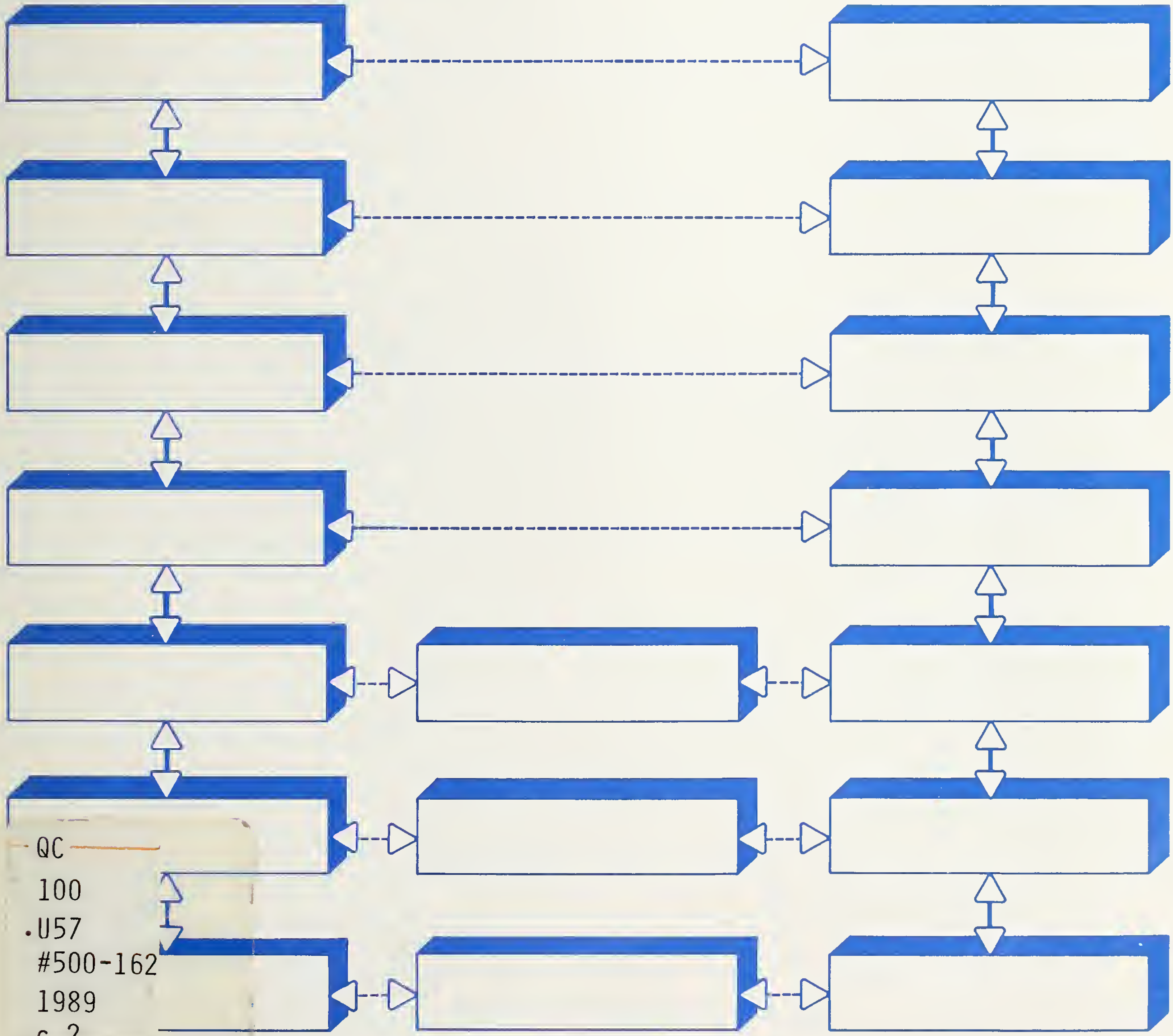
U.S. Department of Commerce
National Institute of Standards and Techn
Computer Systems Techn

ONS

NIST Special Publication 500-16

Stable Implementation Agreements for Open Systems Interconnection Protocols Version 2 Edition 1 December 1988

Based on the Proceedings of the NIST Workshop for Implementors of OSI



The National Institute of Standards and Technology[1] was established by an act of Congress on March 3, 1901. The Institute's overall goal is to strengthen and advance the Nation's science and technology and facilitate their effective application for public benefit. To this end, the Institute conducts research to assure international competitiveness and leadership of U.S. industry, science and technology. NIST work involves development and transfer of measurements, standards and related science and technology, in support of continually improving U.S. productivity, product quality and reliability, innovation and underlying science and engineering. The Institute's technical work is performed by the National Measurement Laboratory, the National Engineering Laboratory, the National Computer Systems Laboratory, and the Institute for Materials Science and Engineering.

The National Measurement Laboratory

Provides the national system of physical and chemical measurement; coordinates the system with measurement systems of other nations and furnishes essential services leading to accurate and uniform physical and chemical measurement throughout the Nation's scientific community, industry, and commerce; provides advisory and research services to other Government agencies; conducts physical and chemical research; develops, produces, and distributes Standard Reference Materials; provides calibration services; and manages the National Standard Reference Data System. The Laboratory consists of the following centers:

- Basic Standards[2]
- Radiation Research
- Chemical Physics
- Analytical Chemistry

The National Engineering Laboratory

Provides technology and technical services to the public and private sectors to address national needs and to solve national problems; conducts research in engineering and applied science in support of these efforts; builds and maintains competence in the necessary disciplines required to carry out this research and technical service; develops engineering data and measurement capabilities; provides engineering measurement traceability services; develops test methods and proposes engineering standards and code changes; develops and proposes new engineering practices; and develops and improves mechanisms to transfer results of its research to the ultimate user. The Laboratory consists of the following centers:

- Computing and Applied Mathematics
- Electronics and Electrical Engineering[2]
- Manufacturing Engineering
- Building Technology
- Fire Research
- Chemical Engineering[3]

The National Computer Systems Laboratory

Conducts research and provides scientific and technical services to aid Federal agencies in the selection, acquisition, application, and use of computer technology to improve effectiveness and economy in Government operations in accordance with Public Law 89-306 (40 U.S.C. 759), relevant Executive Orders, and other directives; carries out this mission by managing the Federal Information Processing Standards Program, developing Federal ADP standards guidelines, and managing Federal participation in ADP voluntary standardization activities; provides scientific and technological advisory services and assistance to Federal agencies; and provides the technical foundation for computer-related policies of the Federal Government. The Laboratory consists of the following divisions:

- Information Systems Engineering
- Systems and Software Technology
- Computer Security
- Systems and Network Architecture
- Advanced Systems

The Institute for Materials Science and Engineering

Conducts research and provides measurements, data, standards, reference materials, quantitative understanding and other technical information fundamental to the processing, structure, properties and performance of materials; addresses the scientific basis for new advanced materials technologies; plans research around cross-cutting scientific themes such as nondestructive evaluation and phase diagram development; oversees Institute-wide technical programs in nuclear reactor radiation research and nondestructive evaluation; and broadly disseminates generic technical information resulting from its programs. The Institute consists of the following divisions:

- Ceramics
- Fracture and Deformation[3]
- Polymers
- Metallurgy
- Reactor Radiation

[1] Headquarters and Laboratories at Gaithersburg, MD, unless otherwise noted; mailing address Gaithersburg, MD 20899.
[2] Some divisions within the center are located at Boulder, CO 80303.
[3] Located at Boulder, CO, with some elements at Gaithersburg, MD.

Computer Systems Technology

NIST Special Publication 500-162

Stable Implementation Agreements for Open Systems Interconnection Protocols Version 2 Edition 1 December 1988

Based on the Proceedings of the NIST Workshop for Implementors of OSI

Workshop Chairman
Tim Boland, NIST

Special Interest Group (SIG) Chairmen

Directory Services	Chris Moore	The Wollongong Group
FTAM	Klaus Truoel	DFN
Lower Layers	Fred Burg	AT&T
Network Management	Paul Brusil	MITRE
ODA	Frank Dawson	IBM
OSINET	Carol Edgar	NIST
Registration	Einar Stefferud	NMA-Northrup
Security	Dennis Branstad	NIST
TLC	J.J. Cinecoe	Wang
Transaction Processing	John Neumann	Consultant
Upper Layers	David Chapell	Cray Research
Virtual Terminal	Cyndi Jung	3COM
X400	Charles Fox	DEC

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary

National Institute of Standards and Technology
Raymond G. Kammer, Acting Director

Issued February 1989

Reports on Computer Systems Technology

The National Institute of Standards and Technology (NIST) (formerly the National Bureau of Standards) has a unique responsibility for computer systems technology within the Federal government. NIST's National Computer Systems Laboratory (NCSL) develops standards and guidelines, provides technical assistance, and conducts research for computers and related telecommunications systems to achieve more effective utilization of Federal information technology resources. NCSL's responsibilities include development of technical, management, physical, and administrative standards and guidelines for the cost-effective security and privacy of sensitive unclassified information processed in Federal computers. NCSL assists agencies in developing security plans and in improving computer security awareness training. This Special Publication 500 series reports NCSL research and guidelines to Federal agencies as well as to organizations in industry, government, and academia.

Library of Congress Catalog Card Number: 89-600726
National Institute of Standards and Technology Special Publication 500-162
Natl. Inst. Stand. Technol. Spec. Publ. 500-162, 508 pages (Feb. 1989)
CODEN: NSPUE2

U.S. GOVERNMENT PRINTING OFFICE
WASHINGTON: 1989

For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington, DC 20402

Table of Contents

List of Figures

List of Tables

1. GENERAL INFORMATION

1.1 PURPOSE OF THIS DOCUMENT

This document records stable implementation agreements of OSI protocols among the organizations participating in the NIST Workshop for Implementors of OSI. Stable in the context of this document means that:

1) The agreements are based on final standards (e.g., ISO-IS or CCITT Recommendations) or nearly final (e.g., ISO-DIS) with no significant changes expected, and,

2) The agreements have been approved by the NIST Workshop Plenary for progression from the Ongoing Agreements document to this document after a period of review. These agreements are considered final; the only changes allowed will be clarifications, errata and the correction of omissions discovered in their implementation.

For these reasons, the agreements are considered advanced enough for use in product and test suite development. This means that readers can use this text as a basis for procurement references for OSI products. All of the text in this document is considered stable as defined above.

Future releases of these Stable Agreements will add and/or extend functionality offered by this version. When required, new versions will be introduced on a yearly basis. It is the NIST Workshop intent that new versions of this Stable Agreements document will be compatible with the present version. If this proves impractical, the agreements will attempt to provide mechanisms and guidelines which maximize interoperability.

Agreements text is either in this Stable Document (Stable) or in the aligned Ongoing Document (not yet stable). It is a goal that the same text not appear in the same position in both documents at once (except for Section 1).

The intended audience for this document is composed of those individuals who are interested in Stable Implementation Agreements for OSI protocols. Each section of the document covers a different subject area, and the sections are presented so as to present a consistent and unified approach. The structure of each section, whenever possible, is divided into the following subsections:

- Introduction,
- Scope and Field of Application,
- Status,

- o Eratta,
- o Protocal and Service Agreements,
- o Conformance, and
- o Appendicies.

The corresponding and aligned document, "Ongoing Implementation Agreements for OSI Protocols: Continuing Agreements", records agreements which are not yet considered stable, in the sense described above. This document will be referenced as the "Ongoing Agreements Document". This Stable document is aligned with the Ongoing Agreements Document in the sense that the structures are identical, and pointers are given in this Stable Document to work in the Ongoing Agreements Document which could become stable in the future. Chapter numbers have been changed from previous Workshop documents to reflect this alignment.

The benefit of this document to the reader is that it gives a complete accounting of current stable agreements. Minor changes (Eratta) to these agreements will be issued as supplements.

Version 2 (this version) is backwards compatible with Version 1 to the maximum extent possible. Version 2 includes all of the material from Version 1, (modified by eratta) as well as now stable material from the previous year; important new functional additions are in the areas of lower layers network technology, virtual terminal capability, office document architecture, and directory services.

1.2 PURPOSE OF THE WORKSHOP

In February, 1983, at the request of industry, NIST organized the NIST Workshop for Implementors of OSI to bring together future users and potential suppliers of OSI protocols. The Workshop accepts as input the specifications of emerging standards for protocols and produces as output agreements on the implementation and testing particulars of these protocols. This process is expected to expedite the development of OSI protocols and promote interoperability of independently manufactured data communications equipment.

1.3 WORKSHOP ORGANIZATION

The Workshop organizes its work through Special Interest Groups (SIGs) that prepare technical documentation. An executive committee of SIG chairpersons led by the overall Workshop chairperson administers the Workshop. NIST invites highly qualified technical leaders from participating organizations to assume leadership roles in the SIGs. The SIGs are encouraged to coordinate with standards organizations and user groups, and to seek widespread technical consensus on implementation agreements through international discussions and liaison activities.

The Workshop meets four times a year at the National Institute of

Standards and Technology in Gaithersburg, Maryland where each SIG is required to convene its meeting. In addition, a plenary assembly of all Workshop delegates is convened for consideration of SIG motions and other Workshop business. SIGs are also encouraged to hold interim meetings at varied locations around the world.

The Workshop is an open public forum. Registration materials, documents, and Workshop schedules are available from:

National Institute of Standards and Technology
NIST Workshop for Implementors of OSI
Building 225, Room B-217
Gaithersburg, Maryland 20899

1.4 USE AND ENDORSEMENT BY OTHER ENTERPRISES

The Workshops are held for those organizations expressing an interst in implementing or procuring OSI Protocols and Open Systems. However, there is no corporate commitment to implementations associated with Workshop participation.

The Workshop and associated agreements have been endorsed by various activities and groups. See the aligned section of the Ongoing Agreements Document for more on this subject.

1.5 RELATIONSHIP OF THE WORKSHOP TO THE NIST

As resources permit, NIST, with voluntary assistance from industry, develops formal protocol specifications, reference implementations, tests, and test systems for the protocols agreed to in the Workshops. The NIST organizes, administers, and makes technical contributions to the Workshop. The NIST bears no other relation to the workshop.

1.6 STRUCTURE AND OPERATION OF WORKSHOP

1.6.1 Plenary

The main body of the workshop is a Plenary Assembly. Any organization may participate. Representation is international. The NIST prefers for the business of Workshops to be conducted informally since there are no corresponding formal commitments within the Workshop to implement the decisions reached. For more information, consult the aligned Section of the Ongoing Agreements Document.

1.6.2 Special Interest Groups

Within the Workshop there are Special Interest Groups (SIGs). The SIGs receive their instructions for their technical program of work from the Plenary. The SIGs meet independently during the Workshop week. As technical work is completed by a SIG, it is presented to the Plenary for disposition. For more information on SIGs (including SIG charters), consult the aligned section of the Ongoing Agreements Document.

1.7 POINTS OF CONTACT

For information concerning the workshop, write to:

Chair, NIST Workshop for Implementors of OSI
at the address given in Section 1.3.

Individual points of contact are given in the aligned section of the Ongoing Agreements Document.

2. SUB NETWORKS

2.1 INTRODUCTION

This chapter provides agreements about subnetwork services used in providing the OSI Network Layer.

2.2 SCOPE AND FIELD OF APPLICATION

These agreements cover subnetwork types including local area networks, packet switched networks, circuit switched networks, ISDN, and others.

2.3 STATUS

This version was completed in December 1988.

2.4 ERRATA

2.5 LOCAL AREA NETWORKS

2.5.1 IEEE 802.2 LOGICAL LINK CONTROL

The following decisions have been reached with respect to this protocol.

1. Link Service Access Point (LSAP)

   The IEEE 802 committee has assigned the code below to address systems using any ISO network layer protocol. Note that bit zero is transmitted first.

   The most significant bit is bit 7, thus this bit pattern represents hexadecimal FE.

0	1	2	3	4	5	6	7
0	1	1	1	1	1	1	1

Figure 2.1 LSAP bit pattern

2. Type and Class

   Only the connectionless type 1, class 1 IEEE 802 link service will be used.

2.5.2 IEEE 802.3 CSMA/CD ACCESS METHOD

The following implementation agreements have been reached with respect to the IEEE 802.3 CSMA/CD Access Method and Physical Layer Specifications:

- o The address length shall be 48 bits

The following implementation agreements have been reached with respect to 10 BROAD 36 Networks:

1. Single Cable Networks

   - The translator frequency shall be 192.25 Mhz
   - The channel allocations are

Reverse Channels	Forward Channels
T12, T13, T14	L, M, N
T13, T14, 2′	M, N, O
T14, 2′, 3′	N, O, P
2′, 3′, 4′	O, P, Q
3′, 4′, 4A′	P, Q, R
4′, 4A′, 5′	Q, R, S

2. Dual Cable Networks

   For nontranslated dual cable networks forward and reverse frequencies are the same. Permissible channel allocations are:

T12, T13, T14
T13, T14, 2′
T14, 2′, 3′
2′, 3′, 4′
3′, 4′, 4A′
4′, 4A′, 5′
L, M, N
M, N, O
N, O, P
O, P, Q
Q, R, S

3. When both IEEE 802.4 and IEEE 802.3 10 BROAD 36 networks coexist on the broadband cable system the preferred channel allocations are:

	Reverse	Forward
IEEE 802.3	T12, T13, T14	L, M, N
IEEE 802.4	6', FM1'	T, U
channels reserved for future use	3', 4' 4A', 5'	P, Q R, S

2.5.3 IEEE 802.4 TOKEN BUS ACCESS METHOD

The following options are agreed to with respect to Draft J of token bus:

- Data Rate:
  - 10 Mb (Broadband)
  - 5 Mb (Carrierband)
- Addressing: 48 bit
- The lmeOption, Priority Mechanism, shall be implemented
- Broadband Channel Assignments

Forward	Reverse
P	3'
Q	4'
R	4A'
S	5'
T	6'
U	FM1'

2.5.4 IEEE 802.5 TOKEN RING ACCESS METHOD

The following implementation agreements have been reached with respect to the IEEE Standard 802.5, Token Passing Ring Access Method and Physical Layer specification.

- The data signalling rate shall be 4 Mbit/s
- The address length shall be 48 bits

- The message priority (PM) of the AMP data unit shall be 7

- The ALL_STATIONS_THIS_RING_ADDRESS shall be X'COOOFFFFFFFF'

- The TRR value shall be 4 milliseconds

- The THT value shall be 8.9 milliseconds

- The TQP value shall be 20 milliseconds

- The TVX value shall be 10 milliseconds

- The TNT value shall be 2.6 milliseconds

- The TAM value shall be 7 seconds

- The TSM value shall be 15 seconds

- The MAC Information field (I-field) shall be defined as follows:

Starting Sequence	I-Field	End Sequence

and the:

1) Starting Sequence includes: SD, AC, FC, DA, SA
2) Ending Sequence includes: FCS, ED, FS

Figure 2.2 I-Field Format

- With the above timer and MAC I-field definitions, the following limits are defined:

  - Protocol limits the I-field to a maximum of 4425 bytes, and

  - All stations shall support I-fields that have a minimum of one byte and a maximum of at least 2000 bytes.

2.5.5 FIBER DISTRIBUTED DATA INTERFACE (FDDI)

2.5.5.1 Token Ring Media Access Control (MAC, X3.139-1987)

(Refer to the Ongoing Implementation Agreements Document)

2.5.5.2 TOKEN RING PHYSICAL LEVEL (PHY,X3.148-1988)

(Refer to the Ongoing Implementation Agreements Document)

2.5.5.3 PHYSICAL LAYER MEDIA DEPENDENT (PMD, X3.166-198X)

(Refer to the Ongoing Implementation Agreements Document)

2.6 WIDE AREA NETWORKS

2.6.1 CCITT RECOMMENDATION X.25

The procedures required to describe the DTE side of a DTE/DCE interface for systems attached to sub-networks providing an X.25 interface shall be as defined in ISO 7776 and ISO 8208 and as supplemented below. (These procedures shall also apply to a DTE operating on a DTE/DTE interface).

2.6.2 ISO 7776

ISO 7776 is used as the Layer 2 Protocol with the agreements defined below.

1 The address assignments are:

DTE = A (=11000000 binary)
DCE = B (=10000000 binary)

On a DTE/DTE interface, one of the DTEs, by a prior agreement, shall use the DCE address.

2 The modulus shall be 8.

3 A window size (k) of 7 shall be supported. In addition, other window sizes may also be supported.

4 The Multilink Procedures are excluded.

2.6.3 ISO 8208

The elements of ISO 8208 applicable for use depend on the OSI role of ISO 8208 (ie., provision of CONS, support of CLNP). Independent of the role, ISO 8208 is used as the Layer 3 protocol, with the following agreements:

1 Virtual Call Service,

2 any mutually agreed window and packet size, however, all DTEs must be capable of supporting a window size of 2, a packet size of 128 octets, and a sequence number modulus of 8,

3 a DTE must be capable of receiving the Flow Control Parameter Negotiation Facility and responding appropriately (per ISO 8208), and

4 (Refer to the Ongoing Implementation Agreements Document.)

When ISO 8208 is used to support CONS, the optional user facilities in Section 5.1 of ISO 8878 shall also be supported.

When ISO 8208 is used to support CLNP (when providing the CLNS), Permanent Virtual Circuit Service may also be used.

2.7 INTEGRATED SERVICES DIGITAL NETWORKS (ISDN)

2.7.1 Introduction

This section defines Implementation Agreements for packet-data transfer in an ISDN context. The agreements provide a set of procedures for accessing an ISDN so that end systems implemented according to these agreements can obtain ISDN services and can successfully interoperate.

The agreements are not meant to preclude vendors from implementing additional procedures as long as they do not create system interoperability problems. Capabilities will vary from ISDN to ISDN and procedures beyond those included here may be necessary to request and utilize network services more effectively and fully.

The agreements cover two fundamental ISDN services for X.25 packet mode ISDN terminals, namely,

CASE I: The ISDN provides a circuit-mode (Layer 1) connection either on demand ("switched") or permanently ("dedicated circuit"). A general description of the corresponding ISDN 64 Kbps circuit-mode bearer service is described in CCITT Recommendation I.231. The circuit-mode connection is between an X.25 ISDN terminal and (i) a PSPDN, or (ii) another X.25 ISDN terminal. The circuit-mode connection to a PSPDN corresponds to CASE A: of CCITT Recommendation X.31.

CASE II: The ISDN provides the X.25 virtual circuit service. A general description of this service is given in CCITT Recommendation I.232. This case corresponds to CASE B: of CCITT Recommendation X.31.

Figures 2.3 and 2.4 give the agreed stacks for X.25 packet transfer over D and B channels, respectively. Some particular aspects are given below.

1. The packet data transfer is on a B channel of a Basic Access or a Primary Rate Interface. In CASE II, it can be on a D channel of a Basic Access Interface.

2. The layer 2 procedures are LAPB (ISO 7776) on a B channel and LAPD (CCITT Recommendation Q.921) on a D channel.

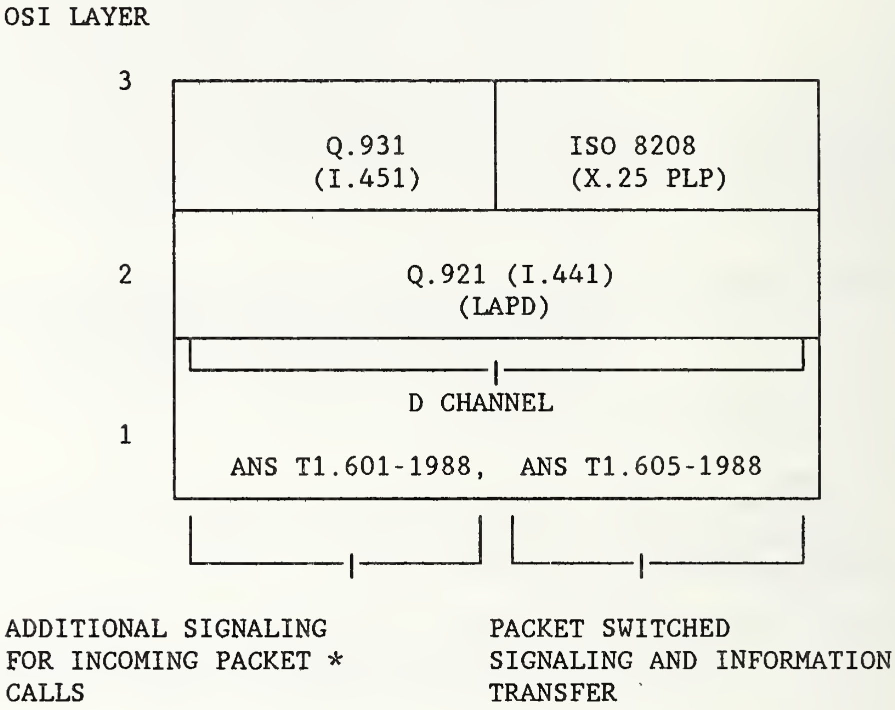
3. X.25 PLP (ISO 8208) procedures are used, including the setting up and clearing of virtual calls.

4. Q.921 and Q.931 procedures on a D channel are used for access signaling, when appropriate, to select the B or D channel for packet data transfer and for establishing and releasing a physical path in the ISDN.

5. Refer to Chapter 3 for the specification of methods for providing OSI Network Services.

2.7.2 Implementation Agreements

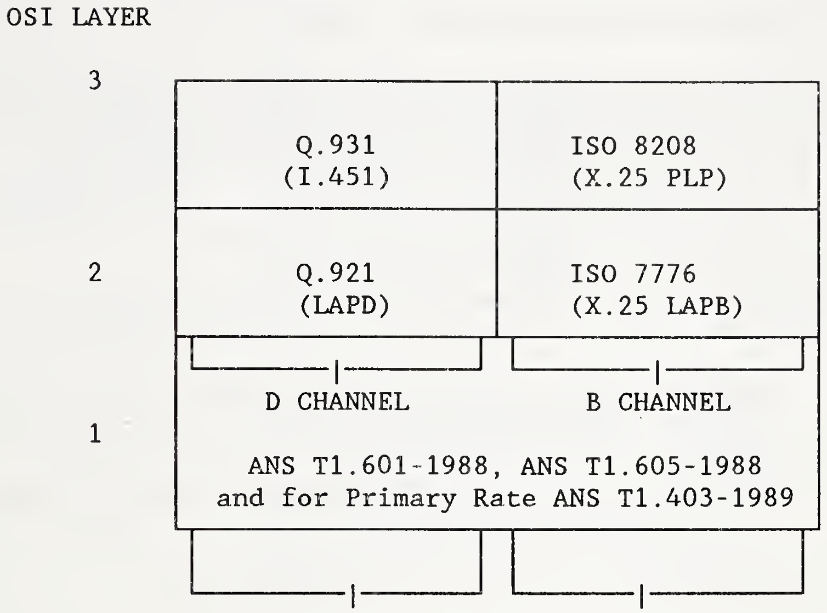
This section gives Implementation Agreements for individual ISDN-related protocols. The relevant protocol stacks are given in Figures 2.3 and 2.4.



* MAY BE NULL

Figure 2.3 Protocol Layers at S, T and U reference points when D Channel is used in ISDN

Editor's Note: The addition of the "U" reference point is noted in this figure.



OSI LAYER

3

Q.931
(I.451)

ISO 8208
(X.25 PLP)

2

Q.921
(LAPD)

ISO 7776
(X.25 LAPB)

1

D CHANNEL

B CHANNEL

ANS T1.601-1988, ANS T1.605-1988
and for Primary Rate ANS T1.403-1989

SIGNALING FOR CIRCUIT SWITCHED ACCESS* ADDITIONAL SIGNALING FOR INCOMING PACKET CALLS*

PACKET SWITCHED SIGNALING AND INFORMATION TRANSFER

* MAY BE NULL

Figure 2.4 Protocol Layers at S, T and U reference points when B Channel is used in ISDN

Editor's Note: The addition of the "U" reference point is noted in this figure.

2.7.2.1 Physical Layer, Basic Access at "U"

ANS T1.601-1988, "Integrated Services Digital Network-Basic Access Interface for Use on Metallic Loops for Application on the Network Side of the NT-Layer 1 Specification" applies.

2.7.2.2 Physical Layer, Basic Access at S and T

ANS T1.605-1988, "Integrated Services Digital Network-Basic Access Interface at S and T Reference Points-Layer 1 Specifications" applies.

2.7.2.3 Physical Layer, Primary Rate at "U"

The physical layer is governed[1] by ANS T1.403-1989, "Carrier-to-Customer Installation Metallic Interface", and CCITT Recommendation I.431-1988, "Primary Rate User-Network Interface - Layer 1 Specification subject to the exceptions given below.

The following portions of ANS T1.403-1988 shall be deleted.

- Section 5.3.1:	The bit rate tolerance option of +/- 200 bps.
- Section 5.6:	The minimum pulse density of this section.
- Section 6.1:	The superframe format.
- Section 6.3:	The complete section.
- Section 8.0:	The reference to the SF format.
- Section 8.3:	The text in paragraph 8.3.1.1 and footnote 7 (8.3.1.2).
- Section 8.4.1:	Footnote 9.
- Section 9/Fig. 9	Provisions for the use of the RJ48M connector.
- Table 2	This table.
- Table 3	The illustration in Table 3 of "Robbed-Bit Signaling".

The following portions of ANS T1.403-1989 shall be modified :

[1] ANSI accredited subcommittee T1E1 is developing a standard for the ISDN primary rate interface at reference points "S" and "T" as well as "U". One of the accepted guidelines for the standard is consistency with ANS T1.403-1988. It is intended that, when this new ISDN-unique standard is adopted, this agreement will be modified to reference it and will be extended to cover interfaces at reference points "S" and "T" as well as "U".

- Section 5.3.2 The text of this section is replaced by:

  "The line code is B8ZS except as noted in Section 7", and

- Section 7.0: The reference to the pulse density requirements of Section 5.6 is inappropriate. The text is replaced by:

  "The provisions of Clear Channel Capability (CCC) depends upon the use of the B8ZS line code, though the use of ZBTSI is one interim method that may be employed by agreement of the network and the user"

The provisions of ANS T1.403-1989 shall be supplemented by the provisions of CCITT Recommendation I.431 - Section 4.4.

2.7.2.4 Data Link Layer, D-Channel

CCITT Recommendation Q.921 (I.441), "ISDN User-Network Interface Data Link Layer Specification" applies.

2.7.2.5 Signaling

CCITT Recommendations Q.931 (I.451), "ISDN User-Network Interface Layer 3 Specification" applies.

The following agreements have been reached concerning the use of Q.931.

1 On a Basic Rate Interface supporting the ISDN virtual circuit service, all of Q.931 Section 6 except for 6.1.1 and 6.2.1 (the sections covering the circuit-switched access case) shall apply. The following sections also apply: 2.2, packet mode access connection states; 3.2, messages for packet mode access connection control; 4-4.5, section specifying general information element handling and encoding; 4.7, information elements for packet communications.

2 On a Primary Rate Interface supporting the ISDN virtual circuit service all of Q.931 Section 6 shall apply except for 6.1.1 and 6.2.1 (the sections

specifying the circuit switched access case), 6.1.2.2, 6.2.1, 6.2.2.2 and 6.4.2 (the sections specifying D-Channel ISDN Virtual Circuit service case). The following sections also apply: 2.2, packet mode access connection states; 3.2 messages for packet mode access connection control; 4-4.5, sections specifying general information element handling and encoding; 4.7, information elements for packet communications.

3 On a Basic or Primary Rate Interface supporting the unrestricted 64-Kbps circuit-mode service, Q.931 sections 6.1.1, 6.2.1, 6.4.1 and 6.4.3 shall apply. The following sections also apply: 2.1, circuit mode connection states; 3.1, messages for circuit mode connection control; 4-4.5, sections specifying general information element handling and encoding.

2.7.2.6 Data Link Layer B-Channel

The agreements on ISO 7776 specified in Section 2.6.2 shall apply here.

If the ISDN provides a circuit-mode service between two ISDN packet-mode devices, then the layer 2 address shall be assigned as follows:

1 For permanent ("non-switched") circuit-mode service, one terminal uses address A and the other terminal uses address B, as arranged by prior agreement, and

2 For demand ("switched") circuit-mode service, the terminal originating the circuit-mode call uses address A and the other terminal uses address B.

2.7.2.7 Packet Layer

The agreements on ISO 8208 specified in Section 2.6.3 shall apply here. When ISO 8208 is used on the D-Channel, the maximum DATA packet size (i.e., actually the maximum size of the User Data Field in a DATA packet) shall be limited to 256 octets.

2.7.3 Rate Adaptation

(Refer to the Ongoing Implementation Agreements Document)

3. NETWORK LAYER

3.1 INTRODUCTION

This chapter presents agreements for providing the OSI network service. Also contained here are agreements on network layer addressing and routing.

3.2 SCOPE AND FIELD OF APPLICATION

These agreements cover both connectionless-mode and connection-mode network services.

3.3 STATUS

This version of the agreements was completed in December, 1988.

3.4 ERRATA

3.5 CONNECTIONLESS-MODE NETWORK SERVICE (CLNS)

3.5.1 ISO 8473

1. Subsets of the protocol:

   o Implementations will not transmit PDUs encoded using the inactive subset. Received PDUs encoded using the inactive subset will be discarded.

   o The non-segmenting subset will not be used. Implementations will not generate data PDUs without a segmentation part. However, implementations will receive and correctly process PDUs which do not contain the segmentation part.

2. Mandatory Functions:

   o The lifetime parameter shall be used as specified in Section 6.4 of ISO 8473. The parameter shall have an initial value of at least three times the network span or three times the maximum transit delay (in units of 500 milliseconds), whichever is greater.

   o The reassembly timer for an initial PDU at the reassembly point shall be no greater than the largest value of all lifetime parameters contained in all derived PDUs.

- The use/non-use of checksums shall be capable of being configured. The default value shall be non-use.

  Editor's Note: The vote on the bullet above has not been presented to the Editor.

- If the implementation supports the generation of an ER PDU, the system shall insert in the destination address field of the ER PDU the contents of the source address field of the PDU that generated the error.

3. Optional Functions:

- The Security parameter is not defined by these Agreements. Implementations shall not transmit the parameter except where defined by bilateral agreements.

- Partial and complete source routing will <u>not</u> be supported.[2]

- Partial record of route will be supported by Intermediate systems.

- ISO 8473 will be followed with respect to QOS.

- For systems implementing the congestion notification function, the following applies.

  A Globally Unique QOS Maintenance parameter shall be included in all PDU originated by End Systems. As specified in ISO 8473, the initial value of the Congestion Experienced flag (CE flag) within the Globally Unique QOS Maintenance Parameter shall be set by the originating End System to zero. All other flags within the Globally Unique QOS Maintenance Parameter shall be set based on the specific local needs of the originating End System.

  Intermediate systems not implementing queue length averaging shall leave the CE flag in the same state as it was received. In particular, no intermediate system (IS) shall ever clear (set to zero) the CE flag.

[2] A defect exists with the Partial Source Routing option which can cause PDUs to loop in the network until their lifetime expires.

All intermediate systems shall monitor all incoming and outgoing queues and compute average queue lengths as shown by example in Table 3.1. The averaging is done from the beginning of the previous cycle to the current time. A cycle begins at the instant of the first NSDU arrival after an idle period.

An IS should set the CE flag in all NSDUs forwarded on a queue which has an average queue length greater than one.

The queue length averaging algorithm computes the average queue length over two cycles, where the two cycles are:

1) the "previous cycle", which is the interval from when the IS becomes busy, until it becomes idle and the idle ends (indicated by the instant the first packet arrives to the idle IS), and

2) the "current cycle", which is the interval from the end of the idle interval to the current time instant when the average queue length is computed.

An embodiment of the averaging algorithm is shown in Table 3.1.

Table 3.1 Queue Length Averaging Algorithm

The algorithm makes use of the following variables:

t = Current time
t_i = time of i^{th} arrival or departure event
q_i = number of packets in the system after the event
T_0 = time at the beginning of the previous cycle
T_1 = time at the beginning of the current cycle

The algorithm consists of three components:

1. Queue Length Update: Beginning with $q_0 = 0$,
   If the i^{th} event is an arrival event, $q_i = q_{i-1}+1$
   If the i^{th} event is a departure event, $q_i = q_{i-1}-1$

2. Queue Area (integral) update:

$$\text{Area of the previous cycle} = \sum_{t_i \epsilon \{T_0, T_1)} q_{i-1}(t_i - t_{i-1})$$

$$\text{Area of the current cycle} = \sum_{t_i \epsilon \{T_1, t)} q_{i-1}(t_i - t_{i-1})$$

3. Average Queue Length Update:

Average Queue length over the two cycles

$$= \frac{\text{Area of the two cycles}}{\text{Time of the two cycles}} = \frac{\text{Area of the two cycles}}{t-T_0}$$

- (Refer to the Ongoing Implementation Agreements document for additional optional functions)

3.5.2 Provision of CLNS over Local Area Networks (LANS)

When providing CLNS over a LAN subnetwork, the following shall apply:

1. The definition of CLNS shall be as specified in ISO 8348/ADD,

2. The protocol used to provide CLNS shall be ISO 8473 with agreements as specified in 3.5.1, and

3. The necessary subnetwork dependent convergence function shall be as defined in ISO 8473 - Section 8.5.1, "SNDCF used with ISO 8802/2 sub-networks".

3.5.3 Provision of CLNS over X.25 Subnetworks

When providing CLNS over X.25 subnetworks, the following shall apply:

1. The definition of CLNS shall be as specified in ISO 8348/ADD,

2. The protocol used to provide CLNS shall be ISO 8473 with agreements as specified in 3.5.1, and

3. The necessary subnetwork dependent convergence function shall be as defined in ISO 8473 - Section 8.5.2, "SNDCF used with ISO 8208 subnetworks for operation over X.25 subnetworks," The default throughput class shall be used if this facility is available, and

4. The X.25 PLP shall be as defined in ISO 8208.

3.5.4 Provision of CLNS over ISDN

When providing CLNS over an ISDN, the following shall apply.

3.5.4.1 CLNP Utilizing X.25 Services

- The definition of CLNS shall be as specified in ISO 8348/ADD.

- The protocol used to provide CLNS shall be ISO 8473 with agreements as specified in Section 3.5.1, and

- The necessary Sub-network Dependent Convergence function shall be as defined in:

  - ISO 8473 for operation of CLNP over X.25 with agreements as specified in 3.5.3, and

  - ISO/DIS 9574 for control of the B and D channels.

  Note: The stated scope of ISO/DIS 9574 does not explicitly cover the operation of CLNP over an ISDN. However, the procedure identified for operating X.25 in conjunction with I.451 are still applicable. The procedures in ISO/DIS

9574 that correspond to 8878 are not utilized when providing CLNS.

- The X.25 PLP shall be as defined in ISO 8208.

- The agreements for the ISDN-related protocols are specified in Section 2.7.

3.5.5 PROVISION OF CLNS OVER POINT-TO-POINT LINKS

(Refer to the Ongoing Implementation Agreements document).

3.6 CONNECTION-MODE NETWORK SERVICE (CONS)

The following agreements concern provision of the connection-mode Network Service.

3.6.1 Mandatory Method of Providing CONS

3.6.1.1 General

Independent of the subnetwork type (of Section 2), when providing the CONS using X.25-1984, the following shall apply as described below.

- The definition of the CONS is as specified in ISO 8348, Network Service Definition.

- The mapping of the elements of the CONS to the elements of the X.25 Packet Level Protocol (PLP) is as specified in ISO 8878, Use of X.25 to Provide the Connection-mode Network Service.

- The general procedures and formats of the X.25 PLP are as specified in ISO 8208, X.25 Packet Level Protocol for Data Terminal Equipment.

- CONS may be provided as part of the subnetwork types mentioned in Section 2. In particular, when CONS is provided in a Local Area Network, ISO 8881, in addition to the documents listed above, shall apply.

3.6.1.2 X.25 WAN

No provisions additional to those in Section 3.6.1.1 apply in an X.25 WAN.

3.6.1.3 LANs

When providing the CONS in a Local Area Network, the following aspects of ISO 8881, in addition to the documents listed in Section 3.6.1.1, shall apply:

- o Clauses 1-6 and 9-11 for LLC Type 1 operation, including the additional nonstandard default packet size listed in Clause 6.3, Note 2

Note: Operation of ISO 8208 in conjunction with LLC Type 2 requires agreement on LLC Type 2 procedures.

3.6.1.4 ISDN

When providing the CONS in an ISDN, the considerations for control of a B and D channel in ISO/DIS 9574, in addition to those provided in Section 3.6.1.1, shall apply.

3.6.2 Additional Option: Provision of CONS over X.25 1980 Subnetworks

When providing CONS over an X.25 1980 subnetwork, the following shall apply:

- o The definition of the CONS is as specified in ISO 8348, Network Service Definition, and

- o The subnetwork dependent convergence protocol required to provide CONS shall be as specified in ISO 8878 Annex A, and referred to as the Alternative Procedures for Network Connection Establishment and Release, with agreements as defined in 3.6.3.2.

3.6.3 Agreements on Protocols

3.6.3.1 ISO 8878

- o The Receipt Confirmation service will not be provided, so the corresponding protocol elements need not be implemented.

- o The Expedited Data service will not be provided, so the corresponding protocol elements need not be implemented.

- o Where the ISO 8208 diagnostic codes are not provided, all Cause/Diagnostic code combinations can be mapped to the Originator/Reason code of "Undefined".

3.6.3.2 Subnetwork Dependent Convergence Protocol (ISO 8878/Annex A)

- o The Receipt Confirmation service will not be provided, so the corresponding protocol elements need not be implemented.

- o The Expedited Data service will not be provided, so the corresponding protocol elements need not be implemented.

3.7 ADDRESSING

Address formats supported will conform to Addendum 2 of ISO 8348.

- o NSAP address formats will have a hierarchical structure. This will reduce the size of routing tables.

- o If used in the Domain Specific Part (DSP), an NSAP selector shall be the least significant component in the hierarchy. The NSAP selector shall not be used to preform routing; it is simply intended to identify the network service user at the destination end system. For those implementations using an NSAP selector, there shall be one and only one selector for each NSAP within the end system. All NSAP addresses identifying a given NSAP will use the same NSAP selector value.

3.8 ROUTING

Editor's Note: The revised text for 3.8 received conditional approval subject to review (with several abstensions) in the SIG vote.

The OSI routing problem has been decomposed into two distinct classes of routing:

- End Systems (ES) to Intermediate Systems (IS) routing.

- IS-to-IS routing.

3.8.1 End System to Intermediate System Routing

For use in conjunction with ISO 8473 over LANs (refer to Section 2.5) and point-to-point links, ISO 9542 shall be used to exchange configuration information.

> **Editor's Note:** The mandatory use of ISO 9542 as stated above received unconditional support from the SIG.

Additionally, a management mechanism capable of adding and deleting entries into the Routing Information Base (RIB) is recommended. When using the management mechanism to add an entry, there should be no holding timer, and the entry should be write protected from alteration by the ES-IS protocol. This mechanism enables routing table entries to be made which are static in nature.

The agreements below apply to the use of ISO 9542.

1. Implementors shall support any valid NSAP format. For the purposes of the protocol, NSAP addresses are treated simply as octet strings.

2. For LANs, implementors shall support both Configuration Information and Route Redirection Information; no subsets are permitted.

3. All timer values shall be configurable.

4. Use or non-use of checksums shall be configurable. It is recommended not to use ISO 9542 checksums when originating PDUs.

5. The QOS, Security and Priority parameters should not be

used for routing.[3] For conformance, intermediate systems must transmit these parameters in RD PDUs if they are present in the data PDU which generated the redirect. However, end systems must ignore them in received RD PDUs.

6. End systems and intermediate systems shall support the configuration notification function in Clause 6.7 of the protocol specification. A mechanism shall be provided to enable/disable this function. For end systems listening to both ISHs and ESHs, this function shall only be invoked upon receipt of an ISH.

7. This protocol employs the same LSAP as ISO 8473.

8. The encoding of the BSNPA address follows the syntax rules for the data link being used. On a LAN, for example, it is a 48-bit MAC address.

9. The multicast addresses corresponding to "All Intermediate Systems on the network" (All_ISN) and "All End Systems on the network" (All_ESN) shall default to the following:

   All_ESN = 0900 2B00 0004
   All_ISN = 0900 2B00 0005

10. The Error Report flag shall be set to zero (0) for NPDUs sent as a result of invoking the QUERY Configuration Function.

3.8.2 Intermediate Systems to Intermediate Systems Routing

Intermediate systems shall provide mechanisms to create and update the required Routing Information Base.

3.9 PROCEDURES FOR OSI NETWORK SERVICE/PROTOCOL IDENTIFICATION

[3] In order to effectively route on these fields, a standard algorithm must be agreed upon. When such an agreement is available, this restriction will be relaxed.

3.9.1 General

The Protocol Identifiers specified in ISO DTR 9577 ("Protocol Identification in the OSI Network Layer") provide a basis from which OSI systems (both end systems and intermediate systems) may derive a set of procedures for indicating which OSI protocols are used in a particular instance of communication. As such, these procedures are only concerned with Initial Protocol Identifiers (IPIs) and Subsequent Protocol Identifiers (SPIs) that identify OSI protocols and pertain to the following types of systems:

A. systems providing/supporting only CONS (using ISO 8208/8878),
B. systems providing/supporting only CLNS (using ISO 8473), and
C. systems providing/supporting both CONS and CLNS.

From this set of definitions, the following possibilities for success (S) or failure (F) of an instance of communication can be defined, as shown in the table below:

Table 3.2 - End Systems Communications

Originating End System Type	Destination End System Type A	B	C
A	S	F	S
B	F	S	S
C	S	S	S

3.9.2 Processing of Protocol Identifiers

The usage of Protocol Identifiers in Network Protocol Data Units (NPDUs) depends on several factors:

- the OSI Network Service to be provided,
- the protocol to be used in providing this service,
- the role the protocol is to be used in (per the Internal Organization of the Network Layer), and
- the type of subnetwork to which the system is connected.

3.9.2.1 Originating NPDUs

The use of a particular OSI Network Service depends on the capabilities of both the origination and destination end systems. It is not the intent of this section to provide guidelines on how to make this choice except for simple obvious criteria; rather, it is intended only to provide guidance on how to convey this choice to the destination system.

Where a priori knowledge exists in the originating end system about the capabilities (with respect to OSI Network Services available) of the destination end system, it should be used. This may result in no communication if the two end systems involved only provide Network Services of different types. A selection is required in cases where both end systems provide both types of network services; this selection is conveyed by the use of the IPI and SPI (but the selection process is an implementation matter). Alternatively, where a priori knowledge does not exist, then the selection of a service to use in an instance of communication depends solely on the capabilities of the originating end system as described below.

- If only CONS-related protocols (e.g., ISO 8208) are available, then this should be used and the Protocol Identifiers specified so as to reflect the chosen protocol(s) and service.

- If only CLNS-related protocols (e.g., 8473) are available, then this should be used and the Protocol Identifiers specified so as to reflect the chosen protocol(s) and service.

- If both services are available, then other criteria are used in deciding which to use in an instance of communication.

Note: The choice of OSI Network Service to be used in an instance of communication is reflected in the Network Service primitives issued by the Network Service user.

Once a selection of Network Service has been made, the use of particular protocols depend on, for example, the subnetwork to which the originating End System is attached. Some specific cases are given in Annex A of ISO DTR 9577. Another case involves use of the Protocol for Providing the Connectionless Network Service directly over the Data Link Service, as given in ISO 8473 (e.g., in a LAN). In this case, the IPI indicates ISO 8473.

3.9.2.2 Destination System Processing

A system receiving an NPDU must first be concerned with the protocol identified by the IPI. Valid values are given in Table 2 of ISO DTR 9577. If the protocol is recognized as one supported by the system, further processing of the protocol is performed according to the rules of that protocol. If not, an error is recognized and may be conveyed to the originating peer entity. With respect to ISO 8208 and ISO 8473, the following would apply for such error conditions.

1. For ISO 8208, the condition is classified as an "invalid General Format Identifier", for which a DIAGNOSTIC packet may be returned. If DIAGNOSTIC packets are not used by the system, the NPDU is discarded without any further action.

2. For ISO 8473, the NPDU is discarded without any further action.

Given acceptance of the protocol identified by the IPI, the system must also determine the acceptability of the subsequent protocols and OSI Network Service being requested. Use of ISO 8473 implies CLNS; however, use of ISO 8208 can imply either CONS or CLNS, as identified by the SPI. In the case of ISO 8208, therefore, further processing is needed to determine the acceptability of the requested protocol/service. If these are not acceptable (e.g., not supported by the system), the call should be cleared with a diagnostic code of "Connection Rejection - unrecognizable protocol identifier in user data" (decimal 249).

Note: In ISO 8208, a call may be refused for reasons other than non-support of the requested OSI Network Service.

3.9.2.3 Further Processing in Originating End System

Further processing on receipt of an NPDU in response to an initial attempt to communicate may be necessary/useful to determine the success of such an attempt.

For ISO 8473, when used directly over the Data Link Service, the success or failure of an attempt to communicate may not be visible/obvious within the Network Layer. On the other hand, use of ISO 8473 over ISO 8208 may provide, via the diagnostic code in a received CLEAR INDICATION packet, an indication of failure to communicate (e.g., the remote system does not support CLNS).

When using ISO 8208 to provide the CONS, the diagnostic code in a received CLEAR INDICATION packet may provide the necessary indication of why a call was refused.In cases where an ISO 8208 call is refused with diagnostic #249, it would not be desirable to re-attempt such calls with the exact same set of parameters; however, how the originating system ensures this is a local matter.

In cases where an originating system is capable of supporting both OSI Network Services, it may wish to re-attempt communications using the other mode of Network Service than that initially attempted.

3.9.3 APPLICABLE PROTOCOL IDENTIFIERS

(Refer to the Ongoing Implementation Agreements Document.)

3.10 MIGRATION CONSIDERATIONS

This section considers problems arising from evolving OSI standards and implementations based on earlier versions of OSI standards.

3.10.1 X.25-1980

Until there is widespread availability of 1984 X.25 service, it will be necessary for X.400 systems to use those existing packet-switched public data networks which offer only pre-1984 X.25 service. While 1980 X.25 does not provide the CONS as defined by ISO 8348, there is no implication of non-conformance to these Agreements resulting there from for systems using 1980 X.25 to interchange data at the Network Layer, provided they conform in all other respects.

This is an exception to the Agreements for providing the OSI Network Service, granted temporarily for practical reasons. This exception will be removed when it is deemed to be no longer necessary, in the judgement of the Workshop. While this provision is in effect, it provides an alternative method of using 1980 X.25 to the provisions of 3.6.2.

3.11 USE OF PRIORITY

(Refer to the Ongoing Implementation Agreements document).

3.11.1 INTRODUCTION

(Refer to the Ongoing Implementation Agreements document).

3.11.2 OVERVIEW

(Refer to the Ongoing Implementation Agreements document).

3.12 CONFORMANCE

(Refer to the Ongoing Implementation Agreements document).

4. TRANSPORT

4.1 INTRODUCTION

These agreements support the integration of LANs, packet networks, and other WANs with the smallest possible set of mandatory protocol sets, in accordance with the other agreements already reached. Nothing here shall preclude vendors from implementing protocol suites in addition to the ones described in this document.

4.2 SCOPE AND FIELD OF APPLICATION

This chapter presents agreements for providing the OSI Transport layer services over both connection mode and connection-less mode.

4.3 STATUS

Completed December 1988.

4.4 ERRATA

4.4.1 ISO/CCITT DEFECT REPORTS

A misalignment between the CCITT and ISO parameter coding values for sequence number and flow control confirmation has been identified. As a short term solution, the following ISO encoding should apply:

subsequence number	1000 1010
flow control conformation	1000 1100

It is intended that, when an ISO/CCITT solution to this defect is available, this agreement will be modified to align with the solution.

4.5 PROVISION OF CONNECTION MODE TRANSPORT SERVICES

Three connection mode protocol classes have been identified for implementation. Transport classes 0, 2 and 4 of X.224 (1988) have been endorsed for use over CONS. Only Transport Class 4 of ISO 8073 ADD[4] has been endorsed for use over CLNS. The following class combinations are endorsed for CONS: (0), (0,2) or (0,2,4).

[4] All references to ISO 8073 in ISO 8073 ADD should be interpreted as applying to X.224 (1988)

4.5.1 TRANSPORT CLASS 4

4.5.1.1 Transport Class 4 Overview

Transport Class 4 is mandatory for communication between systems using the OSI CLNS and may also be used for systems using the OSI CONS (i.e., a private MHS, etc.).

4.5.1.2 Protocol Agreements

A disconnect request shall be issued in response to a connect request when the maximum number of Transport connections is reached or exceeded.

4.5.1.2.1 Rules for Negotiation

- All implementations shall request "use of extended formats" in the CR TPDU. Implementations shall accept the "use of extended formats" in the CC TPDU if it was proposed in the CR TPDU. Implementations shall accept "use of normal formats" if it was proposed in the CR TPDU.

- Negotiation of protection is outside the scope of these agreements. If negotiation of protection is not supported, receipt of the protection parameters in CR TPDU and CC TPDU shall be ignored.

- Implementations shall be capable of proposing and accepting the non-use of checksums.

  Editor's Note: The SIG vote on the above bullet has not been received by the editor.

- Use of the acknowledgement time parameter is optional . If an implementation is operating any policy which delays the transmission of AK TPDUs, the maximum amount of time by which a single AK TPDU may be delayed shall be indicated to the peer Transport service provider using the acknowledgement time parameter. The value transmitted should be expressed in units of milliseconds and rounded up to the nearest whole millisecond.

