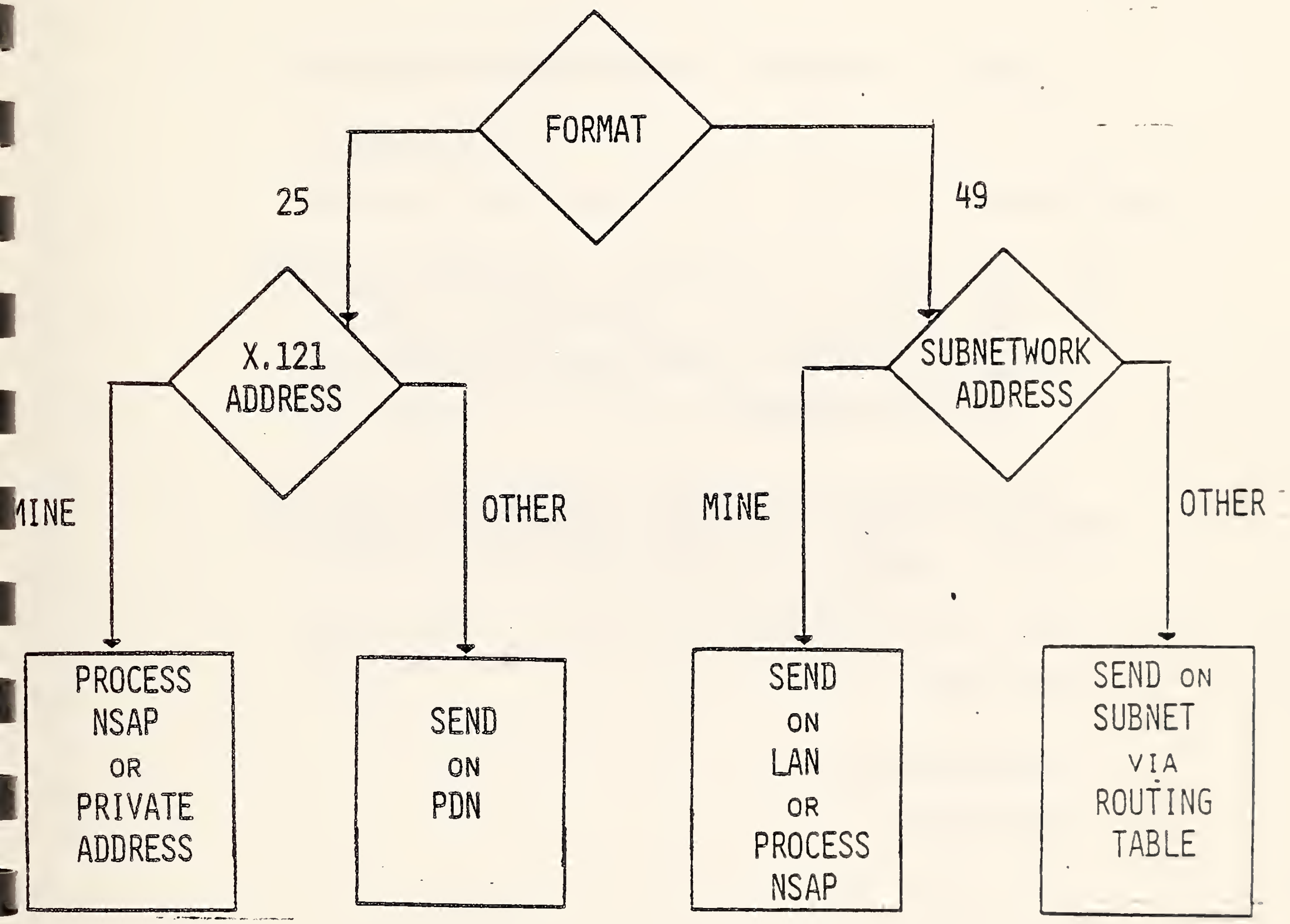
  - 7 OCTETS X.121 DTE ADDRESS
  - PRIVATE ADDRESS

NOTE: THE FIRST TWO ITEMS CONSTITUTES ROUTING AND YOU MAY NOT WANT TO DO THAT.

DESTINATION:

- X' 49'
- 48 BIT STATION ADDRESS
- 8 BIT NSAP

ROUTING


FORMAT
25
49
X.121 ADDRESS
SUBNETWORK ADDRESS
MINE
OTHER
MINE
OTHER
PROCESS NSAP OR PRIVATE ADDRESS
SEND ON PDN
SEND ON LAN OR PROCESS NSAP
SEND ON SUBNET VIA ROUTING TABLE

LAYER 3 - INTERNET: TIME, RESOURCES ESTIMATES

o WHAT

- IP: DEMO SUBSET OF CONFORMANCE SUBSET (NO TYPE 2 OR TYPE 3 FUNCTIONS I.E., NO SECURITY, SOURCE ROUTING, PRIORITY, ROUTE RECORDING, QOS, OR PRIORITY FUNCTIONS)

  COMPLETE: 2/1/85 (INSTALLED AND RUNNING, READY FOR PROTOCOL TESTING BY NBS)

- IP TEST BED

  ARCHITECTURES

  TEST CASES

- TEST BED ARCHITECTURES

  ENCODER/DECODER

  REFERENCE IMPLEMENTATION (RI) WITH TEST HARNESS

- ENCODER/DECODER

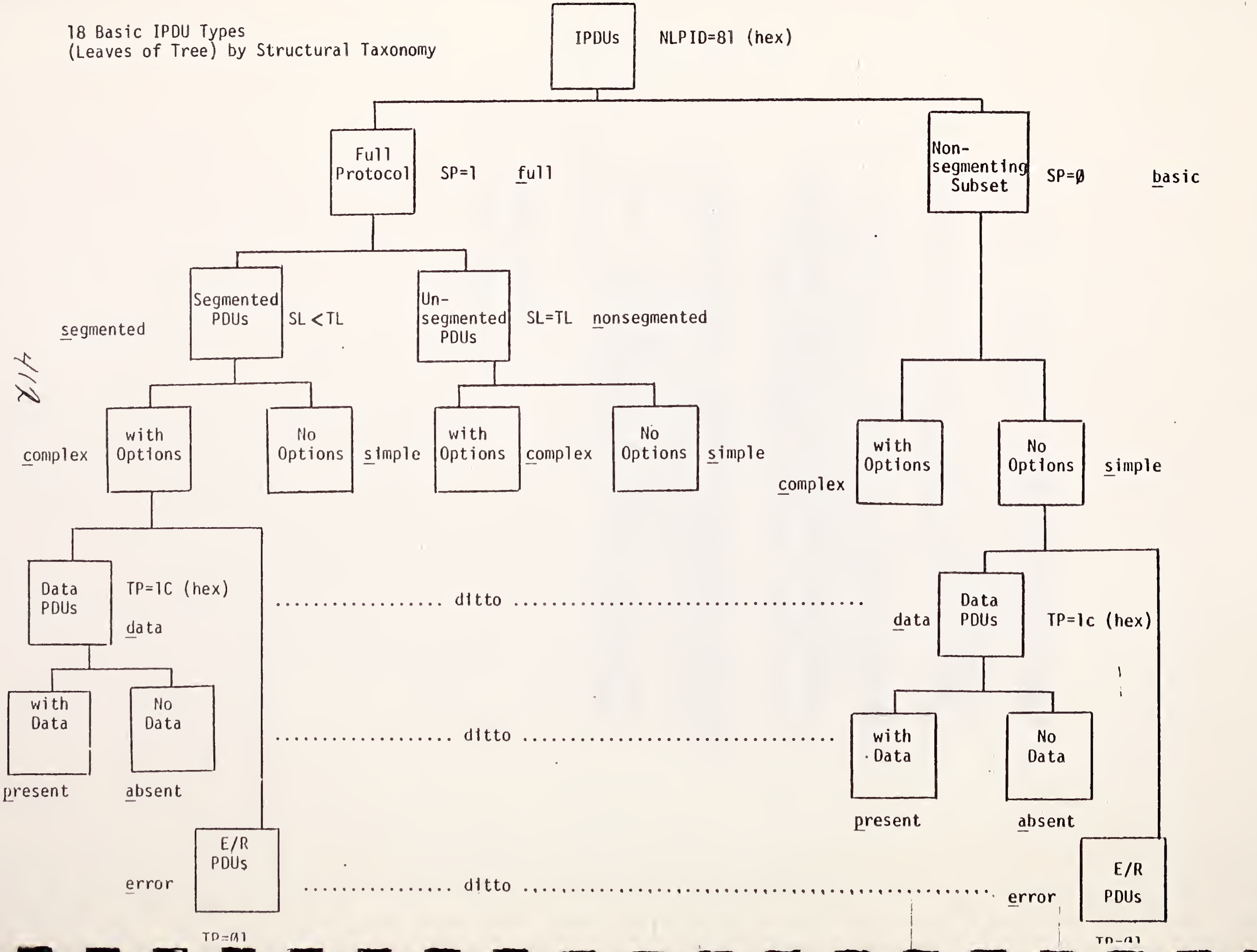
  CAN CONSTRUCT, RECOGNIZE ALL TYPE OF VALID IPDUs