- QoS negotiation is outside the scope of these agreements. If QoS negotiation is not supported, receipt of the parameters "throughput", "residual error rate", "priority", and "transit delay" in the CR and CC TPDUs shall be ignored.

- Implementations shall not send user data in the CR TPDU or the CC TPDU. The disposition of any user data received in a CR TPDU or CC TPDU is implementation dependent.

- An unknown parameter in any received CR TPDU shall be ignored.

- A Transport entity shall accept a DR TPDU and a corresponding DC TPDU with or without a checksum in response to a CR or CC TPDU.

- Transmitted DR TPDUs shall carry a disconnect reason code which pertains to the actual cause of the disconnect. A DR TPDU may carry a reason code of "0" (unspecified) if an appropriate reason code is not defined.

- Known parameters with valid lengths but with invalid values in a CR TPDU shall be handled as follows:

Parameter	Action
TSAP id	Send DR TPDU
TPDU size	ignore parameter, use default
Version	ignore parameter, use default
Checksum	discard CR TPDU
Alternate Protocol Classes	Protocol Error

Editor's Note: The SIG vote to delete the Table entry for acknowledgement timer received 1 abstension.

- The Transport expedited data transfer function shall be supported. Transport expedited service is not available unless both Transport entities negotiate its use.

- Unrecognized or not applicable bits of the Additional Options parameter shall be ignored.

4.5.1.2.2 TRANSPORT CLASS 4 SERVICE ACCESS POINTS OR SELECTORS

The TSAP selector field in the CR and CC TPDUs shall be encoded as a variable length field and will be interpreted as an octet string. The length of the string cannot exceed 32 octets.

4.5.1.2.3 Retransmission Timer

It is recommended that the value used for the retransmission timer be based upon the round-trip delay experienced on a transport connection. The implementation should maintain, and continually update, an estimate of the round-trip delay for the TC. From this estimate, a value for the retransmission timer is calculated each time it is started. An example technique for maintaining the estimate and calculating the retransmission timer is described below. The value of the retransmission timer may be calculated according to the following formula:

$$T1 \longleftarrow kE + AR$$

In this formula, E is the current estimate of the round-trip delay on the transport connection, AR is the value of the acknowledgement time parameter received from the remote transport service provider during connection establishment, and k is some locally administered factor.

A value for k should be chosen to keep the retransmission timer sufficiently small such that lost TPDUs will be detected quickly, but not so small that false alarms are generated causing unnecessary retransmission.

The value of E may be calculated using an exponentially weighted average based upon regular sampling of the interval between transmitting a TPDU and receiving the corresponding acknowledgement. Samples are taken by recording the time of day when a TPDU requiring acknowledgement is transmitted and calculating the difference between this and the time of day when the corresponding acknowledgement is received. New samples are incorporated with the existing average according to the following formula.

$$E \longleftarrow E + (1 - \alpha)(S - E)$$

In this formula, S is the new sample and α is a parameter which can be set to some value between 0 and 1. The value chosen for α determines the relative weighting placed upon the current estimate and the new sample. A large value of α weights the old estimate more heavily causing it to respond only slowly to variations in the round-trip delay.

A small value weights the new sample more heavily causing a quick response to variations. (Note that setting α to 1 will effectively disable the algorithm and result in a constant value for E, being that of the initial seed.)

If α is set to $1-2^{-n}$ for some value of n, the update can be reduced to a subtract and shift as shown below.

$$E \longleftarrow E + 2^{-n} (S - E)$$

When sampling, if an AK TPDU is received which acknowledges multiple DT TPDUs, only a single sample should be taken being the round-trip delay experienced by the most recently transmitted DT TPDU. This attempts to minimize in the sample any delay caused by the remote transport service provider withholding AK TPDUs.

4.5.1.2.4 Keep-Alive Function

The Class 4 protocol detects a failed Transport connection by use of an 'inactivity timer'. This timer is reset each time a TPDU is received on a connection. If the timer ever expires, the connection is terminated.

The Class 4 protocol maintains an idle connection by periodically transmitting an AK TPDU upon expiration of the 'window timer'. Thus, in a simple implementation, the interval of one transport entity's window timer must be less than that of its peer's inactivity timer, and vice versa. The following agreements permit communicating transport entities to maintain an idle connection without shared information about timer values.

o In accordance with ISO 8073, Clause 12.2.3.9.a, all implementations <u>must</u> respond to the receipt of a duplicate AK TPDU not containing FCC by transmitting an AK TPDU containing the 'flow control confirmation' parameter.

- Implementations must always transmit duplicate AK TPDUs without FCC on expiration of the local window timer (see ISO 8073, Clause 12.2.3.8.1). Receipt of this TPDU by the remote Transport entity will cause it to respond with an AK TPDU containing the 'flow control confirmation' parameter. When this is received by the local transport entity, it will reset its inactivity timer. See Figure 4.1.

- It is a local matter for an implementation to set the intervals of its timers to appropriate relative values. Specifically:

- The window timer must be greater than the round-trip delay. See Section 4.5.1.2.3.

- The inactivity timer must be greater than two times the window timer; and should normally be an even greater multiple if the Transport connection is to be resilient to the loss of an AK TPDU.

A duplicate AK TPDU (See Figure 4.1) is one which contains the same values for YR-TU-NR, credit, and subsequence number as the previous AK TPDU transmitted. A duplicate AK TPDU does not acknowledge any new data, nor does it change the credit window.

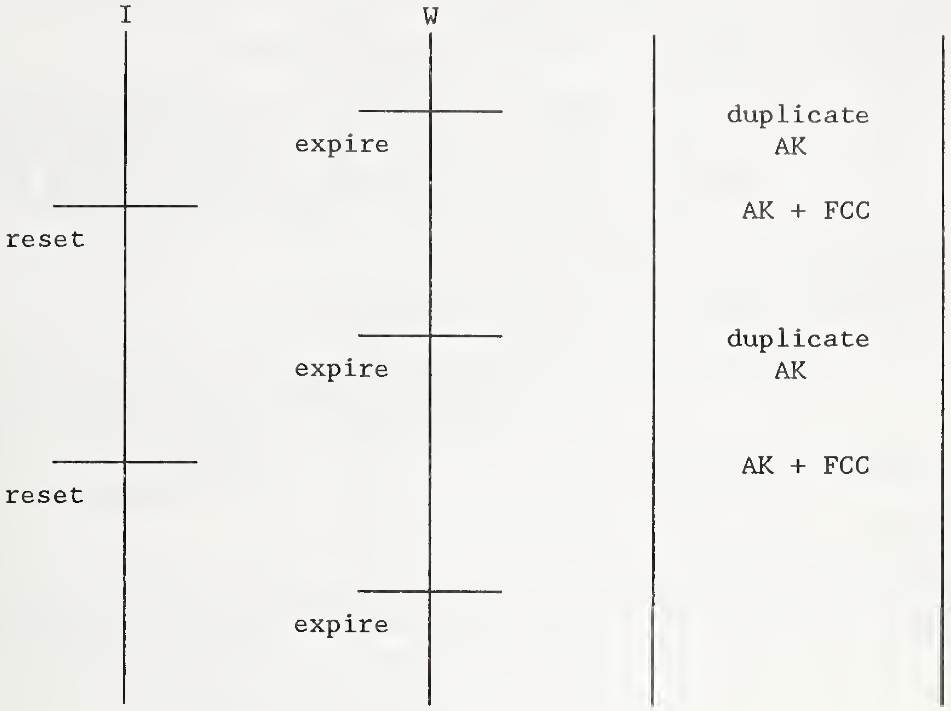


Figure 4.1 AK exchange on idleconnection

4.5.1.2.5 Congestion Avoidance Policies

This section defines both mandatory and optional requirements relating to avoiding congestion in OSI networks and recovering from it when it is experienced. The mandatory requirements specify a minimum approach to congestion avoidance/recovery which can be tuned based upon the specific requirements of the network. The optional requirements specify a dynamic window sizing scheme which, if implemented, will contribute further to the avoidance of congestion in the network.

Mandatory Requirements

(Refer to the Ongoing Implementation Agreements).

Optional Requirements

For systems implementing the dynamic window sizing scheme the following rules apply as described below.

RECEIVING TRANSPORT ENTITY (RTE) RULES:

Rule 1: Initialization of Window

The initial value of WR (known as WR_0) shall have a locally configurable upper bound. This window is sent to the sending transport entity (STE) in the next CDT field transmitted.

Rule 2: Required Sampling Period

All RTEs shall maintain a fixed value for WR until the next 2WR DT TPDU arrive since the last CDT field was transmitted by the RTE.

Rule 3: Required Counting of Received TPDUs in a Sampling Period

All RTEs shall maintain a count, N equal to the total number of TPDUs received and a count, NC equal to the total number of TPDUs received which had the CE Flag set. All types of TPDUs are included in the counts for N and NC, not just DT TPDUs.

Rule 4: Required Action upon the end of a Sampling Period

All RTEs shall take the following action at the end of each sampling period:

- If the count NC is less than fifty percent of the count N, the RTE shall increase WR by adding 1 up to a maximum, WR_1, (that is based on the local buffer management policy) otherwise, it shall decrease WR by multiplying by 0.875 (a minimum of 1).
- Reset N and NC to zero.
- Transmit the new window WR in the next CDT field sent to the sending transport entity.

SENDING TRANSPORT ENTITY (STE) RULES

Rule 1: Initialization of Window

All STEs shall maintain a sending window size (WS). Initially and also as long as there is no loss, WS is set equal to the receiving window value WR received from the remote RTE in the last CDT field.

Rule 2: Required Action on a Timeout

All STEs shall reset WS to one when the retransmissions timer expires and indicates a lost TPDU. WS now limits the number of DT TPDUs that may be transmitted or retransmitted without further acknowledgements.

Rule 3: Required Counting of Acknowledged TPDU

All STEs shall maintain a count, ACKRCVD of the number of DT TPDUs acknowledged, by the RTE, since WS was last adjusted. Therefore each time WS is adjusted, the count ACKRCVD shall be reset to zero.

Rule 4: Increase Window Policy

All STEs shall increase WS by one each time ACKRCVD is equal to or greater than the current value of WS, unless WS exceeds the window permitted by the remote RTE.

4.5.1.2.6 USE OF PRIORITY

(Refer to the Ongoing Implementation Agreements).

4.5.2 TRANSPORT CLASS 0

4.5.2.1 Transport Class 0 Overview

Transport Class 0 over X.25 is mandatory (see X.400) for use in communicating with public MHS systems operating in accordance with the CCITT X.400 series recommendations. The purpose of the agreements concerning Transport Class 0 is to allow connection to these public services. Transport Class 0 over X.25 can also be used in communicating between PRMDs (this choice is prevalent outside North America).

4.5.2.2 Protocol Agreements

Transport Class 0 agreements follow.

- The Error (ER) TPDU may be used at any time and upon receipt requires that the recipient disconnect the network connection, and by extension the transport connection.

- The allowed values for the maximum TPDU size are as specified in ISO 8073. They are: 128, 256, 512, 1024, and 2048.

- The Class 0 protocol does not support multiplexing. At any instant, one Transport corresponds to one Network connection.

- It is recommended that the optional timers TS1 and TS2, if implemented, be settable by local system management. Values in the order of minutes should be supported.

- An unlimited TSDU length must be supported.

4.5.2.2.1 TRANSPORT CLASS 0 SERVICE ACCESS POINTS

For communicating with public MHS systems, Section 5 of X.410 specifies the use and format of TSAP identifiers.

4.5.2.3 Rules for Negotiation

The ISO rules for negotiations will be used.

4.6 CONNECTIONLESS TRANSPORT

(Refer to the Ongoing Implementation Agreements.)

4.7 TRANSPORT PROTOCOL IDENTIFICATION

(Refer to the Ongoing Implementation Agreements).

Editor's Note: This material has not been in the Ongoing Document for the required period of time. The SIG had voted this material into this document (the Stable Document). The SIG feels that this material is stable.

5. UPPER LAYERS

5.1 INTRODUCTION

In this portion of the Implementors' Agreements, the NIST Upper Layers SIG is primarily concerned with providing implementation agreements for ACSE, and the Presentation and Session layers, so that systems implemented according to these agreements can successfully interoperate.

> **Editor's Note:** Subsections have been renumbered to align with Upper Layers section in the Ongoing Agreements Document.

5.1.1 References

All documents referenced in the Upper Layers section of these agreements can be found in the REFERENCES section of this NIST Implementors' Agreements document.

5.2 SCOPE AND FIELD OF APPLICATION

This section does not detail particular conformance statements for ACSE, Presentation, and Session, since what is to be implemented in each case depends on which Application Service Elements (ASE's) and which functional units within each ASE are used with an Application Process. Each ASE's SIG must specify which functional units of each layer it requires. However, the scope of each layer is based on the total indicated requirements of all ASE's for which there is an active NIST SIG. The implementation agreements are not specified beyond that scope.

It is not the intent of this document to specify or reproduce standards, but when a referenced standard is unclear or has known defects, an attempt will be made to remedy the problem herein. Any attempted clarification should be considered as a possible interpretation; the ISO standard still takes precedence if there is any conflict. The situation with respect to defects in a standard is somewhat different; a reported defect may be technically resolved by the appropriate international technical committee with likely approval by the voting members pending for several months. Since relevant defects can't be ignored in an implementation, this document will recommend using defect resolutions which have the tentative approval of the appropriate standards committees.

5.3 STATUS

Final text for this chapter will be available in mid 1989.

> Editor's Note: Upper Layers text was voted into NBS Special Publication 500 - 150 by the Plenary in December, 1987.

5.4 ERRATA

5.4.1 ISO Defect Reports

This section lists the defect reports from ISO which are currently recognized to be valid for the purposes of NIST conformance.

5.4.2 Session Defects

These defects are listed for the benefit of X.400 implementations.

The following 8326 defect reports have been incorporated into version 1 of Session:

004, 006, 007, 009, 011, 012, 013, 014, 015, 016, 017, 020.

The following 8327 defect reports have been incorporated into version 1 of Session:

001, 003, 004, 005, 006, 007, 008, 009, 010, 012, 017, 018, 019, 026, 027, 030, 034, 035.

5.5 ASSOCIATION CONTROL SERVICE ELEMENT

5.5.1 Introduction

This section details the implementation requirements for the Association Control Service Element (ACSE) of the Application layer. It is the intent of this section to follow the ISO ACSE standards. Where those specifications are inadequate, this section should provide the necessary information.

5.5.2 Services

5.5.2.1 ACSE Services

The following ACSE service primitives are within the possible scope of an NIST-conformant system.

1. A_ASSOCIATE request
2. A_ASSOCIATE indication
3. A_ASSOCIATE response
4. A_ASSOCIATE confirm
5. A_RELEASE request
6. A_RELEASE indication
7. A_RELEASE response
8. A_RELEASE confirm
9. A_ABORT request
10. A_ABORT indication
11. A_P_ABORT indication

5.5.2.2 Use of Presentation Layer Services

ACSE services will make use of Presentation layer services in the manner defined in the ACSE Protocol specification.

5.5.3 Protocol Agreements

Implementations shall be based on the ACSE Service definition and the ACSE Protocol specification.

5.5.3.1 Application Context

Specific Application Contexts and their names will be supplied and defined by an appropriate NIST SIG. Other application contexts may be defined and specified as dictated by particular application requirements.

Optional names and specifications are outlined by each application SIG under the heading "Specific ASE Requirements for ACSE, Presentation, and Session". The use of these names implies adherence to the relevant NIST implementors' agreements for a particular application SIG.

The utility of an NIST-defined name (which is an OBJECT IDENTIFIER) is left up to the application. An NIST name may or may not be used in the ACSE APDU. The consequence of the name is left up to the application entities and any a priori agreements that they have. In other words, it is up to the application whether this parameter is ignored or validated for correctness. (Note that the consequence of this name must also be dictated by the particular conformance test).

The UL SIG recognizes that this parameter needs further definition by the appropriate standards bodies. Therefore, the use of this parameter for association negotiation is not recommended at this time.

5.5.3.2 Section Deleted

5.5.3.3 Section Moved to 5.11.1.1.1

5.5.4 ASN.1 ONGOING RULES

(Refer to Ongoing Agreements Document.)

5.6 ROSE

(Refer to Ongoing Agreements Document.)

5.7 RTSE

(Refer to Ongoing Agreements Document.)

5.8 PRESENTATION

5.8.1 Introduction

This section details the implementation requirements for the Presentation layer. It is the intent of this section to follow the ISO Presentation Standards. Where those specifications are inadequate, this section should provide the necessary information.

The task of the Presentation layer is to carry out the negotiation of transfer syntaxes and to provide for the transformation to and from transfer syntaxes. The transformation to and from a particular transfer syntax is a local implementation issue and is not discussed within this section. This section is concerned with the protocol agreements, and thus is entirely devoted to the issues involved with the negotiation of transfer syntaxes and the responsibilities of the Presentation protocol.

5.8.2 Services

5.8.2.1 Presentation Services

The following functional units are within the possible scope of an NIST-conformant system.

Presentation Kernel - This functional unit supports the basic Presentation services required to establish a Presentation connection, transfer normal data, and release a Presentation connection. This is a non-negotiable functional unit.

The Context Management and Context Restoration functional units are not within the scope of an NIST conformant system and need not be supported.

The requirement that the Presentation kernel functional unit be implemented does not imply that any of the Session functional units for expedited data, typed data, and capability data and the corresponding Presentation service primitives are required to be implemented. Any service not supported by the Session layer is also not supported by the Presentation layer; see the section on Session Functional Units for the possible Session functional units. The services provided by the Presentation layer are limited by the services provided by the Session layer as defined in the Session service definition ISO/IS 8326 and the Session protocol definition ISO/IS 8327.

5.8.2.2 Use of Session Layer Services

Presentation layer services shall make use of Session layer services in the manner defined in the Presentation Protocol Specification.

5.8.3 Protocol Agreements

Implementations shall be based on the Presentation Service Definition, ISO 8822 and the Presentation Protocol Definition, ISO 8823.

5.8.3.1 Transfer Syntaxes

- o The following transfer syntax must be supported for all mandatory abstract syntaxes: the basic encoding rules for ASN.1. This syntax is derived by applying the basic encoding rules for ASN.1 to the abstract syntax (see the Basic Encoding Rules for ASN.1, ISO 8825).

- o The number of transfer syntaxes proposed is dependent upon the recognized transfer syntaxes which are available to support the particular abstract syntaxes used by an Application Entity.

5.8.3.2 Abstract Syntaxes

- o Several abstract syntax names may map onto a single transfer syntax name.

- o The ACSE abstract syntax shall always be present in the defined context set.

5.8.3.3 Presentation Context Identifier

- o A conformant implementation shall encode presentation context identifiers in the range 0 to 32,767.

- o Implementations must be able to handle a minimum of two presentation contexts per connection.

5.8.3.4 Section Deleted

5.8.3.5 Default Context

If the Presentation expedited data service is required, the default context must be explicitly present in the P-CONNECT PPDU at Presentation connect time.

5.8.3.6 P-Selectors

Local P-selectors shall be a maximum of four octets. This applies only to P-selectors in PPDUs whose receipt by an NIST-conformant system normally results in either a P-CONNECT indication or a P-CONNECT confirmation being issued.

5.8.3.7 Provider Abort Parameters

No conformance requirements are implied by the use of either the Abort-reason or the Event-identifier component of the ARP-PPDU. The decision to include these parameters is left up to the implementation issuing the abort.

5.8.3.8 Provider Aborts and Session Version

The Presentation Provider Abort PPDU (ARP-PPDU) shall be present regardless of the Session version in effect for a given association. This precludes the use of indefinite length encoding of an ARP-PPDU when Session Version 1 is in effect.

5.8.3.9 CPC-Type

NIST conformant implementations shall not use any CPC-type values in the SS-user data parameter of the S-CONNECT request unless more than one transfer syntax is proposed. Each CPC-type represents a unique transfer syntax.

Note: Currently only one transfer syntax is recognized.

5.8.3.10 Presentation-Context-Definition-Result-List

No semantics are implied by the absence of this optional component of the CPR-PPDU. This component is required if the Provider-reason is absent in the CPR-PPDU. If the Provider-reason is present, then the Presentation-context-definition-result-list is optional.

5.8.3.11 RS-PPDU

The Presentation-context-identifier-list shall not be present when only the kernel functional unit is in effect.

5.8.4 Presentation ASN.1 Encoding Rules

5.8.4.1 Invalid Encoding

If a received PPDU contains any improperly encoded data values (including data values embedded within the User Data field of a PPDU) and an abort is issued, then either an ARU or an ARP shall be issued.

5.8.4.2 Section Deleted

5.8.4.3 Section Deleted

5.8.5 GENERAL

(Refer to Ongoing Agreements Document.)

5.8.6 CONNECTION-ORIENTED

(Refer to Ongoing Agreements Document.)

5.8.7 CONNECTIONLESS

(Refer to Ongoing Agreements Document.)

5.9 SESSION

5.9.1 Introduction

This section details the implementation requirements for the Session layer. It is the intent of this section to follow the ISO Session Standards to the fullest extent possible. Where those specifications are inadequate, this section should provide the necessary information.

5.9.2 Services

5.9.2.1 Session Services

The following functional units are within the possible scope of an NIST-conformant system.

Kernel

Duplex

Expedited Data

Resynchronize

Exceptions

Activity Management

Half-duplex

Minor Synchronize

Major Synchronize

Typed Data

5.9.2.2 Use of Transport Services

The use of Transport layer services by the Session layer functional units listed in the previous section is as specified in the Transport Protocol Specification, ISO/IS 8073.

5.9.3 Protocol Agreements

Implementations shall be based on the Session Service Definition ISO/IS 8326 and the Session Protocol Definition ISO/IS 8327.

5.9.3.1 Concatenation

When a category 0 SPDU is concatenated with a category 2 SPDU, the category 0 SPDU shall contain neither the Token Item field nor User Data. If either a Token Item field or User Data is received in such a concatenated incoming SPDU, the receiving Session Protocol Machine has the option of either properly processing the fields or issuing a provider abort on the connection.
Extended concatenation is not required and can be refused using the normal negotiation mechanisms of the Session protocol.

5.9.3.2 Segmenting

Session segmenting is not required and can be refused using the normal negotiation mechanisms of the session protocol. All conformant implementations shall be able to interwork without Session segmenting.

5.9.3.3 Reuse of Transport Connection

Reuse of a Transport connection is not required and can be refused.

5.9.3.4 Use of Transport Expedited Data

The use of Transport expedited service is as stated in the Session protocol specification: if available, Transport expedited service must be requested and used.

5.9.3.5 Use of Session Version Number

Session Versions 1 and 2 are recognized. Each relevant NIST SIG chooses the version or versions of Session which it requires for a particular implementation phase, and these choices are documented in Section 5.11.

Session Version 2 specifies the use of unlimited user data during connection establishment as dictated by the DAD 2 to ISO 8327 to Incorporate Unlimited User Data.

All Session Version 1 implementations must be able to negotiate Version 1 operation when responding to a CONNECT (CN) SPDU proposing both Version 1 and Version 2.

In addition, all Session Version 1 implementations, upon receipt of a CONNECT (CN) SPDU proposing only Version 2, should respond with a REFUSE (RF) SPDU containing a Reason Code indicating that the proposed version is not supported. Until pending defect reports are adopted, implementations may disconnect.

If Session Versions 1 and 2 are both proposed in the CONNECT (CN) SPDU, then the maximum length of the User Data parameter value in the CONNECT (CN) SPDU shall be 512 octets and a PGI field of 193 shall be associated with this parameter. This implies that an implementation supporting both Session Versions 1 and 2 can establish a connection with an implementation supporting only Version 1.

If only Session Version 2 is proposed in the CONNECT (CN) SPDU, then the maximum length of the Session User Data parameter value of the S-CONNECT service request shall be 10,240 octets. This restriction implies that the OVERFLOW ACCEPT (OA) SPDU and CONNECT DATA OVERFLOW (CDO) SPDU are not used. If the length of the User Data parameter value is no greater than 512 octets, then an associated PGI field of 193 shall be used, otherwise a PGI field of 194 shall be used.

When Session Version 2 is negotiated, then in all SPDUs the maximum length of the User Data parameter value with an associated PGI field of 193 shall be 10,240 octets. NIST-conformant Session Version 2 implementations need only support the maximum data lengths specified in the Specific ASE Requirements section.

5.9.3.6 Receipt of Invalid SPDUs

Upon receipt of an invalid SPDU, the SPM shall take any action in A.4.3 of the Session Protocol Definition ISO/IS 8327 except Action d.

5.9.3.7 Invalid SPM Intersections

If the conditions described in A.4.1.2 of the Session Protocol Definition ISO/IS 8327 are satisfied, the SPM shall always take the actions described by A.4.1.2 a.

Note: This implies that no S-P-EXCEPTION-REPORT indications will be generated nor EXCEPTION REPORT

SPDUs sent due to invalid intersections of the Session state table resulting from received SPDUs.

5.9.3.8 S-Selectors

S-selectors shall be a maximum of 16 octets.

5.9.4 GENERAL

(Refer to Ongoing Agreements Document.)

5.9.5 CONNECTION-ORIENTED

(Refer to Ongoing Agreements Document.)

5.9.6 CONNECTIONLESS

(Refer to Ongoing Agreements Document.)

5.10 UNIVERSAL ASN.1 ENCODING RULES

5.10.1 Tags

The maximum value of an ASN.1 basic encoding tag that need be handled by an NIST-conformant implementation shall be 16,383. This is the maximum unsigned number that can be represented in 14 bits, therefore, the maximum encoding of a tag occupies 3 octets.

5.10.2 Definite Length

The maximum value of an ASN.1 length octets component that need be handled by an NIST-conformant implementation shall be 4,294,967,295. This is the maximum unsigned integer that.can be represented in 32 bits, therefore, the maximum encoding of a length octets component will occupy 5 octets. Also, note this restriction does not apply to indefinite length encoding.

5.10.3 EXTERNAL Type

It is assumed that "Presentation layer negotiation of encoding rules" is always in effect, and therefore clause 32.5 of the Specification of ASN.1, ISO 8824 never applies.

5.10.4 Integer

(Refer to Ongoing Agreements Document.)

5.10.5 String Types

(Refer to Ongoing Agreements Document.)

5.10.6 Bit String

(Refer to Ongoing Agreements Document.)

5.11 CONFORMANCE

In order for an implementation to be in conformance with the NIST implementors' agreements, the rules below shall be followed.

- A conformant implementation must meet all of the requirements of this specification. All documents referenced in the Upper Layers section shall be used as the supporting documents for all implementations of ACSE, Presentation, or Session. The full references for these documents are in the REFERENCES section.

- NIST-conformant implementations shall be ISO conformant. PICS may contain limitations on length or value aspects of a protocol. PICS of NIST-conformant systems shall not contain restrictions more severe than those in these implementation agreements. Note: an implementation may abort a connection if the constraints specified in these agreements are violated.

- Guidelines for implementation of standards' defects will be as per the resolution of such defects by the appropriate ISO standards committee.

5.11.1 Specific ASE Requirements for ACSE Presentation and Session

The following list for each ASE the corresponding NIST SIG's requirements of and restrictions on ACSE, Presentation, and Session.

All listed requirements and restrictions shall be included in an NIST-conformant system and shall be implemented in accordance with these NIST Implementor's agreements.

All OBJECT IDENTIFIERS are specified in terms of their associated ObjectDescriptors. See the chapter on OBJECT IDENTIFIERs for the values of the associated OBJECT IDENTIFIERs.

5.11.1.1 FTAM

5.11.1.1.1 Phase 2

ACSE Requirements:
all

Application Contexts:

- "ISO FTAM" - implies the use of the ACSE and the FTAM ASE.

Abstract Syntaxes:

- "ISO 8650-ACSE1"

  Associated Transfer Syntax:

  - "Basic Encoding of a single ASN.1 type"

A value is defined for the AE Title only to satisfy the FTAM requirement for exchanging fields of this type.

This value does not identify an Application Entity and carries no semantics. The AE title maps onto the AP title and the AE qualifier. If the AE title is used, then both AP title (an OBJECT IDENTIFIER) and AE qualifier (an INTEGER) must be sent.

The value for the AP title is {1 3 9999 1 ftam-nil-ap-title (7)} at this time. Values for the AE qualifier are outside the scope of these agreements.

Presentation Requirements:

Presentation Functional Units:

- kernel

Presentation Contexts:

- At least 3 Presentation Contexts must be supported.

Abstract Syntaxes:

Abstract Syntaxes for conformant Implementations

- "FTAM-PCI"

  Associated Transfer Syntax:

- o "Basic Encoding of a single ASN.1 type.

- o "FTAM unstructured binary abstract syntax"

  Associated Transfer Syntax:
  - o "Basic Encoding of a single ASN.1 type"

Editor's Note: In Definitions below, "NBS" designation will be preserved.

Abstract Syntaxes Depending on Implementation Profile

- o "FTAM-FADU"

  Associated Transfer Syntax:
  - o "Basic Encoding of a single ASN.1 type"

- o "FTAM unstructured text abstract syntax"

  Associated Transfer Syntax:
  - o "Basic Encoding of a single ASN.1 type"

- o "NBS abstract syntax AS1"

  Associated Transfer Syntax:
  - o "Basic Encoding of a single ASN.1 type"

- o "NBS file directory entry abstract syntax"

  Associated Transfer Syntax:
  - o "Basic Encoding of a single ASN.1 type"

Session Requirements:

Session Functional Units:
- o kernel
- o duplex

Version Number: 2

Maximum size of User Data parameter field: 10,240

Session Options:

Session Functional Units:

- resynchronize

  only a Resynchronize Type value of "abandon"

- minor synchronize

  Note: The minor synchronize functional unit is required whenever the resynchronize functional unit is available.

5.11.1.2 MHS

5.11.1.2.1 Phase 1

Session Requirements:

Session Functional Units:

- kernel
- half-duplex
- exceptions
- activity management
- minor synchronize

Version Number: 1

Maximum size of User Data parameter field: 512

Session Notes:

- Restricted use is made by the RTS of the session services implied by the functional units selected. Specifically,

  - No use is made of S-TOKEN-GIVE, and
  - S-PLEASE-TOKENS only asks for the data token.

- In the S-CONNECT SPDU, the Initial Serial Number should not be present.

- The format of the Connection Identifier in the S-CONNECT SPDU is described in Version 5 of the X.400-Series Implementors' Guide.

5.11.1.2.2 Phase 2, PROTOCOL P7

(Refer to Ongoing Agreements Document.)

5.11.1.2.3 Phase 2, PROTOCOL P3

(Refer to Ongoing Agreements Document.)

5.11.1.3 DS

5.11.1.3.1 Phase 1

ACSE Requirements:
all

Application Contexts:

- o "id-ac-directoryAccessAC"
- o "id-ac-directorySystemAC"

Abstract Syntaxes:

- o "ISO 8650-ACSE1"

  Associated Transfer Syntax:
  - o "Basic Encoding of a single ASN.1 type"

- o "id-as-directoryAccessAS"

  Associated Transfer Syntax:
  - o "Basic Encoding of a single ASN.1 type"

- o "id-as-directorySystemAS"

  Associated Transfer Syntax:
  - o "Basic Encoding of a single ASN.1 type"

Presentation Requirements:

Presentation Functional Units:

- o kernel

Presentation Contexts:

- o At least 2 Presentation Contexts must be supported.

Session Requirements:

Session Functional Units:

- o kernel
- o duplex

Version Number: 2

Maximum size of User Data parameter field:
10,240

5.11.1.4 Virtual Terminal

5.11.1.4.1 Phase 1a

ACSE Requirements:
all

Application Contexts:

- "ISO VT" - implies the use of the ACSE and the VT ASE

Abstract Syntaxes:

- "ISO 8650-ACSE1"

  Associated Transfer Syntax:
  - "Basic Encoding of a single ASN.1 type"

Presentation Requirements:

Presentation Functional Units:

- kernel

Presentation Contexts:

- 2

Abstract Syntaxes:

- "VT Basic"

  Associated Transfer Syntax:
  - "Basic Encoding of a single ASN.1 type"

Session Requirements:

Session Functional Units:

- kernel
- duplex
- expedited data
- major synchronize
- resynchronize
  only a Resynchronize Type value of "abandon"
- typed data

Version Number: 2

Maximum size of User Data parameter field: 10,240

Session Options:

- expedited data

5.12 REFERENCES

(Refer to Ongoing Agreements Document.)

5.13 APPENDIX A: RECOMMENDED PRACTICES

Reflect Parameter Values

The optional "Reflect Parameter Values" parameter in the Provider ABORT SPDU shall be encoded so as to represent the Session connection state, the incoming event and the first invalid SPDU field exactly at the moment a protocol error was detected.

The first octet encodes the Session state as a number relative to 0 as detailed in Table 1.

The second octet encodes the incoming event as a number relative to 0

The third octet contains the SI, PGI, or PI Code of any SI field, PGI unit or PI unit in error.

NOTE: The remaining 6 octets are undefined herein.

Table 5.1 Session States

State	rel.#	Description
1	0	Idle, no transport connection
1B	1	Wait for T-connect confirm
1C	2	Idle, transport connected
2A	3	Wait for the ACCEPT SPDU
3	4	Wait for the DISCONNECT SPDU
8	5	Wait for the S-CONNECT response
9	6	Wait for the S-RELEASE response
16	7	Wait for the T-DISCONNECT indication
713	8	Data Transfer state
1A	9	Wait for the ABORT ACCEPT SPDU
4A	10	Wait for the MAJOR SYNC ACK SPDU or PREPARE SPDU
4B	11	Wait for the ACTIVITY END ACK SPDU or PREPARE SPDU
5A	12	Wait for the RESYNCHRONIZE ACK SPDU or PREPARE SPDU
5B	13	Wait for the ACTIVITY INTERRUPT SPDU or PREPARE SPDU
5C	14	Wait for the ACTIVITY DISCARD ACK SPDU or PREPARE SPDU
6	15	Wait for the RESYNCHRONIZE SPDU or PREPARE SPDU
10A	16	Wait for the S-SYNC-MAJOR response
10B	17	Wait for the S-ACTIVITY-END response
11A	18	Wait for the S-RESYNCHRONIZE response
11B	19	Wait for the S-ACTIVITY-INTERRUPT response
11C	20	Wait for the S-ACTIVITY-DISCARD response
15A	21	After PREPARE, wait for the MAJOR SYNC ACK SPDU or the ACTIVITY END ACK
15B	22	After PREPARE, wait for the RESYNCHRONIZE SPDU or the ACTIVITY DISCARD SPDU
15C	23	After PREPARE, wait for the RESYNCHRONIZE ACK SPDU, or the ACTIVITY INTERRUPT ACK SPDU or the ACTIVITY DISCARD ACK SPDU
18	24	Wait for GIVE TOKENS ACK SPDU
19	25	Wait for a recovery request or SPDU
20	26	Wait for a recovery SPDU or request
21	27	Wait for the CAPABILITY DATA ACK SPDU
22	28	Wait for the S-CAPABILITY-DATA response
1D	29	Wait for the CONNECT DATA OVERFLOW SPDU
2B	30	Wait for the OVERFLOW ACCEPT SPDU
15D	31	After PREPARE, wait for the ABORT SPDU

Table 5.2 Incoming Events

Event	Rel.#	Description
SCONreq	0	S-CONNECT request
SCONrsp+	1	S-CONNECT accept response
SCONrsp-	2	S-CONNECT reject response
SDTreq	3	S-DATA request
SRELreq	4	S-RELEASE request
SRELrsp+	5	S-RELEASE accept response
SUABreq	6	S-U-ABORT request
TCONcnf	7	T-CONNECT confirmation
TCONind	8	T-CONNECT indication
TDISind	9	T-DISCONNECT indication
TIM	10	Time out
AA	11	ABORT ACCEPT
AB-nr	12	ABORT - no reuse
AC	13	ACCEPT
CN	14	CONNECT
DN	15	DISCONNECT
DT	16	DATA TRANSFER
FN-nr	17	FINISH - no reuse
RF-nr	18	REFUSE - no reuse
SACTDreq	19	S-ACTIVITY-DISCARD request
SACTDrsp	20	S-ACTIVITY-DISCARD response
SACTEreq	21	S-ACTIVITY-END request
SACTErsp	22	S-ACTIVITY-END response
SACTIreq	23	S-ACTIVITY-INTERRUPT request
SACTIrsp	24	S-ACTIVITY-INTERRUPT response
SACTRreq	25	S-ACTIVITY-RESUME request
SACTSreq	26	S-ACTIVITY-START request
SCDreq	27	S-CAPABILITY-DATA request
SCDrsp	28	S-CAPABILITY-DATA response
SCGreq	29	S-CONTROL-GIVE request
SEXreq	30	S-EXPEDITED-DATA request
SGTreq	31	S-TOKEN-GIVE request
SPTreq	32	S-TOKEN-PLEASE request
SRELrsp-	33	S-RELEASE response reject
SRSYNreq(a)	34	S-RESYNCHRONIZE request abandon
SRSYNreq(r)	35	S-RESYNCHRONIZE request restart
SRSYNreq(s)	36	S-RESYNCHRONIZE request set
SRSYNrsp	37	S-RESYNCHRONIZE response
SSYNMreq	38	S-SYNC-MAJOR request
SSYNMrsp	39	S-SYNC-MAJOR response
SSYNmreq	40	S-SYNC-MINOR request
SSYNmrsp	41	S-SYNC-MINOR response
STDreq	42	S-TYPED-DATA request
SUERreq	43	S-U-EXCEPTION-REPORT request

Table 5.2 - Incoming Events Continued

Event	Rel.#	Description
AD	45	ACTIVITY DISCARD SPDU
ADA	46	ACTIVITY DISCARD ACK SPDU
AE	47	ACTIVITY END SPDU
AEA	48	ACTIVITY END ACK SPDU
AI	49	ACTIVITY INTERRUPT SPDU
AIA	50	ACTIVITY INTERRUPT ACK SPDU
AR	51	ACTIVITY RESUME SPDU
AS	52	ACTIVITY START SPDU
CD	53	CAPABILITY DATA SPDU
CDA	54	CAPABILITY DATA ACK SPDU
ED	55	EXCEPTION DATA SPDU
ER	56	EXCEPTION REPORT SPDU
EX	57	EXPEDITED DATA SPDU
FN-r	58	FINISH - reuse SPDU
GT	59	GIVE TOKENS SPDU
GTA	60	GIVE TOKENS ACK SPDU
GTC	61	GIVE TOKENS CONFIRM SPDU
MAA	62	MAJOR SYNC ACK SPDU
MAP	63	MAJOR SYNC POINT SPDU
MIA	64	MAJOR SYNC ACK SPDU
MIP	65	MINOR SYNC POINT SPDU
NF	66	NOT FINISHED SPDU
PR-MAA	67	PREPARE (MAJOR SYNC ACK) SPDU
PR-RA	68	PREPARE (RESYNCHRONIZE ACK) SPDU
PR-RS	69	PREPARE (RESYNCHRONIZE) SPDU
PT	70	PLEASE TOKENS SPDU with Token Item Parameter
RA	71	RESYNCHRONIZE ACK SPDU
RF-r	72	REFUSE - reuse SPDU
RS-a	73	RESYNCHRONIZE - abandon SPDU
RS-r	74	RESYNCHRONIZE - restart SPDU
RS-s	75	RESYNCHRONIZE - set SPDU
TD	76	TYPED DATA SPDU
CDO	77	CONNECT DATA OVERFLOW SPDU
OA	78	OVERFLOW ACCEPT SPDU
PR-AB	79	PREPARE (ABORT) SPDU

6. OBJECT IDENTIFIERS AND OTHER REGISTRATION ISSUES

Editor's Note: As other registration material becomes stable, it will be included in this section.

All upper layer agreements specified in Chapter 5 of the NIST Special Publication 500-150, "Stable Implementation Agreements for Open Systems Interconnection Protocols" (with errata) are also implicitly included in these agreements.

The following objects need to be administered by an ad hoc registration authority:

Application Context Name
Abstract Syntax Name
Transfer Syntax Name
Document Type Name
Constraint Set Name
File Model
VT Profile
VT Control Object

Since all objects to be administered by the NIST Workshop SIGs are identified by the ASN.1 type OBJECT IDENTIFIER, the following structure shall be used:

Using the NameAndNumberForm (::= identifier (NumberForm)) for an ObjIdComponent we have:

```
ObjectIdentifierValue ::= { identifier1 (NumberForm1)
       identifier2 (NumberForm2)
       identifier3 (NumberForm3)
       identifier4 (NumberForm4)
       identifier5 (NumberForm5)
       identifier6 (NumberForm6) }
```

The assignment of identifiers and NumberForms is as follows:

identifier1	NumberForm1
iso	1

identifier2	NumberForm2
identified-organization	3

identifier3	NumberForm3
issuing-organization	9999

identifier4	NumberForm4
organization-code	1

identifier5	NumberForm5
application-context	1
abstract-syntax	2
file model	3
constraint-set	4
document-type	5
transfer-syntax	6
ftam-nil-ap-title	7
VT profile	8
VT control object	9

Note 1: The value of NumberForm3 is selected for use by implementors of these agreements: it has not been assigned by ISO or by any official Registration Authority. It does correspond to an "ad hoc" issuing organization with an ICD of 9999, as specified by ISO 6523. We intend to use the procedure designated in D.7 of the Specification of ASN.1, ISO 8824 once the appropriate Registration Authority has been established. This mechanism is subject to change dependent upon ISO standards.

Note 2: Specific values for identifier6 and NumberForm6 are chosen as needed by the NIST UL SIG. A table of the currently allocated values is given later.

Note 3: The NIST UL SIG will assign values for identifier5 and NumberForm5 as required by other SIGs.

Note 4: Companies wishing to interoperate may designate themselves with an organization code other than 1 under { iso (1) identified-organization (3) issuing-organization (9999) } for the purpose of defining private OBJECT IDENTIFIERs.

TABLE OF ALLOCATED OBJECT DESCRIPTORS and OBJECT IDENTIFIERS

The values of the first 4 NumberForms are constant, so the following value is defined for use in the table below.

```
NIST-ad-hoc OBJECT IDENTIFIER ::= { 1 3 9999 1 }
```

Note that the only OBJECT IDENTIFIERs herein defined are flagged with an '*'; all other OBJECT IDENTIFIERs and their associated ObjectDescriptors are reproduced here solely for the convenience of implementors. The standards defining these OBJECT IDENTIFIERs and ObjectDescriptors take precedence over the values specified below.

However, note some values defined herein are not entirely arbitrary, i.e., the VT SIG has based its NumberForm6 definitions on the AD1 to ISO 9040.

Editor's Note: The "NBS" designation is preserved for current FTAM entries; for other entries the "NIST" designation is used.

Application Context

```
"ISO FTAM"
     { iso(1) standard(0) 8571 application-context(1) iso-ftam(1) }

"id-ac-directoryAccessAC"
     { joint-iso-ccitt(2) ds(5) 3 1 }

"id-ac-directorySystemAC"
     { joint-iso-ccitt(2) ds(5) 3 2 }

"ISO VT"
      { iso(1) standard(0) 9041 application-context(1) }
```

Abstract Syntax

```
"FTAM PCI"
     { iso(1) standard(0) 8571 abstract-syntax(2) ftam-pci(1) }

"FTAM FADU"
     { iso(1) standard(0) 8571 abstract-syntax(2) ftam-fadu(2) }

"FTAM unstructured text abstract syntax"
     { iso(1) standard(0) 8571 abstract-syntax(2) unstructured-text(3)
     }

"FTAM unstructured binary abstract syntax"
     { iso(1) standard(0) 8571 abstract-syntax(2)
     unstructured-binary(4) }
```

```
"NBS abstract syntax AS1"
       { NBS-ad-hoc abstract-syntax(2) nbs-as1(1) } *

"NBS file directory entry abstract syntax"
       { NBS-ad-hoc abstract-syntax(2) nbs-as2(2) } *

"ISO 8650-ACSE1"
     { joint-iso-ccitt association-control(2) abstract-syntax(1)
       apdus(0) version1(1) }

"id-as-directoryAccessAS"
     { joint-iso-ccitt ds(5) 9 1 }

"id-as-directorySystemAS"
     { joint-iso-ccitt ds(5) 9 2 }

"VT BASIC"
     { iso(1) standard(0) 9041 abstract-syntax(2) }
```

File Model

```
"FTAM hierarchical file model"
     { iso(1) standard(0) 8571 file-model(3) hierarchical(1) }
```

Constraint Set

```
"FTAM unstructured constraint set"
     { iso(1) standard(0) 8571 constraint-set(4) unstructured(1) }

"FTAM sequential flat constraint set"
     { iso(1) standard(0) 8571 constraint-set(4) sequential-flat(2) }

"FTAM ordered flat constraint set"
     { iso(1) standard(0) 8571 constraint-set(4) ordered-flat(1) }

"NBS ordered flat constraint set"
     { NBS-ad-hoc constraint-set(4) nbs-ordered-flat(3) } *
```

Document Type

```
"ISO FTAM unstructured text"
     { iso(1) standard(0) 8571 document-type(5) unstructured text(1) }

"ISO FTAM sequential text"
     { iso(1) standard(0) 8571 document-type(5) sequential-text(2) }

"ISO FTAM unstructured binary"
```

```
     { iso(1) standard(0) 8571 document-type(5) unstructured-binary(3)
     }
"NBS-6 FTAM sequential file"
     { NBS-ad-hoc document-type(5) sequential(6) } *

"NBS-7 FTAM random access file"
     { NBS-ad-hoc document-type(5) random-file(7) } *

"NBS-8 FTAM indexed file"
     { NBS-ad-hoc document-type(5) indexed-file(8) } *

"NBS-9 FTAM file directory file"
     { NBS-ad-hoc document-type(5) file-directory(9) } *
```

Transfer Syntax

```
"Basic Encoding of a single ASN.1 type"
     { joint-iso-ccitt(2) asn1(1) basic-encoding(1) }
```

VT Profile

```
"NIST generic root for OSI VTE-profiles"
     nist-vte-profile OBJECT IDENTIFIER ::= { nist-ad-hoc 8 } *
```

Note: used only with a subsidiary leaf as a specific VTE profile identifier.

```
"NIST VTE-Profile Telnet-1988"
     { nist-vte-profile telnet-1988(0) } *

"NIST VTE-Profile Transparent-1988"
     { nist-vte-profile transparent-1988(1) } *

"NIST VTE-Profile Forms-1988"
     { nist-vte-profile forms-1988(2) } *

"NIST VTE-Profile Scroll-1988"
     { nist-vte-profile scroll-1988(3) } *
```

VT Control Object

```
"NIST root for OSI VT control objects of miscellaneous type"
     nist-vt-co-misc OBJECT IDENTIFIER ::=
         { nist-ad-hoc nist-vt-co(9) cotypemisc(0) } *

"NIST root for OSI VT control object of TCCO type"
     nist-vt-co-tcco OBJECT IDENTIFIER ::=
       { nist-ad-hoc nist-vt-co(9) cotypetcco(4) } *
```

```
"NIST VT CO for conveying Sequenced Application Signals"
     nist-vt-co-misc-sa OBJECT IDENTIFIER ::=
       { nist-vt-co-misc sa(0) } *

"NIST VT CO for conveying Unsequenced Application Signals"
     nist-vt-co-misc-ua OBJECT IDENTIFIER ::=
       { nist-vt-co-misc ua(1) } *

"NIST VT CO for conveying Sequenced Terminal Signals"
     nist-vt-co-misc-st OBJECT IDENTIFIER ::=
       { nist-vt-co-misc st(2) } *

"NIST VT CO for conveying Unsequenced Terminal Signals"
     nist-vt-co-misc-ut OBJECT IDENTIFIER ::=
       { nist-vt-co-misc ut(3) } *
```

Miscellaneous

```
"nil AP Title"
     { NBS-ad-hoc ftam-nil-ap-title(7) } *
```

7. CCITT 1984 X.400 BASED MESSAGE HANDLING SYSTEM

7.1 INTRODUCTION

This is an implementation agreement developed by the Implementor's Workshop sponsored by the U.S. National Institute of Standards and Technology to promote the useful exchange of data between devices manufactured by different vendors. This agreement is based on, and employs protocols developed in accord with, the OSI Reference Model. While this agreement introduces no new protocols, it eliminates ambiguities in interpretations.

This is an implementation agreement for a Message Handling System (MHS) based on the X.400-series of Recommendations (1984) and Version 5 of the X.400 Series Implementor's Guide from the CCITT. It is recommended that product vendors consult later versions of this guide. Figure 7.1 displays the layered structure of this agreement.

This agreement can be used over any Transport protocol class. In particular, this MHS agreement can be used over the Transport protocol class 0 used over CCITT X.25, described in Section 5.2 of this document. In addition, this MHS agreement can be used over the Transport profiles used in support of MAP (Manufacturing Automation Protocol) or TOP (Technical and Office Protocols). Note that the MAP or TOP environment must support the reduced Basic Activity Subset (BAS) as defined in X.410.

The UAs and MTAs require access to directory and routing services. A Directory Service is to be provided for each (vendor-specific) domain. Except insofar as they must be capable of providing addressing and routing described hereunder, these services and associated protocols are not described by this agreement.

User Agent Layer	CCITT X.420
Message Transfer Agent Layer	CCITT X.411
Reliable Transfer Service Layer	CCITT X.410
Presentation Layer	CCITT X.410 Sec. 4.2
Session Layer	See Section 5.9

Figure 7.1 The layered structure of this implementation agreement

7.2 SCOPE

This agreement applies to Private Management Domains (PRMDs) and Administration Management Domains (ADMDs). Four boundary interfaces are specified:

(A) PRMD to PRMD,
(B) PRMD to ADMD,
(C) ADMD to ADMD, and
(D) MTA to MTA (within a PRMD, e.g., for MTAs from different vendors.)

In case A, the PRMDs do not make use of MHS services provided by an ADMD. In cases B and C, UAs associated with an ADMD can be the source or destination for messages. Furthermore, in cases A and B, a PRMD can serve as a relay between MDs, and in cases B and C an ADMD can serve as a relay between MDs. Figure 7.2 illustrates the interfaces to which the agreement applies.

X.400 protocols other than the Message Transfer Protocol (P1) and the Interpersonal Messaging Protocol (P2) are beyond the scope of this agreement. Issues arising from the use of other protocols or relating to P1 components in support of other protocols are outside the scope of this document. This agreement describes the minimum level of services provided at each interface shown in Figure 7.2. Provision for the use of the remaining services defined in the X.400 Series of Recommendations is outside the scope of this document.

With the exception of intra domain connections, this agreement does not cover message exchange between communicating entities within a domain even if these entities communicate via P1 or P2. Bilateral agreements between domains may be implemented in addition to the requirements stated in this document. Conformance to this agreement requires the ability to exchange messages without use of bilateral agreements.

PRMD = Private Management Domain
ADMD = Administration Management Domain

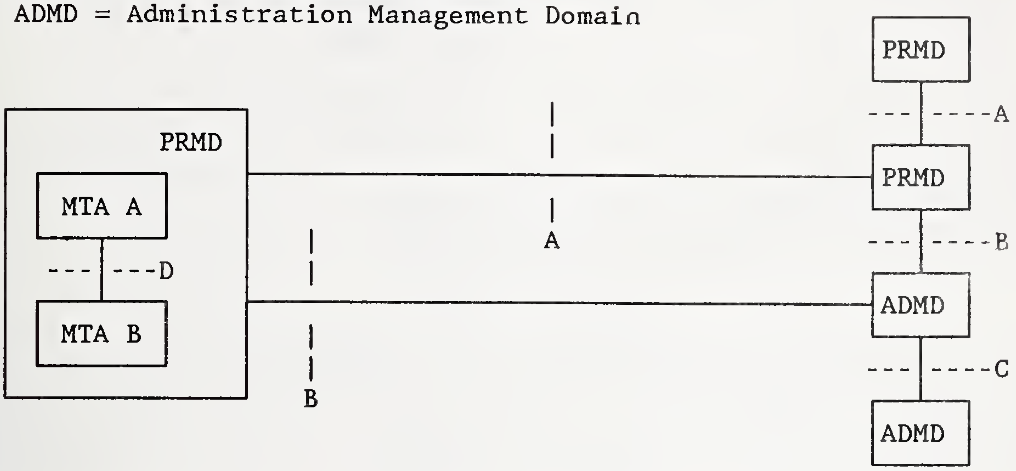


Figure 7.2 This agreement applies to the interface between: (A) PRMD and PRMD; (B) PRMD and ADMD; (C) ADMD and ADMD; and (D) MTA and MTA

7.3 STATUS

This version of the X.400 based Message Handling System implementation agreements was completed on December 16, 1988. No further enhancements will be made to this version. See the next section--Errata.

7.4 ERRATA

7.5 PRMD to PRMD

7.5.1 Introduction

This section is limited in scope to issues arising from the direct connection (interface A in Figure 7.2) of two PRMDs. "Direct" means that no ADMD or relaying PRMD provides MHS services to facilitate message interchange. "Direct" does not exclude those instances for which Administrations happen to be ADMDs but are not providing X.400 services, that is, they are used only to provide lower layer services such as X.25. Figure 7.3 schematically represents the scope of this section.

These issues relate to the use of the UAL (User Agent Layer) and MTL (Message Transfer Layer) services, protocol elements, recommended practices and constraints. In particular, this section addresses the P1 and P2 protocols and their related services in a direct connection environment. This section describes the minimum level of services provided by a PRMD. Provision for the use of the remaining services defined in the X.400 Series of Recommendations is beyond the scope of this section.