  CAN CONSTRUCT, RECOGNIZE ERRONEOUS IPDUs

- RI WITH TEST HARNESS (SCENARIO INTERPRETER AND TRANSPORT OVER IP)

  CAN CONSTRUCT A SUBSET OF VALID PDUs (ONLY THOSE INDUCIBLE VIA THE IP SERVICE INTERFACE)

18 Basic IPDU Types
(Leaves of Tree) by Structural Taxonomy



RI WITH TEST HARNESS

o ASSUMPTIONS:

1) MAXIMUM LEVEL OF STAFFING (AT LEAST 1 STAFF/PERITEM)
2) MAXIMUM CONCURRENCY
3) START DATE 9/4/84
4) AVAILABILITY OF NBS VAX DEVELOPMENT SYSTEM BY START DATE (EARLIEST DELIVERY DATE: OCTOBER '84, EARLIEST AVAILABLE DATE: NOVEMBER '84)
5) SEPARATE DEVELOPMENT PATH FOR INTERNET WITH IP INSTALLED AND RUNNING (READY FOR PROTOCOL TESTING) BY 2/1/85
6) SEPARATE DEVELOPMENT PATHS FOR EXTENDING, UPDATING TRANSPORT AND FOR FTP; SEPARATE TESTING OF MODIFIED TRANSPORT AND OF FTP.

o LIKELY SLIPPAGE:

3 MONTHS (FOR ASSUMPTION #4) TO 7/31/85.

RI WITH TEST HARNESS

	TIME-TO-COMPLETE (PM)			
ITEM	DESIGN, SPECIFY	CODE, UNIT TEST	CRITICAL PATH TIME	CRITICAL PATH COMPLETE DATE
MODIFY INTERFACES FOR IP	1	1	2	10/31/84
MODIFY PDU LOG ANALYSIS TOOLS	1	1	2	
TEST CASES	1		1	10/31/84
PORT TEST HARNESS TO 32-BIT ENVIRONS			2	12/31/84
INTEGRATION TESTING			1	2/28/85
INTERNET PROTOCOL TESTING			1	3/31/85
INSTALLATION TESTING, DISTRIBUTION TO TEST CENTERS, FIELD TESTING			1	4/30/85

ENCODER/DECODER

- ASSUMPTIONS:

  1) MAXIMUM LEVEL OF STAFFING (AT LEAST 1 STAFF PER ITEM).
  2) MAXIMUM DEGREE OF CONCURRENCY
  3) START DATE: 9/4/84
  4) AVAILABILITY OF NBS VAX DEVELOPMENT SYSTEM BY START DATE (EARLIEST DELIVERY DATES OCTOBER '84, EARLIEST AVAILABLE DATE: NOVEMBER '84)
  5) SEPARATE DEVELOPMENT PATH FOR INTERNET WITH IP INSTALLED AND RUNNING (READY FOR PROTOCOL TESTING) BY 2/1/85.
  6) SEPARATE DEVELOPMENT PATHS FOR EXTENDING, UPDATING TRANSPORT AND FOR FTP: SEPARATE TESTING OF MODIFIED TRANSPORT AND OF FTP.

- LIKELY SLIPPAGE:

  3 MONTHS (FOR ASSUMPTION #4) TO 6/30/85

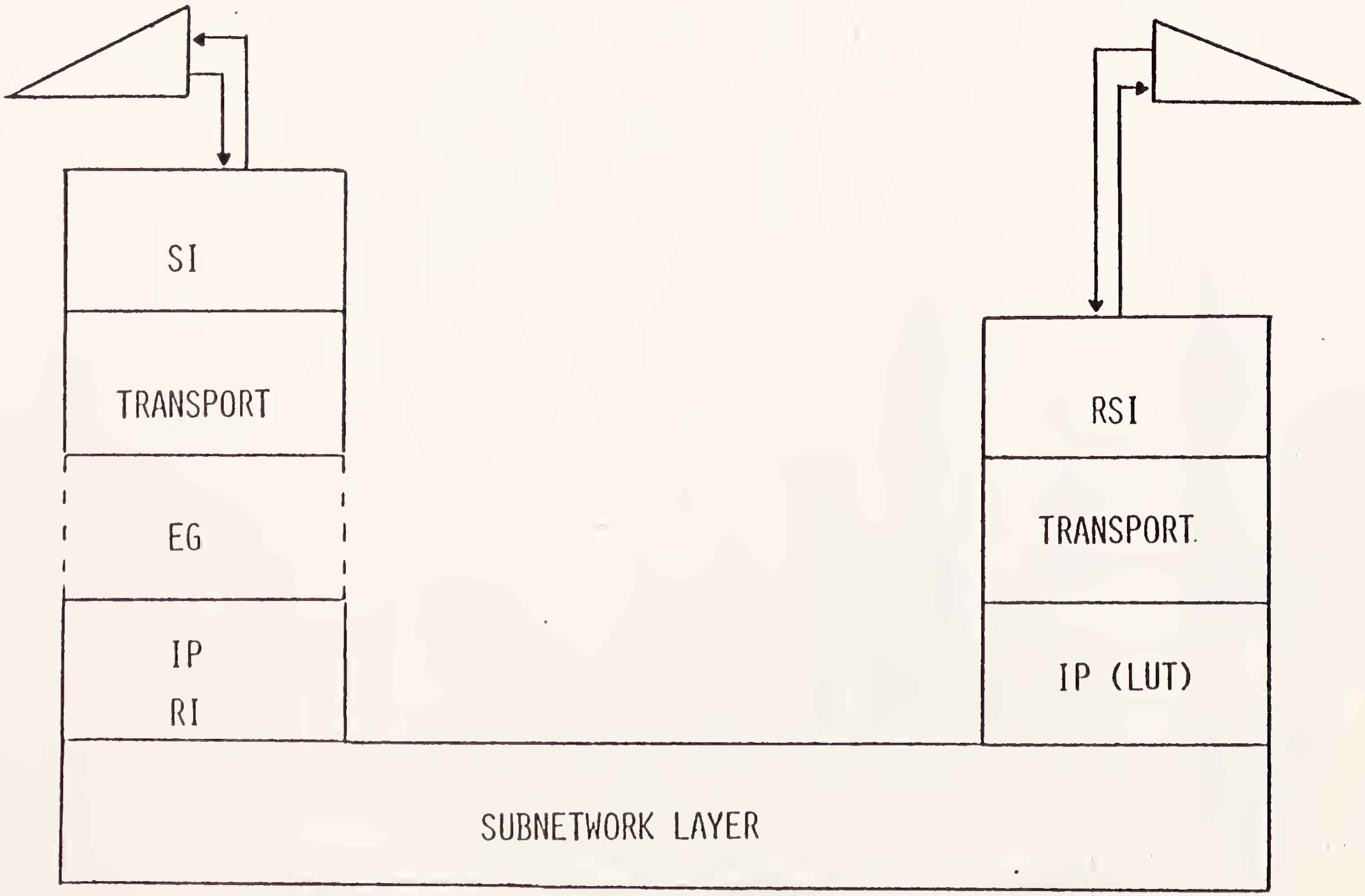
  +4 MONTHS (FOR ASSUMPTIONS #1, #2) TO 10/31/85