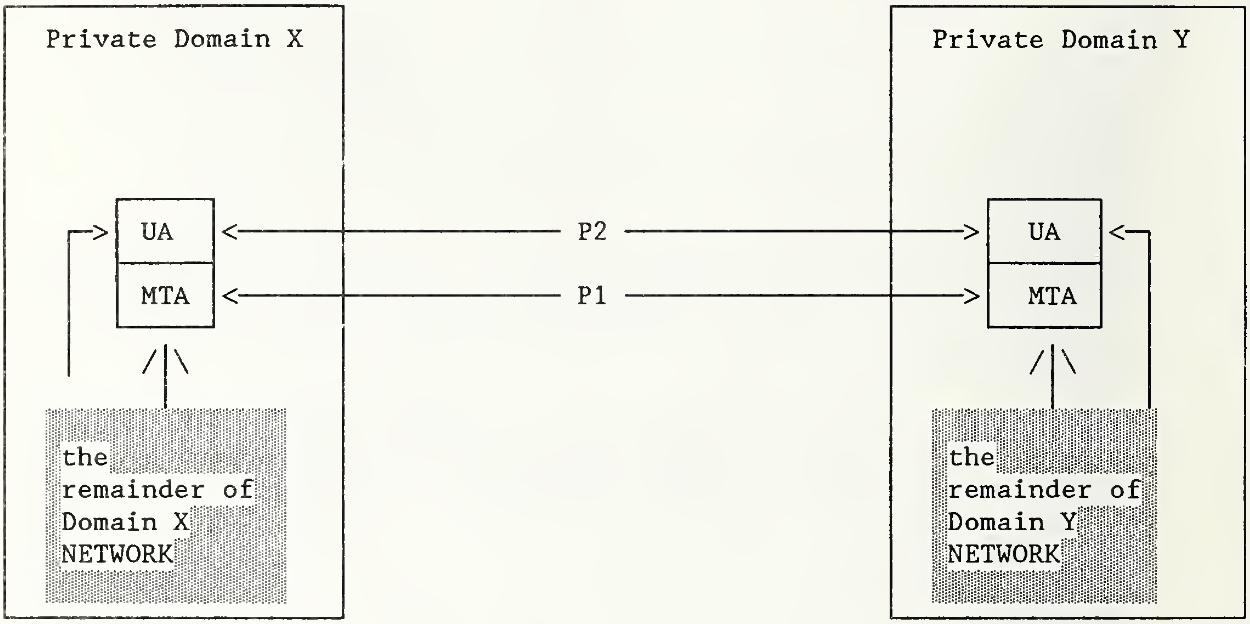

Private Domain X
Private Domain Y
UA
MTA
P2
P1
UA
MTA
the remainder of Domain X NETWORK
the remainder of Domain Y NETWORK

Figure 7.3 Interconnection of private domains

7.5.2 Service Elements and Optional User Facilities

This section identifies those service elements and optional user facilities that must be provided in support of P1 and P2.

7.5.2.1 Classification of Support for Services

The classification of UA and MT-Service elements is used to define characteristics of equipment. Equipment can claim SUPPORT or NON-SUPPORT of a Service; in the case of UA-service elements, a separate classification is given for Origination and Reception.

The service provider is defined as the entity providing the service, in this case, the MTL or the UAL. The service user is either the MHS user or the UAL. The classification of provider and user relates to the sublayer for which the service element is defined.

7.5.2.1.1 Support (S)

a) Support means:

- The service provider makes the service element available to the service user, and

  - The service user gives adequate support tothe MHS to invoke the service element or makes information associated with the service element available.

b) Support for Origination means that:

  - The service provider makes the service element available to the service user for invocation, and

  - The service user gives adequate support to the end user of the MHS to invoke the service element.

c) Support for Reception means that:

  - The service provider makes information associated with the service element available to the service user.

Note: A UA- or MT-service element can carry information from originator to recipient only if:

  - the service element is available to the originator,

  - the service element is available to the recipient, and

  - all intermediate steps carry the information.

7.5.2.1.2 Non Support (N)

This means that the service provider is not required to make the service element available to the service user. However, the service provider should not regard the occurrence of the corresponding protocol elements as an error and should be able to relay such elements. Implementations making a profile available should indicate deviations (additions or deletions) with respect to the requirement in the profile.

7.5.2.1.3 Not Used (N/U)

This means that although the Recommendation allows this service element, this profile does not use it.

7.5.2.1.4 Not Applicable (N/A)

This means that this service element does not apply in this particular case (for originator or recipient).

7.5.2.2 Summary of Supported Services

a) Within a PRMD, a User Agent must support all P2 BASIC IPM Services (X.400) and all P2 ESSENTIAL IPM Optional user facilities (X.401) subject to the qualifiers listed in Appendix 7A.

b) Within a PRMD, a MTA must support all BASIC MT Services (X.400) and all ESSENTIAL MT optional user facilities (X.401) subject to the qualifiers listed in Appendix 7A.

c) No support is required of the additional optional user facilities of X.401.

7.5.2.3 MT Service Elements and Optional User Facilities

Tables 10.1 through 10.3 show the message transfer (MT) service elements and optional user facilities.

Table 7.1 Basic MT service elements

Service Elements	Support (S) or Non-support (N)
Access Management	N/U[1]
Content Type Indication	S
Converted Indication	S
Delivery Time Stamp Indication	S
Message Identification	S
Non-delivery Notification	S
Original Encoded Information Types Indication	S
Registered Encoded Information Types	N/U[1]
Submission Time Stamp Indication	S

[1] Not applicable to co-resident UA and MTA.

Table 7.2 MT optional user facilities provided to the UA-selectable on a per-message basis

MT Optional User Facilities	Categorization	Support (S) or Non-support (N)
Alternate Recipient Allowed	E	S
Conversion Prohibition	E	S
Deferred Delivery	E	N[2]
Deferred Delivery Cancellation	E	N[2]
Delivery Notification	E	S
Disclosure of Other Recipients	E	N[3]
Explicit Conversion	A	N
Grade of Delivery Selection	E	S
Multi-destination Delivery	E	S
Prevention of Non-delivery Notification	A	N[4]
Probe	E	N
Return of Contents	A	N

Table 7.3 MT optional user facilities provided to the UA agreed for any contractual period of time

MT Optional User Facilities	Categorization	Support (S) or Non-Support (N)
Alternate Recipient Assignment	A	N
Hold for Delivery	A	N/U
Implicit Conversion	A	N

E: Essential optional user facility.
A: Additional optional user facility.
[2] A local facility subject to qualifiers in Appendix 7A.
[3] Support not required for an originating MT user; support must be provided for recipient MT users.
[4] Subject to qualifiers in Appendix 7A.

7.5.2.4 IPM Service Elements and Optional User Facilities

Tables 7.4 through 10.5 show the IPM service elements and optional user facilities.

Table 7.4 Basic IPM service elements

Service Elements	Origination by UAs	Reception by UAs
Access Management	N/U[5]	N/U[5]
Content Type Indication	S	S
Converted Indication	N/A	S
Delivery Time Stamp Indication	N/A	S
Message Identification	S	S
Non-delivery Notification	S	N/A
Original Encoded Information Types Indication	S	S
Registered Encoded Information Types	N/A	N/A[5]
Submission Time Stamp Indication	S	S
IP-message Identification	S	S
Typed Body	S	S

[5] Does not apply to co-resident UA and MTA.

Table 7.5 IPM optional facilities agreed for a contractual period of time

Service Elements	Categorization	Support (S) or Non-Support (N)
Alternate Recipient Assignment	A	N
Hold for Delivery	A	N
Implicit Conversion	A	N

Table 7.6 IPM optional user facilities selectable on a per-message basis

IPM Optional User Facilities	Origination by UAs	Reception by UAs
Alternate Recipient Allowed	A (N)	A (N)
Authorizing Users Indication	A (N)	E (S)
Auto-forwarded Indication	A (N)	E (S)
Blind Copy Recipient Indication	A (N)	E (S)
Body Part Encryption Indication	A (N)	E (S)
Conversion Prohibition	E (S)	E (S)
Cross-referencing Indication	A (N)	E (S)
Deferred Delivery	E (N)[6]	N/A
Deferred Delivery Cancellation	A (N/U)[6]	N/A
Delivery Notification	E (S)	N/A
Disclosure of Other Recipients	A (N)	E (S)
Expiry Date Indication	A (N)	E (S)
Explicit Conversion	A (N)	N/A
Forwarded IP-message Indication	A (N)	E (S)
Grade of Delivery Selection	E (S)	E (S)
Importance Indication	A (N)	E (S)
Multi-destination Delivery	E (S)	N/A
Multi-part Body	A (N)	E (S)
Non-receipt Notification	A (N)	A (N)
Obsoleting Indication	A (N)	E (S)
Originator Indication	E (S)	E (S)
Prevention of Non-delivery Notification	A (N)	N/A
Primary and Copy Recipients Indication	E (S)	E (S)
Probe	A (N)	N/A
Receipt Notification	A (N)	A (N)
Reply Request Indication	A (N)	E (S)
Replying IP-message Indication	E (S)	E (S)
Return of Contents	A (N)	N/A
Sensitivity Indication	A (N)	E (S)
Subject Indication	E (S)	E (S)

[6] A local facility subject to qualifiers in Appendix 7A.

7.5.3 X.400 Protocol Definitions

This section reflects the agreements of the NIST/OSI Workshop regarding P1 and P2 protocol elements.

7.5.3.1 Protocol Classification

The protocol classifications are defined below in Table 7.7:

1) UNSUPPORTED = X
These elements may be generated, but no specific processing should be expected in a relaying or delivering domain. A relaying domain must at least relay the semantics of the element. The absence of these elements should not be assumed, in a relaying or delivering domain, to convey any significance.

2) SUPPORTED = H
These elements may be generated. However, implementations are not required to be able to generate these elements. Appropriate actions shall be taken in a relaying or delivering domain.

3) GENERATABLE = G
Implementations must be able to generate and handle these protocol elements, although they are not necessarily present in all messages generated by implementations of this profile. Appropriate actions shall be taken in a relaying or delivering domain.

4) REQUIRED = R
Implementations of this profile must always generate this protocol element. However, its absence cannot be regarded as a protocol violation as other MHS implementations may not require this protocol element. Appropriate actions shall be taken in a relaying or delivering domain.

5) MANDATORY = M
This must occur in each message as per X.411 or X.420 as appropriate; absence is a protocol violation. Appropriate actions shall be taken in a relaying or delivering domain.

Table 7.7 Protocol Classifications

7.5.3.2 General Statements on Pragmatic Constraints

a) Where a protocol element is defined as a choice of Numeric String and Printable String (i.e., Administration Domain Name and Private Domain Identifier), then a numeric value encoded as a printable string is equivalent to the same value encoded as a numeric string. This does not apply to the Country Name protocol element.

b) The maximum number of recipients in a single MPDU is 32K - 1 (that is, 32767). However, no individual limits on the number of occurrences (recipients) are placed on the following protocol elements: Authorizing Users, Primary Recipients, Copy Recipients, Blind Copy Recipients, Obsoletes and Cross References. Additionally, there is no limit on the number of Reply to Users. This is a local matter for the originating system.

c) Use of strings. A Printable String is defined in terms of the number of characters, which is the same number of octets. For T.61 strings the number of octets is twice the number of characters specified.

d) The ability to generate maximum size elements is not required, with the exception of the component fields in the Standard Attribute List, in which case it is required.

7.5.3.3 MPDU Size

The following agreements govern the size of MPDUs:

o All MTAEs must support at least one MPDU of at least two megabyte, and

o The size of the largest MPDU supported by a UAE is a local matter.

7.5.3.4 P1 Protocol Elements

7.5.3.4.1 P1 Envelope Protocol Elements

Table 7.8 contains Protocol Elements and their classes.

Table 7.8 P1 protocol elements

Element	Class	Restrictions and Comments
MPDU		
UserMPDU	G	
DeliveryReportMPDU	G	
ProbeMPDU	H	
UserMDPU		
UMPDUEnvelope	M	
UMPDUContent	M	
UMPDUEnvelope		
MPDUIdentifier	M	
originator ORname	M	
originalEncodedInformationTypes	G	If this field is absent, then the Encoded Information Type is "unspecified".
ContentType	M	
UAContentID	H	Maximum length = 16 characters.
Priority	G	
PerMessageFlag	G	Maximum length = 2 octets.
deferredDelivery	X	
PerDomainBilateralInfo	X	No limit on number of occurrences.
RecipientInfo	M	Maximum number = 32K - 1 occurrences. More severe limitations are by bilateral agreement.
TraceInformation	M	
UMPDUContent	M	
MPDUIdentifier		
GlobalDomainIdentifier	M	
IA5String	M	Maximum length = 32 characters, graphical subset only. Refer to T.50 for clarification of graphical subset.
PerMessageFlag		
discloseRecipients	H	
conversionProhibited	G	
alternateRecipientAllowed	H	
contentReturnRequest	X	

(Continued on next page.)

Table 7.8 P1 protocol elements, Continued

Element	Class	Restrictions and Comments
PerDomainBilateralInfo		
CountryName	M	Maximum length = 3 characters.
AdministrationDomainName	M	Maximum length = 16 characters.
BilateralInfo	M	Maximum depth = 8; maximum length = 1024 octets (including encoding).
RecipientInfo		
recipient	M	
ExtensionIdentifier	M	Maximum value = 32K - 1 (32767).
perRecipientFlag	M	Maximum length = 2 octets.
ExplicitConversion	X	
perRecipientFlag		
ResponsibilityFlag	M	
ReportRequest	M	
UserReportRequest	M	
TraceInformation		Reference should be made to Version 5 of the X.400 Implementor's Guide for information related to Trace sequencing.
GlobalDomainIdentifier	M	
DomainSuppliedInfo	M	
DomainSuppliedInfo		
arrival	M	
deferred	X	
action	M	
0=relayed (value)	G	
1=rerouted (value)	H	Rerouting is not required.
converted	H	
previous	H	
ORName		See Section 7.5.3.5
EncodedInformationTypes		
bit string	M	Delivery can only occur if match is made with Registered Encoded Information Types. Individual vendors may impose limits. Maximum length = 4 octets.
G3NonBasicParameters	X	
TeletexNonBasicParameters	X	
PresentationCapabilities	X	

(Continued on next page.)

Table 7.8 P1 protocol elements, Continued

Element	Class	Restrictions and Comments
DeliveryReportMPDU		
DeliveryReportEnvelope	M	
DeliveryReportContent	M	
DeliveryReportEnvelope		
report	M	
originator	M	
TraceInformation	M	
DeliveryReportContent		
original	M	
intermediate	G	See comment at end of table.
UAContentID	G	
ReportedRecipientInfo	M	Maximum number = 32K - 1 occurrences.
returned	H	Can only be issued if specifically requested in the originating message.
billingInformation	X	Maximum depth = 8; maximum length = 1024 octets (including encoding).
ReportedRecipientInfo		
recipient	M	
ExtensionsIdentifier	M	
PerRecipientFlag	M	
LastTraceInformation	M	
intendedRecipient	H	
SupplementaryInformation	X	Maximum length = 64 characters. Value is pending verification by the CCITT SG VIII or XI.
LastTraceInformation		
arrival	M	
converted	G	
Report	M	

(Continued on next page.

Table 7.8 P1 protocol elements, continued

Element	Class	Restrictions and Comments
Report		
DeliveredInfo	G	Generated if delivery is reported.
NondeliveredInfo	G	Generated if failure to deliver is reported.
DeliveredInfo		
delivery	M	
typeofUA	R	This element must be generated with a PRIVATE value by PRMDs.
NonDeliveredInfo		
ReasonCode	M	
DiagnosticCode	H	Whenever possible, use a meaningful diagnostic code.
ProbeEnvelope		
probe	M	
originator	M	
ContentType	M	
UAContentID	H	Maximum length = 16 characters.
original	G	If this field is absent, then the Encoded Information Type is "unspecified".
TraceInformation	M	
PerMessageFlag	G	
contentLength	H	
PerDomainBilateralInfo	X	
RecipientInfo	M	Maximum number = 32K - 1 occurrences.
GlobalDomainIdentifier		
CountryName	M	Maximum length = 3 characters.
AdministrationDomainName (4)	M	Maximum length = 16 characters or digits.
PrivateDomainIdentifier	R	Maximum length = 16 characters or digits. This element must be generated by PRMDs.
End of Definitions		

Notes on Table 7.8

Comment on intermediate TraceInformation in the DeliveryReportContent in Table 7.8: Audit and confirmed reports should not be requested by other than the originating domain for two reasons. First, the return path of the report may be different from the path taken by the original message, and it may exclude the domain that added the request for audit and confirmed to the message. Second, if the return path is different from the path of the original message, the originating domain would receive intermediate trace information that it did not request.

7.5.3.5 ORName Protocol Elements

Only form 1 variant 1 O/R names are supported.

Table 7.9 contains ORName protocol elements.

These agreements interpret 1984 X.400 Section 3.4 to mean that the attributes in the ORName in the MPDU must identify exactly one UA, and that all the values for the attributes specified in the ORName must be identical to the values of the corresponding attributes associated with the recipient UA. Underspecified names that are unique are deliverable.

Overspecified ORNames, ORNames with mismatching fields, and ambiguous names are to be non-delivered or sent to the alternate recipient as appropriate.

Table 7.9 ORName protocol elements

Element	Class	Restrictions and Comments
ORName		
StandardAttributeList	M	
DomainDefinedAttributeList	X	
StandardAttributeList (1)		
CountryName	R	As defined in X.411, Maximum length = 3 characters.
AdministrationDomainName (4)	R	Maximum length = 16 characters or digits.
X121Address	X	Maximum length = 15 digits.
TerminalID	X	Maximum length = 24 characters.
PrivateDomainName (2)	G	Maximum length = 16 characters.
OrganizationName (2)	G	Maximum length = 64 characters.
UniqueUAIdentifier	X	Maximum length = 32 digits.
PersonalName	G	Maximum length of values of sub-elements = 64 characters. Note: The possibility that this value may be reduced to 40 characters is for further study by the CCITT.
OrganizationalUnit (3)	G	Maximum length = 32 characters per occurrence. A maximum of four occurrences are allowed.
DomainDefinedAttributeList (5)		Maximum = 4 occurrences.
type	M	Maximum length = 8 characters.
value	M	Maximum length = 128 characters.
PersonalName		
surName	M	Maximum length = 40 characters.
givenName	G	Maximum length = 16 characters.
initials	G	Maximum length = 5 characters; excluding surname initial and punctuation and spaces.
generationQualifier	G	Maximum length = 3 characters.

(Continued on next page.)

Table 7.9 ORName Protocol Elements, Continued

Notes:

1. The following apply for comparison of the Standard Attributes of an O/R Name:

   a. Lower case is interpreted as upper case (for IA5).

   b. Multiple spaces may be interpreted as a single space. Originating domains shall only transmit single significant spaces. If multiple spaces are transmitted, non-delivery may occur.

2. At least one of these must be supplied.

3. These should be sent in descending sequence, from the most significant <Organizational Unit> (highest in organization hierarchy) to the least significant. Only those specified should be sent. (That is, an unspecified <Organizational Unit> should not be sent along as a field of [null] content, nor zero length, etc.)

4. This attribute shall contain one space in all ORNames of messages originated in a PRMD that is not connected to an ADMD, and in ORNames of recipients reachable only through a PRMD; otherwise, this attribute shall contain an appropriate ADMD name.

5. Some existing systems which will be accessed via an X.400 service (whether directly connected using X.400 protocols or otherwise) may require the provision of addressing attributes which are not adequately supported by Standard Attributes as defined in these Agreements. In such cases, Domain Defined Attributes are an acceptable means of carrying additional addressing information. Failure to support the specification and relaying of DDAs may prevent successful interworking with such existing systems until such time as all systems are capable of relaying and delivery using only the Standard Attribute list. Specific recommendations on the use of DDAs for particular applications are in the Recommended Practices Section 7.12, Appendix B.

7.5.3.6 P2 Protocol Profile (Based on [X.420])

Tables 7.10 and 7.11 classify the support for the P2 protocol elements required by this profile. The tables give restrictions and comments in addition to X.420.

Restriction on length is one of the types of restrictions. The reaction of implementations to a violation of this restriction is not defined by this Profile.

7.5.3.6.1 P2 Protocol - Heading

Table 7.10 below specifies the support for protocol elements in P2 Headings.

Table 7.10 P2 heading protocol elements

Element	Class	Restrictions and Comments
UAPDU		
IM-UAPDU	G	
SR-UAPDU	X	
IM-UAPDU		
Heading	M	
Body	M	
Heading		
IPMessageId	M	
originator	R	
authorizingUsers	H	
primaryRecipients	G	At least one of primaryRecipients, copyRecipients, or blindCopyRecipients must be present.
copyRecipients	G	
blindCopyRecipients	H	
inReplyTo	G	
obsoletes	H	
crossReferences	H	
subject	G	Maximum length = 128 T.61 characters (256 octets);the ability to generate the maximum size subject is not required.
expiryDate	H	
replyBy	H	
replyToUsers	H	
importance	H	Appropriate action is for further study.
sensitivity	H	Appropriate action is for further study.
autoforwarded	H	

(Continued on next page.)

Table 7.10 P2 heading protocol elements, continued

Element	Class	Restrictions and Comments
IPmessageId		
ORName	H	
PrintableString	M	Maximum length = 64 characters.
ORDescriptor		
ORName	H	Specify the ORName whenever it is possible. See Appendix 7B.
freeformName	H	Maximum length = 64 characters, graphical subset only (128 octets.)
telephoneNumber	H	Maximum length = 32 characters. This allows for punctuation. It does not take into account possible future use by ISDN.
Recipient		
ORDescriptor	M	
reportRequest	X	
replyRequest	H	
Body		No limit on number of BodyParts.
BodyPart	G	No limit on length of any BodyPart or the depth of ForwardedIPMessage BodyParts nested. Classification is subject to pending CCITT resolution
SR-UAPDU		
nonReceipt	H	
receipt	H	
reported	M	
actualRecipient	R	
intendedRecipient	H	
converted	X	
NonReceiptInformation		
reason	M	
nonReceiptQualifier	H	
comments	H	Maximum length = 256 characters.
returned	H	May only be issued if specifically requested by originator.

(Continued on next page.)

Table 7.10 P2 heading protocol elements, continued

Element	Class	Restrictions and Comments
ReceiptInformation		
receipt	M	
typeOfReceipt	H	
SupplementaryInformation	X	Maximum length = 64 characters. Note: This value is pending verification by the CCITT SG VIII or IX.
	End of Definitions	

7.5.3.6.2 P2 Protocol - BodyParts

a) All BodyParts with identifiers in the range 0 up to and including 16K -1 are legal and should be relayed. BodyPart identifiers corresponding to X.121 Country Codes should be interpreted as described in Note 2 of Figure 7.4.

- Implementations are required to generate and image IA5Text.

- Implementations should specify the other BodyPart types supported.

- If an implementation supports a particular BodyPart type for reception, it should also be able to support that BodyPart type for reception if it is part of a ForwardedIPMessage.

- For the BodyPart types currently considered, support for the protocol elements is as indicated in Table 7.11.

b) Privately Defined BodyParts

This section describes an interim means for identifying privately defined BodyParts. This section shall be replaced in a future version taking into account CCITT recommendations with equivalent functionality.

```
BodyPart  :: =  CHOICE {

   [0]IMPLICIT IA5Text,
              [1]IMPLICIT TLX,
                .

                    .

                    .
             [234]IMPLICIT UKBodyParts,
                    .

                    .

                    .
             [310]IMPLICIT USABodyParts,
                    .

                    .

                    .                    }
             Where UKBodyParts and USABodyParts are defined as:

                SEQUENCE {BodyPartNumber, ANY}

                BodyPartNumber ::= INTEGER
```

Note 1: In the EncodedInformationTypes of the P1 Envelope, the undefined bit must be set when a message contains a privately defined BodyPart. Each UA that expects such BodyParts should include undefined in the set of deliverable EncodedInformationTypes it registers with the MTA.

Note 2: All BodyPartNumbers assigned must be interpreted relative to the BodyPart in which they are used, which is that tagged with the value [310] for those defined within the United States. The NIST assigns unique message BodyPartNumbers for privately defined formats within the United States.

Note 3: Refer to Section 7.12.6 for recommendations regarding the implementaion of USABodyParts.

Figure 7.4 X.409 Definition of Privately Defined BodyParts

7.5.3.6.3 P2 BodyPart Protocol Elements

Table 7.11 P2 BodyParts

Elements	Class	Restrictions and Comments
BodyPart		
IA5Text	G	
TLX	X	
Voice	X	
G3Fax	X	
TIFO	X	
TTX	X	
Videotex	X	
NationallyDefined	X	
Encrypted	X	
ForwardedIPMessage	H	
SFD	X	
TIF1	X	
unidentified	X	
IA5Text		
repertoire	H	
IA5String	M	For rendition of IA5Text see Appendix 7C.
TLX		For further study by CCITT.
Voice		
Set		For further study by CCITT.
BitString	M	
G3Fax		
numberOfPages	X	
P1.G3NonBasicParameters	X	
SEQUENCE (OF BIT STRING)	M	
BIT STRING	H	See Note.
P1.G3NonBasicParameters		Support for individual elements is for further study.
TIFO		
T.73Document	M	
T.73ProtocolElement	H	See Note.

(Continued on next page.)

Table 7.11 P2 BodyParts, continued

Elements	Class	Restrictions and Comments
TTX		
numberOfPages	X	
telexCompatible	X	
P1.TeletexNonBasicParams	X	
SEQUENCE	M	
T61String	H	See Note.
P1.TeletexNonBasicParams		
graphicCharacterSets	X	
controlCharacterSets	X	
pageFormats	X	
miscTerminalCapabilities	X	
privateUse	X	
Videotex		
SET		For further study by CCITT.
VideotexString	M	
NationallyDefined		
ANY	M	
Encrypted		
SET		For further study by CCITT.
BIT STRING	M	
ForwardedIPMessage		
delivery	H	
DeliveryInformation	H	
IM-UAPDU	M	
DeliveryInformation		
P1.ContentType	M	
originator	M	
original	M	
P1.Priority	G	
DeliveryFlags	M	
otherRecipients	H	
thisRecipient	M	
intendedRecipient	H	
converted	X	
submission	M	

(Continued on next page.)

Table 7.11 P2 BodyParts, continued

Elements	Class	Restrictions and Comments
SFD		
SFD.Document	M	
TIF1		
T73 Document	M	
T73.ProtocolElement	H	See note.

Note: This element is not an addition to the definition of the BodyPart. It is described here to show that the SEQUENCE may contain zero elements. A Problem Report has been submitted to the CCITT to clarify whether this is permissible. The NIST/OSI Workshop will adopt the CCITT decision.

7.5.4 Reliable Transfer Server (RTS)

7.5.4.1 Implementation Strategy

Based on X.410 Clause 3 and X.411 Clause 3.5.

7.5.4.2 RTS option selection

a) The maximum number of simultaneous associations is not limited in this profile; if the capacity of a system is exceeded, it should not initiate or accept additional associations.

b) Associations are established by the MTA which has messages to transfer.

c) Associations are released when they are not needed. Associations may also be ended prematurely due to internal problems of the RTS.

d) For both monologue and two way alternate associations, the initiator keeps the initial turn.

e) When establishing an RTS association, the following rules apply to the use of parameters in addition to those in X.410 Clause 3.2.1:

Dialogue mode: Monologue must be supported for this profile; two-way alternate is used only if both partners agree.

Initial turn: Kept by the initiator of the association.

f) The 'priority-mechanism' and the 'transfer-time limit' are regarded as local matters.

7.5.4.3 RTS Protocol Options and Clarifications

Realization of the RTS protocol is subject to the following rules in addition to those specified in X.410 Clause 4:

a) One RTS association corresponds to one or more consecutive session connections (not concurrent ones). The first is opened with ConnectionData of type OPEN, and subsequent ones are opened with type RECOVER.

b) Recovery of a Session connection is only by RTS initiator.

c) Checkpoint size:

  - Checkpointing and No Checkpointing should be supported. Whenever possible, checkpointing should be used.

  - The minimum checkpointSize is 1 (that is, 1024 octets).

d) Window size:

  - Minimal value of 1 (if checkpointing is supported).

  - WindowSize = 1 means: After an S-SYNCH-MINOR request is sent, wait until the confirmation is received before issuing an S-DATA, S-SYNCH-MINOR, or S-ACTIVITY-END request.

e) APDUs should not be blocked into one activity.

f) Only one SSDU shall be transferred:

  - Between two adjacent minor synch points.

  - Between minor synch points and adjacent S-ACTIVITY-START and S-ACTIVITY-END requests.

  - Between S-ACTIVITY-START and S-ACTIVITY-END without checkpoints.

g) A monologue association is defined as follows:

- The RTS user responsible for establishing the association is called the initiator.

- The initiator keeps the initial turn.

- APDUs are transferred in the direction of the initiator to the recipient only.

- There shall be no token passing.

- Only the initiator can effect an orderly release of the association.

h) A two-way alternate association is as described in X.410.

i) In the UserData parameter of the S-U-ABORT, the ReflectedParameter will not be used in the AbortInformation element.

j) When the S-ACTIVITY-RESUME is used to resume an activity in the same session connection as the one in which it started, this must happen immediately after the activity has been interrupted (i.e., no intervening activity can occur). Otherwise, X.410 Clause 4.3 paragraph 1 may be violated.

k) When S-ACTIVITY-RESUME is used to resume an activity started in another session connection, the following conditions must be met:

   - The current session connection is of type "recover".

   - The value of OldSessionConnectionIdentifier in S-ACTIVITY-RESUME must match the value of the SessionConnectionIdentifier parameter used in the S-CONNECT of the prior session connection. This value is also identical to the SessionConnectionIdentifier in the ConnectionData (in PConnect, in SS-UserData) for the current session connection.

   - This must occur as the first activity of the next session connection for the same RTS-association. It must be the first, otherwise X.410 Clause 4.5.1 point 1 is violated.

Note: It is in the same RTS-ASSOCIATION because the use of S-ACTIVITY-RESUME only makes sense within the scope of one RTS association.

l) If the transfer of an APDU is interrupted before the confirmation of the first checkpoint, the value of the SynchronizationPointSerialNumber in S-ACTIVITY-RESUME should be zero, and the S-ACTIVITY-RESUME must be immediately followed by an S-ACTIVITY-DISCARD.

m) In S-TOKEN-PLEASE, the UserData parameter shall contain an integer conforming to X.409 which conveys the priority.

n) The receiving RTS can use the value of the Reason parameter in the S-U-EXCEPTION-REPORT to suggest to the sending RTS that it should either interrupt or discard the current activity. S-U-Exception Reports are handled as stated in Version 5 of the Implementors Guide pages 12-13, paragraph E-33.

o) In the case of S-P-ABORT, the current activity (if any) is regarded as interrupted, rather than discarded.

p) Table 7.12 illustrates the legal negotiation possibilities allowed by X.410 Clause 4.2.1 regarding checkpoint size and window size.

q) These agreements include the provisions of Version 6 of the Implementors Guide identical in all respects to Version 5, except that the following points have been added to Section 3.5:

o for Section 4.4.1 of X.410;
"If the receiving RTS receives an S-ACTIVITY-DISCARD indication primitive and has already issued a TRANSFER indication primitive, it aborts the connection (S-U-ABORT request) with the "transfer completed" version code."

o for Section 4.6.2 of X.410
"The "transfer completed (7)" abort reason is used to indicate to the sending RTS that the receiving RTS could not discard the activity because it has already completed the transfer (issued a TRANSFER indication primitive)." Transfer completed (7) is also added to the table of abort reasons in this section.

Table 7.12 Checkpoint window size of IP

		acceptor answer		

		CS = 0
		(or unspecified)
		WS unspecified
			CS = m
			WS = j
			(or unspecified)
				CS = n
				WS = j
				(or unspecified)

initiator proposal	CS = 0
	(or unspecified)
	WS = i
	(or unspecified)
		legal	legal	legal

	CS = k
	WS = i
	(or unspecified)
		legal	legal	not allowed

Legend:

- o CS means CheckpointSize
- o WS means WindowSize
- o i, j, k, m, and n are integer values with the following relations:

$0 \le m \le k < n$ (values assigned to CS)
$0 < j \le i$ (values assigned to WS)

- o For unspecified parameters, the default applies. In this case,the numeric relations apply, that is, the default values substitutefor the unspecified integer.

7.5.4.4 RTS Protocol Limitations

The RTS Protocol Limitations for this profile are listed in Table 7.13.

Table 7.13 RTS protocol elements

Element	Class	Restriction
PConnect	M	
DataTransferSyntax	M	Value = 0.
pUserData	M	
checkpointSize	H	
windowSize	H	
dialogueMode	H	
ConnectionData	M	
applicationProtocol	G	Value = 1.
	H	Value = 8883.
ConnectionData		
open	G	
recover	G	
open		
RTS user data	G	
recover		
SessionConnectionIdentifier	G	
RTS user data		
mTAName	G	Maximum length 32 characters graphic subset of IA5 only.
password	G	Maximum length 64 octets graphic subset of IA5 only.
< null RTS User Data >	G	Generated if other validation methods are used.
SessionConnectionIdentifier		
CallingSSUserReference	M	Maximum length 64 octets including encoding = 62 octets of T.61.
CommonReference	M	
AdditionalReferenceInformation	H	Maximum length 4 octets including encoding = 2 octets of T.61.
PAccept	G	
DataTransferSyntax	M	Value = 0.
pUserData	M	
checkpointSize	H	
windowSize	H	
ConnectionData	M	

(Continued on next page.)

Table 7.13 RTS protocol elements, continued

Element	Class	Restriction
PRefuse	G	
RefuseReason	M	
SS User Data (in S-TOKEN-PLEASE)	G	See Note
AbortInformation (in S-U-ABORT)	G	
AbortReason	H	
reflectedParameter	X	Restricted to 8 bits.
End of Definitions		

Note: Generated if supplied by the RTS-user. The RTS use may specify a priority in the TURN-PLEASE primitive, and if so, it is carried as the SS-User-Data in S-TOKEN-PLEASE.

7.5.5 Use of Session Services

The session requirements and use of session are covered in Section 5 of this document.

7.5.6 Data Transfer Syntax

This section defines Presentation Transfer Syntax and notation rules applicable to these agreements. Implementations must conform EXACTLY as specified in X.409 with no further restrictions. Appendix 7C defines rendition of IA5 Text and T61 characters.

7.6 PRMD to ADMD and ADMD to ADMD

7.6.1 Introduction

This section defines the implementation agreements that apply to the interface between two management domains when at least one is an ADMD. A message arriving at an ADMD has either no recipient within that domain or one or more recipients within that domain. In the former case, the ADMD serves as a relay between two or more domains and the actions required of that ADMD are independent of the nature (PRMD or ADMD) of the domains. In the latter case, the ADMD is responsible for delivering messages to the proper recipient(s) within its jurisdiction, and may also be responsible for relaying the message.

Given the two roles for an ADMD, this section describes two distinct sets of functional requirements for an ADMD. The first is the relaying requirement that is needed to provide PRMD and other ADMD interworking. The second is analogous to the PRMD's support to its customers through the integrated UAs. These are distinct functional differences. The services provided to the UAs of an ADMD are independent of the requirement that an ADMD provide a function for interworking with any type of Management Domain (MD). Figure 7.5 illustrates the two roles played by an ADMD.

This section is presented in the form of deviations from the agreements applicable to PRMD-to-PRMD (Section 7.5). Unless explicitly noted in the remainder of this section, all of the specifications for PRMD to PRMD apply to PRMD to ADMD and ADMD to ADMD.

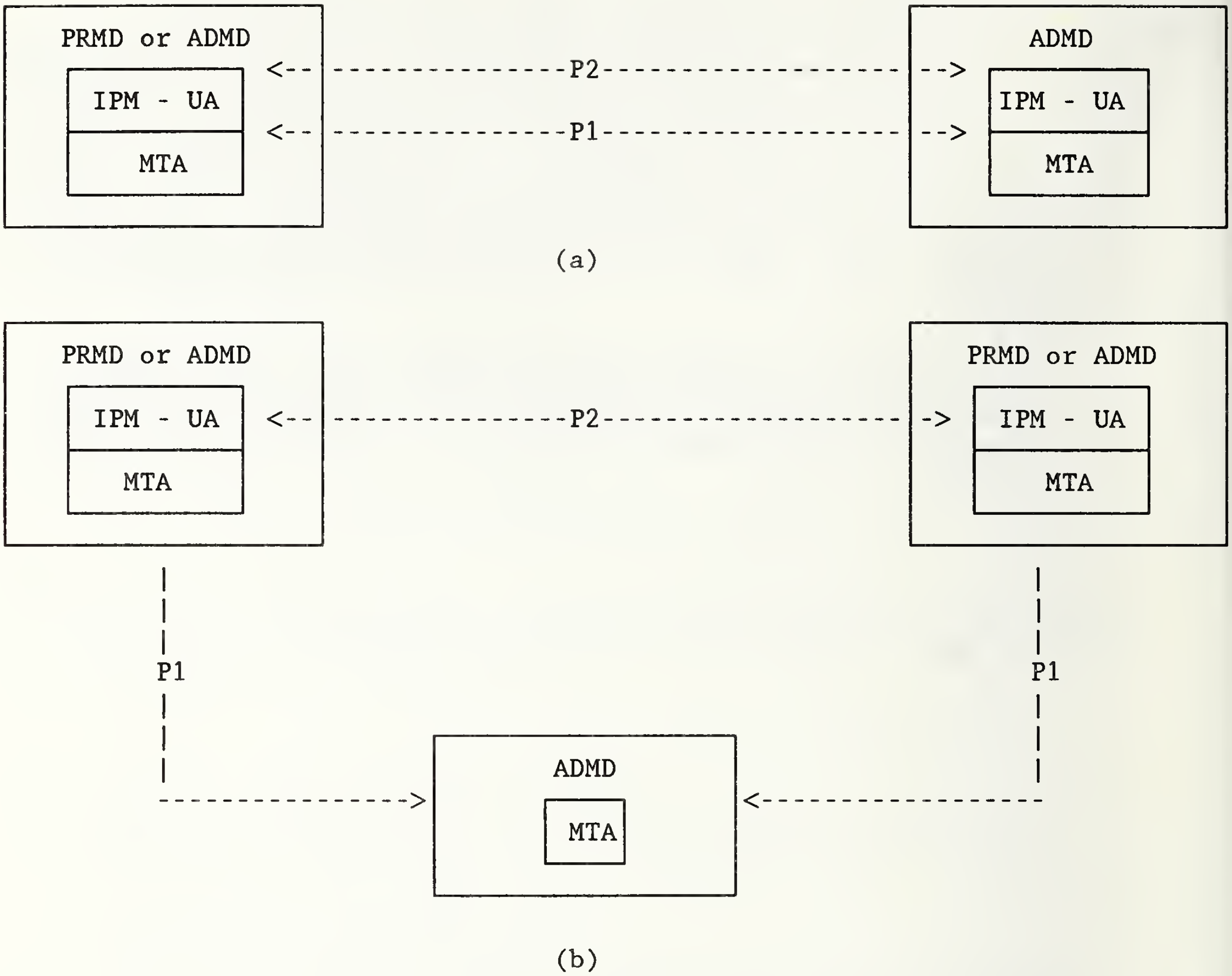


Figure 7.5 An ADMD may (b) or may not (a) serve as a relay

7.6.2 Additional ADMD Functionality

The following defines the additional ADMD specific functionality required over and above that specified in the PRMD section.

7.6.2.1 Relay Responsibilities of an ADMD

ADMDs will relay all content types (not just P2) unchanged in the absence of a request for conversion.

7.6.2.2 P1 Protocol Classification Changes

Table 7.14 describes the changes to the PRMD P1 Protocol classifications required for a delivering Administration domain (with respect to the original message; this means the domain which originates the delivery reports).

Table 7.14 P1 Protocol Classification Changes for a Delivering ADMD

Protocol Elements	Class	
DeliveredInfo		
typeOfUA	H	
ReportedRecipientInfo		
SupplementaryInformation	H	See Note 1.
GlobalDomainIdentifier		
PrivateDomainIdentifier	H	

For relaying Administration domains, the classifications are all "X"

For originating Administration domains, these are all "NOT APPLICABLE".

Note 1: Domains providing access to TELEX/TELETEX recipients, whether directly or indirectly as a result of bilateral agreements between domains, must ensure that this information, when present, is accessible by the recipient of the delivery report.

7.6.2.3 O/R Names

O/R Names shall consist of:

- o CountryName,
- o AdministrationDomainName.

as well as one of the following:

- o PrivateDomainName,
- o PersonalName,
- o OrganizationName,

- o OrganizationalUnit,
- o UniqueUAIdentifier,
- o X121Address.

and permits the optional inclusion of a

- o DomainDefinedAttributeList.

Note: The destination PrivateDomainName or OrganizationName must be present if destined for a PRMD. The ADMD relaying the message to that destination PRMD requires this element.

7.6.2.4 P1 ADMD Name

Management Domains (MDs) must specify in the ADMD name field of the O/R Name StandardAttributeList in P1, the name of the Administration domain:

- o to which the message is being sent (in recipient names)

- o from which the message originated (in the originator name).

7.6.3 Interworking with Integrated UAs

If the message originates at a UA owned by an ADMD, or is delivered to such a UA, the O/R Name follows the same Form 1 Variant 1 constraints as the base specifications; except that the ADMD name is the name of the ADMD that owns the UA and instead of supplying a PRMD Name, one (or more) of the following must be provided:

- o OrganizationName,
- o OrganizationalUnit,
- o PersonalName.

and may optionally include a

- o DomainDefinedAttributeList.

7.6.4 Differences with Other Profiles

7.6.4.1 TTC Profile

There are no outstanding issues regarding interworking between TTC-conformant systems and NIST-conformant systems with the exception of the number of recipients and the supported MPDU sizes. The ExtensionIdentifier field may contain a maximum value of 32K-1; however, according to the current TTC profile, if a message with more than 256 recipients is received, some TTC-conformant domain may generate a nondelivery notification. This also applies to the ReportedRecipientInfo in a delivery report. Further, a TTC MTA is required to handle an MPDU size of at least 32KB. The NIST required MPDU size is 2MB as covered in Section 7.5.3.3. Other differences are shown in Appendix E. TTC is currently based on Version 4 of the Implementor's Guide.

7.6.4.2 CEPT Profile

See Appendix 7E.

7.6.5 Connection of PRMDs to Multiple ADMDs

Given that Management Domain names (both PRMD and ADMD) shall be unique within the U.S., then when an ADMD is presented a message for transfer from a PRMD, it will accept O/R Names (both originator and recipient) which have an AdministrationDomainName field value different than the Administration's name. "Accept" implies the attempt to route/deliver the message shall be made, as appropriate, based upon the knowledge that MD names are unique.

Whether this functionality is required by an Administration for conformance to this agreement is for further study.

If a PRMD is connected to two or more ADMDs which are not effectively connected (either directly or via a third ADMD), full X.400 functionality shall not be available. Problems occur especially in the areas of:

- Naming,
- Routing,
- Replying.

It should be noted that a single PRMD that is connected to more than one ADMD can be represented by more than one combination of country-name, ADMD-name, and PRMD name. For example, it may occur that the PRMD-name for a particular PRMD may take different values, depending on the ADMD-name. Implementors should be aware of the consequences of these possibilities on routing.

7.6.6 Connection of an ADMD to a Routing PRMD

It is possible for a collection of interconnected private domains to establish one domain as the "gateway" to an ADMD, and hence to the world.

If an ADMD is connected to such a gateway PRMD, the individual private domains shall be registered with the Administration. Administrations need not support such connections.

Note also that upon receipt by the ADMD of a message originating somewhere within the PRMD collection, that the TraceInformation may contain more than one element.

The X.400 Recommendations specify that an ADMD should not attempt to relay a message destined for another ADMD through a PRMD, thus an ADMD should ensure that messages destined for another ADMD are not relayed through a PRMD. It should be noted, however, that a relaying PRMD will relay any such message it receives.

7.6.7 Management Domain Names

- All Management Domain Names (both Private and Administration) shall be unique within the U.S.

- A central naming authority shall be established to register domain names.

7.6.8 Envelope Validation Errors

For validation errors, a non-delivery notice shall be generated (if possible) with reason code of 'unableToTransfer' and diagnostic code of 'invalidParameters' (unless specified otherwise).

ADMDs will validate P1 Envelopes in the following areas:

a) The X.409 syntax of all elements should be checked.

b) The pragmatic constraint limits (lengths of fields and number of occurrences of fields) should be checked.

c) Semantic validation of the following elements should be done:

   - originator O/R Name,
   - recipient O/R Name in the RecipientInfo,
   - Priority.

d) Only recipient Names with the responsibility flag set should be validated. The validation of O/R names is defined in 7.8.3.3; the validation of priority is defined in 7.8.3.7.1.

   MPDU Identifier Validation

   - Validation of the GlobalDomainIdentifier component of the MPDU Identifier is performed upon reception of a message (i.e., as a result of a TRANSFER.Indication).

- The country name should be known to the validating domain, and depending on the country name, validation of the ADMD name may also be possible.

- Additional validation of the GlobalDomainIdentifier is performed against the corresponding first entry in the TraceInformation. If inconsistencies are found during the comparison, a non-delivery notice with the above defined reason and diagnostic codes is generated.

- A request will be generated to the CCITT for a more meaningful diagnostic code (such as 'InconsistentMPDUIdentifier').

7.6.9 Quality of Service

7.6.9.1 Domain Availability

7.6.9.1.1 ADMD Availability

The goal is to provide 24 hour per day availability. Note that there will be periods of time when an ADMD may be unavailable due to maintenance windows in its supporting network or in an MTA within the domain.

7.6.9.1.2 PRMD Availability

Although the goal of PRMD availability is also 24 hours per day, business reasons are likely to dictate some different level of availability. ADMDs shall require a profile from the PRMD that indicates its schedule of regular availability to the ADMD.

7.6.9.2 Delivery Times

In the absence of standardized quality of service parameters, the following are agreed to. When standardized parameters from CCITT Study Group I become available, they shall be adopted.

a) In table 7.15 the following delivery time targets are established:

Table 7.15 Delivery Time Targets

Delivery Class	95% Delivered Before
Urgent	3/4 hour
Normal	4 hours
Non-Urgent	24 hours

b) The interval(s) between retries and the number of retry attempts that an ADMD uses in attempting delivery to a PRMD or integrated UA, will be locally determined domain parameters. However, the total elapsed times after which delivery attempts will be stopped are shown in Table 7.16. This implies that, after these times, a Non-Delivery Notice will be generated.

An Administration shall continue to attempt delivery until the forced nondelivery time, even if the recipient domain has scheduled an unavailability window.

Table 7.16 Forced Nondelivery Times

Delivery Class	NonDelivery Forced After
Urgent	4 hours
Normal	24 hours
Non-Urgent	36 hours

Note: Both tables apply to the period between acceptance by the originating MTA in the originating Administration domain to the time of delivery in the destination Administration domain. Transit time within PRMDs is NOT included in the above times.

7.6.10 Billing Information

a) All aspects relating to billing, charging, tariffs, settlement, and in particular to the use of the billingInformation field in the delivery report, is subject to bilateral agreement, and shall not be addressed in these implementation agreements.

b) No ADMD shall require a PRMD to supply or process billing information.

7.6.11 Transparency

a) No P1 extensions, other than the MOTIS extensions are to be allowed (Reference A/3211). Should an ADMD receive a message containing P1 extensions, it shall generate a non-delivery notice (if possible) with reason code of unableToTransfer and diagnostic code of invalidParameters.

If MOTIS elements are present, a relaying MTA can optionally:

- o Relay the message. If the MTA does relay, it must not drop any of the protocol elements.

- o Non-Deliver the message.

A receiving MTA can optionally:

- o Deliver the message

- o Non-Deliver the message.

b) The CCITT has been requested to establish a more meaningful diagnostic code (such as protocolError) for this occurrence. Such a code has now been provided in the Implementors Guide.

c) P2 extensions shall be relayed transparently by ADMDs.

7.6.12 RTS Password Management

RTS password management shall be a local matter. This includes:

- o password length
- o frequency of changes
- o exchange of passwords with communicating partners
- o loading passwords (i.e., the timing of password changes with respect to active associations).

7.6.13 For Further Study

Issues requiring further study are:

- o Intra-Domain Routing
- o Multi-Vendor Domains

7.7 INTER and INTRA PRMD CONNECTIONS

7.7.1 Introduction

This section is limited in scope to issues arising from the indirect connection of a PRMD to another PRMD or to an ADMD, and to the interconnection of MTAs to form inter-PRMD connections. Indirect means that the connection is made via a relaying PRMD. The X.400 Recommendations describe the way that a PRMD connects to a ADMD and the way that an ADMD connects to another ADMD. The Recommendations do not, however, describe the way that a PRMD connects indirectly to an ADMD or another PRMD, nor do they describe the way that MTAs are connected within a PRMD. These configurations (shown in Figures 7.6 and 7.7) are useful, for example, in connecting equipment from different vendors at a single customer site.

The P1 protocol and its related services for both inter and intra PRMD connections are addressed in this section. In addition, a method for routing within a PRMD is given. It is recognized that uniform methods for Administration, maintenance, and quality of service should be developed for such configurations, and this work is for further study.

This section describes the minimum that must be provided in order to implement a relaying PRMD and a MTA within a PRMD.

This section is presented in the form of deviations from agreements applicable to PRMD to PRMD connection (Section 7.5). That is, unless specifically noted in the remainder of this section, the agreements in Section 7.5 apply to both relaying PRMDs and MTAs within a PRMD.

It should be noted that the comments regarding ORNames in Section 7.6.5 also apply to this section.

7.7.2 The Relaying PRMD

A PRMD that has the capability of relaying messages to another PRMD is called a relaying PRMD. A PRMD implementation need not claim to be a relaying PRMD. A PRMD implementation which does claim to be a relaying PRMD must follow the implementation agreements in this section.

7.7.2.1 Relay Responsibilities of a PRMD

The responsibilities of a relaying PRMD are the same as those of an ADMD (as specified in Sections 7.6.8 and 7.6.2.1). In addition, the PRMD will simply deliver messages routed to it from an ADMD, even if this results in routing a message from the ADMD to the PRMD to another ADMD.

7.7.2.2 Interaction with an ADMD

In order for an ADMD to route a message to ADMD A via ADMD B, it must know that A is reachable through B. Similarly, in order for any MD to route a message to PRMD A via a relaying PRMD B, it must know that A is reachable through B (see Figure 7.8).

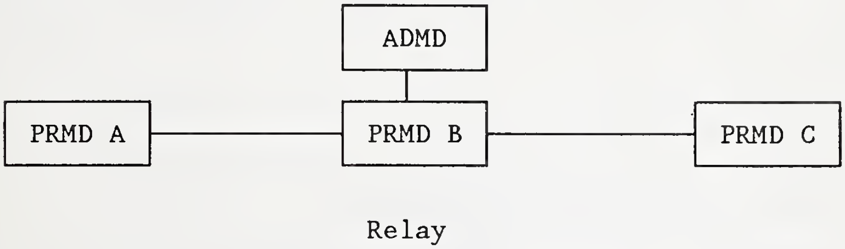


Figure 7.6 Relaying PRMD

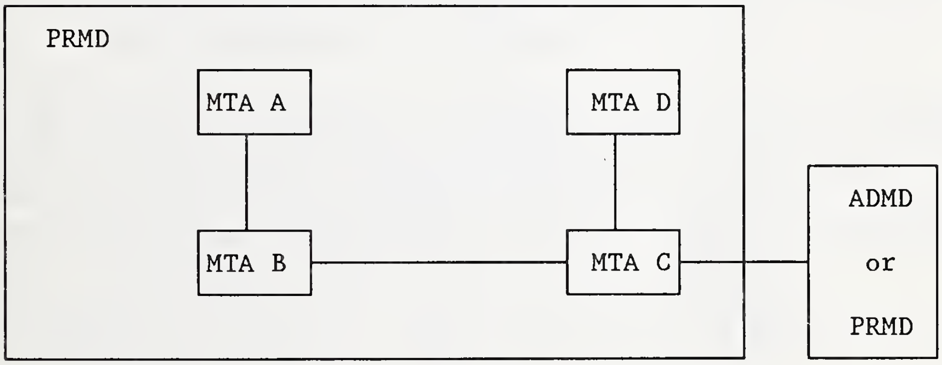


Figure 7.7 Intra PRMD connections

Note 1: Section 7.6.6 specifies that ADMDs are not required to connect to a relaying PRMD, but they are not precluded from doing so.

Note 2: TraceInformation may have more than one sequence on submission of a message by a relaying PRMD to an ADMD.

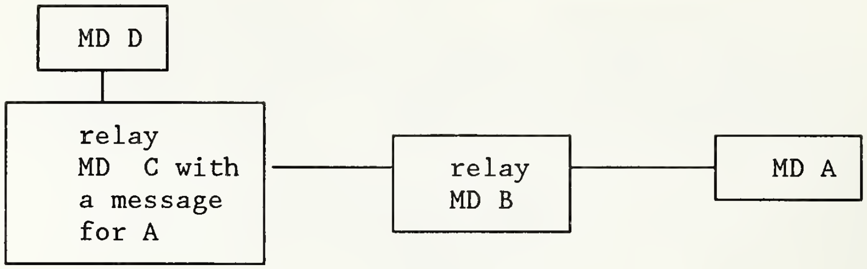


Figure 7.8 MD C must know of A to route the message

7.7.3 Intra PRMD Connections

A PRMD is composed of MTAs which cooperate to perform the functions expected of a domain. An MTA implementation need not claim to follow the implementation agreements of this section.