ENCODER/DECODER

	TIME-TO-COMPLETE (PM)			
	DESIGN, SPECIFY	CODE, UNIT TEST	CRITICAL PATH TIME	CRITICAL PATH COMPLETE DATE
TMM	2	2	4	12/31/84
E/D	2	2	4	
COMPARATOR	2	2	4	
XMIT/RCV	2	2	4	
LOG ANALYZERS	2	2	4	
RER	2	2	4	
USER COMMAND LANGUAGE	1			
TEST CASES	1			
INTEGRATION TESTING			1	1/31/85
IP TESTING			1	2/28/85
INSTALLATION TESTING, DISTRIBUTION TO TEST CENTERS, FIELD TESTING			1	3/31/85

LAYER 3 TIME, RESOURCE ESTIMATES

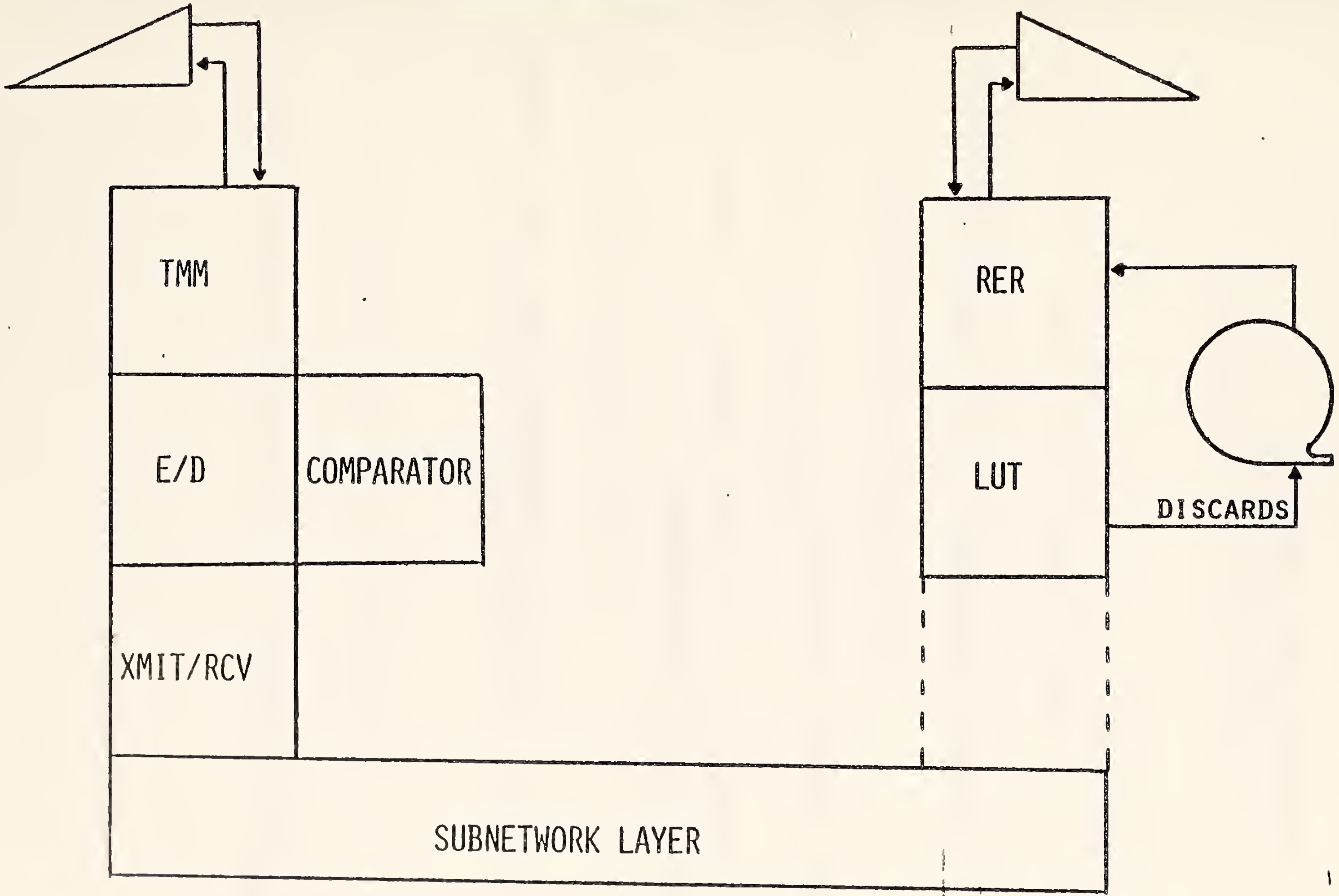
- ACCURACY: PROBLEMATIC
- 3 SECTIONS:
  - TASK AND CRITICAL PATH TIME
  - ASSUMPTIONS
  - LIKELY SLIPPAGE