7.7.3.1 Relay Responsibilities of an MTA

The relaying responsibilities of an MTA are the same as those of an ADMD (as specified in Sections 7.6.8 and 7.6.2.1) with one exception: loop suppression within the domain is done using the MOTIS InternalTraceInfo protocol element. The MTA must validate the InternalTraceInfo (see Section 7.8.3.5 for details on validation). In addition, the PRMD will simply deliver messages routed to it from an ADMD, even if this results in routing a message from the ADMD to the PRMD to another ADMD (please see Section 7.6.6).

7.7.3.2 Loop Suppression within a PRMD

a) The only mechanism defined in the X.400 Recommendations for suppressing loops is TraceInformation, which is added on a per domain basis to detect and suppress loops among domains. Loops among MTAs within a domain need to be detected and suppressed. This implies that each MTA must add trace information that is meaningful within the domain. The MOTIS solution of adding InternalTraceInfo to the P1 Envelope of a message was adopted. The definition of InternalTraceInfo is given in Figure 7.9. The InternalTraceInfo is added by each MTA within a PRMD to handle a message, and it is examined in the same way as TraceInformation to detect and suppress loops.

```
    InternalTraceInfo ::= [APPLICATION 30]
       IMPLICIT SEQUENCE OF
       SEQUENCE {
          MTAName,
          MTASuppliedInfo }

    MTAName ::= PrintableString
```

Figure 7.9 Definition of InternalTraceInfo

If the MTAName and password of X.411 are used for validation, then it is recommended that the MTAName used for validation also be used in the InternalTraceInfo. However, there is a complication: in X.411, MTAName is an IA5String, and the MTAname defined by MOTIS is a PrintableString. Efforts will be made to change the MOTIS definition from PrintableString to IA5String.

b) Three actions are defined in MTASuppliedInfo: relayed, rerouted, and recipientReassignment as shown in Figure 7.10. The recipientReassignment action is not supported in these agreements. The ability to generate it is not required, and if it is present on an incoming message, the action field will be ignored.

```
MTASuppliedInfo ::= SET {
  arrival [0] IMPLICIT Time,
  deferred [1] IMPLICIT Time OPTIONAL,
  action [2] IMPLICIT INTEGER
     { relayed(0), rerouted(1), recipientReassignment(2) }
  previous MTAName OPTIONAL }
```

Figure 7.10 Defined Actions in MTASuppliedInfo

7.7.3.3 Routing Within a PRMD

a) Routing within a PRMD is complicated by the lack of a directory standard. In particular, it constrains intra-domain routing decisions to be based on some combination of the intra-domain attributes of the O/R Name, Organization Name Organizational Units, and Personal Name. In order to enhance interworking and to reduce the difficulty of configuring intra-domain connections, it is useful to restrict the ways in which these may be used in making routing decisions.

b) However, it is recognized that vendors may wish to provide MTAs with varying degrees of routing capability within a PRMD as a temporary expedient until appropriate standards for automated construction of directories and routing tables are available. This section assigns class numbers to certain levels of routing capability and discusses the consequences of using MTAs which fall into each class. The classification scheme will allow some diversity in allocating O/R Name space and in configuring intra-domain routes.

c) When other methods are recommended by standards bodies, the classification scheme described here will become obsolete. Large-scale, multi-vendor PRMDs may not be practical in the absence of standardized methods.

7.7.3.3.1 Class Designations

When it is clear that a message is to be delivered within a domain, the Country Name, ADMD Name, and PRMD Name have already served their purpose in determining the next MTA in the route to the recipient. The remaining fields that might be used on making routing decisions within the PRMD are the Organization Name, Organizational Units, and Personal Name.

MTAs are classified by their ability to discriminate between O/R names when making routing decisions within a PRMD. Conformant MTAs will be classified as shown in Table 7.17.

Table 7.17 Conformant MTA Classifications

	Class 1	Class 2	Class 3
Organization Name	H	H	H
SEQUENCE OF Organizational Unit	X	H	H
Personal Name	X	X	H

a) An 'H' means that the MTA must be able to base its intra-domain routing decisions on the given component of the O/R Name. In particular, both Class 2 and Class 3 MTAs must be able to discriminate on all the members in a supplied sequence of OrganizationalUnits. A Class 3 MTA must be able to discriminate on all of the elements in a PersonalName.

An 'X' means that the MTA need not have the ability to discriminate on the given component.

b) There is a hierarchy in support of components. The ability to discriminate on a given component does not imply the requirement to do so: e.g., a Class 3 MTA is not required to have tables for every PersonalName in the domain. Equally, an MTA which can discriminate on OrganizationalUnits to make routing decisions need not always use the full sequence in an O/R Name if a partial sequence provides enough information.

c) The above classifications only apply to routing decisions in selecting a next hop within a domain. All MTAs are entitled to examine the full O/R Name when identifying their own directly served UAs.

d) The routing table of a Class 1 MTA will be relatively small, because intra-domain routing decisions are based solely on OrganizationName. The routing table of a Class 2 MTA may be substantially larger and more complex because of its ability to discriminate on OrganizationalUnits as well as OrganizationName to make routing decisions. The routing table of a Class 3 MTA may be larger still, because its use of the components of PersonalName in addition to the other information.

7.7.3.3.2 Specification of MTA Classes

If an MTA implementation claims to follow the implementation agreements, it must be either a Class 1, Class 2, or a Class 3 MTA. The class of an MTA implementation should be specified so that PRMD administrators can choose equipment properly.

7.7.3.3.3 Consequences of Using Certain Classes of MTAs

Definition: An MTA which accepts submission requests and furnishes delivery indications to a UA is said to "directly serve" the UA.

a) The presence in a domain of an MTA acting as a Class 1 or Class 2 MTA imposes administrative restrictions on the assignment of O/R Names to UAs and in the configuration of routes within that domain.

- A Class 1 MTA may directly serve UAs from several OrganizationNames. However, if a Class 1 MTA directly serves a UA with a given OrganizationName, no other MTA in the domain may directly serve a user with the same OrganizationName. This means that if all MTAs in a domain are Class 1, then all UAs with a given OrganizationName must be assigned to the same MTA.

- A Class 2 MTA may directly serve UAs from any combination of OrganizationName and sequence of OrganizationalUnits. However, if a Class 2 MTA directly serves a UA with a given combination, no other MTA in the domain may directly serve a user with the same combination. This means that if all MTAs in a domain are Class 2, then all UAs with a given OrganizationName and sequence of OrganizationalUnits must be assigned to the same MTA.

- A domain consisting entirely of Class 3 MTAs is free of all the above restrictions.

b) If Class 1 or Class 2 MTAs are used to perform relaying within a PRMD containing MTAs of other classes, care must be exercised in determining the topology of the domain to avoid leaving certain UAs inacessible from certain MTAs within the domain. The example below shows one of the configurations that should be avoided. The example is intended to stimulate careful examination of the relationship between network and organizational topologies.

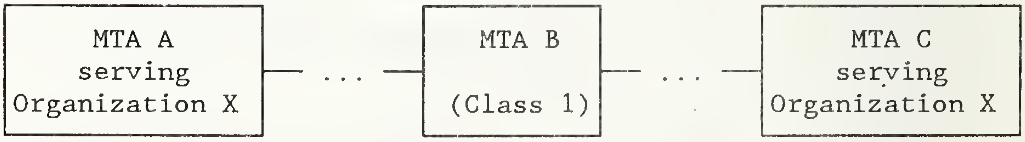


Figure 7.11 Example of a configuration to be avoided

In Figure 7.11, B will route all messages for Organization X to either A or C because B is a Class 1 MTA. The administrator who created this configuration probably wanted B to route some messages for Organization X to A, and some to C. However, B does not have the capability for this because it is only a

Class 1 MTA. The configuration in Figure 7.11 can be corrected by replacing B with a Class 2 or Class 3 MTA.

7.7.3.4 Uniqueness of MPDUidentifiers Within a PRMD

When generating an IA5String in an MPDUIdentifier, each MTA in a domain must ensure that the string is unique within the domain. This shall be done by providing an MTA designator with a length of 12 octets which is unique within the domain, to be concatenated to a per message string with maximum length of 20 octets.

Two pieces of information, the MTA name and MTA designator, need to be registered within a PRMD to guarantee uniqueness. This registration facility need not be automated. If the MTA name is less than or equal to 12 characters, it is recommended that it also be used as the MTA designator.

7.7.4 Service Elements and Optional User Facilities

A PRMD made up of MTAs which support varying sets of service elements in addition to those required in these agreements may appear to provide inconsistent service for these elements. For example, if one MTA supports deferred delivery and another MTA does not, then deferred delivery can be used by some, but not all, users in the PRMD. Similarly, if one MTA supports return of contents and another does not, then a user outside of the PRMD will receive returned contents for messages sent to one user, but not for messages sent to another user. Note that this same inconsistency occurs when sending to two PRMDs which support different additional optional elements.

7.7.5 X.400 Protocol Definitions

This section describes additions and modifications to Section 7.5.3 which are required for implementation of a relaying PRMD or an MTA within a PRMD.

7.7.5.1 Protocol Classification

a) The classification scheme given in Section 7.5.3.1 applies to elements passing from one PRMD to another. For both relaying PRMDs, and MTAs in a PRMD, the same classification scheme will be used, but within a PRMD the classification applies to elements passing from one MTA to another.

b) In addition to the classifications given in Section 7.5.3.1, a classification of Prohibited has been used.

PROHIBITED = P

This element shall not be used. Presence of this element is a protocol violation.

7.7.5.2 P1 Protocol Elements

Table 7.18 contains protocol elements and their classes. An * signifies that the classification of the protocol element has not changed from Table 7.8.

Table 7.18 P1 Protocol Elements

Element	Class	Restrictions and Comments
UMPDUEnvelope		
MPDUIdentifier	M*	This field needs to be unique within a PRMD. See Sections 7.7.3.4 for the method of ensuring uniqueness.
originator	M*	It is recommended that all components of the originator's ORName be included to help ensure that reports can be returned.
TraceInformation	M*	The first MTA in the domain to receive the message adds the TraceInformation. Subsequent MTAs can update the TraceInformation in the event of conversion or deferred delivery. When a message is generated, the originating MTA adds the TraceInformation.
InternalTraceInfo	M/P	This element is mandatory for envelopes transferred between MTAs within a PRMD, and prohibited in messages transferred outside the domain. Elements are always added to the end of the sequence. (See Note 1)
InternalTraceInfo MTAName	M	MTANames within a PRMD must be unique. See Section 7.7.3.4 for the method of assuring uniqueness Maximum length = 32 characters.
MTASuppliedInfo	M	

(Continued on next page.)

Table 7.18 P1 Protocol Elements, continued

Element	Class	Restrictions and Comments
MTASuppliedInfo		
arrival	M	
deferred	X	This field must be generated by MTAs which perform deferred delivery.
action	M	See Section 7.7.3.2 for restrictions on values of this field.
previous	X	This field must be generated by MTAs which perform rerouting.
DeliveryReportEnvelope		
TraceInformation	M*	The first MTA in the domain to receive the report adds the TraceInformation. When a report is generated, the originating MTA adds the TraceInformation.
InternalTraceInfo	M/P	This field is mandatory for envelopes transferred between MTAs within a PRMD, and prohibited in messages transferred outside the domain. (See Note 1)
DeliveryReportContent		
intermediate InternalTraceInfo	P	If it were possible to include this field in the delivery report content, an audit and confirmed report could be provided to detect problems within a PRMD. Efforts are being made to add this field to the MOTIS definition.
DeliveredInfo		
typeOFUA	R*	It is the responsibility of the MTA generating the report to generate this element.

(Continued on next page.)

Table 7.18 P1 Protocol Elements, continued

Element	Class	Restrictions and Comments
ProbeEnvelope		
TraceInformation	M*	The first MTA in the domain to receive the probe adds the TraceInformation. When a probe is generated, the originating MTA adds the TraceInformation.
InternalTraceInfo	M/P	This field is mandatory for envelopes transferred between MTAs within a PRMD, and prohibited in messages transferred outside the domain. (See Note 1)

Note 1: The M classification is only applicable if an implementation is claiming conformance according to Section 7.10.2, point (g) 4th bullet.

7.7.5.3 Reliable Transfer Server (RTS)

In the pUserData of PConnect, the value of applicationProtocol should be 1. This value was chosen because the agreements on intra-domain connections are not strictly P1, nor are they MOTIS. Philosophically, it would be good to choose a new application protocol identifier for these agreements, but this introduces too many practical problems. Since these agreements are closer to P1 than to MOTIS, the value of 1 will be used. This will not cause interworking problems between domains, because the only deviation from P1 is the InternalTraceInfo, which will not be present in messages transferred outside of a domain.

7.8 ERROR HANDLING

This section describes appropriate actions to be taken upon receipt of protocol elements which are not supported in this profile, malformed MPDUs, unrecognized O/R Name forms, content errors, errors in reports, and unexpected values for protocol elements.

7.8.1 MPDU Encoding

The MPDU should have a context-specific tag of 0, 1, or 2. If it does not have one of these tags, it is not possible to figure out who originated the message. Therefore, the way this error is reported is a local matter.

7.8.2 Contents

Once delivery to the UA has occurred, it is not possible to report errors in P2 information to the originator. In addition, it seems unreasonable to insist that the MTA that delivers a message ensure that the P2 content of the message is acceptable. As a result, the handling of content errors is a local matter.

7.8.3 Envelope

This section describes the handling of errors in message envelopes. Some of the error conditions described below may be detected in a recipient's O/R Name. This may limit the reporting MTA's ability to generate a nondelivery notification that accurately reflects the erroneous O/R Name in the ReportedRecipientInfo. This handling of this situation is a local matter.

7.8.3.1 Pragmatic Constraint Violations

In all cases of pragmatic constraint violation, a nondelivery report should be generated with a ReasonCode of unableToTransfer, and a DiagnosticCode of pragmaticConstraintViolation.

7.8.3.2 Protocol Violations

a) If all required protocol elements are not present, a nondelivery report with a ReasonCode of unableToTransfer and a DiagnosticCode of protocolViolation should be generated.

b) If a protocol element is expected to be of one type, but is encoded as another, then a nondelivery report with a ReasonCode of unableToTransfer and a DiagnosticCode of invalidParameters should be generated.

7.8.3.3 O/R Names

a) The domain that has responsibility for delivering a message should also have the responsibility to send the nondelivery notification if the message cannot be delivered. Therefore, each MTA should only validate the O/R Names of recipients with responsibility flags set to TRUE. In addition, a nondelivery notification can only be sent if the originator's O/R Name is valid.

b) If any element in the O/R Name is unrecognized or if the CountryName, AdministrationDomainName, and one ofPrivateDomainName and OrganizationName (and, for ADMDs, PersonalName and OrganizationalUnit) are not all present, then a nondelivery report should be generated with a ReasonCode of unableToTransfer, and a DiagnosticCode of unrecognizedORName. If the message can be delivered even though the ORName is invalid, delivery is a local matter. Note, however, that if the message is delivered, the invalid ORName might be propagated through the X.400 system (e.g., by forwarding).

c) If the O/R Name has all of the appropriate protocol elements and the message still cannot be delivered to the recipient, the following DiagnosticCodes may appear in the nondelivery report: unrecognizedORName,ambiguousORName,and uaUnavailable.

7.8.3.4 TraceInformation

a) Since non-relaying domains need not do loop suppression, domains with responsibility for delivering the message need not be concerned about the semantics of the TraceInformation, that is, arrival time and converted EncodedInformationTypes can be provided to the UA without inspection by the MTAs of the domain as long as the TraceInformation is properly encoded according to X.409.

b) When a message is accepted for relay, the relaying domain must check that a TraceInformation SEQUENCE has been added by the domain that last handled the message. If the appropriate TraceInformation was not added, this should be treated as a protocolViolation (Section 7.8.3.2).

c) In addition, the relaying domain must check that the information was added in the sequence defined by the rules for adding TraceInformation in the CCITT X.400 Implementor's Guide. If the sequence is invalid,then a nondelivery report should be generated with a ReasonCode of unableToTransfer and a diagnosticCode of invalidParameters.

Note: It would be desirable for the CCITT to add a diagnostic code of invalidTraceInformation to allow a more meaningful description of this problem. A request for this new diagnostic code will be submitted.

7.8.3.5 InternalTraceInfo

This section applies only to MTAs which follow the agreements of Section 7.7.

a) When a message is accepted for relay from another MTA in the domain, the relaying MTA must check that an InternalTraceInfo SEQUENCE has been added by the MTA that last handled the message. If the appropriate InternalTraceInfo was not added, this should be treated as a protocolViolation (Section 7.8.3.2).

b) In addition, the relaying MTA must check that the information was added in the sequence defined by the rules for adding TraceInformation in the CCITT X.400 Implementor's Guide. If the sequence is invalid, then a nondelivery report should be generated with a ReasonCode of unableToTransfer and a diagnosticCode of invalidParameters.

Note: It would be desirable for the CCITT to add a diagnostic code of invalidTraceInformation to allow for a more meaningful description of this problem. A request for this new diagnostic code will be submitted.

7.8.3.6 Unsupported X.400 Protocol Elements

The protocol elements defined in X.400 but unsupported by this profile are: the deferredDelivery and PerDomainBilateralInfo parameters of the UMPDUEnvelope, the ExplicitConversion parameter of RecipientInfo, and the alternateRecipientAllowed and contentReturnRequest bits of the PerMessageFlag. Appropriate actions are described below for domains that do not support the protocol elements.

7.8.3.6.1 deferredDelivery

The delivering domain shall do one of the following:

- deliver at once,
- hold for deferred delivery,
- return a nondelivery notification with a ReasonCode of unableToTransfer and a DiagnosticCode of noBilateralAgreement.

7.8.3.6.2 PerDomainBilateralInfo

If a delivering domain receives this element, the element can be ignored.

7.8.3.6.3 ExplicitConversion

If ExplicitConversion is requested the message should be delivered if possible. That is, if the UA is registered to accept the EncodedInformationTypes of the message, then the message should be delivered even though the requested conversion could not be performed along the route. If delivery is not possible, then a nondelivery report should be generated with a ReasonCode of conversionNotPerformed with no DiagnosticCode.

7.8.3.6.4 alternateRecipientAllowed

If a delivering domain receives this element the element can be ignored.

7.8.3.6.5 contentReturnRequest

If a delivering domain receives this element, the element can be ignored.

7.8.3.7 Unexpected Values for INTEGER Protocol Elements

There are three INTEGERs in the P1 Envelope. Appropriate actions are described below for domains receiving unexpected values for Priority, ExplicitConversion, and ContentType.

7.8.3.7.1 Priority

Additional values for Priority have been suggested by at least one group of implementors as upward compatible changes to the X.400 Recommendations. Therefore, if a PRMD receives an unexpected value for Priority, and this value is greater than one byte in length, a nondelivery report should be generated with a ReasonCode of unableToTransfer and DiagnosticCode of invalidParameters. If the value is less than or equal to one byte, the PRMD can either generate a nondelivery report as previously specified or interpret the Priority as normal and deliver or relay the message.

7.8.3.7.2 ExplicitConversion

When an unexpected value is received for ExplicitConversion, it should be handled as in Section 7.8.3.6.3.

7.8.3.7.3 ContentType

If the ContentType is not supported by the delivering MTA, then a nondelivery report should be generated with a ReasonCode of unableToTransfer, and a DiagnosticCode of contentTypeNotSupported.

7.8.3.8 Additional Elements

In the absence of multilateral agreements to the contrary, receipt of privately tagged elements and protocol elements in addition to those defined in X.400 will result in a nondelivery report with a ReasonCode of unableToTransfer and a DiagnosticCode of invalidParameters.

The exceptions to this are the MOTIS elements. The treatment of MPDU's containing these MOTIS extensions is described in Section 7.6.11.

7.8.4 Reports

There is no mechanism for returning a delivery or status report due to errors in the report itself. Therefore the handling of errors in reports is a local matter.

7.9 MHS USE OF DIRECTORY SERVICES

7.9.1 Directory Service Elements

a) Recommendation X.400 recognizes the need of MHS users for a number of directory service elements. Directory service elements are intended to assist users and their UAs in obtaining information to be used in submitting messages for delivery by the MTS. The MTS may also use directory service elements to obtain information to be used in routing messages. Some functional requirements of directories have been identified and are listed below.

o Verify the existence of an O/R name.

o Return the O/R address that corresponds to the O/R name presented.

o Determine whether the O/R name presented denotes a user or a distribution list.

- o Return a list of the members of a distribution list.
- o When given a partial name, return a list of O/R name possibilities.
- o Allow users to scan directory entries.
- o Allow users to scan directory entries selectively.
- o Return the capabilities of the entity referred to by the O/R name.
- o Provide maintenance functions to keep the directory up-to-date.

b) In addition to functionality, a number of operational aspects must be considered. These include user-friendliness, flexibility, availability, expandability, and reliability.

c) Currently, these aspects of directory service elements and procedures are under study by both the CCITT and the ISO. Both organizations are committed to the development of a single Directory Service specification for use by MHS and all other OSI based applications.

Given the incomplete nature of the ongoing activities within the CCITT and the ISO, <u>no implementation details will be provided now</u> <u>for MHS use of Directory Services</u>. Implementation agreements for MHS Use of Directory Services will be issued when current activities within the CCITT and the ISO are stable.

7.9.2 Use of Names and Addresses

a) It is recognized that these agreements enable a wide variety of naming and addressing attributes (see Section 7.5.3.5 ORName Protocol Elements) wherein each PRMD may adopt particular routing schemes within its domain.

b) With the exception of the intra-domain connection agreements:

These agreements make no attempt to recommend a standard practice for electronic mail addressing.

c) Inter-PRMD addressing may be secured according to practices outside the scope of these agreements, such as:

- o manual directories
- o on-line directories

- ORName address specifications
- ORName address translation.

d) Further, each PRMD may adopt naming and addressing schemes wherein the user view may take a form entirely different from the attributes reflected in Table 7.9. And, each PRMD may have one user view for the originator form and another for the recipient form, and perhaps other forms of user addressing. In some cases (e.g., receipt notification) these user forms must be preserved within the constraints of these implementation agreements. However, mapping between one PRMD user form to another PRMD user form, via the X.400 ORName attributes of these agreements, is outside the scope of these agreements.

7.10 CONFORMANCE

7.10.1 Introduction

In order to ensure that products conform to these implementation agreements, it is necessary to define the types and degrees of conformance testing that products must pass before they may be classified as conformant. This section defines the conformance requirements and provides guidelines for the interpretation of the results from this type of testing.

This section is incomplete and will be enhanced in future versions of this Agreement. Later versions will reflect the problems of conformance testing and will outline specific practices and recommendations to aid the development of conformance tests and procedures.

7.10.2 Definition of Conformance

For this section, the term conformance is defined by the following:

a) The tests indicated for this section are intended to establish a high degree of confidence in a statement that the implementation under test (IUT) conforms (or does not conform) to the agreements of this section.

b) Conformance to a service element means that the information associated with the service element is made accessible to the user (person or process) whenever this agreement says that this information should be available.

Accessible means that information must be provided describing how a user (person or process):

- o causes appropriate information to be displayed, or
- o causes appropriate information to be obtained.

c) Conformance to P1, P2, and RTS as part of an X.400 OSI application requires that only the external behavior of that OSI system adheres to the relevant protocol standards.

In order to achieve conformance to this section, it is not required that the inter-layer interfaces be available for testing purposes.

d) Conformance to the protocols requires:

- o that MPDUs correspond to instances of syntactically correct data units,

- o MPDUs in which the data present in the fields and the presence (or absence) of those fields is valid in type and semantics as defined in X.400, as qualified by this profile,

- o correct sequences of protocol data units in responses (resulting from protocol procedures).

e) Statements regarding the conformance of any one implementation to this profile are not complete unless a Protocol Implementation Conformance Statement (PICS) is supplied.

f) The term "Implementation Under Test" (IUT) is interchangeable with the term "system" in the definition of conformance, and may refer to:

- o a domain, which may be one or more MTA's with co-located or remote UA's,

- o a single instance of an MTA and co-located UA with X.400 (P1, P2, RTS and session) software,

- o a relaying product with P1, RTS and session software,

- o a gateway product.

g) Claiming Implementation Conformance

- o An implementation which claims to be conformant as an ADMD must adhere to the agreements in Sections 7.5 and 7.6.

- o An implementation which claims to be conformant as a PRMD must adhere to the agreements in Section 7.5.

- An implementation which claims to be conformant as a relaying PRMD must adhere to the agreements in Section 7.5 and the appropriate Sections of 7.7.

- An implementation which claims to be conformant to the intra-domain connection agreements must adhere to the agreements in Section 7.5 and the appropriate Sections of 7.7.

7.10.3 Conformance Requirements

7.10.3.1 Introduction

Conformance to this specification requires that all the services listed as supported in Sections 7.5, 7.6, and if appropriate, 7.7 of these agreements are supported in the manner defined, in either the CCITT X.400 Recommendations or these agreements. It is not necessary to implement the recommended practices of Section 7.12, Appendix B, in order to conform to these agreements.

It is the intention to adopt, where and when appropriate the testing methodology and/or the abstract test scenarios currently being defined by the CCITT X.400 Conformance Group. However, it is recognized that formal CCITT Recommendations relating to X.400 Conformance Testing will not be available until 1988. It is also recognized that aspects of these agreements are outside the scope of the CCITT, and that other organizations will have to provide conformance tests in these cases.

7.10.3.2 Initial Conformance

This section is intended to provide guidelines to vendors who envisage having X.400 products available prior to any formal mechanism, or "Conformance Test Center" being made accessible that would allow for conformance to this product specification to be tested.

It is feasible that vendors and carriers will want to enter bilateral test agreements that will allow for initial trials to be carried out for the purposes of testing initial interworking capabilities. It is equally feasible that for the purposes of testing interoperability, only a subset of this specification will initially be tested.

Note: By claiming conformance to this subset of information the vendor or carrier CANNOT claim conformance to this entire specification.

There are two aspects to the requirements, interworking and service, as described in the following sections.

7.10.3.2.1 Interworking

The interworking requirements for conformance implies that tests be done to check for the syntax and semantics of protocol data elements for a system as defined by the classification scheme of Sections 7.5.2.1.1 and 7.7.5.2. For a relay system, the correct protocol elements should be relayed as appropriate. For a recipient system, a message with correct protocol elements must not be rejected where appropriate.

7.10.3.2.2 Service

For information available to the recipients via the IPMessage Heading and Body, the following should be made accessible:

- IPMessage ID - only the PrintableString portion of the IPMessageId needs to be accessible.
- subject,
- primaryRecipients,
- copyRecipients,
- blindcopyRecipients,
- authorizingUsers,
- originator,
- inReplyTo,
- replyToUsers,
- importance,
- sensitivity,
- IA5Text Bodypart.

7.11 APPENDIX A: INTERPRETATION OF X.400 SERVICE ELEMENTS

The work on service element definitions is limited to those that are defined as 'supported' in Section 7.5 of this specification. Furthermore it is not the intent of this section to define how information should be made available or presented to a MHS user, nor is it intended to define how individual vendors should design their products. In addition, statements on conformance to a specific service element and the allocation of error codes that are generated as a result of violations of the service should be defined in the sections on conformance and errors as part of the main product specification. The main objective is to provide clarification, where required, on the functions of a service element, and in particular what the original intent of the Recommendations were.

SERVICE ELEMENTS

The following Service Elements defined in X.400 have been examined and require further text to be added to their definitions to represent the proposed implementation of these service elements by the X.400 SIG.

The service element clarifications are to be taken in the context of this profile.

Service elements not referenced in this section are as defined in X.400.

PROBE

A PRMD need <u>not</u> generate probes.

If a probe is addressed to and received by a PRMD, the PRMD <u>must</u> respond with a <u>Delivery Report</u> as appropriate at the time the probe was processed.

DEFERRED DELIVERY

In the absence of bilateral agreements to the contrary, Deferred Delivery and Deferred Delivery Cancellation are local matters (i.e., confined to the originating domain) and need <u>not</u> be provided.

The extension of Deferred Delivery beyond the boundaries of the initiating domain is via bilateral agreement as specified in Section 3.4.2.1 of X.411.

Content Type Indication

It is required that both an originating and recipient domain be able to support P2 content type. The ability for domains to be able to exchange content types other than P2 will depend on the existence of bilateral or multi-lateral agreements.

Original Encoded Information Types Indication

It is required that both an originating and recipient domain be able to support IA5 text. Support for other encoded information types, for the purposes of message transfer between domains, will depend on the existence of bilateral or multi-lateral agreements.

The use of the 'unspecified' form of encoded information type should only be used when the UMPDU content represents an SR-UAPDU or contains an auto-forwarded IM-UAPDU.

The original encoded information type of a message is not meaningful unless a message is converted en route to the recipient. These agreements support only IA5 text, which should not undergo conversion. The original encoded information types should be made accessible to the recipient for upward compatibility with the use of non-IA5 text message body parts.

Registered Encoded Information Types

A UMPDU with an 'unspecified' value for Original Encoded Information Type shall be delivered to the UA.

Delivery Notification

The UAContentID may be used by the recipient of the delivery notification for correlation purposes.

Disclosure of Other Recipients

This service is not made available by originating MTAE's to UAE's, but must be supported by relaying and recipient MTAE's.

By supporting the disclosure of other recipients the message recipient can be informed of the O/R names of the other recipient(s) of the message, as defined in the P1 envelope, in addition to the O/R Descriptors within the P2 header.

These agreements do not support initiation of disclosure of other recipients, but the information associated with it should be made accessible to the recipient for upward compatibility with support for the initiation of this service element.

Typed Body

As defined in X.400 with the addition of the Private Body Types that are to be supported. At present there is no mechanism provided within X.420 that would allow you to respond to reception of an unsupported body type.

Action taken in this situation is a local matter.

Blind Copy Recipient Indication

It should be considered that the recipient's UA acts on behalf of the recipient, and therefore may choose to disclose all BCC recipients to each other. Therefore it is the responsibility of the originating domain to submit two or more messages, depending on whether or not each BCC should be disclosed to each other BCC.

Auto Forwarded Indication

A UA may choose not to forward a message that was previously auto-forwarded. In addition there is no requirement for an IPM UA that does not support non-receipt or receipt notification to respond with a non-receipt notification when a message is auto-forwarded.

Primary and Copy Recipients Indication

It is required that at least one primary recipient be specified; however, for a forwarded message this need not be present. The recipient UA should be prepared to accept no primary and copy recipients to enable future interworking with Teletex, Fax, etc.

Sensitivity Indication

A message originator should make no assumptions as to the semantic interpretation by the recipients UA regarding classifications of sensitivity. For example, a personal message may be printed on a shared printer.

Reply Request Indication

In requesting this service an originator may additionally supply a date by which the reply should be sent and a list of the intended recipients of the reply. If no such list is provided then the initiator of the reply sends the reply to the originator of the message and any recipients the reply initiator wishes to include. The replytoUsers and the replyBy date may be specified without any explicit reply being requested. This may be interpreted by the recipient as an implicit reply request. Note that for an auto-forwarded message an explicit or implicit reply request may not be meaningful.

Body Part Encryption

The original encoded information type indication includes the encoded information type(s) of message body parts prior to encryption by the originating domain. The ability for the recipient domain to decode an encrypted body part is a local matter. Successful use of this facility can only be guaranteed if there exists bilateral agreements to support the exchange of encrypted body parts.

Forwarded IP message Indication

The following use of the original encoded information type in the context of forwarded messages is clarified:

- If forwarding a private message body part the originator of the forwarded message shall set the original encoded information types in the P1 envelope to undefined for that body part.

- The encoded information types of the message being forwarded should be reflected in the new original encoded information types being generated.

- See Appendix 7B on recommended practices for the use of the delivery information as part of Forwarded IP-message.

Multipart Body

It is the intent of multipart bodies to allow for the useful and meaningful structuring of a message that is constructed using differing body part types. For example, it is not recommended that a message made up of only IA5 text should be represented as a number of IA5 body parts, each one representing a paragraph of text.

7.12 APPENDIX B: RECOMMENDED X.400 PRACTICES

It is not necessary to follow the recommended practices when claiming conformance to these agreements.

7.12.1 RECOMMENDED PRACTICES IN P2

1. ORDescriptor

   Vendors following the NIST/OSI Workshop guidelines shall, whenever possible, generate the ORName portion of an ORDescriptor in ALL IPM heading fields.

2. ForwardedIPMessage BodyParts

   ForwardedIPMessage_BodyParts should be nested no deeper than eight. There is no restriction on the number of ForwardedIPMessage BodyParts at any given depth.

3. DeliveryInformation

   It is strongly recommended that DeliveryInformation be supplied in both forwarded and autoforwarded message body parts. DeliveryInformation is useful when a message has multiple forwarded message body parts because without it, the EncodedInformationType(s) of the component forwarded messages cannot be deduced easily. DeliveryInformation is useful for autoforwarded messages because the EncodedInformationType of an autoforwarded message is "unspecified" and the EncodedInformationType(s) of the message cannot be determined easily without it. Absence of the EncodedInformationType(s) makes it difficult for a UA to easily determine whether the message can be rendered.

7.12.2 RECOMMENDED PRACTICES IN RTS

1. In the case where S-U-ABORT indicates a temporaryProblem, reestablishment of the session should not be attempted for a "sensible" time period (typically not less than five minutes).

   In instances where this delay is not required or necessary, report a localSystemProblem.

2. S-U-EXCEPTION-REPORT reason codes can be interpreted as follows:

   o receiving ability jeopardized (value 1)
   Possible meaning: The receiving RTS knows of an impending system shutdown.

- local ss-User error (value 5)
  Possible meaning: The receiving RTS needs to resynchronize the session dialogue.

- irrecoverable procedure error (value 6)
  Possible meaning: The receiving RTS has had to delete a partially received APDU, even though some minor synchronization points have been confirmed.

- non specific error (value 0)
  Possible meaning: The receiving RTS cannot handle the APDU (for example, because it was too large) and wishes to inform the sending RTS not to try again.

- sequence error (value 3):
  Possible meaning: The S-ACTIVITY-RESUME request specified a minor synchronization point serial number which does not match the checkpoint data.

3. For purposes of identifying an MTA during an RTS Open, OSI addressing information should be used. This addressing information is conveyed by lower layer protocols and is reflected by the calling and called SSAP parameters of the S-CONNECT primitives.

   MTA validation and identification are related, but separate, functions. The mTAName and password protocol elements of the RTS user data should be used for validation, rather that identification, of an MTA. The RTS initiator and responder may independently require each other to supply mTAName and password.

   The CallingSSUserReference parameter of the S-CONNECT primitives should only have meaning to the entity that encoded it and should not be used to identify an MTA.

7.12.3 RECOMMENDED PRACTICES FOR ORName

Table 7.9 stipulates that the StandardAttributeList must contain either PrivateDomainName or OrganizationName. It is recommended that, for both originator and recipients in a private domain, the PrivateDomainName field be used.

It is recommended that there should be a DomainDefinedAttribute to be used in addressing UAs in existing mail systems, in order to curtail the proliferation of different types of DomainDefinedAttributes used for the same purpose. The syntax of this DomainDefinedAttribute conforms to the CCITT Pragmatic Constraints, and thus has a maximum value length of 128 octets and a type length of 8 octets, each of type Printable String. Only one occurrence is allowed.

This DomainDefinedAttribute has the type name "ID" (in uppercase). It contains the unique identifier of the UA used in addressing within the domain. This DomainDefinedAttribute is to be exclusively used for routing within the destination domain (i.e., once routed to that domain via the mandatory components of the StandardAttributeList); any other components of the StandardAttributeList may be provided. If they conflict delivery is not made.

The contents of this parameter need not be validated in the originating domain or any relaying domain, but simply transferred intact to the next MTA or domain.

Class 2 and class 3 MTAs in a PRMD should allow administrators to decide the number of OrganizationalUnits that should appear in user names, instead of imposing a software controlled limit which is less than four. This is desirable because when two different vendors impose different limits on the number of OrganizationalUnits in a name, it becomes difficult for the administrator to choose a sensible naming scheme.

There are existing mail systems that include a small set of non-Printable String characters in their identifiers. For these systems to communicate with X.400 messaging systems, either for pass-through service or delivery to X.400 users, gateways will be employed to encode these special characters into a sequence of Printable String characters. This conversion should be performed by the gateway according to a common scheme and before insertion in the ID DDA, which is intended to carry electronic mail identifiers. X.400 User Agents may also wish to perform such conversions.

It is recommended that the following symmetrical encoding and decoding algorithm for non-Printable String characters be employed by gateways. The encoding algorithm maps an ID from an ASCII representation to a PrintableString representation. Any non-printable string characters not specified in the table are covered by the category "other" in the table below.

The principal conversion table for the mapping is as follows:

Table 7B.1 Printable string to ASCII mapping

ASCII Character	Printable String Character
% (percent)	(p)
@ (at sign)	(a)
! (exclamation)	(b)
" (quote mark)	(q)
_ (underline)	(u)
((left paren.)	(l)
) (right paren.)	(r)
other	(3DIGIT)

where 3DIGIT has the range 000 to 377 and is interpreted as the octal encoding of an ASCII character.

To encode an ASCII representation to a PrintableString, the table and the following algorithm should be used:

```
IF current character is in the encoding set THEN
     encode the character according to the table above
ELSE
   write the current character;
continue reading;
```

To decode a PrintableString representation to an ASCII representation, the table and the following algorithm should be used:

```
IF current character is not "(" THEN
   write character
ELSE
   {
   look ahead appropriate characters;
   IF composite characters are in the above table THEN
      decode per above table
   ELSE
      write current character;
   }
continue reading;
```

Class 2 and class 3 MTAs in a PRMD should allow administrators to decide the number of OrganizationalUnits that should appear in user names, instead of imposing a software controlled limit which is less than four. This is desirable because when two different vendors impose different limits on the number of OrganizationalUnits in a name, it becomes difficult for the administrator to choose a sensible naming scheme.

7.12.4 POSTAL ADDRESSING

For domains wishing to support postal (or physical) delivery options, the following interim set of "nationally-defined" domain defined attributes are recommended. The CCITT will define Standard Attributes in support of physical delivery in its 1988 Recommendations; this is only an interim solution.

CCITT will also be addressing the services associated with physical delivery. This interim solution does not address the end-to-end service aspects of physical delivery; in particular, the following IPM service elements do not currently extend outside of the X.400 environment:

- alternate Recipient Assignment
- PROBE
- Receipt Notification / Non-Receipt Notifications
- Grade of delivery

"Delivery" means passing a message from the MTS to the physical delivery system (PDS), and not to the user (or user agent).

The following three DDAs are recommended to be used to specify a postal (or physical) address:

CNTRPC encodes the country and postal code for postal delivery. The DDA value is of the form "Country?Postalcode" (for example, "USA?22096"). The country field is optional, the postal code is optional; the separator ("?") is not. If both country and postal code are missing, this DDA should not be specified.

PDA 1 The country and postal code fields are free-form text.

PDA 2 These two DDA (signifying Postal Delivery Address strings 1 and 2) form a 256 character free-form postal address. Fields are separated by a question mark ("?"). There is no implied separator between PDA1 and PDA2. The meaning of the fields are defined by each domain supporting the physical delivery interface. PDA1 contains the first 128 characters, PDA2 the next 128 characters. If the PDA string is less than 128 characters, PDA2 is not used.

For example, if the domain interprets the PDA fields as lines, the address

Mr. John Smith
Conway Steel
123 Main Street
Reston VA 22096

would be encoded as follows:

```
type = "PDA1"  value = "Mr. John Smith?Conway Steel?123 Main
Street?Reston VA"
CNTRPC = "?22096"
```

7.12.5 EDI use of X.400

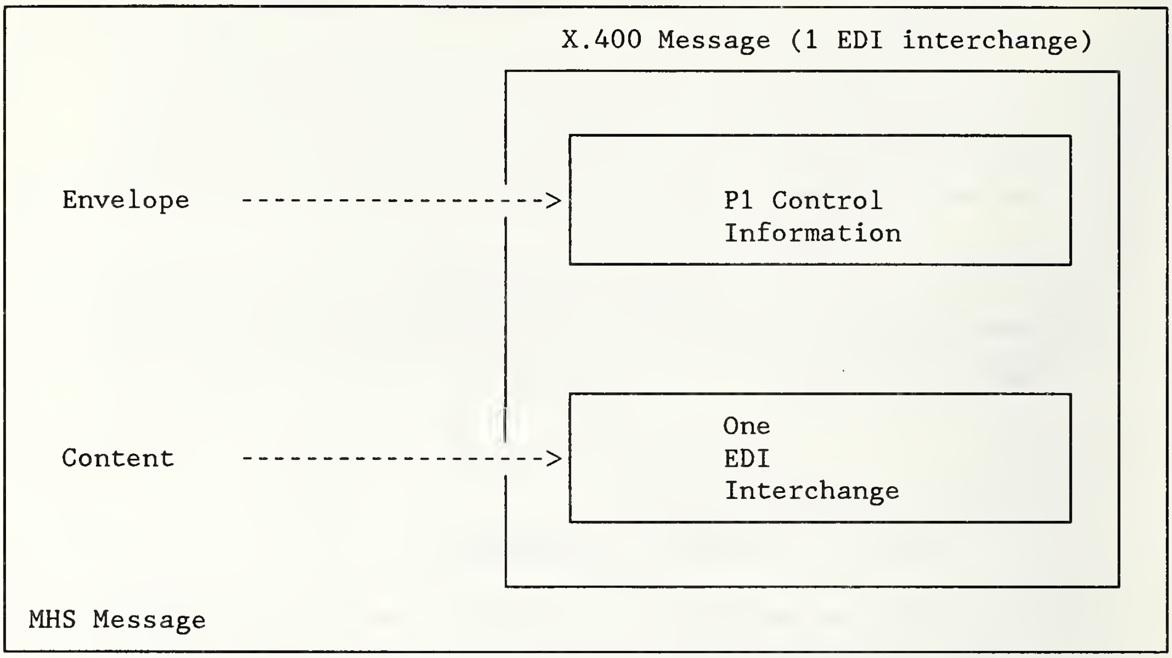
7.12.5.1 Introduction and Scope

This is a guideline for EDI data transfer in an X.400 environment conforming to the NIST agreements. These recommended practices outline procedures for use in transferring EDI transactions between trading partner applications in an attempt to facilitate actual X.400 implementation by EDI users.

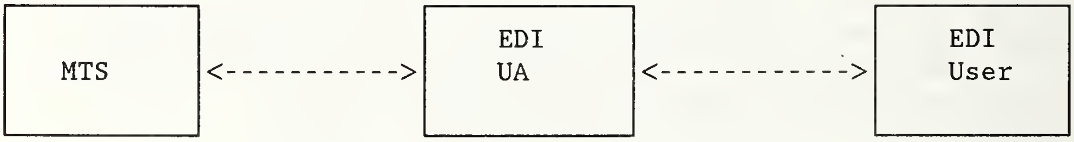
The scope of this guideline is to describe specific recommendations for adopting X.400 as the data transfer mechanism between EDI applications.

7.12.5.2 Model

The MHS recommendations can accommodate EDI through the approach illustrated below. Many Message Transfer (MT) service elements defined in the X.400 recommendations are particularly useful to the EDI application.



This diagram depicts an EDI content (1 EDI interchange) enveloped by the P1 MHS envelope. All the MT Services defined in the X.400 Recommendations may be used for EDI. However, it is not required to support optional or non-essential services to exchange EDI data between EDI users. When an EDI user submits an EDI Trade Document to the EDI User Agent, the EDI UA will submit the EDI content plus P1 envelope to the Message Transfer System (MTS).



The EDI UA must support the essential MT Services as defined in these Agreements; for example, as a minimum, to provide default values for services not elected by the EDI user, such as Grade of Delivery.

Note: MT Services are not necessarily made available by the EDI UA to the EDI user.

7.12.5.3 Protocol Elements Supported for EDI

The following P1 protocol elements will be used to support EDI applications:

Content Type

For EDI applications, the content type will be 0 (undefined content).

Original Encoded Information Types

Any EIT defined in the X.400 Recommendations may be used to specify the encoding of EDI content. However, for ANSI X12 EDI applications in particular, it is expected that the "undefined" and "Ia5Text" EIT's will normally be used, with "undefined" used to signify the EBCDIC character set.

7.12.5.4 Addressing and Routing

It is anticipated that connection of some existing systems to an X.400 service for EDI purposes will be by other than X.400 protocols, at least in the short term.

EDI messages entering the X.400 environment will therefore need to have X.400 O/R Names added to identify the origination and recipient trading partners, typically by means of local directory services in the origination domain which will map EDI identifiers/addresses into O/R Names. Such O/R Names will contain Standard Attributes as defined in Table 7.9 and for recipient trading partners will at least identify the destination domain.

In the case of trading partners outside the X.400 environment, it is expected, however, that there will be cases where message delivery will require the provision of addressing information beyond that which can be carried in Standard Attributes. In such cases, Domain Defined Attributes are recommended to be used.

The syntax of this DDA is as defined in Table 7.9, with a single occurrence having the type name "EDI" (uppercase) and a value containing the identifier/address of the trading partner. For ASC X12 purposes, specifically, this value will comprise the 2 digit interchange ID qualifier followed by the interchange ID (max 15 characters). Routing on this DDA shall only occur, if at all, in the destination domain.

7.12.6 USA Body Parts

It is recommended that UAs can generate any USA Body Part, as defined in Section 7.5.3.6.2, and that they can receive such body parts as well. reception of USA Body Parts does not imply further processing by the UA, but merely that the body part is made available, with a indication of its registered body part identifier, to another process or deposition in a file. Generation implies the reverse of this process.

7.13 APPENDIX C: RENDITION OF IA5Text AND T61String CHARACTERS

7.13.1 GENERATING AND IMAGING IA5Text

The characters that may be used in an IA5String are the graphic characters (including Space), control characters and Delete of the IA5 character repertoire ISO 646.

The graphic characters that may be used with a guaranteed rendition are those related with positions 2/0 to 2/2, 2/5 to 3/15, 4/1 to 5/10, 5/15 and 6/1 to 7/10 in the basic 7-bit code table.

The other graphic characters may be used but have no guaranteed rendition.

The control characters that may be used but have no guaranteed effect are a subset consisting of the format effectors 0/10 (LF), 0/12 (FF) and 0/13 (CR) provided they are used in one of the following combinations:

CR LF	to start a new line
CR FF	to start a new page (and line)
LF .. LF	to show empty lines (always after one of the preceding combinations).

The other control characters or the above control characters in different combinations may be used but have no guaranteed effect.

The character Delete may occur but has no guaranteed effect. The IA5String in a P2 IA5Text BodyPart represents a series of lines which may be divided into pages. Each line should contain from 0 to 80 graphic characters for guaranteed rendition. Longer lines may be arbitrarily broken for rendition. Note that X.408 states that for conversion from IA5Text to Teletex, the maximum line length is 77 characters.

7.13.2 GENERATING AND IMAGING T61String

For further study.

7.14 APPENDIX D: DIFFERENCES IN INTERPRETATION DISCOVERED THROUGH TESTING OF THE MHS FOR THE CeBit 87 DEMONSTRATION

Several interworking problems were discovered through multi-vendor testing. These problems, and recommendations for solutions to them are discussed in this appendix.

7.14.1 ENCODING OF RTS USER DATA

The password is defined as an ANY in the X.400 Recommendations, and implementor's groups have decided to use an IA5String for this field. There was some confusion about what the X.409 encoding for this IA5String would be, and the correct encoding is:

```
class:    context specific
form:     constructor
id code:  1
length:   length of contents
contents: (primitive encoding)
     IA5String:     16
     length:        length of contents
     contents:      the password string
class:    context specific
form:     constructor
id code:  1
length:   length of contents
contents: (constructor encoding) left as an exercise for the
          reader
```

Implementations should be prepared to receive any X.409 type for the password because of its definition as an ANY.

7.14.2 EXTRA SESSION FUNCTIONAL UNITS

One vendor proposed more than the required set of functional units on opening the session connection, and the receiver rejected the connection. All debate aside about whether the initiator should have proposed units outside of the required set, or whether the receiver should have rejected the connection, the set of functional units can be negotiated in a straightforward way. The following is recommended.

If the initiator proposes using more than the required set of functional units, the responder should specify the set of functional units that it would like to use (which should include the required set) in the open response. The session implementations will automatically use the intersection of the units proposed by both sides.

If the initiator proposes using less than the required set of functional units, the responder should reject the connection.

Unfortunately, there is not an appropriate RefuseReason for rejecting the connection, so instead of refusing the connection in the response to the S-CONNECT, the receiver should issue an S-U-ABORT with an AbortReason of protocolError. Note that it is valid to issue an S-U-ABORT instead of responding to the S-CONNECT. A problem report has been submitted to the CCITT requesting the addition of a RefuseReason for this situation.

If the responder proposes using less than the required set of functional units, the session connection is established before the initiator can check for this. If too few functional units have been proposed, the initiator should abort the connection using S-U-ABORT, with an abort reason of protocolError.

7.14.3 MIXED CASE IN THE MTA NAME

The MTA name is frequently exchanged over the telephone when two systems are being configured to communicate with one another. In one such telephone exchange, the casing of the MTA name was not specified, the MTA name consisted of both upper and lower case letters, and one of the implementations compared MTA names for equality in a case sensitive manner. Consequently, connections failed until the problem was detected and repaired. It is recommended that the MTA name be compared for equality in a case insensitive manner, and that the password be compared for equality in a case sensitive manner.

7.14.4 X.410 ACTIVITY IDENTIFIER

The X.400 Implementor's Guide recommends that the activity identifier be X.409 encoded, but this is only a recommendation and not a requirement. Consequently, receiving systems cannot assume that the activity identifier will be X.409 encoded.

7.14.5 ENCODING OF PER RECIPIENT FLAG AND PER MESSAGE FLAG

In the definition of the PerRecipientFlag in X.411, there is a statement that the last three bits are reserved, and should be set to zero. It is unclear whether those bits are unused in the X.409 encoding. Receivers should accept encodings with either zero or three unused bits. A problem report has been submitted to the CCITT asking for clarification.

Though there is not any statement in X.411 about the last four bits of the PerMessageFlag, some vendors have encoded this with zero unused bits, and some have encoded it with four unused bits. The PerMessageFlag should be encoded with at least four unused bits.

7.14.6 ENCODING OF EMPTY BITSTRINGS

There are three valid encodings for an empty bitstring: a constructor of length zero, a constructor of indefinite length followed by the end-of-contents terminator, and a primitive of length one with a zero octet as the value.

7.14.7 ADDITIONAL OCTETS FOR BITSTRINGS

Nothing in X.409 constrains an implementation from sending two, three, four, or even more octets for a bitstring that fits into one octet, with the undefined bits set to zero. Note that the number of excess octets is bounded by the pragmatic constraints guidelines of the CCITT X.400 Implementor's Guide for all of the bitstrings in P1.

7.14.8 APPLICATION PROTOCOL IDENTIFIER

If a value other that 1 is received in the applicationProtocol of the pUserData in the PConnect, NIST implementations will reject the connection. If CEN/CENELEC implementations receive a value other than 8883 for this field, they will reject the connection. This is an unfortunate state of affairs, because if NIST implementations accept the value of 8883 without supporting the MOTIS service elements, they would be misrepresenting themselves. To make matters worse, CEPT uses a value of 1, but relays MOTIS elements, which means that MOTIS elements will be relayed to implementations using a value of 1 to demonstrate that they do not support MOTIS. Work is continuing to try to find a solution that will allow European implementations to interwork with U.S implementations.

7.14.9 INITIAL SERIAL NUMBER IN S-CONNECT

This should be implemented in accordance with Section 3.5.1 E4 of the Implementors' Guide.

7.14.10 CONNECTION DATA ON RTS RECOVERY

It is clarified that the ConnectionData is identical in both the S-CONNECT.request and the S-CONNECT.response. The value of the ConnectionData is the old Session Connection Identifier.

7.14.11 ACTIVITY RESUME

If an activity is being resumed on a new session connection, it is not clear from X.410 and X.225 whether all four of the called-ss-user reference, the calling-ss-user reference, the common reference, and the additional reference information should be specified in the S-ACTIVITY-RESUME, or whether one of the ss-user-references should be absent. It is also unclear whether the called-ss-user reference should be identical to the calling-ss-user reference if both are present. Consequently, receivers should be tolerant of this situation. Appropriate problem reports will be submitted to the CCITT asking for clarification.

7.14.12 OLD ACTIVITY IDENTIFIER

The Old Activity Identifier in S-ACTIVITY-RESUME refers to the original activity identifier.

7.14.13 NEGOTIATION DOWN TO TRANSPORT CLASS 0

For European implementations, X.410 specifies that class 0 transport must be supported. However, it is permissible for an initiator to propose a higher class as the preferred class, provided that class 0 appears as the alternate class in the T-Connect PDU. A responding implementation can choose to use either the preferred or alternate class, but again, must be able to use class 0. In other words, for private to private connections in Europe, class 0 transport is required.

This conflicts with the NIST agreements, since class 0 is only required if one of the partners in a connection is an ADMD.

7.15 APPENDIX E: WORLDWIDE X.400 CONFORMANCE PROFILE MATRIX

Y CONFORMANCE (E)
implies a conformance problem for European products in the U.S.