SI - SCENARIO INTERPRETER
EG - EXCEPTION GENERATOR
IP RI - INTERNET PROTOCOL
REFERENCE IMPLEMENTATION

RSI - REMOTE SCENARIO INTERPRETER
LUT - LAYER UNDER TEST

ENCODER/DECODER



TMM - TEST MANAGEMENT MODULE
E/D - ENCODER/DECODER
XMIT/RCV - SENDER/RECEIVER

RER - REMOTE ECHOER/RESPONDER
LUT - LAYER UNDER TEST (IP)

NETWORK LAYER ADDRESSING

CONCEPTS AND TERMINOLOGY

PRINCIPLES FOR CREATING THE NETWORK LAYER ADDRESSING SCHEME

NETWORK ADDRESS SEMANTIC STRUCTURE

REPRESENTATION AS BINARY AND DECIMAL

RELATIONSHIP BETWEEN SEMANTICS. REPRESENTATION. AND ENCODING

BASIC PRINCIPLES OF THE NETWORK LAYER ADDRESSING SCHEME

HIERARCHICAL STRUCTURE OF NSAP ADDRESSES

- ROUTING
- ADMINISTRATION OF ADDRESS SPACE
- MULTI-LEVEL HIERARCHY
- CONCEPT OF ADDRESS "DOMAINS" AND "SUBDOMAINS"

GLOBAL IDENTIFICATION OF ANY NSAP

ROUTE AND SERVICE TYPE INDEPENDENCE

BINARY AND DECIMAL ADDRESSES ACCOMMODATED

VARIABLE LENGTH ADDRESSES UP TO A DEFINED MAXIMUM SIZE

NETWORK ADDRESS SEMANTIC STRUCTURE

INITIAL DOMAIN PART (IDP)

- AUTHORITY AND FORMAT IDENTIFIER (AFI)
    - CONVEYS FORMAT, LENGTH, AND "ABSTRACT SYNTAX" OF THE REST OF NSAP ADDRESS
    - SPECIFIES AUTHORITY RESPONSIBLE FOR ALLOCATING THE INITIAL DOMAIN IDENTIFIER
- INITIAL DOMAIN IDENTIFIER (IDI)
    - FOLLOWS ONE OF EIGHT FORMATS (SEE NEXT VIEWGRAPH)
    - SPECIFIES THE NETWORK ADDRESSING SUBDOMAIN FROM WHICH VALUES OF THE DSP ARE ALLOCATED
    - SPECIFIES THE AUTHORITY RESPONSIBLE FOR ALLOCATING VALUES OF THE DSP

DOMAIN SPECIFIC PART (DSP)

- SEMANTICS IS (LOCALLY) SIGNIFICANT IN THE CONTEXT SPECIFIED BY THE IDP
- MAY BE BASED ON DECIMAL, BINARY, CHARACTER, OR "NATIONAL CHARACTER"

INITIAL DOMAIN IDENTIFIER FORMATS

X.121-DTE

- IDI IS AN X.121 ADDRESS (UP TO 14 DIGITS)

X.121-DCC

- IDI IS AN X.121 DATA COUNTRY CODE (3 DIGITS)

F.69

- IDI IS A TELEX NUMBER (UP TO 8 DIGITS)

E.163

- IDI IS A TELEPHONE NETWORK (PSTN) NUMBER (UP TO 12 DIGITS)

E.164

- IDI IS AN ISDN NUMBER (UP TO 15 DECIMAL DIGITS)

ISO-6523

- IDI IS ALLOCATED ACCORDING TO ISO 6523. CONSISTING OF A 4 DIGIT INTERNATIONAL CODE DESIGNATOR (ICD). FOLLOWED BY UP TO 28 DIGITS DERIVED FROM AN ORGANIZATION CODE