Y CONFORMANCE (US)
implies a conformance problem for U.S. products in Europe.

- The A/311 profile is specified in Env 41 202, the A/3211 profile in Env 41 201

- No TTC protocol classification for RTS exists.

- The notation X/Y indicates "X" for PRMDs and "Y" for ADMDs, i.e. "M/G" would be Mandatory for PRMDs and Generatable for ADMDs.

Table 7E.1 Protocol element comparison of RTS

RTS element	NIST	A/311	A/3211	PROBLEM Y/N
PConnect	M	M	M	N
DataTransferSyntax	M 0	M 0	M 0	N
PUserData	M	M	M	N
checkpointSize	H	H	H	N
windowSize	H	H	H	N
dialogueMode	H	H	H	N
connectdata	M	M	M	N
applicationProtocol	G 1 H 8883	H 1	R 8883	N
ConnectionData				
Open	G	G	G	N
Recover	G	H	G	N
Open				
RTSUserData	G	G	G	N
Recover				
SessionConnectionID	G	G	G	N
RTSUserData				
MTAName	G	G	G	N
Password	G	G	G	N
null	G	G	G	N
SessionConnectionID				
CallingUserReference	M	M	M	N
CommonReference	M	M	M	N
AdditionalRefInfo	H	H	H	N
PAccept	G	G	G	N
DataTransferSyntax	M 0	M 0	M 0	N

(Continued on next page.)

Table 7E.1 Protocol element comparison of RTS, continued

RTS element	NIST	A/311	A/3211	PROBLEM (Y/N)
PUserData	M	M	M	N
CheckpointSize	H	H	H	N
WindowSize	H	H	H	N
ConnectionData	M	M	M	N
PRefuse	G	G	G	N
RefuseReason	M	M	M	N
SSUserData (in S-TOKEN-PLEASE)	G	G	G	N
AbortInformation (in S-U-ABORT)	G	G	G	N
AbortReason	H	H	H	N
reflectedParameter	X	X	X	N

Table 7E.2 Protocol element comparison of P1

P1 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
ORname					
StandardAttributeList	M	M	M	M	N See Note 4
DomainDefAttributeList	X	X	X	G	Y See Note 5
StandardAttributeList					
CountryName	R	R	R	M	N
		ISO R	R		N
		X.121 H	H		Y Conformance (E)
		Other X	X		Y Prot Vio
AdministrationDomainName	R	R	G	M	N
... if PrintableString		R	G		N
... if numericString		H	H		Y Conformance (E)
X.121 Address	X	X/R	X		Y Conf(US)See Note 1
Terminal ID	X	X/G	X		Y Conf(US)See Note 1
PrivateDomainName	G	G	G	G	N
OrganizationName	G	G	G	G	N
UniqueUAidentifier	X	X/G	X		Y Conf(US)See Note 1
PersonalName	G	G	G	G	N
OrganizationalUnit	G	G	G	G	N
DomainDefinedAttribute	X	X	X	G	N
Type	M	M	M	M	N
Value	M	M	M	M	N
PersonalName					
Surname	M	M	M	M	N
GivenName	G	G	G	G	N
Initials	G	G	G	G	N
GenerationQualifier	G	X	X	X	Y Conformance (E)
GlobalDomainIdentifier					
CountryName	M	M	M	M	N
AdministrationDomainName	M	M	G	M	Y Proto Vio
PrivateDomainIdentifier	R/H	H	R	M/X	N
MPDU					
UserMPDU	G	G	G	G	Y TTC required MPDU size is 32K
DeliveryReportMPDU	G	G	G	G	N
ProbeMPDU	H	H	H	H	N

(Continued on next page

Table 7E.2 Protocol element comparison of P1, continued

P1 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
UserMPDU					
UMPDUenvelope	M	M	M	M	N
UMPDUcontent	M	M	M	M	N
UMPDUenvelope					
MPDUidentifier	M	M	M	M	N
originatorORname	M	M	M	M	N
originalEncodedTypes	G	H	H	G	Y Conformance (E)
ContentType	M	M	M	M	N
UAcontentID	H	H	H	H	N
Priority	G	G	G	G	N
PerMessageFlag	G	G	G	G	N
DeferredDelivery	X	X	X	X	N
PerDomainBilatInfo	X	X	X	X	N
RecipientInfo	M	M	M	M	Y TTC MPDU 32K
TraceInformation	M	M	M	M	N
MOTIS-> LatestDelivery			X		N
MOTIS-> InternalTraceInfo	M/P		P		N
UMPDUcontent	M	M	M	M	N
MPDUidentifier					
GlobalDomainIdent	M	M	M	M	N
IA5string	M	M	M	M	N
PerMessageFlag					
DiscloseRecipients	H	G @ MTL	H	H	Y Conformance (US)
		H at UA	?		Y Conformance (US)
ConversionProhibited	G	G	G	G	N
AlternatRecipAllowed	H	G @ MTL	H	X	Y Conformance (US)
		H at UA	?		Y Conformance (US)
ContentReturnRequest	X	X	X	X	
MOTIS-> redirectionProhibited			X		N
PerDomainBilateralInfo					
CountryName	M	M	M	M	N
AdminDomainName	M	M	G	M	Y Prot Vio
MOTIS-> PrivateDomainName			G		N
BilateralInfo	M	M	M	M	N

(Continued on next page

Table 7E.2 Protocol element comparison of P1, continued

P1 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
DeliveryReportContent					
original MPDUident	M	M	M	M	N
intermediate Trace	X/G	X	X	X	Y Conformance (E)
UAcontentID	G	G	G	G	N
ReportedRecipientInfo	M	M	M	M	Y TTC 256 max
returned	H	H	X	X	Y Conformance (E)
billing information	X	X	X	X	N
ReportedRecipientInfo					
recipient ORname	M	M	M	M	N
extensionsIdentifier	M	M	M	M	N
PerRecipientFlag	M	M	M	M	N
LastTraceInformation	M	M	M	M	N
intendedRecipient	H	H	H	H	N
SupplementaryInfo	X/H	X	X	X	Y Conformance (E)
MOTIS-> ReassignmentInfo			X		N
MOTIS-> ReassignmentInfo					
MOTIS-> intendedRecipient			M		N
MOTIS-> reasonForReassignment			H		N
LastTraceInformation					
arrival	M	M	M	M	N
convertedEncInfoTypes	G	G	H	G	Y Conformance (E)
Report	M	M	M	M	N
Report					
DeliveredInfo	G	G	G	M	N See Note 6
NonDeliveredInfo	G	G	G		N
DeliveredInfo					
delivery	M	M	M	M	N
TypeofUA	R/H	H	R	M/G	N
NonDeliveredInfo					
ReasonCode	M	M	M	M	N
DiagnosticCode	H	H	H	H	N
MOTIS-> UAprofileIdentifier			X		N
MOTIS-> UAprofileIdentifier					
MOTIS-> ContentType			M		N
MOTIS-> EncodedInfoTypes			M		N

(Continued on next page

Table 7E.2 Protocol element comparison of P1, continued

P1 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
ProbeEnvelope					
probe	M	M	M	M	N
originator	M	M	M	M	N
ContentType	M	M	M	M	N
UAcontentID	H	H	H	H	N
originalEncInfoTypes	G	H	H	G	Y Conformance (E)
TraceInformation	M	M	M	M	N
PerMessageFlag	G	G	G	G	N
ContentLength	H	H	H	H	N
PerDomainBilatInfo	X	X	X	X	N
RecipientInfo	M	M	M	M	Y TTC 256 max
MOTIS-> InternalTraceInfo	M/P		P		N
RecipientInfo					
RecipientORname	M	M	M	M	N
ExtensionIdentifier	M	M	M	M	N
PerRecipientFlag	M	M	M	M	N
ExplicitConversion	X	X	X	X	N
MOTIS-> OriginatorReqAlternatRecip			X		N
MOTIS-> ReassignmentInfo			X		N
PerRecipientFlag					
ResponsibilityFlag	M	M	M	M	N
ReportRequest	M	M	M	M	N
UserReportRequest	M	M	M	M	N
TraceInformation					
GlobalDomainIdent	M	M	M	M	N
DomainSuppliedInfo	M	M	M	M	N

(Continued on next page

Table 7E.2 Protocol element comparison of P1, continued

P1 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
DomainSuppliedInfo					
arrival	M	M	M	M	N
deferred	X	X	X	X	N
action	M	M	M	M	N
(0=relayed)	G	G	G		N Note: Re-routing not required.
(1=rerouted)	H	H	H		N
MOTIS-> (2=recipientReassigned)			H		N
converted	H	G	H	H	Y Conformance(US)
previous	H	G	G	X	Y Conformance(US) (Note: G is inconsistent with action (relayed) being "H".)
ORname					
EncodedInformationTypes					
BitString	M	M	M	M	N See Note 3
G3NonBasicParameters	X	X	X	X	N
TeletexNonBasicParams	X	R	X	X	Y Conformance(US)
PresentationAbilities	X	X	X	X	N
DeliveryReportMPDU	G	G	M	G	N
DeliveryReportEnvelop	M	M	M	M	N
DeliveryReportContent	M	M	M	M	N
DeliveryReportEnvelope					
report	M	M	M	M	N
originator ORname	M	M	M	M	N
TraceInformation	M	M	M	M	N
InternalTraceInfo	M/P		P		N

(Continued on next page

Table 7E.3 Protocol element comparison of P2

P2 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
UAPDU					
IM_UAPDU	G	G	G	G	N
SR_UAPDU	X	X	X	X	N
IM_UAPDU					
Heading	M	M	M	M	N
Body	M	M	M	M	N
Heading					
IPmessageID	M	M	M	M	N
Originator ORname	R	R	R	M/G	N
AuthorizingUsers	H	H	H	H	Y TTC 16 max
PrimaryRecipients	G	G	G	G	Y TTC 256 max
CopyRecipients	G	G	G	G	Y TTC 256 max
BlindCopyRecipients	H	H	H	H	Y TTC 256 max
InReplyTo	G	G	G	G	N
Obsoletes	H	H	H	H	Y TTC 8 max
CrossReferences	H	H	H	H	Y TTC 8 max
Subject	G	G	G	G	N
ExpiryDate	H	H	H	H	N
ReplyBy	H	H	H	H	N
ReplyToUsers	H	H	H	H	Y TTC 32 max
Importance	H	H	H	H	N
Sensitivity	H	H	H	H	N
Autoforwarded	H	H	H	H	N
MOTIS-> CirculationList			X		N
MOTIS-> ObsoletingTime			X		N
IPmessageID					
ORname	H	H	H	H	N
PrintableString	M	M	M	M	N
ORdescriptor					
ORname	H	H	H	M	N See Note 6
FreeFormName	H	H	H		N
TelephoneNumber	H	H	H	G	N
Recipient					
ORdescriptor	M	M	M	M	N
ReportRequest	X	X	X	X	N
ReplyRequest	H	H	H	H	N

(Continued on next page

Table 7E.3 Protocol element comparison of P2, continued

P2 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
MOTIS-> CirculationList					
MOTIS-> CirculationMember			X		N
MOTIS-> checkmark			M		N
MOTIS-> membername			M		N
MOTIS-> OBsoletingTime					
MOTIS-> Time			H		N
MOTIS-> IP_MessageID			H		N
Body					
BodyPart	G	M	M	G	Y Conformance (US)
SR_UAPDU					
NonReceipt	H	H	H	M	N
Receipt	H	H	H		N
Reported	M	M	M	M	N
ActualRecipient	R	R	R	G	N
IntendedRecipient	H	H	H	H	N
Converted	X	X	X	G	N
MOTIS-> CirculationStatus			X		N
NonReceiptInformation					
Reason	M	M	M	M	N
NonReceiptQualifier	H	H	H	H	N
=expired (value)	0	0	0	0	N
=obsoleted (value)	1	1	1	1	N
=subscriptionTerminated	2	2	2	2	N
MOTIS-> =timeobsoleted (value)			X		N
Comments	H	H	H	X	N
returned	H	X	X	X	Y Conformance (E)
ReceiptInformation					
Receipt	M	M	M	M	N
TypeOfReceipt	H	H	H	G	N
SupplementaryInfo	X	X	X	X	N

(Continued on next page

Table 7E.3 Protocol element comparison of P2, continued

P2 Protocol	NIST	A/311	A/3211	TTC	PROBLEM (Y/N)
BODYPART SUPPORT					
o IA5 Text	G	G	G		N See Note 7
o TLX	X	X	X		N
o Voice	X	X	X		N
o G3FAX	X	X	X		N
o TIFO	X	X	X		N
o TTX	X	X/H	X		Y Conf(US)See Note 2
o VideoTex	X	X	X		N
o NationallyDefined	X	X	X		N
o Encrypted	X	X	X		N
o ForwardedIPmessage	H	H	H		N
o SFD	X	X	X		N
o TIFI	X	X	X		N
MOTIS-> o ODA			X		N
MOTIS-> o ISO6937 Text			H		N

Note 1: It should be noted that the A/311 profile states: For routing all ADMDs should support all Form 1 Variants of O/R Name. All PRMDs should support at least Form 1, Variant 1 form of OR Name.

Note 2: It should also be noted that the A/311 profile requires that all ADMDs should support the reception of Teletex body parts for delivery to their own UAEs.

Note 3: An A/3211 implementation may generate MOTIS encoded information types. See 7.6.11.

Note 4: Only Form 1 Variant 1 of O/Rname shown for TTC, but TTC defines other forms and variants. Form 1 Variant 1 recommended for PRMDs and ADMDs, Form 1 Variant 2 also recommended for ADMDs.

Note 5: DDA's can be used to specify recipients in any Japanese domains other than TTC. Assignment of DDAs for UAs within TTC domains is not recommended.

Note 6: One of [DeliveredInfo/NonDeliveredInfo] must be present. TTC encodes this as shown. Other profiles represent this by classifying both protocol elements as generatable. A similar situation exists with the P2 ORdescriptor.

Note 7: TTC is expected to support IA5 for some international MHS communications.

7.16 APPENDIX F: INTERWORKING WARNINGS

ADMD name is to be encoded as a single space when configurations with no ADMD's are present. It should be noted that this may change in January 1988 so that the ADMD name is encoded as a zero length element in such cases.

The NIST agreements allow implementation to generate MPDUs with no body parts. Such MPDUs will be rejected by European-conformant systems. (Note this situation may change in January 1988)

In order to optimize the number of recipients you can read and reply to, it is advisable to be able to generate all standard O/R name attributes.

8. FUTURE MESSAGE HANDLING SYSTEMS

Editor's Note: This Section is reserved for future Stable Message Handling Systems text based upon the 1988 CCITT recommendations. As this material is declared stable, it will be moved from the aligned section in the Ongoing Agreements Document to the same section in this stable document. Consult Section 7 in the Ongoing Agreements document for more information on this subject.

9. ISO FILE TRANSFER, ACCESS AND MANAGEMENT PHASE 2

Editor's Note: In document type names, constraint set names, and abstract syntax definitions, the "NBS" designation will be preserved.

9.1 INTRODUCTION

This section defines Implementors' Agreements based on ISO File Transfer, Access and Management (FTAM), as defined in ISO 8571. This International Standard has four parts. Part 1 of the IS gives general concepts, Part 2 defines the Virtual Filestore (VFS), Part 3 defines the File Service, and Part 4 defines the File Protocol.

FTAM, as described in the IS, is based on the following ISO documents: ACSE Service and Protocol (ISO 8649, ISO 8650), Presentation Service and Protocol (ISO 8822, ISO 8823), ASN.1 Abstract Syntax Notation and Basic Encoding Rules (ISO 8824, ISO 8825), and Session Service and Protocol (ISO 8326, ISO 8327). These services and protocols are defined architecturally in the OSI Reference Model (ISO 7498). These Agreements provide detailed guidance for the implementor, and eliminate ambiguities in interpretations.

The general agreements reached with respect to the ISO File Transfer, Access and Management Protocol (FTAM) are that the Phase 2 FTAM specification (this section) is based on the International Standard (IS).

9.2 SCOPE AND FIELD OF APPLICATION

These FTAM Phase 2 Agreements cover transfer of and access to files between the Filestores of two end systems, including the management of a Virtual Filestore. One end system acts in the Initiator role and initiates the file transfer/access, while the other end system acts in the Responder role and provides access to the file in the Virtual Filestore. This paper describes Agreements for the actions and attributes of the Virtual Filestore, and the service provided by the file service provider to file service users, together with the necessary communications between the Initiator and Responder.

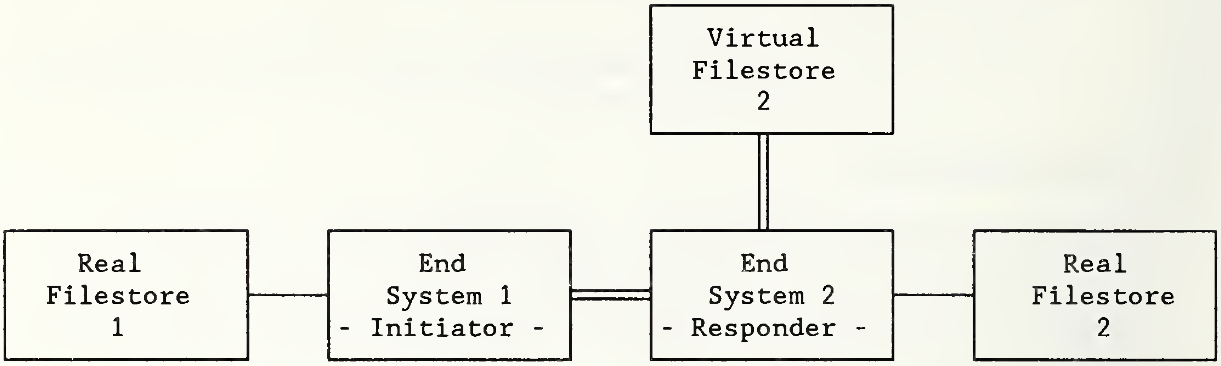


Figure 9.1 Model of file transfer/access

Note: Agreements apply on the double lines of Figure 1. The mapping between the Virtual Filestore and the Real Filestore together with the local data management system is not part of these Agreements.

These Agreements define General Agreements in Section 9.5 through 9.16, minimum functionality (Conformant Implementations) in Section 9.17, and functionality for several Implementation Profiles which are tailored to different classes of user requirements in Sections 9.18 and 9.19.

9.3 STATUS

This version of the FTAM Implementation Agreements was completed December 16, 1988. No further enhancements will be made to this version. See the next section, ERRATA.

Note: These Agreements were updated from the previous March 1987 DIS based Agreements.

9.4 ERRATA

Editor's Note: All changes over the past year have been included directly into the text. For specific descriptions of the errata, consult the NIST Workshop editor.

9.5 ASSUMPTIONS

1. FTAM protocol machines must be able to parse and process at a minimum 7K octets of FTAM PCI, FTAM structuring (FTAM-FADU) and FTAM user data (including grouped FPDUs) as they would be encoded with the ASN.1 Basic Encoding Rules. It is recommended, however, that Presentation user data not be restricted in size.

2. In order to maximize interoperability, it is important that the implementations of FTAM service providers do not unnecessarily restrict the service user's ability to generate arbitrary file service requests. Otherwise, they may not be able to work with FTAM Responders whose operation is constrained by their mapping of the FTAM Virtual Filestore to their local filestore. For example, error procedures should only be invoked when an error actually occurs, not at the point of the specification of options which might result in an error.

3. Implementations must be able to parse all valid optional parameters if they are present in the PDU. Only those optional parameters specified as supported in these Agreements are required to be implemented. If these parameters are not present, a default value is assigned locally. A Responder should not refuse a request solely because a parameter that is optional in the FTAM standard, but is supported in these Agreements, is not present.

4. Consideration of any standardized service interface is not covered by these Agreements.

5. These Agreements define no restrictions for the values used for the <communication quality of service> parameter in <F-INITIALIZE>.

6. FTAM is defined in phases. The Phase 1 FTAM implementation specification is based on the second ISO Draft Proposal, dated April 1985, and the ISO Draft Proposal 8824 and 8825.

   The Phase 2 FTAM specification (this section) is based on the International Standard (IS). THERE IS NO BACKWARD COMPATIBILITY WITH NIST FTAM PHASE 1. Backward compatibility is impossible, since Phase 1 uses Session services directly, while Phase 2 uses ACSE and Presentation services. Furthermore, there are differences in Filestore, PDU Abstract Syntax, FADU Abstract

Syntax, and Transfer Syntax. There also are differences in the transparency mechanisms and service class negotiations.

The <implementation information> parameter of <F-INITIALIZE> FPDU as defined in ISO 8571-4, 20.3 is used to pass 'user version' information with respect to different FTAM phases of the NIST Implementors Agreements or with respect to FTAM profiles of other bodies (see Section 9.12 of this document). It is the goal of these Agreements to use the 'user version' mechanism to provide at least one level of backward compatibility for all future NIST FTAM Phases, facilitating backward compatibility for future FTAM products, assuming different new versions of the respective IS's also enable backward compatibility.

7. Section 5.11.1.1.1 of this document defines a value (that carries no semantics) for the AETitle that can be used by FTAM ASEs for communication. Other values for the AETitle are outside the scope of these Agreements.

   For the Called-AETitle, Calling-AETitle and Responding AETitle the association shall not be rejected/aborted if the value specified in 5.11.1.1.1 is sent or if any of the parameters are not sent. The association may be rejected/aborted if a value other than specified in 5.11.1.1.1 is sent.

   Use of values outside the scope of these Agreements is discouraged until agreed upon semantics have been associated with AETitles.

8. Use of <shared ASE information> parameter and <charging> parameter is not defined within the scope of the Agreements.

9. Use of <application context name> parameter is not defined within the scope of these Agreements. This parameter does not prohibit the establishment of an FTAM association.

10. These Agreements use the term 'supported' for a parameter to mean that the syntax and semantics of that parameter shall be implemented. However, it is not a requirement that the parameter be used in all instances of communication, unless stated otherwise.

    Also these Agreements use the term 'optionally supported' for a parameter to mean that it is left to the implementation whether the semantics of that parameter are implemented or not.

9.6 PRESENTATION AGREEMENTS

The following Abstract Syntaxes are recognized in these agreements:

"FTAM FADU"
"FTAM PCI"
"FTAM unstructured text abstract syntax"
"FTAM unstructured binary abstract syntax"
"NBS abstract syntax AS1"
"NBS file directory entry abstract syntax"

The following Transfer Syntax is supported:

"Basic Encoding of a single ASN.1 type"
(See Appendix A, Part 3)

9.7 SERVICE CLASS AGREEMENTS

Implementation Agreements have been reached for the following service classes.

- File Transfer
- File Access
- File Management
- File Transfer and Management
- Unconstrained

9.8 FUNCTIONAL UNIT AGREEMENTS

Implementation agreements have been reached for the following functional units.

- Kernel
- Read
- Write
- File Access
- Limited File Management
- Enhanced File Management
- Grouping

Implementation of the Recovery, Restart Data Transfer, and FADU Locking functional units is not specified.

9.9 FILE ATTRIBUTE AGREEMENTS

Implementation of the Kernel Group of file attributes is defined. If the optional Storage Group and Security Group are implemented, aspects of their implementation are defined. Implementation of the Private Group is not specified.

Responses to an attribute value request shall always include one of the following (as specified in ISO 8571-2, Clause 9.4):

- An actual file attribute value.
- A value indicating that no value is available, optionally with a diagnostic.
- No value and an error code, optionally with a diagnostic indicating that the attribute is not supported.

9.9.1 Mandatory Group

Only the Kernel Group of attributes is required. A value for <filename>, <permitted actions>, and <contents type> will always be available.

A minimum range is required for <filename> values as specified in ISO 8571-2. No maximum length or format restrictions apply. A system that does not support <filename> values with a sequence of more than one Graphic String or extended <filename> characteristics may reject a request involving such a <filename>. All systems must be able to interpret a <filename> value with a sequence of one Graphic String. Requests using such a single component <filename> value with a sequence of one Graphic String are responded to using a single component <filename> value. Responses to requests involving <filename> values having two or more Graphic Strings are not defined here but may be interpreted via bilateral or other external agreements. Use of <filename> values with a sequence of more than one Graphic String is discouraged.

Apart from the minimum conformance requirements specified in ISO 8571-2, file names have to be specified in the naming convention of the responding FTAM implementation. It is a local implementation matter of the FTAM Responder, whether or not an additional name mapping onto the real FIlestore's file name convention is supported.

In order to enable interworking with all FTAM Responders' virtual Filestores, it is recommended that FTAM Initiators impose no restrictions on the attribute range supported for file names beyond those specified in ISO 8571-2.

For the purpose of interworking according to these Agreements the <contents type> attribute is limited to the <document type name> format. The <constraint set name, abstract syntax name> form is outside the scope of these Agreements. It should always be parsed correctly when received, but may result in an error.

9.9.2 Optional Groups

If the optional Security Group of file attributes is implemented, an actual value must be available for the <access control> attribute.

The <access control> attribute is a SET OF <access control element>. The minimum requirement in these Agreements is the support of one <access control element>, according to the base standard. The terms <concurrency access>, <identity>, and <passwords> are each optionally supported. Details of their use shall be specified in the PICS. Use of the <location> term is not specified in these Agreements.

Implementation of the Private Group is not specified.

9.10 DOCUMENT TYPE AGREEMENTS

These document types are defined.

FTAM-1	"ISO FTAM unstructured text"
FTAM-2	"ISO FTAM sequential text"
FTAM-3	"ISO FTAM unstructured binary"
NBS-6	"NBS-6 FTAM sequential file"
NBS-7	"NBS-7 FTAM random access file"
NBS-8	"NBS-8 FTAM indexed file"
NBS-9	"NBS-9 FTAM file directory file"

Detailed document type definitions are given in Appendix 6A and in ISO 8571-2, Annex B.

Note: Document types NBS-1 to NBS-5 are not defined in these Agreements. The numbering starts with NBS-6 because of the original DIS version of these Agreements.

An implementation claiming conformance to these Agreements which also supports any or all of the document types FTAM-1, FTAM-2, and FTAM-3 as defined in ISO 8571-2, Annex B, must minimally support the combinations of parameter values as specified in Table 9.1.

Table 9.1 Parameters for FTAM-1, -2, -3

	Universal Class Number	Maximum String Length[6]	String Significance
FTAM-1	General String [1](27) IA5String [2](22)	134 or less	'not-significant'
FTAM-2	Graphic String [3,4](25)	134 or less[5]	'not-significant'
FTAM-3	<not applicable>	512 or less	'not-significant[4]

Notes:

1. The minimum level of support for General String is the IA5 G0 character set and the 8859-1 G0 and G1 character sets, and IA5 C0 set.

2. The support for IA5 String is the IA5 G0 character set and the IA5 C0 set.

3. The minimum level of support for Graphic String is the IA5 G0 character set and the 8859-1 G0 and G1 sets.

4. This is the default when the parameter is not specified.

5. The implementation need not support Data Units whose total character count exceeds 134.

6. As per Table 9.3.

For document types which use the sequential flat constraint set, conformant implementations must minimally support FADU identities as follows:

o for Transfer service class: 'begin', 'end'

o for Transfer and Management service class: 'begin', 'end'

o for Access service class: 'begin', 'end', 'first', 'next'

For the document types NBS-6, NBS-7 and NBS-8 parameters are used for which the Agreements apply as specified in Table 9.2.

Table 9.2 Parameters for NBS-6, NBS-7, NBS-8

Parameter	PrimType	String-length	Length-1	Length-2
int	INTEGER	Number of octets required to represent, in 2's complement format, the largest integer to be passed		
bit	BIT STRING	Number of bits in string (non-varying)		
ia5	IA5String	Max number of characters in string		
graphic	Graphic String	Max number of characters in string		
general	General String	Max number of characters in string		
octet	OCTET STRING	Max numbers of octets in string		
private-class-number	Floating Point Number		The minimum number of bits required to be maintained in the mantissa for relative precision	Number of bits required to represent the largest unbiased integer exponent in 2's complement
univer-time	UTCTime	<not applicable>		
gen-time	Generalized Time	<not applicable>		
boolean	BOOLEAN	<not applicable>		
null	NULL	<not applicable>		

Note: The string length parameter specifies the actual number of from the referenced character set. It does not include any escape sequences or overhead from the encoding.

The primitive data types and minimal size ranges that an implementation must accept for storage are given in Table 9.3.

Table 9.3 FTAM primitive data types

Primitive Data Type		Minimum Range (Octets)
ASN.1	INTEGER	1 - 2
ASN.1	BIT STRING	0 - 1
ASN.1	IA5String	0 - 134
ASN.1	GeneralString	0 - 134
ASN.1	GraphicString	0 - 134
ASN.1	OCTET STRING	0 - 512
ASN.1	BOOLEAN	
ASN.1	NULL	
ASN.1	GeneralizedTime	
ASN.1	UniversalTime	
NBS-AS1	FloatingPointNumber	mantissa 1-23 bits exponent 0-8 bits

Notes:

1. The primitive data types and their maximum ranges for a specific file as described by the parameters above are maintained in the <contents type> file attribute. The <contents type> file attribute value is established at the file's creation and cannot be changed via FTAM for the life of the file. This implies that the data element types and ranges and data unit formats are fixed for all accessors of that file as long as the file exists.

2. The syntax for floating point numbers is part of the definition of NBS abstract syntax AS1 in Annex 9A Part 3. It is derived from existing standards IEC 559 and IEEE 754.

9.10.1 Character Sets

Implementation of a character set in FTAM is understood as:

- a transfer syntax is defined for the character set

- document types are defined using the character set in their abstract syntactic definition

- documents of those types are stored in the Virtual File Store as defined in the character set specification.

> They are written into the VFS and read from the VFS as defined by the abstract syntax and the transfer syntax for the document type. It is not in the scope of FTAM Agreements to specify the local representation of those documents in the Real Filestore, nor to specify rendition of graphic characters or control characters on character imaging devices. These renditions are possible agreements between applications using FTAM for their communication.

The character sets IA5 and ISO 8859-1 shall always be implemented.

9.10.1.1 IA5 Character Set

The International Reference Version (IRV) of IA5 is available for use when there is no requirement to use a national or an application-oriented version. In information interchange, the IRV is assumed unless a particular agreement exists between sender and receiver of the data. The graphic characters allocated to the IRV are as specified in Table 9.4.

Table 9.4. IRV Graphic Character Allocations

Graphic	Name	Coded Representation
#	Number sign	2/3
o	Currency sign	2/4
@	Commercial at	4/0
[	Left square bracket	5/11
\	Reverse solidus	5/12
]	Right square bracket	5/13
^	Circumflex accent	5/14
'	Grave accent	6/0
{	Left curly bracket	7/11
|	Vertical line	7/12
}	Right curly bracket	7/13
~	Tilde, overline	7/14

It should be noted that no substitution is allowed when using the IRV and that the facility of combined vertical and horizontal movements of the active position does not apply to any format effectors.

It is permitted to use composite graphic characters and there is no limit to their number. Because of this freedom, their processing and imaging may cause difficulties at the receiving end. Therefore agreement between sender and receiver of the data is recommended if composite characters are used.

Note: Attention is drawn to the fact that different national character sets exist.

(See ISO 646 or CCITT Recommendation T.50 for more information)

9.10.1.2 Format Effectors

Implementations conforming to these Agreements shall not use a single format effector for affecting a combined vertical and horizontal movement.

Notes:

1. This is discouraged by ISO 646-1983, Clauses 4.1.22 and 6.4; and by ISO 6429-1983, Clauses 7.2.5 and E.3. It is disallowed by ISO 4873-1986, Clause A.3.2.

2. The Agreements require only support of CO control characters of ISO 646, containing among others the format effectors <CR> and <LF>. It is recommended that NIST implementations use <CR> <LF> pairs as line terminators.

9.10.1.3 8859-1 Character Set

The Latin Alphabet No.1 (ISO 8859-1) is used to specify the printable characters of G0 and G1. C0 control characters and their associated rules are taken from the IA5 definition.

9.10.2 Document Type Negotiation Rules

9.10.2.1 Connection Establishment

In connection establishment the <contents type list> parameter is used only to establish presentation contexts. Both the <document type name> form and the <abstract syntax name> form are supported.

9.10.2.2 File Creation

An <F-CREATE request> FPDU must contain a <document type name> value in its <initial attributes> parameter.

If the specified document type requires parameterization, then these parameters must be supplied, otherwise the <F-CREATE request> may be rejected.

Notes:

1. It is understood that <permitted actions> sub-field of <initial attributes> parameter

will always be used at <F-CREATE request>. The value may be changed by the Responder.

2. If the <document type name> used requires DU syntax parameters and one of the parameters specifies 'FloatingPointNumber' as a primitive data type, the request may be rejected, in case the optional type 'FloatingPointNumber' is not supported by the Responder.

9.10.2.3 File Opening

The <document type name> form (with appropriate parameters as specified in 8871-2, Clause 12.3) shall always be used when proposing a <contents type>; as an alternative the 'ContentsTypeUnknown' value may be used in the <F-OPEN request>. An <F-OPEN response> shall use the <document type name> option (with appropriate parameters) in the <contents type> field.

This allows the receiving entity to use the <document type name> attributed to the file instead of receiving a <constraint set name> and <abstract syntax name> pair, which does not reflect the file information contained in the FTAM and NIST document types.

This document type name is either a value from the set of base document type names as negotiated upon connection establishment or a document type name, for which an appropriate presentation context was established.

Notes:

1. An <F-OPEN response> without a <document type name> (but carrying the <constraint set name> and <abstract syntax name> form) may cause the Initiator to issue an <F-CLOSE request>.

2. If the <document type name> used requires DU syntax parameters and one of the parameters specifies 'FloatingPointNumber' as a primitive data type, the request may be rejected, in case the optional type 'FloatingPointNumber' is not supported by the Responder.

9.10.3 Relationship Between DUs, DEs and Document Types

"Abstract Syntax" is used to refer to the syntactic information which is architecturally passed between the Application and Presentation Layers. The Abstract Syntax defines Data Element (DE) types which are not necessarily ASN.1 primitive types. A Data Element (DE) is the smallest piece of data whose identity is necessarily preserved by the Presentation Service. Data types may be made up of other data types. Data Elements are not defined in terms of other Data Elements.

A Data Unit (DU) is a sequence of one or more Data Elements. Architecturally, entire, single DEs are passed into and out of the application process. In a real implementation, DUs may be passed.

To maintain DU boundaries during transfer, file structuring information must be passed (ISO8571-FADU definition in ISO 8571-2, Clause 7.5). A Data Element is referred to as a File-Contents- Data-Element in the ISO8571-FADU definition.

Document types refer to aspects of local processing and storage. They describe:

- structural relationship between DUs,
- structure of DUs, called DU syntax, and
- DE types found in the file.

Because document types pertain to local processing and storage, the DU syntax makes assertions about the syntax and the size of DUs (records) in storage. Parameters on the document types provide this information about the syntax and size of the DUs.

9.11 F-CANCEL ACTION

When an F-CANCEL is sent or received, the following occurs:

- no more data is sent,
- checkpoint numbers are removed, and
- state of the file is implementation dependent.

Note: When mapping F-CANCEL on P-RESYNCHRONIZE (abandon) it is required that P-SYNC-MINOR be used after F-READ/F-WRITE (see ISO 8571-4 Clauses 13, 14).

9.12 IMPLEMENTATION INFORMATION AGREEMENTS

- The <implementation information> parameter of <F-INITIALIZE> FPDU is not required by these Agreements.

- It may be used to pass user version information as a series of values, separated by ';'.

- The following will indicate conformance to the NIST Phase 2 Agreements: NBS-Phase2.

  Note: The list of possible values may be enlarged for future FTAM phases or FTAM profiles of other bodies.

- This parameter is for information only; it is not used for negotiation.

  The establishment of an FTAM regime should not be rejected only because of an unknown <implementation information> value.

9.13 DIAGNOSTIC AGREEMENTS

- The <diagnostic> parameter is supported; a value in the <response> PDU is needed when the <action result> or <state result> is not zero. (The nature of these agreements is to provide <diagnostic> information when any result parameter is not 'success'.)

- General catch-all diagnostic action is discouraged.

- The <further details> subfield is supported. It will be encoded as GraphicString, but is restricted to IA5 (IRV, graphic characters) and ISO 8859-1 only.

- Use of F-P-ABORT for other than protocol errors and catastrophic situations is discouraged.

- When returning an error status in a file management related diagnostic (i.e., <F-READ-ATTRIBUTE response> or <F-CHANGE-ATTRIBUTE response>), identify the erroneous attribute by using the first two characters of <further details> to hold a 2-digit number (encoded as IA5String) from the <F-READ-ATTRIBUTE request> attributes abstract syntax definition (ISO 8571-4, Clause 20.3).

00	Filename
01	Permitted Actions
02	Contents Type
03	Storage Account
04	Date and Time of Creation
05	Date and Time of Last Modification
06	Date and Time of Last Read Access
07	Date and Time of Last Attribute Modification
08	Identity of Creator
09	Identity of Last Modifier
10	Identity of Last Reader
11	Identity of Last Attribute Modifier
12	File Availability
13	File Size
14	Future Filesize
15	Access Control
16	Legal Qualifications
17	Private Use

The set of file management diagnostics, found in ISO 8571-3 Annex A, must be supported.

- In the case where a specific parameter can in no way be accommodated then the request fails and a <diagnostic> indicating one such parameter should be returned by the responder. In the case where a negotiable parameter cannot be accommodated with exactly the value requested but is negotiated to a different value (as defined in the standard) then the request formally succeeds but informative <diagnostics> indicating those parameters negotiated should be returned.

- In order to provide for robust applications using FTAM, well defined and precise diagnostics are required to be returned by responding implementations whenever an action cannot be carried out precisely as requested with respect to non-negotiable parameters. All such applicable diagnostics will be returned in those cases. An action is carried out precisely as requested with respect to a parameter when the value of that parameter on the <request> FPDU is equal to the value in effect during or subsequent to the action, depending on whether the action is regime control.

  Diagnostics exist to signal 'parameter not supported' and Responder implementations shall issue all appropriate diagnostics. The <further details> subfield of the <diagnostic> parameter shall specify the parameter which is not implemented.

9.14 CONCURRENCY

The <concurrency control> used by default on actions requested by an <F-SELECT indication> or <F-CREATE indication> service are:

'shared'	for read and read attribute
'exclusive'	for all other actions

The default for actions not requested is specified as 'not required' as per ISO 8571-3.

Note: A local implementation may choose to be more restrictive in order to assure file consistency for concurrent accessors.

FADU locking is not required.

9.15 REQUESTED ACCESS

The <requested access> parameter on <F-SELECT> or <F-CREATE> is used to specify the actions which the Initiator may perform during the file selection. The value of the <requested access> parameter is compared by the Responder to the <access control> and <permitted actions> file attributes and concurrency controls (including those requested by the Initiator) currently in place on the file. If the value of the <requested access> parameter is not consistent with either <access control>, <permitted actions>, or concurrency controls in place, then the <F-SELECT> or <F-CREATE> must be rejected.

<requested access> is consistent with <access control> if, for each action requested, that action either requires no password, or the required password has been specified on the <F-SELECT request> or <F-CREATE request>.

<requested access> is consistent with <permitted actions> if, for each action requested, that action is allowed by the <permitted actions> file attribute.

<requested access> is consistent with <concurrency control> requested on the <F-SELECT> or <F-CREATE> if, for each action requested, that action has not been specified as 'not required' or 'no access' in the <concurrency control> parameter.

<requested access> is consistent with concurrency controls in place on the file if for each action requested no other accessor of the file has set the concurrency control for that action to either 'exclusive' or 'no access'.

9.16 SECURITY

9.16.1 Initiator Identity and Filestore Password

The <initiator identity> and <filestore password> parameters for an implementation acting as an Initiator are supported. These parameters are optional for an implementation acting as a Responder.

The syntax of <initiator identity> and <filestore password> is system-dependent. <initiator identity> and <filestore password> will represent account information on the local system, which may be different from the <account> parameter.

9.16.2 Access Passwords

The <access passwords> and <create password> parameters for an implementation acting as an Initiator are supported if the Security Group of attributes is supported. These parameters for an implementation acting as a Responder are optionally supported if the Security Group is supported.

9.16.3 Implementation Responsibilities

It is the responsibility of each local system to provide security for its own real filestore. Encryption of passwords will not be done by FTAM.

A user of the file service must be known by the Responder. "Known" is defined by the local Filestore, and is dependent on the level of security provided by the local Filestore.

9.17 REQUIREMENT FOR CONFORMANT IMPLEMENTATIONS

This section gives the criteria to be satisfied by every implementation of FTAM that conforms to these Agreements.

Conformance to these Agreements is stated in terms of the different roles occupied by FTAM implementations. The interoperability of certain configurations of these roles motivates this approach. Interoperable configurations of these roles are given in Section 9.17.1.

The only function provided by every conformant implementation is the transfer of unstructured binary files in their entirety. It must be recognized that such simple transfer, while commonly understood and generally important, will not support all applications of FTAM. Section 9.18 defines Implementation Profiles of FTAM services and protocol that can provide other specific functions. Those other functions exploit the access and management capabilities of FTAM. The unconstrained service class (with appropriately chosen functional units) can be used to provide the functions of any of the Implementation Profiles. Users of FTAM must consider carefully what functions they require. They must examine all the Implementation Profiles and select according to their needs.

Implementation conforming to these Agreements require adherence to the General Agreements in Sections 9.5 through 9.16 of these Agreements.

9.17.1 Interoperable Configurations

Any implementation conforming to this specification must be able to act in at least one of the following role combinations:

1. initiator and receiver,

2. initiator and sender,

3. responder and sender,

4. responder and receiver.

Minimal implementations of combination 1 will interoperate with minimal implementations of combination 3. Minimal implementations of combination 2 will interoperate with minimal implementations of combination 4.

Any implementations of roles 1 and 3 will be able to interoperate at the intersection of their capabilities (which will be at least the minimal capabilities described in Sections 9.17.3 to 9.17.8). Any implementations of roles 2 and 4 will be able to interoperate at the intersection of their capabilities (which will be at least the minimal capabilities described in Sections 9.17.3 to 9.17.8).

These role combinations and this interoperability are shown in Table 9.5 below.

Table 9.5 Interoperable configurations

		Initiator		Responder	
		sender	receiver	sender	receiver
Initiator	sender				x
	receiver			x	
Responder	sender		x		
	receiver	x			

9.17.2 Relationship to ISO 8571--The FTAM Standard

Any implementation in conformance to ISO 8571 (as defined in ISO 8571-4, Clause 22 (Conformance)), in addition to the implementation of the minimal protocols and roles enumerated in Sections 9.17.3 to 9.17.8, is considered to be in conformance with these Agreements. Any implementation violating any of the conformance statements in ISO 8571-4 is considered to be in violation of these Agreements.

9.17.3 Requirements for Document Type Support

The document type FTAM-3 shall be supported for purposes of transfer and storage. The details regarding support for FTAM-3 in the FTAM dialogue are given in Section 9.10.

Support of document types other than FTAM-3 is not required for conformant implementations. Support for document types described in these Agreements also entails support for:

- the semantics given in their description and further qualified in 9.10

- the preferred transfer syntax "Basic Encoding of a single ASN.1 type"

9.17.4 Initiators

Every implementation of an FTAM Initiator shall support:

- the kernel protocol and its mandatory parameters with minimum ranges [Minimum required ranges are specified in Section 9.17.8.],

- the grouping protocol and the <threshold> parameter with a value of at least 2 for use in the file transfer class,
- at least one of the read or write protocols [Specific conformance for reading and writing is defined in Sections 9.17.6 and 9.17.7.],

and support the applicable procedures defined in ISO 8571-4 Clauses 8.1 (FTAM regime establishment), 8.2 (FTAM regime termination), 8.3 (File selection), 8.4 (File deselection), 8.9 (File open), 8.10 (File close), 8.11 (Begin group), 8.12 (End group), and 10 (File general actions). To support the above protocols and procedures the implementation shall always support the kernel functional unit and additionally shall be able to:

- request the grouping and at least one of the read or write functional units,
- request the file transfer class with the <service class> parameter,
- request the document type FTAM-3 using the <document type name> form of the <contents type> parameter,
- request the <FTAM quality of service> parameter with value 0 and accept in all cases the returned value 0, and
- request a <communication quality of service> consistent with the transport definition in these Agreements

as part of the Filestore initialization procedures in ISO 8571-4 Clause 8.1, FTAM regime establishment.

Initiators must be able to operate under all circumstances if the above minimum values are successfully negotiated and returned on an <F-INITIALIZE response> PDU. Initiators must be able to operate with any downward negotiation of requested parameter values as described in the standard.

Should the supporting services break down, such that FTAM communication is impossible, the FTAM protocol machine shall notify the user with an <F-P-ABORT indication> and <diagnostic> value with identifier 1011, as well as any known <further details>.

Note: Interworking may not be possible between Initiators not supporting attributes of the Storage Group and Security

Group, and Responders requiring these attributes to be used.

9.17.5 Responders

Every implementation of an FTAM Responder shall support:

- the kernel protocol and its mandatory parameters with minimum ranges [Minimum required ranges are specified in Section 9.17.8.],
- the grouping protocol and the <threshold> parameter with a value of at least 2 for use in the file transfer class,
- at least one of the read or write protocols [Specific conformance for reading and writing is defined in Sections 9.17.6 and 9.17.7],

and support the applicable procedures, defined in ISO 8571-4 Clauses 9.1 (FTAM regime establishment), 9.2 (FTAM regime termination), 9.3 (File selection), 9.4 (File deselection), 9.9 (File open), 9.10 (File close), 9.11 (Begin group), 9.12 (End group), and 10 (File general actions). To support the above protocols and procedures the implementation shall always support the kernel functional unit and additionally shall be able to:

- accept requests for the grouping and at least one of the read or write functional units,
- accept requests for the file transfer class with the <service class> parameter,
- accept the document type FTAM-3 using the <document type name> form of the <contents type> parameter,
- accept requests for an <FTAM quality of service> parameter with any value but may respond with the value 0, and
- accept requests for a <communication quality of service> consistent with the transport definition in these agreements

as part of the filestore initialization procedures in ISO 8571-4 Clause 9.1, FTAM regime establishment.

Responders must be able to operate under all circumstances if the above minimum values are requested on an <F-INITIALIZE request> PDU. Responders must not negotiate upward in the sense described in the standard.

Responders must complete each action requested and supported in a manner consistent with its description in ISO 8571-2 Clauses 10 (Actions on complete files) and 11 (Actions for file access), and must interpret each supported attribute in a manner consistent with its definition in ISO 8571-2 Clause 12 (File attributes).

Under circumstances where actions cannot be carried out either as requested or consistently with ISO 8571-2 Clause 10 (Actions on complete files) and 12 (Actions for file access), the Responder must return at least one diagnostic indicating:

- o if the failure was due to either a protocol or Filestore failure, and then:

  - -- precisely which action failed,
  - -- at least one of the parameters that could not be accommodated with the diagnostic type indicating at least the degree of failure, as given by the action and state result parameter, or

- o that the failure was due to unforeseen system shutdown.

Should the supporting services break down, such that FTAM communication is impossible, the FTAM protocol machine shall notify the user with an <F-P-ABORT indication> and <diagnostic> with identifier 1011, as well as inform the user of any known <further details>.

9.17.6 Senders

Every implementation of an FTAM sender shall support the read functional unit as Responder or the write functional unit as Initiator, and support the applicable procedures defined in ISO 8571-4 Clauses 11 (State of the bulk data transfer activity), 12 (Bulk data transfer protocol data units), 15 (Bulk data transfer sending entity actions), 17.1 (Discarding), and 17.2 (Cancel).

To support those procedures the implementation shall be able to send files of the document type FTAM-3 and shall be able to send them as user data in PPDUs in blocks of up to 7168 octets.

9.17.6.1 Initiator Senders

Every implementation of an FTAM sender which is also an FTAM Initiator shall support:

- o the write functional unit and protocol, and

- for the document type FTAM-3 the following bulk data transfer specification parameters:

FADU operation	replace
FADU identity	first

and support the applicable procedures, defined in ISO 8571-4 Clause 13 (Bulk data transfer initiating entity actions).

9.17.6.2 Responder Senders

Every implementation of an FTAM sender which is also an FTAM Responder shall support:

- the read functional unit and protocol, and

- for the document type FTAM-3 the following bulk data transfer specification parameters:

FADU identity	first
Access context	UA

and support the applicable procedures, defined in ISO 8571-4 Clause 14 (Bulk data transfer responding entity actions).

9.17.7 Receivers

Every implementation of an FTAM receiver shall support the read functional unit as Initiator or the write functional unit as Responder, and support the applicable procedures, defined in ISO 8571-4 Clauses 11 (State of the bulk data transfer activity), 12 (Bulk data transfer protocol data units), 16 (Bulk data transfer receiving entity actions), 17.1 (Discarding), and 17.2 (Cancel).

To support those procedures the implementation shall be able to receive files of the document type FTAM-3 and shall be able to receive them as user data in PPDUs in blocks of at least 7168 octets.

9.17.7.1 Initiator Receivers

Every implementation of an FTAM receiver which is also an FTAM Initiator shall support:

- the read functional unit and protocol, and

- for the document type FTAM-3 the following bulk data transfer specification parameters:

FADU identity	first
Access context	UA

and support the applicable procedures, defined in ISO 8571-4 Clause 13 (Bulk data transfer initiating entity actions).

9.17.7.2 Responder Receivers

Every implementation of an FTAM receiver which is also an FTAM Responder shall support:

- the write functional unit and protocol, and

- for the document type FTAM-3 the following bulk data transfer specification parameters:

FADU operation	replace
FADU identity	first

and support the applicable procedures, defined in ISO 8571-4 Clause 14 (Bulk data transfer responding entity actions).

9.17.8 Minimum Ranges

Any implementation of any conformant FTAM configuration shall be able to receive and meaningfully process all mandatory parameters for all functional units supported as well as the <diagnostic> parameter within at least the minimum ranges of values given in Table 9.6. A conforming implementation may support a wider range of values for any parameter.

Table 9.6 Required minimal parameter support

Parameter		Minimum Range
general	diagnostic	Values as specified in ISO 8571-3 Annex A (Diagnostic parameter values) Tables 44, 45 and 46 which correspond directly to mandatory parameters.
	action result	All values.
	state result	All values.
F_INITIALIZE		
	functional units[1]	'read' (for initiator/receivers and responder/senders) and 'grouping' or 'write' (for initiator/senders and responder/receivers) and 'grouping'
presentation context management[2]		'False'
	all others	As specified in Sections 9.17.4 and 9.17.5 above.
F_SELECT		
	attributes	Only <filename> is used with a minimum supportable length of 8 characters. Any other attribute supported for other services must have minimum supported lengths as in ISO 8571-2 Clause 15 (Minimum attribute ranges) Table 2.
	requested access	'read' for initiator/receivers 'read' for responder/senders 'replace' for initiator/senders 'replace' for responder/receivers

(Continued on next page.)

Table 9.6 Required minimal parameter support, continued

Parameter		Minimum Range
F_OPEN		
	processing mode	'read' for initiator/receivers 'read' for responder/senders 'replace' for initiator/senders 'replace' for responder/receivers
	contents type	'FTAM-3'
F_READ		
	FADU identity	'first'
	access context	'UA'
F_WRITE		
	FADU operation	'read' for initiator/receivers 'read' for responder/senders 'replace' for initiator/senders 'replace' for responder/receivers
	FADU identity	'first'
F_BEGIN_GROUP		
	threshold[3]	For file transfer (a minimal required function) 2.

For any other supported parameters, minimum ranges are taken from the minimum ranges for the attribute corresponding to each as in ISO 8571-2 Table 4.

Notes:

1. The parameters, functional units, and presentation context management are not ordered, so "minimum value" cannot be formally defined. The above values are those required for conformance to these Agreements but no value conformant to ISO 8571 for use in other applications is regarded to be in violation of these Agreements.

2. Other functional units (and service classes) for defined implementations may also be valid provided that they are implemented in accordance with these Agreements, specifically Section 9.17.8.

3. Every implementation must support the <threshold> value 2 to provide the basic required function of file transfer; any other value in other applications is acceptable.

9.17.9 Use of Lower Layer Services

- Support for the Presentation Context Management functional unit is not required.
- Implementations will support the Session, Presentation, and ACSE requirements as stated in Section 5 of this document.

Note: Implementation of the Session Resynchronize and the Minor Synchronize functional units is highly recommended, since the F-CANCEL service may be less effective when mapped to S-DATA.

9.18 IMPLEMENTATION PROFILES

This section defines Implementation Profiles for the specific functions of:

- File Transfer
- File Access
- File Management.

Those definitions are expressed in terms of:

- Document Types
- Attributes
- Service Classes (both service elements and their parameters).

This by no means defines all possible Implementation Profiles.

The following Implementation Profiles are defined:

T1: Simple File Transfer
T2: Positional File Transfer
T3: Full File Transfer
A1: Simple File Access
A2: Full File Access
M1: Management.

Implementation Agreements have been reached for the following service classes. Note that any given implementation may support more than one service class.

- File Transfer
- File Access
- File Management
- Unconstrained
- File Transfer and Management

Support of an Implementation Profile requires adherence to:

1. corresponding definition in 8571-3 Clause 8 and any related procedures in 8571-4 Clause 8-17,

2. requirements given in Sections 9.5-9.18 of these Agreements, and

3. requirements for parameter and attribute support as defined in Section 9.17.8.

9.18.1 General Requirements for the Defined Implementation Profiles

- Implementations will be able to act either as Initiator or Responder or both.

- Implementations must support diagnostics as described in Section 9.13 of these Agreements.

- Implementations that support the file access service class will support access to sequential files. Support of sequential files entails hierarchy of depth and arc length equal to 1. Other hierarchy depth and arc lengths are not precluded by these agreements.

9.18.2 (deleted)

9.18.3 Document Type Requirements for the Defined Implementation Profiles

Implementations conformant to Implementation Profiles defined in Table 9.7 will support the following document types with the caveats and procedures given. Those document types are defined in Appendix 9A and Section 9.10 of these Agreements, and in ISO 8571-2.

- FTAM-1

- FTAM-2

- FTAM-3

- NBS-6

- NBS-7

  Note: Support of this document type entails the naming of FADUs by their position in preorder traversal.

Caveat: Other methods of naming FADUs depend on the system, application, and specific file, and as such are not described here.

- NBS-8

- NBS-9

Support for any document type requires the ability to transfer and store the abstract syntax given in its definition. These Agreements do not specify techniques or formats for storage.

Caveat: Specific abstract syntaxes for the parameterized document types NBS-6,7,8 are not specified in these Agreements.

Any document type supported must be identifiable by its document type name as given in ISO 8571-2 and in Appendix 9A of these Agreements and, where defined, the parameterization scheme given in Section 9.10 of these Agreements.

For conformance to NBS-9 a Responder is only required to return the <filename> attribute, subject to local security and access control. All other requested attributes need not be returned.

Systems supporting the NBS-9 document type shall make available an NBS-9 document called 'DIRLIS'. This document can be used to obtain a listing of files and their associated attributes from a remote Filestore.

File security issues related to NBS-9 are subject to the security agreements outlined in Section 9.16.

9.18.4 Parameters for the Defined Implementation Profiles

- Implementations will support the <contents type list> parameter on the <F-INITIALIZE> service element. The initiating service must supply a value for this parameter.

- Implementations will support the <diagnostic> parameter as stated in Section 9.13 of these Agreements.

- The <initiator identity> parameter is supported. Use must be consistent with Section 9.16 of these Agreements.

- Implementations are not precluded from using other parameters for security and/or accounting. Responders must state the syntax and the semantics applying to <account> and <charging> parameters. The Responder's minimum implementation is to accept but ignore the <account>.

9.18.5 Parameter Ranges for the Defined Implementation Profiles

Parameter ranges for Implementations Profiles are as stated for primitive data types in Section 9.10 of these agreements.

9.18.6 File Attribute Support for Implementations

Implementations of the Implementation Profiles will support file attributes or attribute groups in the following ways.

- mandatory

  This feature is mandatory in the ISO 8571-2 standard and shall therefore be implemented by all implementations claiming conformance to these Agreements.

- supported

  This feature shall be implemented by all implementations claiming conformance to these Agreements (for attributes, this implies that at least the minimum range of attribute values, as defined in ISO 8571-2 Clause 15, shall be supported). Conformant implementations shall also be able to interwork with other implementations that do not support this feature by negotiating out the corresponding features.

- optionally supported

  Implementations claiming conformance to these Agreements may or may not implement this feature (for attributes, this implies that at least either the minimum range of attribute values, as defined in ISO 8571-2 Clause 15, shall be supported or that the 'no value available' result shall be supplied). If an attribute group with a support level of 'optionally supported' is chosen to be supported, then all the attributes of this group that are classified as 'mandatory' or 'supported' shall be supported.

- not supported

This feature is outside the scope of these Agreements.

Kernel Group	mandatory
o Filename	mandatory
o Permitted Actions	mandatory
o Contents Type	mandatory
Storage Group	optionally supported
o Storage Account	optionally supported
o Date and Time of Creation	optionally supported
o Date and Time of Last Modification	optionally supported
o Date and Time of Last Read Access	optionally supported
o Date and Time of Last Attribute Modification	optionally supported
o Identity of Creator	optionally supported
o Identity of Last Modifier	optionally supported
o Identity of Last Reader	optionally supported
o Identity of Last Attribute Modifier	optionally supported
o File Availability	supported
o Filesize	supported
o Future Filesize	optionally supported
Security Group	optionally supported
o Access Control	supported
o Legal Qualifications	optionally supported
Private Group	not supported

Table 9.7 Implementation profile support requirements

	Service Class (See Note 8)				
Functional Unit	T	M	A	T&M	UNCST
Kernel	T1,T2,T3	M1	A1,A2		
Read (See note 3.)	T1,T2,T3		A1,A2		
Write (See note 3.)	T1,T2,T3		A1,A2		
Limited File Mgmnt.	SeeNote 6	M1	SeeNote 6	See	See
Enhanced File Mgmnt.		M1			
Grouping	T1,T2,T3	M1			
File Access			A1,A2		
Document Types					
FTAM-1	T1,T2,T3	[M1]	A1,A2		
FTAM-2	T2,T3	[M1]	A1,A2		
FTAM-3	T1,T2,T3	[M1]	A1,A2	Note	Note
NBS-6	[T2],T3	[M1]	[A1],A2	4	5
NBS-7	[T2],T3	[M1]	[A1],A2		
NBS-8	T3	[M1]	A2		
NBS-9	[T1],[T2] [T3]	[M1]			

Notes: to 9.18.3 and Table 9.7

1. The Management Implementation Profile is only to be implemented in conjunction with one of the Transfer or Access Profiles.

2. Profile T2 is subset of T3. A1 and T1 are subsets of A2 and T2, respectively.

3. Profiles T1, T2, and T3 require the support of read and/or write functional units.

4. Support of the <File Transfer and Management> service class is optional. The rules for including it in a request and for the response to it are as given in ISO 8571-3, Clause 10.1. Any implementation including TM in the request must be prepared for the possibility that it might be removed from the response.

5. The support of the <Unconstrained> service class is optional. There are no constraints on this service class beyond those of ISO 8571.

6. Limited File Management is not required for the T- and A- Implementation Profiles, but very often it will be a user request to have limited file management functionality available together with file transfer and file access functions. So Limited File Management may be added as an option to the T- and A- Implementation Profiles.

7. [] in Table 9.7 specifies that the document type is optional for the respective Implementation Profile. For M1 the support level depends on the T- or A- Implementation Profile, in conjunction with which M1 is implemented.

8. The Implementation Profiles specify functionality which includes the requirements for conformant implementations as specified in Section 9.17. This is a general basic requirement and is not also reflected in Table 9.7.

9.19 PROVISION OF SPECIFIC FUNCTION

9.19.1 Implementation Profile T1: Simple File Transfer

Implementation Profile T1 provides the function of transferring entire files at the external file service level for files with an unstructured constraint set. This includes support of the document types:

o	FTAM-1	"ISO FTAM unstructured text"	
o	FTAM-3	"ISO FTAM unstructured binary"	
o	NBS-9	"NBS-9 file directory file"	(optional)

This Implementation Profile supports file transfer and not file access, that is, the ability to:

- read a complete file

  and/or

- write (replace, extend) to a file.

9.19.2 Implementation Profile T2: Positional File Transfer

Implementation Profile T2 provides the function of transferring files at the external file service level for files with an unstructured or flat constraint set. This includes support of the document types:

o	FTAM-1	"ISO FTAM unstructured text"

o	FTAM-2	"ISO FTAM sequential text"	
o	FTAM-3	"ISO FTAM unstructured binary"	
o	NBS-6	"NBS-6 FTAM sequential file"	(optional)
o	NBS-7	"NBS-7 FTAM random access file"	(optional)
o	NBS-9	"NBS-9 file directory file"	(optional)

This Implementation Profile supports file transfer and not file access, that is, the ability to:

- read a complete file or a single FADU which is identified by position

  and/or

- write (replace, extend, insert depending on constraint set and document type) to a file or an FADU.

This Implementation Profile is upwardly compatible to T1 for the transfer of unstructured files.

9.19.3 Implementation Profile T3: Full File Transfer

Implementation Profile T3 provides the function of transferring files at the external file service level for files with an unstructured, flat or general hierarchical constraint set. This includes support of the document types:

o	FTAM-1	"ISO FTAM unstructured text"	
o	FTAM-2	"ISO FTAM sequential text"	
o	FTAM-3	"ISO FTAM unstructured binary"	
o	NBS-6	"NBS-6 FTAM sequential file"	
o	NBS-7	"NBS-7 FTAM random access file"	
o	NBS-8	"NBS-8 FTAM indexed file"	
o	NBS-9	"NBS-9 file directory file"	(optional)

This Implementation Profile supports file transfer and not file access, that is, the ability to:

- read a complete file or a single FADU which is identified by key or by position

  and/or

- write (replace, extend, insert depending on constraint set and document type) to a file or an FADU.

This Implementation Profile is upwardly compatible to T1 for the transfer of unstructured files.

9.19.4 Implementation Profile A1: Simple File Access

Implementation Profile A1 provides the function of transfer of and access to files with unstructured or flat constraint sets at the external file service level. This includes support of the document types:

o	FTAM-1	"ISO FTAM unstructured text"	
o	FTAM-2	"ISO FTAM sequential text"	
o	FTAM-3	"ISO FTAM unstructured binary"	
o	NBS-6	"NBS-6 FTAM sequential file"	(optional)
o	NBS-7	"NBS-7 FTAM random access file"	(optional)

This Implementation Profile supports file transfer and file access, that is the ability to:

- read a complete file or FADUs which are identified by position,

- write (replace, extend, insert depending on constraint set and document type) to a file or an FADU,

- locate and erase within files.

9.19.5 Implementation Profile A2: Full File Access

Implementation Profile A2 provides the function of transfer of and access to files with unstructured or flat constraint sets at the external file service level. This includes support of the document types:

o	FTAM-1	"ISO FTAM unstructured text"
o	FTAM-2	"ISO FTAM sequential text"
o	FTAM-3	"ISO FTAM unstructured binary"
o	NBS-6	"NBS-6 FTAM sequential file"
o	NBS-7	"NBS-7 FTAM random access file"
o	NBS-8	"NBS-8 FTAM indexed file"

This Implementation Profile supports file transfer and file access, that is, the ability to:

- read from a complete file, or from a series of FADUs which are identified by key or by position,

- write (replace, extend, insert depending on constraint set and document type) to a file or an FADU,

- locate and erase within files.

9.19.6 Implementation Profile M1: Management

Implementation Profile M1 provides the function for an Initiator to manage the files within the Virtual Filestore, to which access is provided by the Responder. Management includes the services of:

- creating a file
- deleting a file
- reading attributes of a file
- changing attributes of a file.

9.20 HARMONIZATION

The Implementation Profiles for File Transfer, File Access and Management correspond to the Profiles of SPAG (Standards Promotion and Application Group) in Europe, so that interworking will be possible. Those Profiles are described in the 'Guide to the Use of Standards' (GUS); they are the basis for the Functional Standards as defined by CEN/CENELEC (Comite Europeenne de Normalization).

Table 9.8 Implementation Profiles (NIST) and Profiles (SPAG/CEN-CLC)

Implementation Profile	SPAG/CEN-CLC
T1	A/111
T2	A/112
T3	A/113
A1	A/122
A2	A/123
M1	A/13

9.21 APPENDIX A: FTAM DOCUMENT TYPES

Part 1: Document Types
Part 2: Constraint Sets
Part 3: Abstract Syntaxes
Part 4: (deleted)

Part 1: Document Types

NBS-6 Sequential file document type

1. Entry Number: NBS-6

2. Information objects

Table 9.9 Information objects in NBS-6

document type name	{iso identified-organization icd (9999) organization-code (1) document type (5) sequential (6)} "NBS-6 FTAM sequential file"

abstract syntax names:
a) name for asname1
b) name for asname2
	{iso identified-organization icd (9999) organization-code (1) abstract-syntax (2) nbs-as1 (1)}
	"NBS abstract syntax AS1"
	{iso standard 8571 abstract-syntax(2) ftam-fadu (2)}
	"FTAM FADU"

transfer syntax names:	{joint-iso-ccitt asn1 (1) basic-encoding (1) } "Basic Encoding of a single ASN.1 type"

```
parameter syntax:
  PARAMETERS ::= SEQUENCE OF CHOICE {Parameter0, Parameter1, Parameter2}
  Parameter0 ::=  [0] INTEGER {univer-time (23),     |
                                                       gen-time (24),
                                                       boolean (1),
                                                       null (5) }
 Parameter1  ::= [1] SEQUENCE  {
                 universal-class-number-1 INTEGER { int      (2),
                                                    bit      (3),
                                                    ia5     (22),
                                                    graphic (25),
                                                    general (27),
                                                    octet    (4)},
                 string-length INTEGER }
 Parameter2  ::= [2] SEQUENCE {
                 private-class-number INTEGER {float (0)},
                 length-1     INTEGER,
                 length-2     INTEGER }
```

file model	{iso standard 8571 file-model (3) hierarchical (1)} "FTAM hierarchical file model"
constraint set	{iso standard 8571 constraint-set (4) sequential-flat (2)} "FTAM sequential flat constraint set"

```
file contents:
            Datatype1 ::= PrimType -- as defined in Annex 9A, Part 3
            Datatype2 ::=          Node-Descriptor-Data-Element
```

3. Scope and field of application

The document type defines the contents of a file for storage, for transfer and access by FTAM.

Note: Storage refers to apparent storage within the Virtual Filestore.

4. References

ISO 8571, Information Processing Systems - Open Systems Interconnection - File Transfer, Access and Management

5. Definitions

This definition makes use of the terms data element, data unit and file access data unit as defined in ISO 8571-1.

6. Abbreviations

FTAM File Transfer, Access and Management

7. Document semantics

The document consists of zero, one or more file access data units, each of which consists of zero, one or more data elements. The order of each of these elements is significant.

The document structure takes any of the forms allowed by the FTAM hierarchical file model as constrained by the sequential flat constraint set (see Table 9.9) These definitions appear in ISO 8571-2. As additional constraints FADU identity will be limited to 'begin', 'end', 'first' and 'next'.

For a specific file the number of data elements in a data unit is given by the parameters. Each data element is a data type from the set of primitive data types defined in the Annex 9A, Part 3 of this document. Each data unit contains the same data element types in the same order as all other data units. These types are determined by the parameters 0 through 2.

Note: The string length values are the actual number of characters from the specified character set, they do not include any escape sequences or overhead from the encoding.

8. Abstract syntactic structure

The abstract syntactic structure of the document is a hierarchically structured file as defined in the ASN.1 module ISO8571-FADU in ISO 8571, in which each of the file access data units has the abstract syntactic structure of NBS-AS1 as defined by the parameters.

9. Definition of transfer

9.1 Datatype definitions

The file consists of data values which are of either

a) Datatype1 defined in Table 9.9, where the PrimType in the datatype is given by the NBS-AS1 definition; or

b) Datatype2 defined in Table 9.9, the ASN.1 datatype declared as "Data-Element" in the ASN.1 module IS08571-FADU.

9.2 Presentation data values

The document is transferred as a series of presentation data values, each of which is either

a) one value of the ASN.1 datatype "Datatype1", carrying one of the data elements from the document. All values are transmitted in the same (but any) presentation context defined to support the abstract syntax name "asname1" or

b) a value of "Datatype2". All values are transmitted in the same (but any) presentation context defined to support the abstract syntax name "asname2".

Notes:

1. Specific carrier standards may impose additional constraints on the presentation context to be used, where the above permits a choice

2. Any document type defined in this entry either makes no use of Datatype2, or starts with a Datatype2 transmission.

Boundaries between presentation data values in the same presentation context, and boundaries between P-DATA primitives, are chosen locally by the sending entity at the time of transmission, and carry no semantics of the document type. Receivers which support this document type shall accept a document with any of the permitted transfer options (e.g. document type parameters and transfer syntaxes).

9.3 Sequence of presentation data values

The sequence of presentation data values of type a) and the sequence of presentation data values of types a) and b) is the same as the sequence of data elements within a Data Unit, and Data Units in the hierarchical structure, when flattened according to the definition of the hierarchical file model in ISO 8571-2.

10. Transfer syntax

An implementation supporting this document type shall support the transfer syntax generation rules named in Table 9.9 for all presentation data values transferred. An implementation may optionally support other named transfer syntaxes.

11. ASE specific specifications for FTAM

11.1 Simplification and relaxation

11.1.1 Structural simplification

This simplification loses information.

The document type NBS-6 may be simplified to the document type FTAM-3 (allowed only when reading the file). The octet representation of the transferred data is unpredictable. It will usually correspond to the data values as stored in the local Real Filestore of the Responder.

11.2 Access context selection

A document of type NBS-6 may be accessed in any one of the access contexts defined in the sequential flat constraint set. The presentation data units transferred in each case are those derived from the structuring elements defined for that access context in ISO 8571-2.

11.3 The INSERT operation

When the <INSERT> operation is applied at the end of file the transferred material shall be the series of FADUs which would be generated by reading any NBS-6 document with the same parameter values in access context FA.

NBS-7 Random access file

1. **Entry number: NBS-7**

2. **Information objects**

Table 9.10 Information objects in NBS-7

document type name	{iso identified-organization icd (9999) organization-code (1) document type (5) random-file (7)} "NBS-7 FTAM random access file"
abstract syntax names: a) name for asname1 b) name for asname2	{iso identified-organization icd (9999) organization-code (1) abstract-syntax (2) nbs-as1 (1)} "NBS abstract syntax AS1" {iso standard 8571 abstract-syntax(2) ftam-fadu (2)} "FTAM FADU"
transfer syntax names:	{joint-iso-ccitt asn1 (1) basic-encoding (1) } "Basic Encoding of a single ASN.1 type"

parameter syntax:
PARAMETERS ::= SEQUENCE OF CHOICE {Parameter0, Parameter1, Parameter2}
Parameter0 ::= [0] INTEGER {univer-time (23),
gen-time (24),
boolean (1),
null (5) }
Parameter1 ::= [1] SEQUENCE {
universal-class-number-1 INTEGER { int (2),
bit (3),
ia5 (22),
graphic (25),
general (27),
octet (4)},
string-length INTEGER }
Parameter2 ::= [2] SEQUENCE {
private-class-number INTEGER {float (0)},
length-1 INTEGER,
length-2 INTEGER }	

file model	{iso standard 8571 file-model (3) hierarchical (1)} "FTAM hierarchical file model"
constraint set	{iso identified-organization icd (9999) organization-code (1) constraint-set (4) nbs-ordered flat (2)} "NBS ordered flat constraint set"
file contents: Datatype1 ::= PrimType -- as defined in Annex 9A, Part 3 Datatype2 ::= CHOICE { Node-Descriptor-Data-Element, Enter-Subtree-Data-Element } Exit-Subtree-Data-Element }	

3. **Scope and field of application**

The document type defines the contents of a file for storage, for transfer and access by FTAM.

Note: Storage refers to apparent storage within the Virtual Filestore.

4. **References**

ISO 8571, Information Processing Systems - Open Systems Interconnection -File Transfer, Access and Management

5. **Definitions**

This definition makes use of the terms data element, data unit and file access data unit as defined in ISO 8571-1.

6. **Abbreviations**

FTAM File Transfer, Access and Management

7. **Document semantics**

The document consists of zero, one or more file access data units, each of which consists of zero, one or more data elements. The order of each of these elements is significant.

The document structure takes any of the forms allowed by the FTAM hierarchical file model as constrained by the NBS-ordered-flat constraint set (see Table 9.10). These definitions appear in Appendix 9A, Part 2 of this document.

For a specific file the number of data elements in a data unit is given by the parameters. Each data element is a data type from the set of primitive data types defined in the Annex 9A, Part 3 of this document. Each data unit contains the same data element types in the same order as all other data units. These types are determined by the parameters 0 through 2.

Note: The string length values are the actual number of characters from the specified character set, they do not include any escape sequences or overhead from the encoding.

8. **Abstract syntactic structure**

The abstract syntactic structure of the document is a hierarchically structured file as defined in the ASN.1 module ISO8571-FADU in ISO 8571, in which each of the file access data units has the abstract syntactic structure of NBS-AS1 as defined by the parameters.

9. Definition of transfer

9.1 Datatype definitions

The file consists of data values which are of either

a) Datatype1 defined in Table 9.10, where the PrimType in the datatype is given by the NBS-AS1 definition; or

b) Datatype2 defined in Table 9.10, the ASN.1 datatype declared as "Data-Element" in the ASN.1 module ISO8571-FADU.

9.2 Presentation data values

The document is transferred as a series of presentation data values, each of which is either

a) one value of the ASN.1 datatype "Datatype1", carrying one of the data elements from the document. All values are transmitted in the same (but any) presentation context defined to support the abstract syntax name "asname1" or

b) a value of " Datatype2". All values are transmitted in the same (but any) presentation context defined to support the abstract syntax name " asname2".

Notes:

1. Specific carrier standards may impose additional constraints on the presentation context to be used, where the above permits a choice
2. Any document type defined in this entry either makes no use of Datatype2, or starts with a Datatype2 transmission.

Boundaries between presentation data values in the same presentation context, and boundaries between P-DATA primitives, are chosen locally by the sending entity at the time of transmission, and carry no semantics of the document type. Receivers which support this document type shall accept a document with any of the permitted transfer options (e.g. document type parameters and transfer syntaxes).

9.3 Sequence of presentation data values

The sequence of presentation data values of type a) and the sequence of presentation data values of types a) and b) is the same as the sequence of data elements within a Data Unit, and Data Units in the hierarchical structure, when flattened according to the definition of the hierarchical file model in ISO 8571-2.

10. Transfer syntax

An implementation supporting this document type shall support the transfer syntax generation rules named in Table 9.10 for all presentation data values transferred. Implementation may optionally support other named transfer syntaxes.

11. ASE specific specifications for FTAM

11.1 Simplification and relaxation

11.1.1 Structural simplification

This simplification loses information.

The document type NBS-7 may be accessed as a document type FTAM-3 (allowed only when reading the file) by specifying document type FTAM-3 in the <contents type> parameter in <F-OPEN request>, and limiting access context to UA on F-READ.

The octet representation of the transferred data is unpredictable. It will usually correspond to the data values as stored in the local Real Filestore of the Responder.

A document of type NBS-7 can be accessed as a document of type NBS-6 (allowed only when reading the file) by specifying document type NBS-6 with appropriate data type parameters in the <contents type> parameter on the <F-OPEN request>.

11.2 Access context selection

A document of type NBS-7 may be accessed in any one of the access contexts defined in the NBS-ordered-flat constraint set. The presentation data units transferred in each case are those derived from the structuring elements defined for that access context in ISO 8571-2.

11.3 The INSERT operation

When the <INSERT> operation is applied at the end of file the transferred material shall be the series of FADUs which would be generated by reading any NBS-7 document with the same parameter values in access context FA.

NBS-8 Indexed sequential file

1. **Entry Number: NBS-8**
2. **Information objects**

Table 9.11 Information objects in NBS-8

document type name	{iso identified-organization icd (9999) organization-code (1) document type (5) indexed-file (8)} "NBS-8 FTAM indexed file"

abstract syntax names:
a) name for asname1
b) name for asname2
	{iso identified-organization icd (9999) organization-code (1) abstract-syntax (2) nbs-as1 (1)}
	"NBS abstract syntax AS1"
	{iso standard 8571 abstract-syntax(2) ftam-fadu (2)}
	"FTAM FADU"

transfer syntax names:	{joint-iso-ccitt asn1 (1) basic-encoding (1) } "Basic Encoding of a single ASN.1 type"
parameter syntax: PARAMETERS ::= SEQUENCE {DataTypes, KeyType, KeyPosition} DataTypes ::= SEQUENCE OF CHOICE {Parameter0, Parameter1, Parameter2} KeyType ::= CHOICE {Parameter0, Parameter1, Parameter2} -- Parameter0, Parameter1, Parameter2, as defined for the -- document types NBS-6, NBS-7 KeyPosition::= INTEGER	
file model	{iso standard 8571 file-model (3) hierarchical (1)} "FTAM hierarchical file model".
constraint set	{iso standard 8571 constraint-set (4) ordered-flat (3) } "FTAM ordered flat constraint set"
file contents: Datatype1 ::= PrimType -- as defined in Annex 9A, Part 3 Datatype2 ::= CHOICE { Node-Descriptor-Data-Element, Enter-Subtree-Data-Element } Exit-Subtree-Data-Element }	

3. Scope and field of application

The document type defines the contents of a file for storage, for transfer and access using FTAM.

Note: Storage refers to apparent storage within the Virtual Filestore.

4. References

ISO 8571, Information Processing Systems - Open Systems Interconnection -File Transfer, Access and Management

5. Definitions

This definition makes use of the terms data element, data unit and file access data unit as defined in ISO 8571-1.

6. Abbreviations

FTAM File Transfer, Access and Management

7. Document semantics

The document consists of zero, one or more file access data units, each of which consists of zero, one or more data elements. The order of each of these elements is significant.

The document structure takes any of the forms allowed by the FTAM hierarchical file model as constrained by the FTAM ordered flat constraint set (see Table 9.11). These definitions appear in ISO 8571-2.

The following additional requirements are specified for the use of the ordered flat constraint set:

- o The FADU identities 'first', 'last', and 'traversal number' are not required for conformant implementations
- o The identities 'next' and 'previous' are allowed for all FADUs

Each data element is a data type from the set of primitive data types defined in Appendix 9A, Part 3 of this document. Each data unit contains the same data element types in the same order as all other data units. These types and their respective maximum lengths are defined by the <DataTypes> parameter.

Note: The length values refer to the number of characters from the applicable type, not to the number of octets in the encoding, nor to the line length in any rendition of the document, where these are different.

Each data unit in the file has a key associated with it. The key of each data unit is of the same data type as the key of all other data units in the file and is a single data element from the set of primitive data types defined in Appendix 9A, Part 3.

The type and length of the key are defined by the <KeyType> parameter.

The primitive data types and minimum size ranges of each unit which an implementation must accept as a key value are given in the following Table 9.12.

Table 9.12 Datatypes for keys

Key Type	Minimum Range (octets)	Order
ASN.1 INTEGER	(1-2)	increasing numeric value
ANS.1 IA5String	(0-16)	lexical order
ASN.1 GraphicString	(0-16)	lexical order
ANS.1 GeneralString	(0-16)	lexical order
ANS.1 OCTET STRING	(0-16)	increasing value
ASN.1 GeneralizedTime		increasing time value
ASN.1 UniversalTime		increasing time value
NBS-AS1 FloatingPointNumber		increasing numeric value

The position of the key in the data unit is specified by the <position> parameter.
position = 0 implies the key is not part of the data
position > 0 specifies the actual data element in the data unit.

8. Abstract syntactic structure

The abstract syntactic structure of the document is a hierarchically structured file as defined in the ASN.1 module ISO8571-FADU in ISO 8571, in which each of the file access data units has the abstract syntactic structure of NBS-AS1 as defined by the parameters.

9. Definition of transfer

9.1 Datatype definitions

The file consists of data values which are of either

a) Datatype1 defined in Table 9.11, where the PrimType in the datatype is given by the NBS-AS1 definition; or

b) Datatype2 defined in Table 9.11, the ASN.1 datatype declared as "Data-Element" in the ASN.1 module ISO8571-FADU.

9.2 Presentation data values

The document is transferred as a series of presentation data values, each of which is either

a) one value of the ASN.1 datatype "Datatype1", carrying one of the data elements from the document. All values are transmitted in the same (but any) presentation context defined to support the abstract syntax name "asname1" or

b) a value of "Datatype2". All values are transmitted in the same (but any) presentation context defined to support the abstract syntax name "asname2".

Notes:

1. Specific carrier standards may impose additional constraints on the presentation context to be used, where the above permits a choice
2. Any document type defined in this entry either makes no use of Datatype2, or starts with a Datatype2 transmission.

Boundaries between presentation data values in the same presentation context, and boundaries between P-DATA primitives, are chosen locally by the sending entity at the time of transmission, and carry no semantics of the document type. Receivers which support this document type shall accept a document with any of the permitted transfer options (e.g. document type parameters and transfer syntaxes).

9.3 Sequence of presentation data values

The sequence of presentation data values of type a) and the sequence of presentation data values of types a) and b) is the same as the sequence of data elements within a Data Unit, and Data Units in the hierarchical structure, when flattened according to the definition of the hierarchical file model in ISO 8571-2.

10. Transfer syntax

An implementation supporting this document type shall support the transfer syntax generation rules named in Table 9.11 for all presentation data values transferred. Implementation may optionally support other named transfer syntaxes.

11. ASE specific specifications for FTAM

11.1 Simplification and relaxation

11.1.1 Structural simplification

This simplification loses information.

The document type NBS-8 may be accessed as a document type FTAM-3 (allowed only when reading the file) by specifying document type FTAM-3 in the <contents type> parameter in <F-OPEN request>, and limiting access context to UA on F-READ.

The octet representation of the transferred data is unpredictable. It will usually correspond to the data values as stored in the local Real Filestore of the Responder.

A document of type NBS-8 can be accessed as a document of type NBS-6 (allowed only when reading the file) by specifying document type NBS-6 with appropriate data type parameters in the <contents type> parameter on the <F-OPEN request>. The traversal order of the FADUs must be maintained.

Note: The traversal order is as reading the file as NBS-8 in key order.

11.2 Access context selection

A document of type NBS-8 may be accessed in any one of the access contexts defined in the FTAM ordered flat constraint set. The presentation data units transferred in each case are those derived from the structuring elements defined for that access context in ISO 8571-2.

11.3 The INSERT operation

When the <INSERT> operation is applied the transferred material shall be the series of FADU which would be generated by reading any NBS-8 document with the same parameter values in access context FA.

The insertion of a new FADU after an already existing FADU will be indicated via a diagnostic on TRANSFER-END.

11.4 The EXTEND operation

This operation is excluded for the use with this document type.

NBS-9 File directory file

1. **Entry Number:** **NBS-9**

2. **Information objects**

Table 9.13 Information objects in NBS-9

document type name	{iso identified-organization icd (9999) organization-code (1) document type (5) file-directory (9)} "NBS-9 FTAM file directory file"
abstract syntax names:	{iso identified-organization icd (9999) organization-code (1) abstract syntax (2) nbs-as2 (2)} "NBS file directory entry abstract syntax"
transfer syntax names:	{joint-iso-ccitt asn1 (1) basic-encoding (1)} "Basic Encoding of a single ASN.1 type"

parameter syntax

```
PARAMETERS ::=  [0] IMPLICIT BIT STRING {

            -- Kernel group

                 read-filename (0),
                 read-permitted-actions (1),
                 read-contents-type (2),

            -- Storage group

                 read-storage-account (3),
                 read-date-and-time-of-creation (4),
                 read-date-and-time-of-last-modification (5),
                 read-date-and-time-of-last-read-access (6),
                 read-date-and-time-of-last-attribute-modification(7),
                 read-identity-of-creator (8),
                 read-identity-of-last-modifier (9),
                 read-identity-of-last-reader (10),
                 read-identity-of-last-attribute-modifier (11),
                 read-file-availability (12),
                 read-filesize (13),
                 read-future-filesize (14),

            -- Security group

                 read-access-control (15),
                 read-legal-qualifications (16),

            -- Private group

                 read-private-use (17) }
```

(Continued on next page.)

Table 9.13 Information objects in NBS-9 continued.

file model	{iso standard 8571 file-model (3) hierarchical (1)} "FTAM hierarchical file model"
constraint-set	{iso standard 8571 constraint-set (4) unstructured (1)} "FTAM unstructured constraint set"
File contents: Datatype1 ::= FileDirectoryEntry --As defined by NBS-AS2 in Appendix A, --Part 3 of this document	

3. Scope and field of Application

This document defines the contents of a file for transfer (not for storage) using FTAM.

4. References

ISO 8571, Information Processing Systems - Open Systems Interconnection -File Transfer, Access and Management.

5. Definitions

This definition makes use of the terms data element, data unit and file access data unit as defined in ISO 8571-1

6. Abbreviations

FTAM File Transfer, Access and Management.

7. Document Semantics

The document consists of one file access data unit, which consists only of zero, one or more data elements of type <FileDirectoryEntry> (defined in NBS-AS2).

The document structure takes any of the forms allowed by the FTAM hierarchical file model as constrained by the unstructured constraint set. These definitions appear in ISO 8571-1.

The parameter of the document type is used on <F-OPEN request> to specify the desired attributes of each of the files on the Filestore, when reading the document.

8. Abstract syntactic structure

The abstract syntactic structure of the document is a series of file directory entries, each of which is defined by the <FileDirectoryEntry> definition in NBS-AS2.

Additional constraints are defined for this document type: File access actions are restricted to Read. File-directory files may be Selected, Opened, Read, Closed, and Deselected. They may not be Created or Deleted. They may not be Written or Modified (except as a side effect of actions performed on individual files contained within a file directory).

9. Definition of transfer

9.1 Datatype definition

The file consists of zero or more values of Datatype1 defined in Table 9.13.

9.2 Presentation data values

The document is transferred as a series of presentation data values. Each presentation data value shall consist of one value of the ASN.1 data type "Datatype1", carrying one of the file directory entries from the document.

All values are transmitted in the same (but any) presentation context established to support the abstract syntax name "asname1" declared in Table 9.13.

9.3 Sequence of presentation data values

The sequence of presentation data values is the same as the sequence of file directory entries within the Data Unit in the file.

10. Transfer syntax

An implementation supporting this document type shall support the transfer syntax generation rules named in Table 9.13 for all presentation data values transferred. Implementations shall optionally support other named transfer syntaxes.

11. ASE specific specifications for FTAM

11.1 Simplification and relaxation

Relaxation is allowed to any bitstring combination of the document type parameter.

Part 2: Constraint Sets

NBS Ordered flat constraint set

1. Field of application

The NBS-ordered flat constraint set applies to files which are structured into a sequence of individual FADUs and to which access may be made on an FADU basis by position in the sequence.

2. Basic constraints

Table 9.14 Basic constraints for NBS Ordered flat

Constraint set descriptor	"NBS ordered flat constraint set"
Constraint set identifier	{iso identified-organization icd (9999) organization-code (1) constraint-set (4) nbs-ordered-flat(1)}
Node name	None
File access actions	Locate, Read, Insert, Erase, Replace
Qualified action	None
Available access contexts	HA, FA, UA
Creation state	Root node without an associated data unit
Location after open	Root node
Beginning of file	Root node
End of file	No node selected; 'previous' gives last node in traversal sequence,'current'and 'next'give an error.
Read whole file	Read in access context FA or UA with FADU identity of 'begin'.
Write whole file (append)	Transfer the series of leaf FADUs which would be generated by reading the whole file in access context FA; perform the transfer with an FADU identity of 'end' and a file access action of 'insert'.
Write whole file (replace)	Transfer the series of leaf FADUs which would be generated by reading the whole file in access-context HA; perform the transfer with FADU identity 'begin' and file action of 'replace'.

3. **Structural constraints**

The root node shall not have an associated data unit; all children of the root node shall be leaf nodes and may have an associated data unit; all arcs from the root node shall be of length one.

4. **Action constraints**

Insert: The <Insert> action is allowed only at the end of file. If the FADU identity is 'end' the new node is inserted following all existing nodes in the file. If the FADU identity is 'traversal number', the number must be at least one greater than the traversal number of the last existing node. Any nodes between the last existing node and the new node are empty, i.e. nodes without data. If the FADU identity is a 'traversal number' not greater than that of the last existing node, an error will occur.

Erase: The Erase action is only allowed at the root node to empty the file, with FADU identity of 'begin'. The result is a solitary root node without an associated data unit.

Note: It is the intention when using this constraint set to allow for emptying an FADU, i.e. leaving an FADU with a DU of data length 0 (or without a DU); afterwards data may be reinserted into this hole. In order to empty an FADU, the <Replace> operation may be used with new data of length zero (or with an FADU whose <data exists> bit is set to 'false' and no DU). Refilling the hole is accomplished by a <Replace> operation with the new DU (or with the new FADU, whose <data exists> bit is set to 'true' and the new DU).

5. Identity constraints

The FADU identity associated with the file action shall be one of the identities 'begin', 'end', 'first', 'last', 'current', 'next', 'previous' or a 'traversal number' greater than or equal to one. The actions with which these identities can be used are given in the following table.

Table 9.15 Identity constraints in NBS Ordered flat

Action	Begin	End	First	Last	Current	Next	Previous	Traversal
Locate	valid	valid	valid	valid	valid	valid	valid	valid
Read	whole		leaf	leaf	leaf	leaf	leaf	leaf
Insert		valid						leaf
Erase	whole							
Replace	whole		leaf	leaf	leaf	leaf	leaf	leaf

Part 3: Abstract Syntaxes

Abstract Syntax NBS-AS1

```
Abstract syntax name:  {iso identified-organization icd (9999)
                       organization-code  (1) abstract-syntax (2) nbs-as1
                       (1)}
                       "NBS abstract syntax AS1"
```

This is an abstract syntax for the set of presentation data values, each of which is a value of the ASN.1 type NBS-AS1.PrimType

```
NBS-AS1 DEFINITIONS ::=
BEGIN
PrimType ::= CHOICE {  INTEGER,
                       BIT STRING,
                       BOOLEAN,
                       IA5String,
                       GraphicString,
                       GeneralString,
                       OCTET STRING,
                       UTCTime,
                       GeneralizedTime,
                       NULL,
                       FloatingPointNumber }

                       --   The support for IA5String is the IA5 G0
                       --   character set and the IA5 C0 set
                       --   The minimum level of support for
                       --   GraphicString is the IA5 G0 character set and
                            the 8859-1 G0 and G1 sets
                       --   The minimum level of support for
                            GeneralString is the IA5 G0 character set and
                            the 8859-1 G0 and G1 character sets, and IA5
                            C0 set.

FloatingPointNumber ::=     [PRIVATE 0]    CHOICE {
                                 finite [0] IMPLICIT SEQUENCE
                                      {    Sign,
                                           mantissa BIT STRING,
                                      --   first bit must be 1
                                           exponent INTEGER},
                                 infinity [1] IMPLICIT Sign,
                                 signalling-nan [2] IMPLICIT NaN,
                                 quiet-nan [3] IMPLICIT NaN,
                                 zero [4] IMPLICIT NULL }

Sign    ::= INTEGER { positive (0), negative (1) }
NaN     ::= INTEGER
END
```

For this abstract syntax the following transfer syntax can be used
{joint-iso-ccitt asn1 (1) basic-encoding (1)}
"Basic Encoding of a single ASN.1 type"

Notes:
1. The mantissa is a number in the range (1/2 <mantissa<1).
2. The value is equal to mantissa * 2 $^{\text{exponent}}$.
3. The first bit in the mantissa is most significant.
4. See IEEE 754 for definitions of terminology, such as NaN.
5. A minimum length range (in bits) is required for the components of <FloatingPointNumber>, as follows: mantissa 1-23 bits, and exponent 0-8 bits.

Abstract Syntax NBS-AS2

Abstract syntax name: { iso identified-organization icd (9999) organization-code (1) abstract-syntax (2) nbs-as2 (2) }

"NBS file directory entry abstract syntax"

This is an abstract syntax for the set of presentation data values, each of which is a value of the ASN.1 Type NBS-AS2.FileDirectoryEntry.

```
NBS-AS2 DEFINITIONS ::=

BEGIN

FileDirectoryEntry  ::=[PRIVATE 2] Read-Attributes

Read-Attributes     ::=ISO8571-FTAM.Read-Attributes

END
```

For this abstract syntax the following transfer syntax will be used

{ joint-iso-ccitt asn1 (1) basic-encoding (1) }

"Basic Encoding of a single ASN.1 type"

Abstract Syntax "FTAM unstructured text abstract syntax"

This abstract syntax is defined as DataType1 (File Contents) in Table 19 of ISO 8571-2, Annex B.

Abstract Syntax "FTAM unstructured binary abstract syntax"

This abstract syntax is defined as DataType1 (File Contents) in Table 21 of ISO 8571-2, Annex B.

Part 4: (deleted)

10. FUTURE FILE TRANSFER, ACCESS AND MANAGEMENT (FTAM) PHASE 3

Editor's Note: This section is reserved for future stable File Transfer, Access and Management (FTAM) Phase 3 Agreements. These agreements will provide enhancements to services defined in the Stable FTAM Phase 2 Agreements (Section 9). As this Phase 3 material is declared stable, it will be moved from the aligned section in the Ongoing Agreements Document to the same section in this stable document. Consult section 10 in the Ongoing Agreements Document for more information on this subject.

11. DIRECTORY SERVICES PROTOCOLS

11.1 INTRODUCTION

This is an Implementation Agreement developed by the Implementor's Workshop sponsored by the U.S. National Institute of Standards and Technology to promote the useful exchange of data between devices manufactured by different vendors. This agreement is based on and employs protocols developed in accord with the OSI Reference Model. While this agreement introduces no new protocols, it eliminates ambiguities in interpretations.

This is an Implementation Agreement for Directories based on the ISO documents cited in the Reference Section 11.1 (hereafter referenced as Directory Documents). Versions of this document will stay consistent with the latest drafts of those Directory Documents. These Agreements are based on the ISO DIS 9594:1988 (E) text. Figure 11.1 displays the structure of this Implementation Agreement. References to corresponding CCITT documents are included for information.

<table>
<tr><td>Directory Access Protocol
(DAP)</td><td>Directory System Protocol
(DSP)</td></tr>
<tr><td colspan="2">Remote Operations Services and Protocols
(CCITT X.219 and X.229/ISO 9072/1 and 9072/2)</td></tr>
<tr><td colspan="2">Association Control Services and Protocols
(CCITT X.217 and X.227/ISO 8649 and 8650)</td></tr>
</table>

Figure 11.1 Structure of this Implementation Agreement

The Directory User Agents (DUAs) and Directory System Agents (DSAs) provide access to The Directory on behalf of humans and applications such as Message Handling and File Transfer, Access, and Management. See the Scope and Field of Application section for more information on the model used in Directories.

This document covers both the Directory Access Protocol (DAP) and the Directory System Protocol(DSP) defined in the Directory Documents. A good working knowledge of the Directory Documents is assumed by this chapter. All terminology and abbreviations used but not defined in this text may be found in those documents.

11.2 SCOPE AND FIELD OF APPLICATION

Centralized and distributed directories can both be accommodated in this Agreement by the appropriate choice of protocols and pragmatic constraints from those specified. Figure 11.2 illustrates a centralized directory and Figure 11.3 illustrates a distributed directory.

This agreement does not cover interaction between co-located entities, such as a co-resident DUA and DSA. It also does not specify the interface between a user (person or application) and a DUA. Bilateral agreements between a DUA and DSA or DSA and DSA may be implemented in addition to the requirements stated in this document. Conformance to this agreement requires the ability to interact without the use of bilateral agreements other than those required in the Directory Documents.

The logical structure of the Directory Information Base (DIB) is described in the Directory Documents. The manner in which a local portion of the DIB is organized and accessed by its DSA is not in the scope of this agreement.

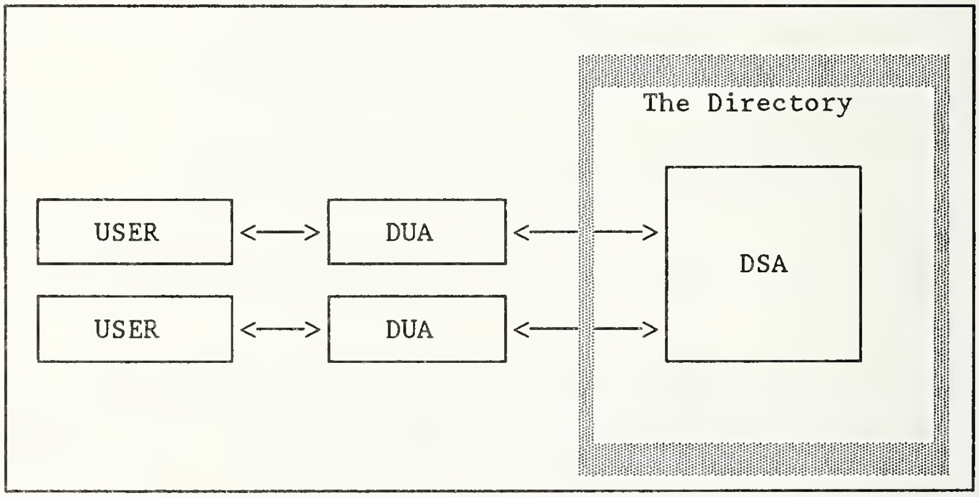


Figure 11.2 Centralized Directory Model

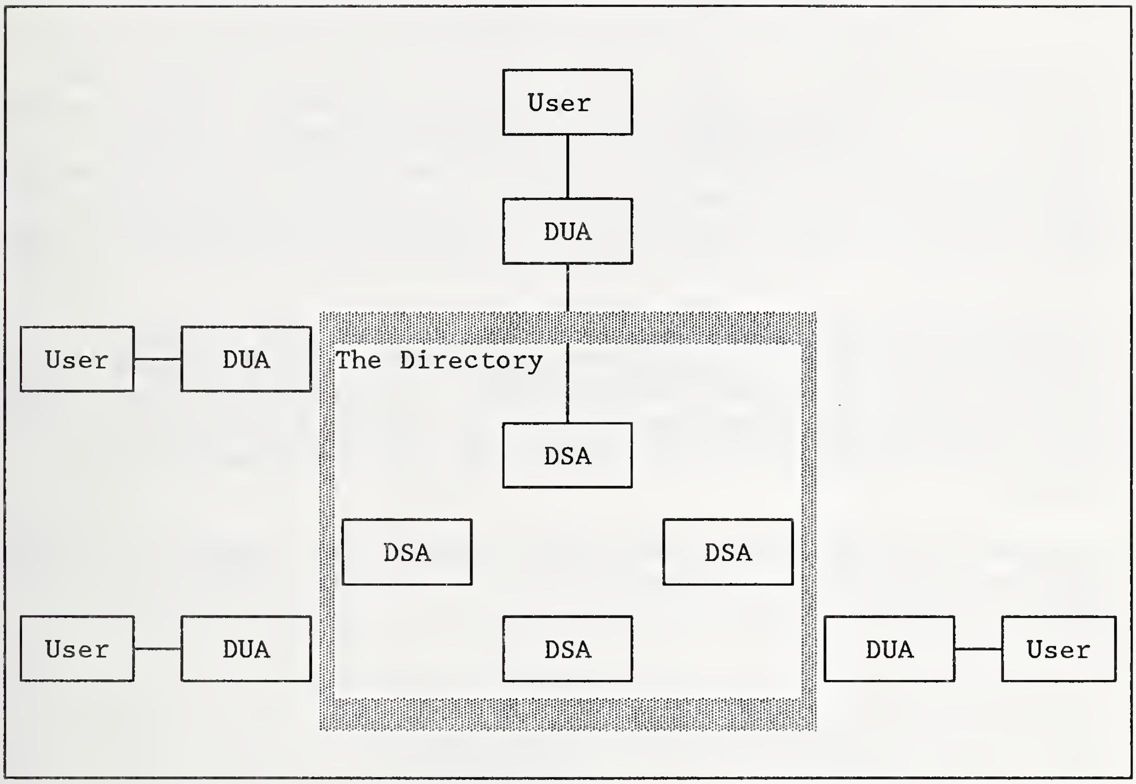


Figure 11.3 Distributed Directory Model

11.3 STATUS

This version was completed in December, 1988.

11.4 Use of Directories

Given the rapid multiplication and expansion of OSI applications, telecommunication systems and services, there is growing need for users of OSI applications, as well as the applications themselves, to communicate with each other. In order to facilitate their communications, a Directory protocol, as referenced in these agreements, has been tailored to meet their respective needs.

In one instance, The Directory will be used as a service to provide humans, in an on-line fashion, rapid and easy retrieval of information useful for determining what telecommunications services are available, and/or how to access, and address their correspondents. Further, service providers offering such a Public Directory may also use this service internally with other various telecommunications services (e.g., MHS) for the proper addressing of calls or messages. Likewise, this does not preclude the usage of these agreements to similarly generate a privately operated Directory that supports both human and application information exchanges.

In another instance, The Directory, will be used as a service by computer applications without direct human involvement. One important service is to provide Presentation Address resolution for named objects, on behalf of OSI applications. The Directory may be used by applications to search for objects (i.e., Application Entities), without direct human involvement, by the use of the "search" or "list" operations.

To support the many possible usages, The Directory is a general purpose system. It is capable of storing data of many different forms as attributes within entries, and is also capable of supporting simple or complex hierarchical structures, with variations in structure possibly occurring between one part of The Directory and another.

Compliant DSA implementations should safeguard this generality, where possible, by placing the minimum of restrictions in "hard-wired" form.

11.5 Directory ASEs and Application Contexts

This section highlights the ASEs (Application Service Elements) and Application Contexts defined in the Directory Documents and of concern to these Agreements. The functionality of the Directory AEs (DUAs and DSAs) is defined by a set of ASEs, each Directory ASE specifying a set of Directory operations.

The interaction between these AEs is described in terms of their use of ASEs. This specific combination of a set of ASEs and the rules for their usage defines an application context.

The following ASEs are described in the Directory Documents:

- Directory Read ASE
- Directory Chained Read ASE
- Directory Search ASE
- Directory Chained Search ASE
- Directory Modify ASE
- Directory Chained Modify ASE

ROSE and ACSE also form part of the Directory Application Contexts.

The following Application Contexts are described in the Directory Document:

- Directory Access Context
- Directory System Context

11.6 Schemas

There are four (4) major topics that relate to schemas:

11.6.1 Support of Structures and Naming Rules

DSAs shall be cable of supporting (subject to refinements laid down in these Agreements) the structure and naming rules defined in the Directory Documents, Part 7, Annex B.

Part 7, Annex B of the Directory Documents provides a framework for the basic use of the Directory in terms of the objects defined in Part 7. It does not, however, form part of the standard and, in any case, permits structures and practices which may be undesirable. The guidelines below provide tighter control within Annex B framework.

Accordingly, it is recommended that Locality entries may be subordinate to Root, Locality, Country, Organization, or Organization Unit. Only an entry subordinate to Root or Country may use a StateOrProvinceName AVA as an RDN. An entry of class GroupOfNames may not be subordinate to an entry of class Locality which is, in turn, part of an Organization/OrganizationalUnit Structure(i.e. the Locality is subordinate to an OrganizationganizationalUnit).

11.6.2 Support of Object Classes and Subclasses

DSAs shall be able to support the storage and use of the following object classes defined in the Directory Documents Part 7:

top
country
locality
organization
organizational unit
DSA
alias
application entity
organizational-person
residential-person
application-process

The support for the use of unregisterd object classes to support multiple superclasses is mandatory; however, the support of additional "MANDATORY" or "OPTIONAL" attributes as part of the unregistered object class definition is optional.

11.6.3 Support of Attribute Types

DSAs shall be able to support the storage and use of attribute type information, as defined in Part 6, including their use in naming and access to entries; they shall also support the definition of new attribute types, making use of pre-existing attribute syntaxes.

11.6.4 Support of Attribute Syntaxes

Suggested methods for the interpretation of selected Attribute Syntaxes are defined in Appendix A.

11.7 Classification of Support for Attribute Types

This section classifies directory support for selected attribute type specified in the Directory Documents.

Classification of support for selected attribute types is either mandatory or optional.

11.7.1 Mandatory Support

The Directory must be able to support these Attribute Types:

Aliased Object Name
Business Category
Common Name
Country Name
Locality Name
ISDN Address
Member
Object Class
Organization Name
Organizational Unit Name
Owner
Post Office Box

Postal Address
Postal Code
Preferred Delivery Mode
Presentation Address
Role Occupant
See Also
Serial Number
State or Province Name
Street Address
Supported Application Context
Surname
Telephone Number
Title
User Password
X.121 Address

Note: Support of these Attribute Types implies full support of the relevant Attribute Syntaxes.

11.7.2 Optional Support

Directory support of these attribute types is considered optional:

Authority Revocation List
CA Certificate
Certificate Revocation List
Description

Destination Indicator
Facsimile Telephone Number
Knowledge Information
Non-Basic Parameters
Physical Delivery Office Name
Registered Address
Search Guide
Teletex Terminal Identifier
Telex Number
User Certificate

Note: DSAs should consider initial support of the Attribute Syntax relevant to any Attribute Type for which future support is planned, in addition to those required for mandatory Attribute Types.

11.8 Introduction to Pragmatic Constraints

The following sections of this document define the pragmatic constraints to which a conformant implementation must adhere in addition to those specified in the Directory Documents. The pragmatic constraints are divided into two areas. The first includes those aspects of pragmatic constraints which apply to the scope of service. The second includes those aspects of pragmatic constraints which are specific to particular attribute types.

11.9 General Constraints

11.9.1 Character Sets

It is a requirement to support all character sets and other name forms defined in the Directory Documents Part 6. Those character sets include:

- T.61
- PrintableString
- NumericString

11.9.2 APDU Size Considerations

In the process of chaining requests it is possible that a chaining DSA may receive invoke or return APDUs that exceed its capacity:

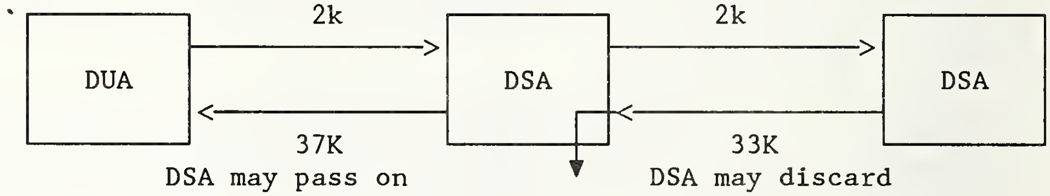


Figure 11.4 APDU Exchange

It is a minimum requirement that invoke and return result APDUs must be accepted unless they exceed 32767 octets in size; in this case they may be discarded as illustrated in the right side of Figure 11.5, and an "unwillingToPerform" error reporting service shall be used.