ISO-6523-ICD

- IDI IS ALLOCATED ACCORDING TO THE ICD FROM ISO 6523

LOCAL

- IDI IS NULL (FOR USE IN A CLOSED COMMUNITY)

ACCOMMODATION OF BINARY AND DECIMAL

BINARY OR DECIMAL ADDRESS ISSUE HAS BEEN CONTROVERSIAL

- IEEE 802 AND MANY PRIVATE NETWORK ADDRESSES BASED ON BINARY

- X.121, PSTN, AND TELEX ADDRESSES BASED ON DECIMAL

DECISION TO ACCOMMODATE BOTH

- ADDRESS IDP (AFI AND IDI) BASED ON DECIMAL

- DSP BASED ON DECIMAL, BINARY, CHARACTER, OR NATIONAL CHARACTER

- EVERY ADDRESS CAN BE FULLY REPRESENTED IN BOTH PURE BINARY AND PURE DECIMAL

- ALGORITHMIC CONVERSION BETWEEN PURE BINARY AND PURE DECIMAL REPRESENTATIONS DEFINED

INTERNETWORK PROTOCOL USES BINARY REPRESENTATION, CARRIES DECIMAL BASED FIELDS AS BCD

- TRANSFORMATIONS NOT REQUIRED IN THIS CASE

RELATIONSHIP BETWEEN SEMANTICS, REPRESENTATION, AND ENCODING

WHAT HAVE WE STANDARDIZED?

- SEMANTICS:
    - AFI (TWO DECIMAL DIGITS)
    - IDI (VARIABLE DEPENDING ON AFI, DECIMAL DIGITS)
    - DSP (VARIABLE, BASED ON DECIMAL, BINARY, CHARACTER, OR NATIONAL CHARACTER)
- PURE DECIMAL REPRESENTATION
- PURE BINARY REPRESENTATION
- ALGORITHMIC TRANSFORMATIONS.

WHAT IS ENCODED IN PROTOCOL HEADERS?

- UP TO PROTOCOL DEFINITION (NOT ADDRESS STANDARD)
- MUST CONVEY SEMANTICS OF ADDRESS STRUCTURE
- MAY USE PURE DECIMAL OR BINARY REPRESENTATION
- MAY DEFINE OTHER WAY TO CONVEY SEMANTICS (E.G., SHORTHAND FOR LOCAL ADDRESSES)

NBS-114A (REV. 2-80)

U.S. DEPT. OF COMM. **BIBLIOGRAPHIC DATA SHEET** *(See instructions)*	1. PUBLICATION OR REPORT NO. NBSIR 84-2984	2. Performing Organ. Report No.	3. Publication Date September 1984

4. TITLE AND SUBTITLE

Minutes of the Seventh NBS Workshop for Implenentors of ISO Open Systems Interconnection September 5-7, 1984

5. AUTHOR(S)

Kenneth Dymond

6. PERFORMING ORGANIZATION *(If joint or other than NBS, see instructions)*

NATIONAL BUREAU OF STANDARDS
DEPARTMENT OF COMMERCE
WASHINGTON, D.C. 20234

7. Contract/Grant No.

8. Type of Report & Period Covered

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS *(Street, City, State, ZIP)*

10. SUPPLEMENTARY NOTES

□ Document describes a computer program; SF-185, FIPS Software Summary, is attached.

11. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here)*

The National Bureau of Standards has so far (September 1984) hosted seven in a series of workshops for computer manufacturers and other organizations implementing ISO protocols for computer communications. The minutes of the seventh workshop--held September 5-7, 1984, in Gaithersburg, MD--are presented. The minutes include a brief summary of the three days' discussion, but the great bulk comprises attachments prepared by participants detailing proposals for implementing various protocols and for demonstrating their interworking. Agreements reached by participants are recorded.

12. KEY WORDS *(Six to twelve entries; alphabetical order; capitalize only proper names; and separate key words by semicolons)*

Computer communications; ISO (International Organization for Standardization) protocols; OSI (Open Systems Interconnection); NBS-OSI Workshops.

13. AVAILABILITY

□ Unlimited

☒ For Official Distribution. Do Not Release to NTIS

□ Order From Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402.

□ Order From National Technical Information Service (NTIS), Springfield, VA. 22161

14. NO. OF PRINTED PAGES

15. Price

$32.50