11.9.3 Service Control (SC) Considerations

This agreement recognizes that DUAs may automatically supply defaults for any SC parameter. The choice of default values selected (if any) is seen to be a matter of local policy and consumer needs.

11.9.4 Priority Service Control

Priority is specified as a service control argument in the Directory Documents. The following statements represent a clarification of the semantics that may be used by a DSA in interpreting and operating on this parameter.

The logical model in Figure 11.5 may be considered as an example by DSAs that implement this Service Control.

- The DSA maintains three logical queues corresponding to the three priority levels.
- The DSA Scheduler is separate and distinct from any scheduling function provided by the underlying operating system or control program services.
- The DSA Scheduler presents jobs to the Underlying Operating Services for execution and always presents

jobs of a higher priority before those of a lower priority.

- The DSA Scheduler will not preempt a request once ithas been passed to the underlying operating system service.

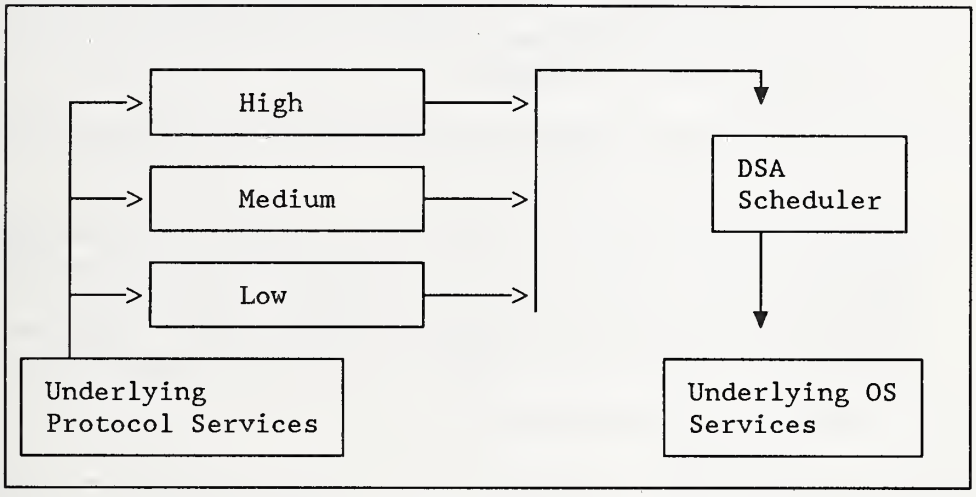


Figure 11.5 Logical DSA Application Environment

11.10 Constraints on Operations

There are no overall constraints upon service arguments or results except those implied in Section 11.9.2 of this document.

11.10.1 Filters

It is required that DSAs, at a minimum, support 8 nested "Filter" parameters, and a total limit of 32 Filter Items. If these limits are exceeded, the recipient of that SearchArgument may return the ServiceProblem "unwillingToPerform".

11.10.2 Errors

There are no constraints upon any Error service except the APDU size limit as defined in Section 11.9.2.

11.10.3 Error Reporting

Detection of Search Loop

A search operation may encounter a looping situation when the search encompasses "whole-subtree", and an alias is encountered which is a superior to some other subtree that has been encountered during the search.

DSAs should be able to detect this situation. One possible method is by:

1. Maintaining a list of the base objects of searches initiated as a consequence of Step 5 of Part 4 Section 18.7.2.2.1 of the Directory Documents (this may require an analysis of the TraceInformation field).

2. Determining whether a new base object is superior to any base object on this list.

A new base object which would cause a loop in this way should be discarded (i.e. should not cause a new search), but no error should be reported by an error-reporting service. The circumstances should be logged so that it may be reported to an appropriate Administrative Authority for rectification.

11.11 Constraints on Attribute Types

This section defines the pragmatic constraints specific to particular attribute types.

11.11.1 Attribute Values

This section describes the pragmatic constraints for attribute values. Each constraint is given in terms of a Length Constraint. The Length Constraint for a given attribute value is the number of units which a sending entity must not exceed and which a receiving entity must accept and process. A sending entity need not be capable of sending attribute values as large as the Length Constraints.

Use of Pragmatic Constraints for Strings

The Length Constraint for strings is expressed as the number of allowable characters.

Attribute Types

Table 14.1 specifies the pragmatic constraints for selected attribute types specified in the Directory Documents; many of these constraints also appear and are the same in the CCITT version of the Directory Documents.

Alphabets

T.61 Strings used as attribute values shall only encode graphic characters and spaces. They must not contain formatting characters (such as subscript) or other control characters.

Table 11.1 Pragmatic Constraints for Selected Attributes

Attribute Type	Content	Constraints	Primary Source	Notes
Aliased Object Name	Distinguished Name			Note 3
Business Category	T.61 or Printable String	ub-business-category 128	CCITT X.520	
Common Name	T.61 or Printable String	ub-common-name 64	CCITT X.520	
Country Name	Printable String	2	ISO 3166	
Description	T.61 or Printable String	ub-description 1024	CCITT X.520	About 1 Screen full
Facsimile Telephone Number	Facsimile Telephone Number	ub-telephone-number 32	CCITT X.520	Optionally includes G3 Non Basic Parameters (Upper bounds ffs
ISDN Address	Numeric String	ub-isdn-address 16	CCITT X.520	E.164 Internat'l ISDN Number
Knowledge Information	T.61 or Printable String	1024	NBS	About 1 screen full
Locality	T.61 or Printable String	ub-locality-name 128	CCITT X.520	
Member	Distinguished name			Note 3
Object Class	Object Identifier	256 octets	NBS	

Table 11.1 Pragmatic Constraints for Selected Attributes (continued)

Attribute Type	Content	Constraints	Primary Source	Notes
Organization Name	T.61 or Printable String	ub-organization-name 64	CCITT X.520	
Organizational Unit Name	T.61 or Printable String	ub-organizational-unit-name 64	CCITT X.520	
Owner	Distinguished Name			Note 3
Physical Delivery Office Name	T.61 or Printable String	ub-physical-office-name 128	CCITT X.520	
Post Office Box	T.61 or Printable String	ub-post-office-box 40	CCITT X.520	
Postal Address	Postal Address	ub-postal-line 6 ub-postal-string 30	CCITT X.520	UPU
Postal Code	T.61 or Printable String	ub-postal-code 40	CCITT X.520	UPU
Presentation Address	Presentation Address	224 octets	NBS	Note 2, ISO 7498.3 & X.200
Role Occupant	Distinguished Name			Note 3
See Also	Distinguished Name			Note 3
Serial Number	Printable String	ub-serial-number 64	CCITT X.520	
State or Province Name	T.61 or Printable String	ub-state-name 128	CCITT X.520	
Street Address	T.61 or Printable String	ub-street-address 128	CCITT X.520	

Table 11.1 Pragmatic Constraints for Selected Attributes (continued)

Attribute Type	Content	Constraints	Primary Source	Notes
Surname	T.61 or Printable String	ub-surname 64	CCITT X.520	
Telephone Number	Printable String	ub-telephone-number 32	CCITT X.520	E.123
Teletex Terminal	Teletex Terminal	ub-teletex-terminal-id 1024	CCITT X.520	Optionaly includes Teletx non-basic parameters upper bound ffs)
Telex	Telex Number	ub-telex-number 14 ub-country-code 4 ub-answerback 8	CCITT X.520	Contains sequence of telex number, country code, and answerback
Title	T.61 or Printable String	ub-title 64	CCITT X.520	
User Password	Octet String	ub-user-password 128	CCITT X.520	Allow long passwords generated by machine
X.121 Address	Numeric String	ub-x121-address 15	CCITT X.520	X.121

Notes:

1. The pragmatic constraints of these parameters are defined in other standards. We will accommodate these values in our pragmatic constraints.

2. Presentation address is composed of "X" NSAP addresses, and three selectors, (20X + 32 + 16 + 16), e.g. if X = 1, this would be 84. These numbers are based on the most recent implementor's agreements. With 8 NSAP addresses this value is 224.

3. Pragmatic constraints are only applied to the individual components of DistinguishedName as defined in the Directory Documents Part 2. Not all components of a DN will necessarily be understood by an implementation.

4. Implementors should be aware that constraints on Postal Address may not be sufficient for some markets.

11.12 Conformance

The following sections will describe various aspects of Directory conformance. It should be noted that conformance to the various ASEs and conformance to the Authentication Framework are viewed as separate issues and are presented in that context.

11.12.1 DUA Conformance

Conformance requirements for DUAs are adequately specified in the Directory Documents, Part 5, Section 9.1. It should be noted that DUA conformance can only be demonstrated by exercising the interface that it provides to its user.

It is recognized that DUAs will be widely differing in nature:

- Some are intended to support human users, some application users

- Particular DUAs may not support particular operations because the application that they support has no requirement; others will be general purpose, and will support all operations.

- Some DUAs will have a fixed view of the Directory content and structure, reflecting the usage of The Directory by a particular application; others will have a more flexible view which can be adapted to new usages.

- Some DUAs will provide automatic referral services with automatic establishment and release of associations; others will place the burden on the user.

- Some DUAs will provide a variety of authentication means; others will support simple authentication only

- Some DUAs will handle operations synchronously; others will have the capability of maintaining several identifiable dialogues with The Directory at one time.

No general implementation agreements are spelled out in respect of these possibilities.

11.12.2 DSA Conformance

Basic conformance requirements for a DSA are defined in the Directory Documents, Part 5, Section 9.2.

Centralized

A centralized DSA is defined as one that contains its entire relevant DIT; it follows that it will not make use of the DSP or generate referral responses. Since this model only contains a single DSA it is not subject to DSA interworking issues and will always provide a consistent level of service and results. A centralized DSA must be fully "protocol" conformant to the DAP.

Cooperating

In a distributed directory, responsibility for various portions of the DIT may be "distributed" among multiple DSAs. On a per operation basis we define a DSA to be "holding" when it is responsible for the fragment of the DIB in which a given entry will appear if it exists; we define a DSA to be "propagating" when it is unable to complete the name resolution process. All DSAs must be capable of acting as a "holder" and a"propagator."

11.12.3 Directory Systems Conformance Classes

A DSA implementation shall satisfy the conformance requirements as defined in the Directory Documents, Part 5, Section 9.2, and shall support the "Versions" argument of "Bind".

Per the conformance Section of the Directory Documents, a DSA shall conform to the abstract syntax of the attribute types for which conformance is claimed. These attribute types shall include those required by Section 11.7.1 of this Implementor's Agreement.

Additionally, an implementation conformant to these agreements shall state which of the following conformance classes it implements:

Conformance Class 0 -- Centralized DSA

A DSA conformant to this class only supports the DirectoryAccessAC.

As the performance of Search and List operations can consume significant resources, the policies of some centralized DSAs may be such that these operations will not be performed. For these cases, the reply to requests for such operations would be a ServiceError with the "unwillingToPerform" ServiceProblem.

Conformance Class 1 -- Distributed DSA

A DSA implementation conformant to this class must implement all the operations in the ASEs that are part of the Application context for which it claims conformance. It **must** support the DirectoryAccessAC and it may optionally support the DirectorySystemAC.

11.12.4 Authentication Conformance

A Directory System may choose to implement various levels of authentication (Directory Documents, Part 8). We define the following levels of authentication in the DS:

- No authentication at all; (**None**)
- **Simple uncorroborated** identification **without** verification
- **Simple uncorroborated** authentication **with** verification: verified identification without a password.
- **Simple corroborated** authentication:
  Verified identification with a password;
  intended to make masquerading difficult.
- **Strong authentication:**
  identification with verification using cryptographic techniques intended to make masquerading, in practical terms, nearly impossible.

The "Authentication Framework" document describes the specific goal of each authentication level; we have listed several practical uses of the various levels:

NONE

No authentication may be required for associations with a DSA containing public information; DSAs operating on a private network in a controlled environment may implicitly trust all connections andhave no requirement for authentication.

SIMPLE UNCORROBORATED

authentication may be desired to maintain access statistics or in a private network where the intitiator is implicitly trusted and there is no need to incur the additional overhead of more sophisticated authentication methods.

SIMPLE CORRORORATED authentication may be necessary in situations where strong authentication is not practical, (i.e. international connection, no knowledge of algorithms in use, etc).

STRONG authentication will be required for secure environments.

A DSA that implements Simple Corroborated Authentication will check the user password by means of a compare operation on the user's entry. If no user password is supplied (Simple Uncorroborated Authentication) the DSA will validate the presence of the entry for the user, by a read operation or otherwise. The authentication will fail if the password is incorrect or if the user's entry does not exist.

A DSA that implements Simple Uncorroborated Authentication without verification will accept simple credentials without validating them.

11.12.5 Authentication Conformance Classes

We define the following two (2) conformance classes for the support of authentication:

- o None, Simple Uncorroborated Authentication without verification

- o None, all forms of Simple Authentication (unprotected)

Consideation of protected simple and strong authentication is not part of this agreement.

11.13 Distributed Operations

The following requirements apply to DSAs supporting distributed operations:

DSAs supporting authentication (e.g. simple authentication by name and password) must be able to invoke DSP operations to carry out authentication by reference to other DSAs. Thus all such DSAs must support the DSP protocol. The requirement is implied by the Directory Documents.

11.13.1 Referrals and Chaining

It is recommended that a DSA which has chained a request act upon any referrals it receives rather than returning them to the requestor if the "PreferChaining" service control is present.

11.14 Underlying Services

Note: The subsections following specify requirements over and above those given in the Directory Documents.

11.14.1 ROSE

It should be noted that support of "abandon" implies support of operation class 2.

11.14.2 Session

All directory implementations are required to support Session Version 2.

Editor's Note: The movement of 11.14.3 from Ongoing to Stable was not presented to Plenary for approval.

11.14.3 ACSE

The A-ABORT service is required by association-accepting DSAs to escape unwanted associations, which, under the ROSE protocol, they cannot release. In all other cases (association-initiating DSAs and DUAs) it may be preferable (though not required) to escape associations using UNBIND rather than abort.

The aborting DUA or DSA may optionally use the user information field of the A-ABORT. Such information, however, is only meaningful for diagnostic purposes and its use is not covered by these Agreements.

11.15 Access Control

Guidelines relation to access control can be found in Annex F of the Directory Documents, Part 2.

11.16 Test Considerations

This section outlines some items that implementors may wish to consider in terms of testing expectations; additionally, future conformance testers may wish to consider these items when developing tests.

11.16.1 Major elements of Architecture

One important aspect of testing is to confirm the correct behavior of DSAs and DUAs in respect of major elements of the directory architecture.

Such major elements include:

o Conformance Statement

o Distinguished names (e.g., name resolution, equivalence of various forms)

o Entries and Attributes (e.g., accessibility by operations, compliance with rules)

o Handling of distributed operations (e.g.,naming contexts and knowledge)

o Schemas

- Structure rules (e.g., storage and maintenance of structure and of naming rules)
- Object classes and sub-classes (e.g., storage and extension of rules for object attributes)
- Attribute types (e.g., storage and maintenance of syntax classes and rules for multi or single valued attributes)
- Attribute syntax (e.g. maintenance and support for attribute value testing and matching, to specification for a defined set of attribute types)

oOperations

- all operations
- correct function
- correct result
- correct responses

- o Aliases (e.g., correct resolution, error responses)
- o Authentication and Access Control (e.g., limitation of modify access)
- o ROSE (e.g., correct handling of invokes, results, rejects, and invoke ids)
- o ACSE (e.g., association establishment / refusal for invalid application contexts, etc.)

11.16.2 Search Operation

Testing of support for filter items should be reasonable. It is not expected that DSAs will be able to handle worst case testing in this area.

11.17 Errors

This section provides clarification of the semantics of various operation errors and implementation guidelines on their usage.

11.17.1 Permanent vs. Temporary Service Errors

The usage of the Service Errors busy, unavailable and unwillingToPerform requires some clarification:

The error busy is particularly transient. It is returned when one or more of The Directory's internal resources are being used to their capacity and, hence, the requested operation cannot, for the moment, be performed. The Directory should be able to recover from this type of resource depletion after a short while.

The error unavailable is also temporary but somewhat less transient. It indicates that The Directory (or some part of it) is currently unavailable and may continue to be unavailable for a reasonably long period of time. for example, this error is returned when a given DSA is functionally disabled, or when a specific part of the DIB is undergoing reconfiguration.

The error unwillingToPerform has a permanent connotation. It indicates that The Directory cannot perform the requested operation because it would require resources beyond its capacity. For example, this error may be returned by a DSA if, satisfying a requested would result in the generation of an APDU in excess of 32767 octets.

11.17.2 Guidelines for Error Handling

11.17.2.1 Introduction

This subsection provides a recommended mapping of error situations which may be encountered to ROSE Rejects or to the errors provided in the DAP and DSP protocols of the Directory Documents.

The Directory Documents are not adequately definitive about the handling of errors. In this document, more explicit guidelines are given.

Error situations are defined by:

- o Symptom, the manner in which the error was detected

- o Situation, the circumstance or phase during which the error was detected. For each possible situation, the error-handling procedure needs to be defined.

11.17.2.2 Symptoms

This subsection describes a set of symptoms (not necessarily exhaustive). Each is identified by a title for reference later in the section; this title is not intended to imply any particular usage in a particular implementation.

E_ACCESS

The initiator has insufficient access rights to carry out this operation.

E_ALIAS_DEREF

An alias has been encountered while a previous alias was being dereferenced, or a name contained an alias plus one or more additional RDNs when the dontDereferenceAliases service control was being used or the name supplied in an operation that precludes alias dereferencing contained an alias plus one or more additional RDNs.

E_ALIAS_LOOP

During a whole-subtree search operation, an alias has been encountered which would lead to a loop (i.e. the alias points to an entry which is superior to entries which have already been evaluated in carrying out the search).

E_ALIAS_PROBLEM

An Alias has been encountered, but the entry to which it points does not exist.

E_ARG_BOUNDS

The argument does not comply with pragmatic constraints (defined locally or by functional standards).

E_ARG_SYNTAX

An operation argument either has incorrect ASN.1 encoding or it has correct ASN.1 encoding, but does not comply to the syntax as defined in the Directory Documents.

Notes: 1. Within BindArgument, additional elements are permitted, to allow future extensions, and do not create an error situation.

2. Errors within attribute values are not included in this codification (see E_ATT_SYNTAX).

E_ARG_VIOL

An operation argument has correct syntax, but it violates additional rules and constraints laid down by the Directory Documents (such as the use of a Priority integer value whose meaning is undefined).

Notes: 1. Within a Relative Distinguished Name, having two AVAs of the same attribute type is an error which is covered by E_DN, and not by E_ARG_VIOL.

2. Errors within attribute values are not included in this codification (see E_ATT_SYNTAX).

E_ATT_BOUNDS

An attribute value does not comply with bounds specified either by the Directory Documents, or by functional standards.

E_ATT_OR_VALUE_EXISTS

Within an entry, an attribute or attribute value already exists, causing an error situation.

E_ATT_SYNTAX

An attribute value either has incorrect ASN.1 encoding or it has correct ASN.1 encoding but does not comply with the ASN.1 encoding defined by the attribute type.

E_ATT_VALUE

An attribute value, although of correct ASN.1 encoding, and conformant with the syntax defined for the attribute type, is not compliant with other rules (e.g. a non-ISO 3166 country name encoding).

E_AUTHENTICATION

The authentication offered does not match that required by the object being authenticated.

E_BUSY

The DSA is unable to handle this operation at this time (but it may be able to do so after a short while).

E_CHAIN

The DSA must use chaining to carry out this operation, but is prohibited from doing so by Service Controls.

E_CREDENTIALS

The credentials offered do not match those of the object with which authentication is taking place.

E_DBE

An inconsistency has been detected in the DSA's data base, which may be localised to a particular entry or set of entries

E_DN

A DN contains an RDN with two AVAs of the same attribute type.

E_DSA

A DSA to which chaining is taking place is unable to respond.

E_ENTRY_EXISTS

An entry of the given name already exists, causing an error.

E_ILLEGAL_ROOT_OBJ

Root's DN has been supplied as the object of a Read, Compare, AddEntry, RemoveEntry, ModifyEntry, ModifyRDN, or as the Base Object of a single level search.

E_ILLEGAL_ROOT_VAL

Root's DN has been supplied illegally as an attribute value (e.g. as an Aliased Object Name).

E_LIST_ON_LEAF

An attempt has been made to carry out a list operation on a leaf entry.

E_LOOP

A loop has been detected in the knowledge information within the system.

E_MATCH

The attribute specified does not support the required matching capability.

E_MISSING_AVA

When creating, or after modifying, an entry, an AVA in the entry's RDN is not represented within the entry's set of attributes.

E_MISSING_OBJECT_CLASS

When creating an entry, the entry does not possess an object class

E_MULTI_DSA

The operation is an update operation which affects other DSAs.

E_NAMING_VIOLATION

The name of the new or modified entry is incompatible with its object class.

E_NO_SUCH_ATT

The specified attribute has not been found.

E_NO_SUCH_OBJECT

The specified entry has not been found.

E_NON_LEAF_OPERATION

The operation being attempted is illegal except on a leaf.

E_NOT_SINGLE_VALUED

An attribute, registered as single-valued, has been found with more than one value.

E_OBJECT_CLASS_MOD

An (illegal) attempt has been made to alter or remove an object class attribute.

E_OBJECT_CLASS_VIOL

There is a schema violation (e.g. missing mandatory attribute, or non-allowed attribute present).

E_REFERENCE
An erroneous reference has been detected (e.g. DSA cannot handle name even as far as the number of RDNs that have already been resolved).

E_SYSTEM_PERM

A serious and permanent software or system error has been detected which prevents completion of the operation.

E_SYSTEM_TEMP

A serious but temporary software or system error has been detected which prevents completion of the operation.

E_TIMEOUT

The operation has not completed within the allotted time.

E_UNABLE_TO_COMPLETE

The DSA is unable to complete this operation, or others like it. (This applies particularly to search.)

E_UNABLE_TO_PROCEED

The DSA cannot satisfy the operation after receiving it on the basis of a valid non-specific subordinate reference.

E_UNDEFINED_ATT

An unregistered attribute has been encountered.

E_UNSUPPORTED_OC

The object class of the entry is not supported as a valid object class for entries within this DSA.

E_VERSION

An unexpected version has been found in Bind.

E_ZERO_VALUES

An attribute has been found (for example, as a result of a modify-entry operation) with no values.

11.17.2.3 Situations

The following situations are recognized within which particular symptoms may give rise to distinct error actions:

BIND-LOCAL

A bind is being attempted; either the entry named is (or should be) within a local naming context, or name resolution is being carried out on the part of the name that is known locally.

BIND-REMOTE

A bind is being attempted, and the entry named is not within a local naming context; remote validation of credentials is being carried out.

NAME-RESOLUTION

Name resolution is being carried out.

ADD-ENTRY-NAME-RESOLUTION

During an add entry operation, name resolution has been successfully accomplished on the superior object, and is now being carried out to determine whether the new entry already exists.

ADD-ENTRY

The entry is being generated.

MODIFY-ENTRY

The entry is being modified.

MODIFY-RDN

The RDN is being modified.

REMOVE-ENTRY

The entry is being removed.

READ

The entry is being read.

COMPARE

A Compare operation is being carried out on the entry.

LIST

A List operation is being carried out on the entry.

SEARCH-FILTER

A Search operation is being carried out; the filter is being evaluated or acted upon.

SEARCH-ENTRY

A Search operation is being carried out; the required entry information is being evaluated or acted upon.

ABANDON

An Abandon operation is being carried out.

TRACE-EVALUATION

The trace element is being evaluated for loops.

11.17.2.4 Error Actions

In the following tables, the recommended actions are identified for all the error symptoms in each situation in which it may be encountered.

The notation is as follows:

Rej - A Reject operation is generated, with problem mistyped-argument.

Ab(ppp) - Abandon Failed Error is generated. ppp may take values codified as follows:

CA - Cannot abandon
NSO - No such operation
TL - Too late

A(ppp) - Attribute Error is generated. ppp may take values:

AVE - Attribute or value already exists
CV - Constraint violation
IAS - Invalid attribute syntax
IM - Inappropriate matching
NSA - No such attribute
UAT - Undefined attribute type

N(ppp) - NameError is generated. ppp may take values:

ADP - Alias dereferencing problem
AP - Alias problem
IAS - Invalid attribute syntax
IL - Is leaf
NSO - No such object

SC(ppp) - Security Error is generated. ppp may take values:

IA - Inappropriate authentication
IAR - Insufficient access rights
IC - Invalid credentials
IS - Invalid signature
PR - Protection required

S(ppp) - Service Error is generated. ppp may take values:

B - Busy
CR - Chaining required
IR - Invalid reference
LD - Loop detected

TLE - Time limit exceeded
UA - Unavailable
UAP - Unable to proceed
UWP - Unwilling to perform

U(ppp) - Update Error is generated. ppp may take values:

AMD - Affects multiple DSA
EAE - Entry already exists
NAN - Not allowed on non-leaf
NAR - Not allowed on RDN
NV - Naming violation
OCV - Object class violation
OMP - Object class modification prohibited

In addition, bracketed numerals give notes.

		Bind		Add
	Bind	Remote	Name	Entry
	Local	Resol'n	Resol'n	Name
E_ACCESS	-	-	SC(IAR)	SC(IAR)
E_ALIAS_DEREF	SC(IC)	SC(IC)	N(ADP)	-
E_ALIAS_LOOP	-	-	-	-
E_ALIAS_PROBLEM	SC(IC)	SC(IC)	N(AP)	-
E_ARG_BOUNDS	(8)	(7)	S(UWP)(12)	S(UWP)(12)
E_ARG_SYNTAX	(1)	(1)	Rej	Rej
E_ARG_VIOL	(1)	(1)	Rej	Rej
E_ATT_BOUNDS	SC(IC)	(7)	N(IAS)	N(IAS)
E_ATT_OR_VALUE_EXISTS	-	-	-	-
E_ATT_SYNTAX	SC(IC)	(7)	N(IAS)	N(IAS)
E_ATT_VALUE	SC(IC)	(7)	N(IAS)	N(IAS)
E_AUTHENTICATION	SC(IA)	SC(IA)	-	-
E_BUSY	S(UA)	S(UA)	S(B)	S(B)
E_CHAIN	-	-	S(CR)	-
E_CREDENTIALS	SC(IC)	SC(IC)	-	-
E_DBE	S(UA)	-	S(UA)	S(UA)
E_DN	SC(IC)	SC(IC)	N(NSO)	U(NV)
E_DSA	-	S(UA)	S(UA)	S(UA)
E_ENTRY_EXISTS	-	-	-	U(EAE)
E_ILLEGAL_ROOT_OBJ	SC(IC)	SC(IC)	-	N(NSO)
E_ILLEGAL_ROOT_VAL	SC(IC)	(7)	N(IAS)	N(IAS)
E_LIST_ON_LEAF	-	-	-	-
E_LOOP	-	S(UA)	S(LD)	-
E_MATCH	SC(IC)	SC(IC)	A(IM)	A(IM)
E_MISSING_AVA	-	-	-	-
E_MISSING_OBJECT_CLASS	-	-	-	-
E_MULTI_DSA	-	-	-	S(AMD)
E_NAMING_VIOLATION	-	-	-	U(NV)
E_NON_LEAF_OPERATION	-	-	-	-
E_NOT_SINGLE_VALUED	-	-	-	-
E_NO_SUCH_ATT	-	-	-	-
E_NO_SUCH_OBJECT	SC(IC)	SC(IC)	N(NSO)	-
E_OBJECT_CLASS_MOD	-	-	-	-
E_OBJECT_CLASS_VIOL	-	-	-	-
E_REFERENCE	-	S(UA)	S(IR)	-
E_SYSTEM_PERM	S(UA)	-	S(UWP)	S(UWP)
E_SYSTEM_TEMP	S(UA)	-	S(UA)	S(UA)
E_TIMEOUT	S(UA)	(9)	S(TLE)	S(TLE)
E_UNABLE_TO_COMPLETE	-	-	-	-
E_UNABLE_TO_PROCEED	-	(2)	(2)	-
E_UNDEFINED_ATT	SC(IC)	-	(3)	U(NV)
E_UNSUPPORTED_OC	-	-	-	-
E_VERSION	S(UA)	-	-	-
E_ZERO_VALUES	-	-	-	-

	Add- Entry	Modify Entry	Trace Evaluation
E_ACCESS	SC(IAR)	SC(IAR)	-
E_ALIAS_DEREF	-	-	-
E_ALIAS_LOOP	-	-	-
E_ALIAS_PROBLEM	-	-	-
E_ARG_BOUNDS	S(UWP)(12)	S(UWP)(12)	-
E_ARG_SYNTAX	Rej	Rej	Rej
E_ARG_VIOL	Rej	Rej	Rej
E_ATT_BOUNDS	A(CV)	A(CV)	(7)
E_ATT_OR_VALUE_EXISTS	A(AVE)	A(AVE)	-
E_ATT_SYNTAX	A(IAS)	A(IAS)	(7)
E_ATT_VALUE	A(IAS)	A(IAS)	(7)
E_AUTHENTICATION	-	-	-
E_BUSY	S(B)	S(B)	-
E_CHAIN	-	-	-
E_CREDENTIALS	-	-	-
E_DBE	S(UA)	S(UA)	-
E_DN	-	-	-
E_DSA	-	-	-
E_ENTRY_EXISTS	-	-	-
E_ILLEGAL_ROOT_OBJ	N(NSO)	N(NSO)	-
E_ILLEGAL_ROOT_VAL	A(IAS)	A(IAS)	(7)
E_LIST_ON_LEAF	-	-	-
E_LOOP	-	-	-
E_MATCH	-	A(IM)	(7)
E_MISSING_AVA	U(NAR)	U(NAR)	-
E_MISSING_OBJECT_CLASS	U(OCV)	U(OMP)	-
E_MULTI_DSA	-	-	-
E_NAMING_VIOLATION	-	-	-
E_NON_LEAF_OPERATION	-	-	-
E_NOT_SINGLE_VALUED	A(CV)	A(CV)	-
E_NO_SUCH_ATT	-	A(NSA)	-
E_NO_SUCH_OBJECT	-	-	-
E_OBJECT_CLASS_MOD	-	U(OMP)	-
E_OBJECT_CLASS_VIOL	U(OCV)	U(OCV)	-
E_REFERENCE	-	-	-
E_SYSTEM_PERM	S(UWP)	S(UWP)	S(UWP)
E_SYSTEM_TEMP	S(UA)	S(UA)	S(UA)
E_TIMEOUT	S(TLE)	S(TLE)	-
E_UNABLE_TO_COMPLETE	-	-	-
E_UNABLE_TO_PROCEED	-	-	-
E_UNDEFINED_ATT	A(UAT)	A(UAT)	(7)
E_UNSUPPORTED_OC	U(OCV)	-	-
E_VERSION	-	-	-
E_ZERO_VALUES	A(CV)	A(CV)	-

	Modify RDN	Remove Entry	Read	Compare
E_ACCESS	SC(IAR)	SC(IAR)	SC(IAR)	SC(IAR)
E_ALIAS_DEREF	-	-	-	-
E_ALIAS_LOOP	-	-	-	-
E_ALIAS_PROBLEM	-	-	-	-
E_ARG_BOUNDS	S(UWP)(12)	-	S(UWP)(12)	S(UWP)(12)
E_ARG_SYNTAX	Rej	Rej	Rej	Rej
E_ARG_VIOL	Rej	Rej	Rej	Rej
E_ATT_BOUNDS	N(IAS)	-	-	A(CV)
E_ATT_OR_VALUE_EXISTS	-	-	-	-
E_ATT_SYNTAX	N(IAS)	-	-	A(IAS)
E_ATT_VALUE	N(IAS)	-	-	A(IAS)
E_AUTHENTICATION	-	-	-	-
E_BUSY	S(B)	S(B)	S(B)	S(B)
E_CHAIN	-	-	-	-
E_CREDENTIALS	-	-	-	-
E_DBE	S(UA)	S(UA)	S(UA)	S(UA)
E_DN	A(CV)	-	-	A(IAS)
E_DSA	-	-	-	-
E_ENTRY_EXISTS	U(EAE)	-	-	-
E_ILLEGAL_ROOT_OBJ	N(NSO)	N(NSO)	N(NSO)	N(NSO)
E_ILLEGAL_ROOT_VAL	N(IAS)	-	-	A(IAS)
E_LIST_ON_LEAF	-	-	-	-
E_LOOP	-	-	-	-
E_MATCH	A(IM)	-	-	A(IM)
E_MISSING_AVA	-	-	-	-
E_MISSING_OBJECT_CLASS	-	-	-	-
E_MULTI_DSA	S(AMD)	S(AMD)	-	-
E_NAMING_VIOLATION	U(NV)	-	-	-
E_NON_LEAF_OPERATION	U(NAN)	U(NAN)	-	-
E_NOT_SINGLE_VALUED	A(CV)	-	-	-
E_NO_SUCH_ATT	-	-	A(NSA)(4)	A(NSA)(4)
E_NO_SUCH_OBJECT	-	-	-	-
E_OBJECT_CLASS_MOD	-	-	-	-
E_OBJECT_CLASS_VIOL	U(OCV)	-	-	-
E_REFERENCE	-	-	-	-
E_SYSTEM_PERM	S(UWP)	S(UWP)	S(UWP)	S(UWP)
E_SYSTEM_TEMP	S(UA)	S(UA)	S(UA)	S(UA)
E_TIMEOUT	S(TLE)	S(TLE)	S(TLE)	S(TLE)
E_UNABLE_TO_COMPLETE	-	-	-	-
E_UNABLE_TO_PROCEED	-	-	-	-
E_UNDEFINED_ATT	A(UAT)	-	A(NSA)(4)	A(NSA)
E_UNSUPPORTED_OC	-	-	-	-
E_VERSION	-	-	-	-
E_ZERO_VALUES	(11)	-	-	-

	List (Filter)	Search (Entry)	Search	Abandon
E_ACCESS	SC(IAR)	SC(IAR)	SC(IAR)	-
E_ALIAS_DEREF	-	(5)	-	-
E_ALIAS_LOOP	-	(5)	-	-
E_ALIAS_PROBLEM	-	(5)	-	-
E_ARG_BOUNDS	S(UWP)(12)	S(UWP)(12)	S(UWP)(12)	-
E_ARG_SYNTAX	Rej	Rej	Rej	Rej
E_ARG_VIOL	Rej	Rej	Rej	-
E_ATT_BOUNDS	-	A(CV)	-	-
E_ATT_OR_VALUE_EXISTS	-	-	-	-
E_ATT_SYNTAX	-	A(IAS)	-	-
E_ATT_VALUE	-	A(IAS)	-	-
E_AUTHENTICATION	-	-	-	-
E_BUSY	S(B)	S(B)	S(B)	-
E_CHAIN	-	-	-	-
E_CREDENTIALS	-	-	-	-
E_DBE	S(UA)	S(UA)	S(UA)	-
E_DN	-	A(IAS)	-	-
E_DSA	-	(5)	-	-
E_ENTRY_EXISTS	-	-	-	-
E_ILLEGAL_ROOT_OBJ	-	(10)	-	-
E_ILLEGAL_ROOT_VAL	-	A(IAS)	-	-
E_LIST_ON_LEAF	N(IL)	-	-	-
E_LOOP	-	(5)	-	-
E_MATCH	-	A(IM)	-	-
E_MISSING_AVA	-	-	-	-
E_MISSING_OBJECT_CLASS	-	-	-	-
E_MULTI_DSA	-	-	-	-
E_NAMING_VIOLATION	-	-	-	-
E_NON_LEAF_OPERATION	-	-	-	-
E_NOT_SINGLE_VALUED	-	-	-	-
E_NO_SUCH_ATT	-	-	-	-
E_NO_SUCH_OBJECT	-	-	-	-
E_OBJECT_CLASS_MOD	-	-	-	-
E_OBJECT_CLASS_VIOL	-	-	-	-
E_REFERENCE	-	-	-	-
E_SYSTEM_PERM	S(UWP)	S(UWP)	S(UWP)	Ab(CA)
E_SYSTEM_TEMP	S(UA)	S(UA)	S(UA)	Ab(CA)
E_TIMEOUT	S(TLE)	S(TLE)	S(TLE)	-
E_UNABLE_TO_COMPLETE	S(B)	S(B)	S(B)	Ab(CA)
E_UNABLE_TO_PROCEED	-	-	-	-
E_UNDEFINED_ATT	-	(6)	(6)	-
E_UNSUPPORTED_OC	-	-	-	-
E_VERSION	-	-	-	-
E_ZERO_VALUES	-	-	-	-

Notes:

1. Use A-U-ABORT. Note, however, that extra elements are permitted here.

2. An "unable-to-proceed" error becomes SC(IC) for bind and N(NSO) for operations if no DSA contacted can located the object.

3. An undefined attributed encountered during name resolution is only an error - N(NSO) - if the entry is identified as local. See also Note 10 below.

4. The A(NSA) condition is reserved in the case of "read" for the situation when no attribute of the specific list provided can be returned (for reasons that include security errors).

5. Any failure to propagate a search causes abandonment of that part of the search.

6. Undefined attributes are regarded as not matched or found, but cause no errors in search.

7. This error, if detected, should be ignored; processing continues.

8. This error would occur as a result of a bind argument with a name containing too many RDNs for the DSA. Use either S(UA) or S(IC).

9. DSAs should use the time-limit service control with local timeout to limit the remote validation of credentials; if the operation fails as a result, S(UA) is used.

10. For a single-entry search, N(NSO) may be used.

11. Either the whole attribute should be removed, or the deleteOldRDNflag should be ignored.

12. Wherever S(UWP) appears in the above tables beside E_ARG_BOUNDS, a ROSE "Rej" is also admissable.

<u>11.17.2.5 Reporting</u>

In addition to the use of error-reporting services, DSAs should implement logging services to assist in management of the Directory. The following (not necessarily complete) classes of error should be logged as described below.

Errors indicating attempted breaches of security.

Errors indicating local software or hardware malfunction.

Errors indicating malfunction or other unacceptable behavior on the part of the invoker of an operation.

Errors indicating loss of chaining service by another DSA.

Error conditions that would be difficult to diagnose with the level of detail supplied over the protocol.

Aborts and other exceptional communications events.

The form and accessibility of any such logs is for further study.

11.18 Appendix A: Maintenance of Attribute Syntaxes

11.18.1 Introduction

The attribute types defined in the Directory Documents, Part 6, and listed in Table 11.1 have requirements, in DSAs which support them, for underlying algorithms that:

- Check attribute values for syntactical correctness and compliance with pragmatic constraints.

- Match attribute values (comparing for equality, for matching substrings, and for relative ordering).

11.18.2 General Rules

A DSA may receive a legitimately encoded attribute or AVA that is unsupported by the DSA. If the DSA is not required to act on it, or to store it within an entry, it shall handle it by passing it on without error. Such attributes may also be used in search filter-item definitions: in this case, no error is reported, but the attribute type or value shall be deemed to be unmatched for all entries in the DSA. This rule applies to occurrences of attributes in both operation arguments and results.

Conversely, a DSA must return a suitable error if an operation requires it to act on or store an attribute or AVA of type unsupported by the DSA. This constraint applies even for AVAs that are contained in attributes that take names as values, since the DSA will be unable correctly to match the attribute values without this attribute information.

11.18.3 Checking Algorithms

The subsections below give additional checks (beyond those directly implied by the Directory Documents) which shall be applied to attributes before they are stored in the DSA.

11.18.3.1 distinguishedNameSyntax

Each component AVA must be checked, unregistered attribute types comprising an error; check also that no two AVAs in the same RDN have the same attribute type.

11.18.3.2 integerSyntax

Local implementations may apply local limitations.

11.18.3.3 telephoneNumberSyntax

The value of policing further rules is for further study (this applies also to telexNumber, teletexTerminalIdentifier, facsimileTelephoneNumber, G3FacsimileNonBasicParameters, x121Address, and iSDNAddress).

11.18.3.4 countryName

The value must be checked for compliance with ISO3166: 1981 (E/F). (Note that from time to time further codes may be allocated.)

11.18.3.5 preferredDeliveryMethod

The values of the integer elements should not be restricted.

11.18.3.6 presentationAddress

No further checks should be applied.

11.18.4 Matching Algorithms

Matching algorithms are conveniently defined in terms of a two-step process:

1. Take the checked reference value, and the value to be matched, and, if necessary, reduce them to a canonical (i.e. standard) form (normalization) appropriate to each attribute syntax.

2. Carry out the comparison in the specified way (e.g. equality, substrings or ordering) using the appropriate rules for the value - character string, integer, boolean, etc.

   Note that the lexical ordering of character strings (when supported) may be subject to local rules.

IMPORTANT NOTE

The combination of normalization and comparison may be replaced, in a particular implementation, by equivalent procedures.

Additional notes on normalization are given below.

11.18.4.1 UTCTimeSyntax

If the "seconds" field is absent, it shall be inserted, and set to "00", and the form converted to the "Z" form. Note. The normalization strategy does not match times where the stored form omits the seconds field, and the compared form contains it, e.g,.

```
8804261919260Z
     88042619192636Z
```

(It might have been expected that these two forms, which coincide in time to within a few seconds, would be considered identical.)

11.18.4.2 distinguishedNameSyntax

For each attribute value, carry out normalization in accordance with the normalization rules defined for the type (if registered); values corresponding to unregistered attribute types are left unchanged at this stage.

11.18.4.3 caseIgnoreListSyntax

To facilitate matching, particularly for substrings, normalization may be considered in terms of a representation which replaces the separate ASN.1 elements by a single string with a delimeter.

11.19 APPENDIX B Glossary

The following abbreviations may be useful; not all are used within these agreements.

ACL	Access Control List
ACSE	Association Control Service Element
ADDMD	Administration Directory Management Domain
AETitle	Application Entity Title
APDU	Application Protocol Data Unit
ASE	Application Service Element
ASN.1	Abstract Syntax Notation - 1
AVA	Attribute Value Assertion
B-RM	Basic Reference Model
CA	Certification Authority
CCITT	The International Telegraph and Telephone Consultative Committee
CEN	Committee for European Normalization
CENELEC	Committee for European Normalization Electronique
CEPT	(Committee of European Posts and Telephones)
COS	Corporation for Open Systems
DAP	Directory Access Protocol
DIB	Directory Information Base
DIT	Directory Information Tree
DMD	Directory Management Domains
DSA	Directory System Agent
DSP	Directory System Protocol
DUA	Directory User Agent
EWOS	European Workshop for Open Systems
FTAM	File Transfer, Access & Management
INTAP	Interoperability Technical Association for Information Processing, Japan
ISDN	Integrated Services Digital Network
ISO/IEC	International Organization for Standardization
KT	Knowledge Tree
LL	Lower layers of OSI model (layers 1-4)
MAP	Manufacturing Automation Protocol
MHS	Message Handling Systems
NIST	National Institute of Standards and Technology
NSAP	Network Services Access Point
OSI	Open Systems Interconnection
PKCS	Public Key Crypto System
POSI	Promotion for Open System Interconnection
PRDMD	Private Directory Management Domain
PSAP	Presentation Service Access Point
RDN	Relative Distinguished Name
ROSE	Remote Operations Service Element
SSAP	Session Service Access Point
SIG	Special Interest Group
SPAG	Standards Promotion & Application Group
TOP	Technical and Office Protocols

TSAP	Transport Service Access Point
UL	Upper layers of OSI model (layers 5-7)
UPU	Universal Postal Union

11.20 Appendix C: Requirements for Distributed Operations

The following material is included for tutorial purposes, and does not represent material additional to the Directory Documents. It is also not intended as a complete statement of requirements (the Distributed Operations part of the Directory Documents should be referred to for a complete treatment).

11.20.1 General Requirements

DSAs supporting distributed operations and claiming support of chaining must fully support DSP, as defined by the Directory Documents. DSAs supporting distributed operations must always be able to accept incoming DSP associations and invocations. DSAs claiming support of chaining must support:

- Loop detection
- Loop avoidance

In passing on operations (when chaining or multi-casting), the original DAP-supplied invocation must be passed on without change of content. In particular, there must be no alteration in any way of any primitive content.

The support of a facility for returning cross-references (Directory Documents Part 4/10.4.1) is optional.

To ensure that traceInformation can be analyzed properly, DSAs shall only possess names that are compliant with the recommendations of the Directory Documents Part 7 (including Annex B).

11.20.2 Protocol Support

11.20.2.1 Usage of ChainingArguments

originator need not be used if requestor in CommonArguments is used.

targetObject shall not be used unless the target object differs from object/base object (if it is present, object/base object are ignored for purposes of name resolution.

operationProgress, traceInformation, aliasDereferenced, aliasedRDNs, referenceType and timeLimit shall be generated, accepted, and used in accordance with the Directory Documents.

returnCrossReferences and info may optionally be generated, and shall always be accepted.

11.20.2.2 Usage of ChainingResults

crossReferences and info may optionally be generated, and shall always be accepted.

12. SECURITY

The Security Architecture specified in ISO 7498/Part 2 - Security Architecture (as presented in ISO/TC 97/SC 21/N1528) shall be used as a basis for further work in the Special Interest Group on Security.

The security services that are to be implemented first shall include confidentiality, integrity, authentication and access control. Non-repudiation of the source shall also be included for consideration for implementation. These services are defined and discussed in more detail in ISO 7498/Part 2 - Security Architecture.

12.1 Definitions

The following definitions, based on the definitions in ISO 7498/Part2, are to be used when interpreting Chapter 12.

Access Control: The provision of a security system that establishes and enforces which users or processes can get access to what data or processing facilities.

Authentication Information: Information used to establish the validity of a claimed identity.

Authorization: The granting of access rights.

Confidentiality: A security service that protects data from unauthorized disclosure.

Connection: A state of communication that exists between two communicating entities by establishing an association between them, providing one or more data paths between them allowing sequential transfers of data, and then terminating the association.

Connectionless: A state of communication that provides transfer of data from one entity to another without a preestablished association.

Data Integrity: The property that data has not been altered or destroyed in an unauthorized manner.

Data Origin Authentication: The corroboration that the source of data received is as claimed.

Digital Signature: Data that allows a recipient of information to verify the source and integrity of the information.

Peer-entity Authentication: The corroboration that peer entities in an association are as claimed.

Repudiation: Denial by one or both of the entities of an association of having participated in all or part of the association or communication of the association.

Selective Field Protection: The protection of specified fields of data in a communication.

Traffic Analysis: The inference of information from observation of traffic flow in communications (presence, absence, amount, direction and frequency).

Traffic Flow Confidentiality: A confidentiality service to protect against traffic analysis.

12.2 Matrix of Security Services and OSI Layers

The following matrix shows the layers of the OSI architecture at which certain security services are considered to be desirable. The entries in the matrix are "H" for high level of desirability, "M" for medium desirability, and "L" for low level of desirability. No entry in the matrix means that the service is not considered desirable. This matrix was produced from a similar matrix in ISO 7498/Part 2 which showed the layers of the architecture that could be used to provide the security service. The level of desirability was established by the members of the Special Interest Group in Security of the OSI Implementors Workshop.

Note: The Matrix is a consensus of the opinions of the members as to where selected security services should be placed. It should not be considered restrictive and interpreted as meaning that the security services cannot be placed elsewhere in the OSI architecture or have other implementation priorities. This will depend upon the differing security needs of specific applications. Also, it should not be considered complete in that other security services may exist that should be incorporated in the architecture.

Table 12.1 OSI Layers Desirable for Placing Security

SERVICE	1	2	3	4	5	6	7
1.(a) Peer entity authentication			L	H			H
(b) Data origin authentication			L	L			H
2. Access Control Service			M	M			H
3.(a) Connection confidentiality	L	L	L	H		H	H
(b) Connectionless confidentiality		L	H	L		H	H
(c) Selective field confidentiality						H	H
(d) Traffic flow confidentiality	M		L				L
4.(a) Connection integrity with recovery				H			L
(b) Connection integrity without recovery			N	N			N
(c) Selective field connection integrity							N
(d) Connectionless integrity			H	L			L
(e) Selective field connectionless integrity							H
5.(a) Non-repudiation: originator							L
(b) Non-repudiation: receiver							L

Implementation priority:	H (high)
	M (medium)
	L (Low)
	N (No Priority)

Table 1 ISO 7498/Part 2: Security Addendum -- NIST/OSI Workshop Summary Of SIG-SEC Discussions of Security Service Placement, May, 1987

Notes: The following notes are for explanation of the above matrix and comments.

A security system should be considered to be an integrated set of security services that are placed at selected OSI layers. The services should be selected based on a risk analysis for the computer system being protected. Security mechanisms must`be then chosen that will provide the security services and incorporated in the software and hardware of the computer system and controlled by the OSI software and hardware at the selected layer(s).

For example, authentication, access control, confidentiality and integrity are selected as the major security goals for an OSI system. A connection oriented transport protocol is being implemented. An example of the use of the Matrix could be in an electronic mail system, to illustrate this the following specific services and layers were chosen:

Peer entity authentication: Layer 4
Data origin authentication: Layer 7

Access Control: Layer 7

Connection confidentiality: Layer 4
Selective field confidentiality: Layer 7

Connection integrity with recovery: Layer 4
Connectionless integrity: Layer 7

The layer 7 services were chosen to support the mail system that would protect the selective paragraphs of an electronic message as directed by the user. A mail system is considered connectionless. Access control is a function of only layer 7.

The layer 4 services were chosen to provide a reliable transport service from the sender to the intended receive of the electronic message. A full connection integrity and confidentiality service with peer entity authentication will assure that all information gets to the receiver correctly and confidentially.

Note: The security protocols and mechanisms that provide these services are beyond the scope of this Chapter at this time. The mechanisms and standards for their interoperability are presently being defined and will be added to this Chapter as they become available.

13. FUTURE SECURITY

Editor's Note: This section is reserved for future stable Security Architecture Agreements. These agreements may be found in the aligned section of the Ongoing Implementation Agreements Document. When these agreements become stable, they will be moved into this Section 13 of this document. Consult Section 13 of the Ongoing Agreements Document for current information.

14. ISO VIRTUAL TERMINAL PROTOCOL

14.1 INTRODUCTION

The NIST/OSI Workshop Virtual Terminal (VT) SIG is making implementation agreements for the OSI Basic Class VT Service and Protocol, ISO 9040 and ISO 9041.

These implementation agreements fall into the following categories.

- Functionality to be implemented, i.e., functional units, etc.
- Identification and specification of VT profiles to be supported by conforming implementations.
- Agreements with regard to implementation issues not specified in ISO 9040 and ISO 9041.
- Resolution of problems with ISO 9040 and ISO 9041 identified during implementation.
- Statement of requirements to meet conformance to these agreements.

14.2 SCOPE AND FIELD OF APPLICATION

14.2.1 Phase Ia Agreements

The Telnet profile is intended to support the following usage:

- a simple line at a time or character at a time dialogue, and
- an application level gateway supporting Internet Telnet and ISO VTP interoperation.

The Transparent profile supports the exchange of uninterpreted sequences of characters. This includes support of VT-users who wish to control terminals directly through the use of embedded control characters and escape sequences.

14.2.2 Phase Ib Agreements

The Forms profile is intended to support forms-based applications with local entry and validation of data by the terminal system. It is the intention of the VT SIG to align this profile with the UK GOSIP forms profile and with the EWOS VT EG Functional Standard.

14.3 STATUS

14.3.1 Status of phase IA

Phase Ia of the VT Agreements was completed May 5, 1988. This phase covers the Telnet and Transparent profiles. No future enhancements will be made to this phase.

14.3.2 Status of phase IB

Phase Ib of the VT Agreements was completed December 16, 1988. Only the Forms profile was stabilized out of phase Ib. Alignment with UK GOSIP and EWOS is anticipated.

14.3.3 Status of phase II

Phase II is still in progress and includes the remaining profile work for Scroll, Page (S-mode), Page (A-mode), and the X3 profiles.

14.4 ERRATA

None at time of publication. (See Ongoing Agreements.)

14.5 CONFORMANCE

Conformant VT implementations are required to support the ISO 9041 Clause 13 requirements plus the additional conformance requirements identified below.

Figure 14.1 shows conformance status for VT facilities which are optional in the ISO VT standard. The terms used in the figure are defined as indicated below.

- "Mandatory" indicated the facility must be provided by all implementations which conform to these agreements.

- "Optional" indicates that the VT facility is not required to meet minimum conformance requirements but has been identified as providing additional useful capabilities.

- "Profile Dependent" indicates that the requirement for the facility, if any, is included in the profile definitions.

- "Not Addressed" indicates that the VT facility is outside the scope of these agreements.

Conformance Status	Mandatory	Optional	Profile Dependent	Not Addressed
Switch Profile **		X		
Multiple Interaction Negotiation **				X
Negotiated Release **				X
Urgent Data **		X		
Break **	X			
Delivery Control*			X	
Enhanced Access Rules **			X	
Structured COs **			X	
Blocks **				X
Fields **			X	
RIOs **			X	
S-mode			X	
A-mode			X	
Mode Switching Capability		X		

*It is not anticipated that new profiles will use quarantined delivery control.

**Functional Units.

Figure 14.1 Conformance Status for VT Facilities

For each mode of operation, (A-mode and S-mode) which is implemented, the default profile for that mode as defined in ISO 9040 must be supported. Implementations that support A-mode must support the A-mode default profile and at least one additional Workshop approved A-mode profile. The Transparent profile does not count as an additional A-mode profile. Implementations that support S-mode must support the S-mode default profile and at least one additional Workshop approved S-mode profile.

For each profile implemented, VTE parameter ranges or values specified in the Workshop-agreed profile and associated notes must be supported.

14.6 PROTOCOL

14.6.1 Protocol Elements

All protocol elements required by the ISO 9040 VT kernel and Break functional units are selected.

All protocol elements required by the Switch Profile functional unit are selected if this functional unit is used. See Figure 14.1.

All protocol elements required by the Urgent Data functional unit are selected if this functional unit is used. See Figure 14.1.

14.6.2 Mapping of Protocol Elements

Mapping of protocol elements on to ACSE or Presentation Services is as defined in ISO 9041.

14.6.3 Protocol Data Unit Structure

Protocol data unit structure is as defined in ISO 9041.

14.7 NIST Registered Control Objects

The following Control Objects are used by more than one profile. Some of the CO parameters are left with undefined values that must be assigned by the profile in which the Control Object is used.

14.7.1 Sequenced Application (SA)

This is a Control object used to convey signals from the application to the terminal in sequence with other updates.

14.7.1.1 Entry Number

14.7.1.2 Name of Sponsoring Body

NIST/OSI Workshop for Implementors of OSI, VTSIG.

14.7.1.3 Date

The date of submission of this proposal is August 26, 1988.

14.7.1.4 Identifier

```
nist-vt-misc-co sa(0)
nist-vt-co-misc-sa OBJECT IDENTIFIER ::=
                              {nist-vt-co-misc sa(0)}
```

14.7.1.5 Descriptor Value

"NIST VT CO for conveying Sequenced Application Signals"

14.7.1.6 CO Parameters

CO-structure	1
CO-priority	"normal"
CO-category	"integer"
CO-size	65535

14.7.1.7 CO Values and Semantics

audible_alarm	[0]	IMPLICIT NULL
newlines_disabled	[1]	IMPLICIT BOOLEAN
restore	[2]	IMPLICIT NULL
visual_alarm	[3]	IMPLICIT NULL
keypad_enabled	[4]	IMPLICIT BOOLEAN
keyboard_unlocked	[5]	IMPLICIT BOOLEAN
device_disconnect	[6]	IMPLICIT NULL
break_signal	[7]	IMPLICIT NULL

14.7.1.8 Additional Information

14.7.1.9 Usage

Profile defined.

14.7.2 Unsequenced Application (UA)

This is a Control object used to convey urgent signals from the application to the terminal.

14.7.2.1 Entry Number

14.7.2.2 Name of Sponsoring Body

NIST/OSI Workshop for Implementors of OSI, VTSIG.

14.7.2.3 Date

The date of submission of this proposal is August 26, 1988.

14.7.2.4 Identifier

nist-vt-misc-co ua(1)

nist-vt-co-misc-ua OBJECT IDENTIFIER::={nist-vt-co-misc ua(1)}

14.7.2.5 Descriptor Value

"NIST VT CO for conveying Unsequenced Application Signals"

14.7.2.6 CO Parameters

CO-structure	1
CO-priority	"urgent"
CO-category	"integer"
CO-size	65535

14.7.2.7 CO Values and Semantics

Same as in SA.

14.7.2.8 Additional Information

14.7.2.9 Usage

Defined in profile.

14.7.3 Sequenced Terminal (ST)

A keyboard can generate many signals that may be given special meaning to the application. This CO is general enough to convey any keyboard event.

14.7.3.1 Entry Number

14.7.3.2 Name of Sponsoring Body

NIST/OSI Workshop for Implementors of OSI, VTSIG.

14.7.3.3 Date

The date of submission of this proposal is August 26, 1988.

14.7.3.4 Identifier

nist-vt-misc-co st(2)

nist-vt-co-misc-st OBJECT IDENTIFIER ::={nist-vt-co-misc st(2)}

14.7.3.5 Descriptor Value

"NIST VT CO for conveying Sequenced Terminal Signals"

14.7.3.6 CO Parameters

CO-structure	1
CO-priority	"normal"
CO-category	"integer"
CO-size	65535

14.7.3.7 CO Values and Semantics

The values of the CO are composite, with values from table below giving meaning to the values in the range 00-FF when added to them.

value	meaning
100	special key - labeled (see list)
200	function key depressed
400	control key depressed
800	shift key depressed
1000	alt key depressed

The special key and the function key are mutually exclusive.

If neither the function keys nor the special keys are pressed, then the value in the range 00-FF will be that of the normal, unshifted code combination generated by the alpha-numeric key.

The control, shift, and alt keys may appear in any combination with the special or function keys.

In general, one of these five keys must be pressed or the value will fall in the range of the repertoire. The shift key must occur in combination with at least one of the other keys in the above table to cause the value to fall outside the repertoire of the display object.

When the special key is depressed, the value of the CO content will use the table below for the value in the range of 00-FF. Otherwise, the value will be defined to be the IA5 value associated with the key.

labeled key	identifier
Break	0
Bell	1
BackSpace	2
Tab	3
BackTab	4
LineFeed	5
CarReturn	6
Cancel	7
Substitute	8
Escape	9
Plus	10
Minus	11
Multiply	12
Divide	13
LeftArrow	14
RightArrow	15
UpArrow	16
DownArrow	17
Insert	18
Delete	19
InsertLine	20
DeleteLine	21
Home	22
End	23
PageUp	24
PageDown	25
PA1	26
PA2	27
PA3	28
HELP	29
StatusProcess	30
InterruptProcess	31
TerminateProcess	32
AbortOutput	33
Formfeed	34
Clear	35
Print	36
Refresh	37
SystemRequest	38

14.7.3.8 Additional Information

14.7.3.9 Usage

Defined in profile.

14.7.4 Unsequenced Terminal (UT)

Keyboard events may need to be conveyed urgently, out of sequence with normal updates. This CO is used to signal the application of the occurrence of such events from the terminal.

14.7.4.1 Entry Number

14.7.4.2 Name of Sponsoring Body

NIST/OSI Workshop for Implementors of OSI, VTSIG.

14.7.4.3 Date

The date of submission of this proposal is August 26, 1988.

14.7.4.4 Identifier

nist-vt-misc-co ut(3)

nist-vt-co-misc-ut OBJECT IDENTIFIER ::= {nist-vt-co-misc ut(3)}

14.7.4.5 Descriptor Value

"NIST VT CO for conveying Unsequenced Terminal Signals"

14.7.4.6 CO Parameters

CO-structure	1
CO-priority	"urgent"
CO-category	"integer"
CO-size	65535

14.7.4.7 CO Values and Semantics

Same as in ST.

14.7.4.8 Additional Information

None

14.7.4.9 Usage

Defined in profile.

14.8 NIST Defined Profiles

14.8.1 Telnet Profile

NIST VTE-Profile Telnet-1988 (r1, r2)

14.8.1.1 Introduction

This profile provides support for TELNET-like operation for users of the ISO Virtual Terminal Service. It is based on the IS version of ISO 9040 and ISO 9041.

14.8.1.2 Association Requirements

14.8.1.2.1 Functional Units

This profile has no mandatory Functional Units.

The Urgent Data Functional Unit is optional, but should be used whenever available.

14.8.1.2.2 Mode

This profile can be used only in A-mode.

14.8.1.3 Profile Body

```
Display-objects=
{
     {
     display-object-name = D, *(DISPLAY)*
     do-access           = "WACA",
     dimensions          = "two",
          x-dimension    =
          {
               x-bound        = "unbounded",
               x-addressing   = "no constraint",
               x-absolute     = "no",
               x-window       = profile argument-r1
          },
          y-dimension    =
          {
               y-bound        = "unbounded",
               y-addressing   = "higher only",
               y-absolute     = "no",
               y-window       = 0
          },
     erasure-capability    = "yes",
     repertoire-capability = 2,
     repertoire-assignment = profile-argument-r2,
     repertoire-assignment = <ESC> 2/5 2/15 4/2
     },
     {
     display-object-name = K, *(KEYBOARD)*
     do-access           = "WACI",
     dimensions          = "two",
          x-dimension    =
          {
               x-bound        = "unbounded",
               x-addressing   = "no constraint",
               x-absolute     = "no",
               x-window       = profile-argument-r1
          },
          y-dimension    =
          {
               y-bound        = "unbounded",
               y-addressing   = "higher only",
               y-absolute     = "no",
               y-window       = 0
          },
     erasure-capability    = "yes",
     repertoire-capability = 2,
     repertoire-assignment = profile-argument-r2,
     repertoire-assignment = <ESC> 2/5 2/15 4/2
     },
},
Control-object =
{
    {
        CO-name         = SY,*(SYNCHRONIZE)*
        CO-access       = "NSAC",
        CO-category     = "symbolic",
        CO-size         = 1,
        CO-priority     = "urgent"
    },
    {
        CO-name         = DI,*(DISPLAY-SIGNAL)*
        CO-access       = "WACA",
        CO-category     = "boolean",
        CO-size         = 5,
        CO-priority     = "normal",
        CO-trigger      = "selected"
    },
    {
        CO-name         = KB,*(KEYBOARD-SIGNAL)*
        CO-access       = "WACI",
        CO-category     = "boolean",
        CO-size         = 5,
        CO-priority     = "normal",
        CO-trigger      = "selected"
    },
    {
        CO-name         = NI,*(negotiation by initiator)*
        CO-access       = "WACI",
        CO-category     = "boolean",
        CO-size         = 4,
        CO-priority     = "normal",
        CO-trigger      = "selected"
    },
    {
        CO-name         = NA,*(negotiation by acceptor)*
        CO-access       = "WACA",
        CO-category     = "boolean",
        CO-size         = 4,
        CO-priority     = "normal",
        CO-trigger      = "selected"
    },
    {
        CO-name         = GA,*(GO-AHEAD)*
        CO-access       = "NSAC",
        CO-category     = "boolean",
        CO-size         = 1,
        CO-priority     = "normal",
        CO-trigger      = "selected"
    }
},
Device-objects=
{
     {
     device-name = DISPLAY-DEVICE,
     device-display-object = D,
     device-default-CO-initial-value = 1."true",*("on")*
     device-minimum-X-array-length = 1,*(no constraint)*
     device-minimum-Y-array-length = 1,*(no constraint)*
     device-control-object = {SY,NA,DI,GA},
          *(SYNC,NEGOTIATE-ACCEPTOR,DISPLAY-SIGNAL,
               GO-AHEAD)*
     device-default-CO-access = "WACA",
     device-default-CO-priority = "normal"
     *(other device object parameters assume corresponding
     DO values)*
     },
     {
     device-name = KEYBOARD-DEVICE,
     device-display-object = K,
     device-default-CO-access = "WACI",
     device-default-CO-priority = "normal",
     device-default-CO-initial-value = 1."true",*("on")*
     device-minimum-X-array-length = 1,*(no constraint)*
     device-minimum-Y-array-length = 1,*(no constraint)*
     device-control-object = {SY,NI,KB,GA}
          *(SYNC,NEGOTIATE-INITIATOR,KEYBOARD-SIGNAL,
               GO-AHEAD)*
     *(other device object parameters assume corresponding
     DO values)*
     }
},
Type of delivery control = "simple-delivery-control".
```

14.8.1.4 Profile Arguments

r1 - is used to represent the line length as the value of VTE parameter x-window for both display objects. This argument is mandatory and takes a nonnegative integer value. This argument is identified by the identifier for x-window for display object D.

r2 - is used to designate the default repertoire for both display objects. This argument is optional, if not present the full US ASCII set is the default. This argument is identified by the identifier for repertoire assignment for the display object D.

14.8.1.5 Profile dependent Control Object Information

This profile does not reference any Control Objects which are defined within this profile.

14.8.1.6 Profile Notes

14.8.1.6.1 Definitive Notes

1. Booleans in the KB and DI control objects are used in this profile to correspond to TELNET commands as follows:

Control Object	Boolean	TELNET
DI/KB	1	IP
DI/KB	2	AO
DI/KB	3	AYT
DI/KB	4	DM
DI/KB	5	BREAK

   The equivalent of a TELNET command is achieved by sending a control object update toggling the boolean that corresponds to the desired TELNET command.

2. The equivalent of a TELNET SYNCH command is achieved by updating the SY control object with the single symbolic value of "SYNCH" (which is mapped onto the integer value 1), and immediately updating the DI (or KB) control object with the DM boolean set "true". IP, AO, AYT, or BREAK commands may be accompanied by a SYNCH command by updating the SY control object and then updating the DI or KB control object toggling both the DM and the other desired boolean. When an update to the SY control object is received subsequent display object updates are discarded until an update to the DI or KB control object is received toggling the DM bit. If a VT-BREAK is received after an SY CO update has been received and prior to the corresponding DI or KB CO update toggling the DM boolean, the discarding of updates is terminated. This is necessary because the VT-BREAK may have caused the DI or KB CO update to be purged.

3. The NI and NA control objects are used to emulate the TELNET option negotiation facility. The facility is symmetric, allowing either party to open negotiation for a change of mode, and every negotiation must be accepted or rejected by the opposite party. The rules for negotiation for each of the option controls are as stated in RFC 854 and as given below.

   a. Only open negotiation for a change from the current state.

   b. Only acknowledge negotiation for a change from the current state.

   c. Do not send any other object updates with a negotiation outstanding.

   For full symmetry, both the NI and NA control objects have the same value definition and consist of 4 booleans with the semantics given below.

BIT	Option	Value
1	Remote Echo	"false" Echo is local; "true" Echo is remote
2	Suppress Go Ahead	"false" Go Ahead; "true" Suppress Go Ahead
3	Binary WACA	"true" use binary WACA; "false" use default or negotiated repertoire for WACA display object
4	Binary WACI	"true" use binary WACI; "false" use default or negotiated repertoire for WACI display object

   Booleans 3 and 4 control the use of the Transparent character set for the D and K display objects respectively. A value of "true" indicates the use of the binary repertoire; "false" indicates the use of the negotiated repertoire. When a party wants to change a repertoire assignment, it must complete a successful TELNET negotiation to change the option control. Then the party with the access rights to the display object in question is required to perform the corresponding secondary attribute modal update.

4. The TELNET EC (erase character) command will be mapped to a pointer relative (x:= x-1) update and an erase current update. The TELNET EL (erase line) command should be mapped to an erase-full-x-array update (an erase operation where the extent is defined as <"start-x",(Yc,Xc-1)> and a pointer update to set x = 1. These X dimension updates are the only times when backward explicit addressing is permitted.

5. The X address of the pointer can be moved forward only by implicit pointer addressing. Addressing of the Y dimension is limited to the next X-array update operation.

6. The VT next X-array update operation will be sent in place of the TELNET NVT "CR,LF" sequence.

7. While the "binary" repertoire is being used no mapping to pointer addressing or erase operations will be done.

8. The repertoire designation "7-bit ASCII (G0+C0)" refers to the repertoire invoked by ISO 2022 defined character set designating escape sequences <ESC> 2/8 4/2, "void", <ESC> 2/1 4/0. The repertoire designation "7-bit ASCII (G0 only)" refers to the repertoire invoked by the ISO 2022 defined character set designating escape sequence <ESC> 2/8 4/2. The designation "binary" refers to the "Virtual Terminal Service Transparent Set" registered in the International Register under ISO 2375 register value 125 and invoked by the escape sequence <ESC> 2/5 2/15 4/2.

9. No termination event list is specified so that data buffering and delivery can be controlled according to context. If local echoing is enabled, the local newline or enter event shall trigger a VT-DELIVER request. With remote echo a timeout or buffer length may be used to trigger a VT-DELIVER request. This buffer length may be 1.

14.8.1.6.2 Informative Notes

1. Users of this profile should refer to the TELNET specification (MIL-STD-1782) and RFCs 854 and 855 for semantics of the TELNET commands. These documents can be obtained by contacting SRI International, DDN Network Information Center, 333 Ravenswood Ave., Menlo Park, CA 94025, (415) 859-3695.

2. An update to the GA control object is equivalent to the TELNET Go Ahead command.

3. If the "go ahead" facility has been negotiated then following a VT-BREAK, only the association acceptor has the right to send data. In the event of VT-BREAK the echo control objects are reinitialized to "false", meaning local echo. If remote echo is desired it must be re-negotiated following VT-BREAK.

4. Negotiation of TELNET options other than echo, transmit binary, and SUPPRESS GO AHEAD is not supported by this profile. Negotiations for these three options can take place at any time during a session.

14.8.1.7 Specific Conformance Requirements

The following character sets are required:

- The G0 character set for U.S. 7-bit ASCII (values 32-126),

- The full U.S. 7-bit ASCII (values 0-127), and

- The transparent character set, (see Note 13).

14.8.2 Transparent Profile

NIST VTE-Profile Transparent-1988 (r1)

14.8.2.1 Introduction

This profile is intended to provide a transparent mode of operation which allows VT-users to exchange transparently uninterpreted sequences of characters but with the added benefit of delivery control to enable the VT-users to determine when the character sequences are to be delivered. This profile may be used when VT-users wish to control terminals directly through the use of embedded control characters.

14.8.2.2 Association Requirements

14.8.2.2.1 Functional Units

No functional units are required by this profile. The VT-BREAK Functional Unit is optional.

14.8.2.2.2 Mode

Use of this profile requires that the value of service parameter VT-mode for the VT-association is "A-Mode".

14.8.2.3 Profile Body

```
Display-objects *(double occurrence)* =
{
     {
     display-object-name = D1,
     do-access           = "WACA",
     dimensions          = "one",
          x-dimensions   =
          {
               x-addressing = "not-permitted"
          },
     repertoire-assignment = profile-argument-r1
     },
     {
     display-object-name = D2,
     do-access           = "WACI",
     dimensions          = "one",
          x-dimension    =
          {
               x-addressing = "not-permitted"
          },
     repertoire-assignment = profile-argument-r1
     }
},
type-of-delivery-control = "simple-delivery-control".
```

14.8.2.4 Profile Arguments

Profile argument r1 is optional and enables negotiation of a value for the VTE-parameter repertoire-assignment for the two display objects (which always have the same value of repertoire assignment when the profile is called). The default value of this argument is the "Virtual Terminal Transparent Set" registered in the International Register under ISO 2375 register value 125, invoked by the escape sequence <ESC> 2/5 2/15 4/2. This argument is identified by the identifier for repertoire-assignment for display object D1.

14.8.2.5 Profile dependent Control Object Information

This profile does not reference any Control Objects.

14.8.2.6 Profile Notes

1. This profile is intended primarily for applications requiring a simultaneous two way exchange of sequences of uninterpreted characters. The semantics usually associated with the display object are not used; for the purposes of this profile, the primary attributes of the character-box graphic elements are actually octets which are passed directly to the real device. There is no relationship between the elements of the X-array and the character boxes of the real device; the secondary attributes of the display object are not utilized. The only operation on the display object which must be supported is the text operation. An alternative repertoire may be selected.

14.8.2.7 Specific Conformance Requirements

Support for the default (transparent) character set is required. It is strongly recommended that the profile argument not be used.

14.8.3 Forms Profile Definition

NIST VTE-Profile Forms-1988 (r1,r2, . . . r29)

14.8.3.1 Introduction

This S-mode VTE-profile is intended for supporting the use of forms based, field oriented data entry applications between a terminal and a host system.

It provides facilities for:

- defining and using screen forms,
- defining field validation and field entry rules, and
- controlling and validating field entry.

This VTE-profile includes support of an optional terminal-end locally attached printer.

This profile is defined using the conventions specified in Annex A of ISO 9040.

14.8.3.2 Association Requirements

14.8.3.2.1 Functional Units

The following VT functional units are required for operation with this profile:

- Enhanced access-rules,
- Structured COs, and
- Fields

The following VT functional units are optional for operation with this profile:

- Urgent Data, and
- Reference Information Objects

14.8.3.2.2 Mode

This is an S-mode profile. Use of this profile requires that the value of the VT-ASSOCIATE VT-mode parameter be one of "S-mode", "either-S" or "either-A".

14.8.3.3 Profile Body

```
Display-objects *(single occurence)* =
     {
     display-object-name = A,
     DO-access           = "WAVAR",
     dimensions          = "three",
          x-dimension    =
               {
               x-bound        = profile-argument-r1,
               x-addressing   = "no constraint",
               x-absolute     = "yes",
               x-window       = x-bound
               },
          y-dimension    =
               {
               y-bound        = profile-argument-r2,
               y-addressing   = "no constraint",
               y-absolute     = "yes",
               y-window       = y-bound
               },
          z-dimension    =
               {
               z-bound        = "unbounded",
               z-addressing   = "no constraint",
               z-absolute     = "no",
               z-window       = profile-argument-r3
               },
     erasure-capability = "yes",
     repertoire-assignment = profile-argument-r4,

     font-capability = 1,
     font-assignment = "device-dependent",

     DO-emphasis = profile-argument-r5,

     foreground-colour-capability = profile-argument-r6,
     foreground-colour-assignment = profile-argument-r7,
     background-colour-capability = profile-argument-r6,
     background-colour-assignment = profile-argument-r8,
block-definition-capability = "no",
field-definition-capability = "yes",
max-fields = profile-argument-r9,
max-field-elements = profile-argument-r10,
access-outside-fields = profile-argument-r11
},
Control-objects =
{
     { *(Field Definition CO)*
     CO-name                = FD,
     CO-type-identifier     = vt-b-sco-fdco,
     CO-structure           = "non-parametric",
     CO-access              = "WAVAR" + profile argument-r12,
     CO-priority            = "normal",
     CO-trigger             = "not-selected"
     },

     { *(Field Entry Instructions CO)*
     CO-name                = EI,
     CO-type-identifier     = "mandatory-feico",
     CO-structure           = "non-parametric",
     CO-access              = "WAVAR" + profile-argument-r12,
     CO-priority            = "normal",
     CO-trigger             = "not-selected"
     },

     { *(Optional registered FEICOs)*
     CO-name                = profile-argument-r13,
     CO-type-identifier     = profile-argument-r14,
     CO-structure           = "non-parametric",
     CO-access              = "WAVAR" + profile-argument-r12,
     CO-priority            = "normal",
     CO-trigger             = "not selected"
     },

     { *(Field Entry Pilot CO)*
     CO-name                = EP,
     CO-type-identifier     = "mandatory-fepco",
     CO-structure           = "non-parametric",
     CO-access              = "WAVAR" + profile-argument-r12,
     CO-priority            = "normal",
     CO-trigger             = "not-selected"
     },

     {  *(Optional registered FEPCOs)*
     CO-name                = profile-argument-r15,
     CO-type-identifier     = profile-argument-r16,
     CO-structure           = "non-parametric",
     CO-access              = "WAVAR" + profile-argument-r12,
     CO-priority            = "normal",
     CO-trigger             = "not-selected"
     },
{ *(Context CO)*
CO-name             = CC,
CO-type-identifier  = vt-b-sco-cco,
CO-structure        = 6,
CO-access           = "WAVAR",
CO-priority         = "normal",
CO-trigger          = "not-selected"
},

{ *(Transmission Policy CO)*
CO-name             = TP,
CO-type-identifier  = vt-b-sco-tpco,
CO-structure        = 1,
CO-access           = "WAVAR" + profile-argument-r12,
CO-priority         = "normal",
CO-trigger          = "not-selected",
CO-category         = "boolean",
CO-size             = 4
},

{ *(Optional registered RIOs)*
CO-name             = profile-argument-r17,
CO-type-identifier  = profile-argument-r18,
CO-structure        = "non-parametric",
CO-access           = "WAVAR" + profile-argument-r12,
CO-priority         = "normal",
CO-trigger          = "not-selected"
},

{ *(Form Waiting Time CO)*
CO-name             = WT,
CO-type-identifier  = "waiting-time",
CO-structure        = 1,
CO-access           = "WAVAR",
CO-category         = "integer",
CO-size             = 65535,
CO-priority         = "normal",
CO-trigger          = "not-selected"
},
```

(The initial value for WT is zero, implying that a Form Waiting Time is not to be used.)

(The following four COs, (SA, UA, ST, and UT), are registered with the NIST registration authority and are referenced by this profile.)

```
{
CO-name              = SA,
CO-type-identifier   = nist-vt-co-misc-sa,
CO-structure         = 1,
CO-access            = "WAVAR" + profile-argument-r12,
CO-priority          = "normal",
CO-trigger           = "not-selected",
CO-category          = "integer",
CO-size              = 65535
},

{
CO-name              = UA,
CO-type-identifier   = nist-vt-co-misc-ua,
CO-structure         = 1,
CO-access            = profile-argument-r12,
CO-priority          = "urgent",
CO-category          = "integer",
CO-size              = 65535,
CO-trigger           = "not-selected"
},

{
CO-name              = ST,
CO-type-identifier   = nist-vt-co-misc-st,
CO-access            = "WAVAR" + opposite of
                                 profile-argument-r12,
CO-priority          = "normal",
CO-trigger           = "not-selected",
CO-category          = "integer",
CO-size              = 65535
},

{
CO-name              = UT,
CO-type-identifier   = nist-vt-co-misc-ut,
CO-access            = opposite of profile-argument-r12,
CO-priority          = "urgent",
CO-trigger           = "not-selected",
CO-category          = "integer",
CO-size              = 65535
}
```

```
Device-objects  *(single or double occurrence)* =
{
     {
     device-name = D,
     device-default-CO-access = "WAVAR",
     device-default-CO-priority = "normal",
     device-default-CO-trigger = "not-selected",
     device-default-CO-initial-value = 1."true",
     device-emphasis = profile-argument-r19,
     device-foreground-colour-assignment =
                               profile-argument-r20,
     device-background-colour-assignment =
                               profile-argument-r21,
     device-minimum-X-array-length = profile-argument-r22,
     device-minimum-Y-array-length = profile-argument-r23,
     device-control-object = {FD,CC,SA,UA,ST,UT,WT,TP},
     device-display-object = A
     },

     IF r24 = "true" THEN   *(define printer)*
     {
     device-name = P,
     device-default-CO-access = "NSAC",
     device-default-CO-priority = "high",
     device-default-CO-trigger = "not-selected",
     device-default-CO-initial-value = 1."false",
     device-emphasis = profile-argument-r25,
     device-foreground-colour-assignment =
                               profile-argument-r26,
     device-background-colour-assignment =
                               profile-argument-r27,
     device-minimum-X-array-length = profile-argument-r28,
     device-minimum-Y-array-length = profile-argument-r29,
     device-control-object = {FD,SA,UA},
     device-display-object = A
     }
}

type-of-delivery-control = "simple delivery control".
```

FIXED Field Entry Instruction Definitions - not updateable

Optional Field
- field entry is optional.

Mandatory Field
- field entry is mandatory.

Selectable Field
- the field is selectable as opposed to requiring entry.

Protected Field
- the field is protected from field entry.

Fill Field
- all character positions k=1 through k=LAST must be entered into the field.

Echo Received Character
- allowed field entry characters are to be echoed as received.

Echo Off
- received field entry characters should not be echoed.

Ignore Case
- upper and lower case alpha characters should be considered valid field input even though other field entry instructions may be specified in terms of a non-matching case.

Inhibit Logical Rendition Attribute Operation
- no form of logical attribute operation, with the exception of character repertoire switching as given below, can be performed on the field. Character repertoire changes are permitted if also permitted by Allowed First Character or Allowed Character, see below. This FEI is intended to be used when the rendition secondary attributes are to be kept under "application" control. See, for example, Allowed First Character for a case of reference to the field modal values.

DYNAMIC Field Entry Instruction Definitions - updateable

Echo Specified Character
- specifies the character which is to be echoed to the user in response to each allowed character entered into the field. The secondary attributes of the echoed character may be specified. If any secondary attribute is not given an explicit value in the FEI, then when the FEI is referenced by activity on a field the field modal attributes for that field are used to supply the attribute value; normal defaulting for field modal attributes applies. See 14.3.1.2 in ISO 9040 AD1.

Minimum Entry
- specifies the minimum number of characters that must be entered into the field. When a field is associated with both the Optional Field FEI and a Minimum Entry FEI, the field is optional but if entry is elected, the number of characters specified by the Minimum Entry FEI must then be entered.

Allowed First Characters
- specifies a set of allowed characters for the first character position of the field. The secondary attributes for the Allowed First Characters may be specified. If any secondary attribute is not given an explicit value in the FEI, then when the FEI is referenced by activity on a field the field modal attributes for that field are used to supply the attribute value; normal defaulting for field modal attributes applies. See 14.3.1.2 in ISO 9040 AD1.

Allowed Characters
- specifies a set of allowed characters for all character positions within the field. The secondary attributes for the set of Allowed Characters may be specified. If Allowed First Characters and Allowed Characters are both specified for a particular field, then the Allowed First Characters applies to the first character position of the field and the set of Allowed Characters applies to the second through last character positions of the field. If any secondary attribute is not given an explicit value in the FEI, then when the FEI is referenced by activity on a field the field modal attributes for that field are used to supply the attribute value; normal defaulting for field modal attributes applies. See 14.3.1.2 in ISO 9040 AD1.

Disallowed Characters
- specifies a set of disallowed characters for all character positions within a field. The secondary attributes for the Disallowed Characters may be specified. If Allowed First Characters and Disallowed Characters are both specified for a particular field, then the Allowed First Characters applies to the first character position of the field and the Disallowed Characters applies to the second through last character positions of the field. When a field is associated with Allowed Characters FEI(s) and Disallowed Characters FEI(s) that have characters in common, the common characters are considered as disallowed. If secondary attributes are not specified, then any characters in the set of Disallowed Characters is prohibited, irrespective of the secondary attribute values it may be given.

Entry Invoke Character
- specifies the character to be used for showing or highlighting to the user, where the next field entry is to be made. The secondary attributes of the specified character may be specified. Fields that are not linked to an Entry Invoke Character FEI, utilize a device dependent entry invoke character which may or may not be represented in the character repertoire negotiated for the device. If any secondary attribute is not given an explicit value in the FEI, then when the FEI is referenced by activity on a field the field modal attributes for that field are used to supply the attribute value; normal defaulting for field modal attributes applies. See 14.3.1.2 in ISO 9040 AD1.

Select Display Mode
- specifies the secondary attributes to be used for showing or highlighting to the user the field which has been selected.

Visit Display Mode
- specifies the secondary attributes to be used for showing or highlighting to the user the field which is the current candidate for being selected. Typically, a selectable field will be considered as a candidate for selection when the cursor is placed on a character within the selectable field.

Waiting Time
- specifies the number of seconds to wait for field entry to complete after the cursor has been positioned within the field. Fields that are not associated with a Waiting Time FEI are not subject to the "Field Waiting Time Expired" Field Entry Event.

Allowed String Values
- specifies a list of strings which identify valid field values. The strings are specified as either a discrete OCTETSTRING or a range of OCTETSTRINGs, or combination of both.

Ranges are specified using a lower "value" OCTETSTRING and a higher "value" OCTETSTRING. The "value" of an OCTETSTRING is the integer value derived from the collating sequence corresponding to the repertoire explicitly or implicitly specified for the OCTETSTRING. For example, the ISO 646 string 'AB' has the integer value 4142(16) and the string 'ABC' has the value 414243(16).

When strings of unequal length are compared, the smaller string is filled on the right with enough spaces to make the strings of equal length. The comparison of ISO 646 strings 'AB' and 'ABC' would be accomplished by first converting the string 'AB' to 'AB ' thus creating the value 414220(16) to be compared against the value 414243(16). The value of the space character is derived from the collating sequence corresponding to the repertoire identified in the field modal attributes. If this repertoire does not contain a space, then the value 20(16) is used.

The secondary attributes for a discrete OCTETSTRING or range of OCTETSTRINGs may be specified. If any secondary attribute is not given an explicit value in the FEI, then when the FEI is referenced by activity on a field the field modal attributes for that field are used to supply the attribute value; normal defaulting for field modal attributes applies. See 14.3.1.2 in ISO 9040 AD1.

Allowed Numeric Values
- specifies a list of numeric strings which identify valid field values. The strings are specified as either a discrete OCTETSTRING or a range of OCTETSTRINGs, or a combination of both.

Ranges are specified using a lower "value" OCTETSTRING and a higher "value" OCTETSTRING. The "value" of an OCTETSTRING is the integer value derived from the collating sequence corresponding to the repertoire explicitly or implicitly specified for the OCTETSTRING. For example, the ISO 646 string '12' has the integer value 3132(16) and the string '123' has the integer value 313233(16).

When strings of unequal length are compared, the smaller string is filled on the left with enough zero characters to make the strings of equal length. The comparison of ISO 646 strings '12' and '123' would be accomplished by first converting the string '12' to '012' thus creating the value 303132(16) to be compared against the value 313233(16). The value of the zero character is derived from the collating sequence corresponding to the repertoire identified in the field modal attributes. If this repertoire does not contain a zero, then the value 30(16) is used.

The secondary attributes for a discrete OCTETSTRING or range of OCTETSTRINGs may be specified. The secondary attributes for a discrete OCTETSTRING or range of OCTETSTRINGs may be specified. If any secondary attribute is not given an explicit value in the FEI, then when the FEI is referenced by activity on a field the field modal attributes for that field are used to supply the attribute value; normal defaulting for field modal attributes applies. See 14.3.1.2 in ISO 9040 AD1.

Mutually Exclusive FEIs

Some FEIs specify field entry validation rules that are in conflict with the rules specified by other FEIs. For example, a particular field cannot be both "protected" and "mandatory". Such conflicting FEIs cannot be associated with the same field. The following table defines the sets of conflicting FEIs.

Table 14.1 SETS OF CONFLICTING FEIs

FEI	Conflicting FEIs
Optional Field	Mandatory Field, Selectable Field, Protected Field.
Mandatory Field	Optional Field, Selectable Field, Protected Field.
Selectable Field	All, except Select Display Mode, Visit Display Mode, Waiting Time.
Protected Field	All.
Fill Field	Selectable Field, Protected Field, Allowed String Values, Allowed Numeric Values.
Echo Received Character	Echo Off, Echo Specified Character.
Echo Off	Echo Received Character, Echo Specified Character.
Ignore Case	Selectable Field, Protected Field.
Echo Specified Character	Echo Off, Echo Received Character
Minimum Entry	Selectable Field, Protected Field.
Allowed First Characters	Selectable Field, Protected Field, Allowed String Values, Allowed Numeric Values.
Allowed Characters	Selectable Field, Protected Field, Allowed String Values, Allowed Numeric Values.
Disallowed Characters	Selectable Field, Protected Field, Allowed String Values, Allowed Numeric Values.
Entry Invoke Character	Protected Field.
Select Display Mode	All, except Selectable Field, Entry Invoke Character, Visit Display Mode, Waiting Time.
Visit Display Mode	All except Selectable Field, Entry Invoke Character, Select Display Mode, Waiting Time.
Waiting Time	Protected Field.

Allowed String Values	Selectable Field, Protected Field, Fill Field, Allowed First Characters, Allowed Characters, Disallowed Characters, Allowed Numeric Values.
Allowed Numeric Values	Selectable Field, Protected Field, Fill Field, Allowed First Characters, Allowed Characters, Disallowed Characters, Allowed String Values.

FEICO Update Syntax

```
FEI DEFINITIONS ::= BEGIN

FEI ::= CHOICE {
echoSpecifiedCharacter    [0]  IMPLICIT Character,
minimumEntries            [1]  IMPLICIT INTEGER,
allowedFirstCharacters    [2]  IMPLICIT CharacterValues,
allowedCharacters         [3]  IMPLICIT CharacterValues,
disallowedCharacters      [4]  IMPLICIT CharacterValues,
entryInvokeCharacter      [5]  IMPLICIT Character,
selectDisplayMode         [6]  IMPLICIT SecAttributes,
visitDisplayMode          [7]  IMPLICIT SecAttributes,
waitingTime               [8]  IMPLICIT INTEGER,
allowedStringValues       [9]  IMPLICIT CharacterValues,
allowedNumericValues      [10] IMPLICIT CharacterValues }

Character ::=SEQUENCE {
     primaryValue    [0]  IMPLICIT OCTETSTRING,
     -- The octet string value specified for primaryValue is
     -- constrained by the repertoires negotiated for the
     -- associated display object.  The octetstring is
     -- restricted to length 1 except for the
     -- allowedStringValues and allowedNumericValues.
     attributes      [1]  IMPLICIT SecAttributes OPTIONAL }

CharacterValues ::= SEQUENCE OF SEQUENCE {
     lowValue        [0]  IMPLICIT Character,
     highValue       [1]  IMPLICIT primaryValue < Character
                                    OPTIONAL }
     -- The default for highValue is the associated
     -- lowValue.  Octet values specified for highValue
     -- are constrained by the repertoire corresponding
     -- to the lowValue value.  The relationship
     -- [lowValue <= highValue] must be true.


SecAttributes ::= IMPLICIT SEQUENCE {
     repertoire           [0]  IMPLICIT INTEGER OPTIONAL,
     foregroundColour     [1]  IMPLICIT INTEGER OPTIONAL,
     backgroundColour     [2]  IMPLICIT INTEGER OPTIONAL,
     emphasis             [3]  IMPLICIT PrintableString
                                              OPTIONAL,
     font                 [4]  IMPLICIT INTEGER OPTIONAL }

     -- Defaults for the secondary attributes are provided
     -- by the field modal attribute values.

END *(FEI DEFINITIONS)*
```

FEICO "mandatory-feico" Initial Content

For each FEIRxx, xx identifies the integer value to be used as "feirList recordIndex" in FDCOUpdate operations. FEICOUpdate operations must use an "index" greater than 127. Note that the character oriented FEIRs or the initial FEICO utilize the default secondary attributes.

```
{
FEIR00    Not Used,

FEIR01    Optional Field,

FEIR02    Mandatory Field,

FEIR03    Selectable Field,

FEIR04    Protected Field,

FEIR05    Fill Field,

FEIR06    Echo Received Character,

FEIR07    Echo Off,

FEIR08    Ignore Case,

FEIR09    Allowed Characters = {A, B, ..., Z},
          *(Integer values '41'H - '5A'H)*

FEIR10    Allowed Characters = {a, b, ..., z},
          *(Integer values '61'H - '7A'H)*

FEIR11    Allowed Characters = {0, 1, ..., 9},
          *(Integer values '30'H - '39'H)*

FEIR12    Disallowed Characters = {A, B, ..., Z},
          *(Integer values '41'H - '5A'H)*

FEIR13    Disallowed Characters = {a, b, ..., z},
          *(Integer values '61'H - '7A'H)*

FEIR14    Disallowed Characters = {0, 1, ..., 9},
          *(Integer values '30'H - '39'H)*

FEIR15-FEIR127 Reserved}
```

Field Entry Event Definitions

{
FEE00 Not used,

FEE01 Unsequenced Terminal event,

FEE02 Sequenced Terminal event,

FEE03 Field entry complete,

FEE04 Field selected,

FEE05 Field waiting time expired,

FEE06 Field Entry Instruction Violation}

Field Entry Condition Definitions

{
FEC00 Not used,

FEC01 The field is currently at the end of the navigation path,

FEC02 The field is not currently at the end of the navigation path,

FEC03 Unconditional}

Field Entry Reaction Definitions

{
FER00 Not used,

FER01 Prevent Further Entry,

FER02 Transmit undelivered updates,

FER03 Relinquish WAVAR,

FER04 Position to character address k=1 in the next field in the navigation path,

FER05 Position to character address k=1 in the previous field in the navigation path,

FER06 Erase current field and position to character address k=1 in the current field,

FER07 Erase all unprotected fields and position to initial character address and first field as indicated in CCO,

FER08 Execute RIO,

FER09 Call RIO,

FER10 Present a visual indication in response to Field Entry Instruction Violations,

FER11 Present an audible indication in response to Field Entry Instruction Violations}

Field Entry Pilot Update Syntax

```
FEPR DEFINITIONS ::= BEGIN

FEE ::=   CHOICE {
     unsequencedTermEvent      [1]  IMPLICIT INTEGER,
     sequencedTermEvent        [2]  IMPLICIT INTEGER,
     fieldEntryComplete        [3]  IMPLICIT NULL,
     fieldSelected             [4]  IMPLICIT NULL,
     fieldWaitTimeExpired      [5]  IMPLICIT NULL,
     feiViolation              [6]  IMPLICIT NULL }

FEC ::=   CHOICE {
     lastField                 [1]  IMPLICIT NULL,
     notLastField              [2]  IMPLICIT NULL,
     unconditional             [3]  IMPLICIT NULL }

FER ::= CHOICE {
     preventFurtherEntry       [1]  IMPLICIT NULL,
     transmitUpdates           [2]  IMPLICIT NULL,
     relinquishWAVAR           [3]  IMPLICIT NULL,
     positionToNextField       [4]  IMPLICIT NULL,
     positionToPreviousField   [5]  IMPLICIT NULL,
     eraseRestartField         [6]  IMPLICIT NULL,
     eraseRestartForm          [7]  IMPLICIT NULL,
     executeRIO                [8]  IMPLICIT RIORecordID,
     callRio                   [9]  IMPLICIT RIORecordID,
     visualIndication          [10] IMPLICIT NULL,
     audibleIndication         [11] IMPLICIT NULL}

RIORecordID ::= SEQUENCE {
     rioNAME         [1]  IMPLICIT PrintableString OPTIONAL
     -- optional if there is only one RIO in the VTE
     recordID        [2]  IMPLICIT PrintableString }

END *(FEPR DEFINITIONS)*
```

FEPCO "mandatory-fepco" Initial Content

For each FEPRxx, xx identifies the integer value to be used as "feprList recordIndex" in FDCOUpdate operations. FEPCOUpdate operations must use an "index" greater than 127.

```
{FEPR00 = Not used,
FEPR01 =  FEE02,    *(Sequenced Terminal Event)*
          FEC03,    *(Unconditional)*
          {
          FER01,    *(Prevent Further Entry)*
          FER02,    *(Transmit Undelivered Updates)*
          FER03}    *(Relinquish WAVAR)*
FEPR02 =  FEE03,    *(Field Entry Complete)*
          FEC01,    *(Last Field)*
          {
          FER01,    *(Prevent Further Entry)*
          FER02,    *(Transmit Undelivered Updates)*
          FER03}    *(Relinquish WAVAR)*
FEPR03 =  FEE03,    *(Field Entry Complete)*
          FEC02,    *(Not Last Field)*
          FER04     *(Position to Next Field)*
FEPR04 =  FEE04,    *(Field Selected)*
          FEC03,    *(Unconditional)*
          {
          FER01,    *(Prevent Further Entry)*
          FER02,    *(Transmit Undelivered Updates)*
          FER03}    *(Relinquish WAVAR)*
FEPR05 =  FEE05,    *(Field Waiting Time Expired)*
          FEC01,    *(Last Field)*
          {
          FER01,    *(Prevent Further Entry)*
          FER02,    *(Transmit Undelivered Updates)*
          FER03}    *(Relinquish WAVAR)*
FEPR06 =  FEE05,    *(Field Waiting Time Expired)*
          FEC02,    *(Not Last Field)*
          FER04     *(Position to Next Field)*
FEPR07 =  FEE06,    *(Field Entry Instruction Violation)*
          FEC03,    *(Unconditional)*
          FER10     *(Visual Indication)*
FEPR08 =  FEE06,    *(Field Entry Instruction Violation)*
          FEC03,    *(Unconditional)*
          FER11     *(Audible Indication)*
FEPR09 - FERP127 Reserved}
```

14.8.3.4 Profile Arguments

r1 - is optional and provides for the negotiation of a value for the VTE parameter x-bound. It takes an integer value greater than zero. Default is 80.

r2 - is optional and provides for the negotiation of a value for the VTE parameter y-bound. It takes an integer value greater than zero. Default is 24.

r3 - is optional and provides for the negotiation of a value for the VTE parameter z-window. It takes a positive integer value. Default is 0.

r4 - is optional and provides for the negotiation of a value(s) for the VTE parameter repertoire-assignment. The value for repertoire-capability is implied by the number of occurrences of profile argument r4. Default specified by ISO 9040.

r5 - is optional and provides for the negotiation of a value for the VTE parameter DO-emphasis. The default value is that defined by ISO 9040, B.17.3. Refer to ISO 9040, B.17.4 for rules governing the selection of non-default values.

r6 - is optional and provides for the negotiation of a value for the VTE parameters foreground-colour-capability and background-colour-capability. Default is 8. This arguement is identified by the identifier for foreground-colour-capability for display object A.

r7 - is optional and provides for the negotiation of a value for the VTE parameter foreground-colour-assignment. Default is {"white", "black", "red", "cyan", "blue", "yellow", "green", "magenta"}.

r8 - is optional and provides for the negotiation of a value for the VTE parameter background-colour-assignment. Defaults {"black", "white", "cyan", "red", "yellow", "blue", "magenta", "green"}.

r9 - is optional and provides for the negotiation of a value for the VTE parameter max-fields. Default is "unbounded".

r10 - is optional and provides for the negotiation of a value for the VTE parameter max-field-elements. Default is 1.

r11 - is optional and provides for the negotiation of a value for the VTE parameter access-outside-fields. Default is "not allowed".

r12 - is mandatory and provides for the negotiation of a value for the VTE parameter CO-access for the Field Definition, Field Entry Instruction, Field Entry Pilot, Transmission Policy, Reference Information, Waiting Time, Sequenced Application, Unsequenced Application, Sequenced Terminal, and Unsequenced Terminal control objects. If the VT-association initiator is the terminal VT-user, it takes the value "WACA", otherwise it takes the value "WACI". This argument is identified by the identifier for CO-access for control object FD.

r13 - is optional and provides for the negotiation of a value for the VTE parameter CO-name for registered FEICOs.

r14 - is optional and provides for the negotiation of a value for the VTE parameter CO-type-identifier for registered FEICOs. An occurrence of r14 is required for every occurrence of r13.

r15 - is optional and provides for the negotiation of a value for the VTE parameter CO-name for registered FEPCOs.

r16 - is optional and provides for the negotiation of a value for the VTE parameter CO-type-identifier for registered FEPCOs. An occurrence of r16 is required for every occurrence of r15.

r17 - is optional and provides for the negotiation of a value for the VTE parameter CO-name for registered RIOs.

r18 - is optional and provides for the negotiation of a value for the VTE parameter CO-type-identifier for registered RIOs. The value vt-b-sco-nullrio selects an empty RIO. An occurrence of r18 is required for every occurrence of r17.

r19 - is optional and provides for the negotiation of a value for the VTE parameter device-emphasis. Default is the value of DO-emphasis. Refer to ISO 9040, B.17.4 for rules governing the selection of non-default values.

r20 - is optional and provides for the negotiation of a value for the VTE parameterdevice-foreground-colour-assignment. Default is the value of foreground-colour-assignment for the display object.

r21 - is optional and provides for the negotiation of a value for the VTE parameter device-background-colour-assignment. Default is the value of background-colour-assignment for the display object.

r22 - is optional and provides for the negotiation of a value for the VTE parameter device-minimum-X-array-length. It takes an integer value greater than zero. Default is the value of x-bound for the display object.

r23 - is optional and provides for the negotiation of a value for the VTE parameter device-minimum-Y-array-length. It takes an integer value greater than zero. Default is the value of y-bound for the display object.

r24 - is a special profile argument. It is optional and provides for the negotiation of a printer device. Default is "false".

r25 - is optional and provides for the negotiation of a value for the VTE parameter device-emphasis for the printer device. Default is the value of DO-emphasis. Refer to ISO 9040, B.17.4 for rules governing the selection of non-default values.

r26 - is optional and provides for the negotiation of a value for the VTE parameter device-foreground-colour-assignment for the printer device. Default is {"black" * 8}.

r27 - is optional and provides for the negotiation of a value for the VTE parameter device-background-colour-assignment for the printer device. Default is {"white" * 8}.

r28 - is optional and provides for the negotiation of a value for the VTE parameter device-minimum-X-array-length for the printer device. It takes an integer value greater than zero. Default is the negotiated value of x-bound for the display object.

r29 - is optional and provides for the negotiation of a value for the VTE parameter device-minimum-Y-array-length for the printer device. It takes an integer value greater than zero. Default is the negotiated value of y-bound for the display object.

14.8.3.5 Profile Dependent Control Objects

This profile uses the NIST registered Control Objects SA, UA, ST and UT. The profile defined values are specified in the body of this profile.

14.8.3.6 Profile Notes

14.8.3.6.1 Definitive Notes

1. The WT control object provides a mechanism for the application VT-user to specify a time in which all the fields of a form must be completed. The terminal VT-user starts the timer at the time when the form is presented to the real device. If the time expires, further entry by the device is stopped, all undelivered updates are transmitted, and WAVAR is relinquished. The undelivered updates are transmitted followed by an update to this control object. The WT update is made using the current value of the WT control object.

2. A selected field is indicated by an addressing operation setting k=1 and f and z to indicate the selected field.

3. When the Character oriented FEIs associated with a particular field have characters in common, the precedence algorithm given below is used.

The Allowed First Characters FEI takes precedence over the Allowed Characters and Disallowed Characters FEIs for field character position k=1.

The Disallowed Characters FEI takes precedence over the Allowed Characters FEI for all field character positions.

The following example illustrates the conflict resolution algorithm. When a particular field is linked to the following three Character oriented FEIs:

```
Allowed First Characters = a
Allowed Characters       = a,b
Disallowed Characters    = a
```

the field must be entered with the letter "a" in the first character position of the field. The remaining character positions in the field are limited to containing the letter "b". Therefore field entry would be limited to a form such as "abbbb. ..".

14.8.3.7 Informative Notes

1. Updates by the application VT-user (only possible within the z-window) are not necessarily immediately imaged to the (human user of the) terminal VT-user unless the real window of the device is currently positioned over such an update. Such updates may move the real window if a VT-DELIVER indication is received.

   When WAVAR is relinquished by the application VT-user the window may be moved so that the field addressed by the CCO is within the window.

   Application VT-user addressing operations that advance z to a higher address which is outside of the z-window cause the z-window to move and include one or more new y-arrays for which no fields are defined. As the z-window moves, one or more y-arrays at lower addresses will no longer be included in the z-window. The field definition records for such y-arrays are implicitly deleted.

2. Some of the defined FEIs imply Field Entry Validation by the Terminal VT-user. Fields linked to such FEIs are candidates for erroneous field entry for which the Terminal VT-user must recover in some local manner. This local manner can only be influenced by use of the Visual Indication and/or Audible Indication FERs. Use of registered FEICO's and or FEPCOs defined outside of this profile may provide enhanced error recovery capabilities.

3. Functions such as next-field, previous-field, tab, etc. are commonly used to provide a one keystroke advance to other fields within the form. The initial content for the default FEICO and FEPCO defined within this profile assumes such a capability is provided locally by the terminal VT-user. At the same time the FEICO and FEPCO update syntax does provide a method for associating function keys with field navigational functions.

14.8.3.8 Specific Conformance Requirements

For further agreement.

14.9 APPENDIX A

14.9.1 Specific ASE Requirement

For specific ASE Requirements identified by the Upper Layer SIG for Virtual Terminals, see Chapter 5, entitled "Upper Layers" in this document. For object identifiers, see Chapter 6 entitled "Object Identifiers and Other Registration Issues", also in this document.

15. FUTURE TRANSACTION PROCESSING

Editor's Note: This section is a placeholder for future stable Transaction Processing Agreements. This a new Special Interest Group whose first regular meeting is in March, 1989. Consult the aligned section of the Ongoing Agreements Document for the latest status of this work.

16. OFFICE DOCUMENT ARCHITECTURE AND INTERCHANGE FORMAT

16.1 PART I - DOCUMENT ARCHITECTURE AND INTERCHANGE FORMAT

16.1.1 Introduction

Section two defines an Implementation Agreement based on Office Document Architecture (ODA) and Interchange Format, as defined in International Standard (ISO) 8613 and provides detailed specification for the implementor. (Note that throughout this agreement, references to ISO 8613 indicate the IS version, 1988). This Implementation Agreement is structured such that Part I addresses implementation issues independent of any specific DAP. Subsequent parts are intended to address implementation issues concerning individual DAPs. At present this Implementation Agreement specifies one DAP as given in Part II.

ISO 8613 has seven parts as given below.

Part 1 of the IS gives an introduction to the standard as a whole and provides a description of the general principles of ODA.

Part 2 defines the document structure model and a reference document processing model.

Part 4 defines the document profile.

Part 5 defines the interchange formats.

Part 6 defines the character content architectures.

Part 7 defines the raster graphics content architectures;

Part 8 defines the geometric graphics content architectures.

ISO 8613 is equivalent to the CCITT T.410 series of Recommendations. This Implementation Agreement uses ISO 8613 as the base standard.

16.1.2 Primary References

For documents referenced in the statement of the agreements relating to Office Document Architecture, consult the References section of this document.

16.1.3 SCOPE AND FIELD OF APPLICATION

This implementation agreement defines a document application profile (DAP) suitable for interchanging documents in formatted form, processable form, or formatted processable form in accordance with ISO 8613.

The DAP is intended for interchange of compound documents between document processing systems. It is intended for documents potentially containing character text, raster graphics, and geometric graphics.

The documents addressed by this DAP range from simple memos to highly structured technical documents. Other DAPs may be defined in the future for other levels of document processing requirements.

16.1.4 STATUS

This is Version 2, Edition 1 of the stable agreement dated December 1988.

16.1.5 ERRATA

None

16.1.6 CONFORMANCE

Introduction

This section defines conformance requirements and provides guidelines for the interpretation of results of conformance testing.

Conformance Requirements

Conformance to this document application profile is defined in terms of an implementation's ability to act as an originator and/or recipient of data streams conformant to the syntax and semantics of ISO 8613 as well as meet all the requirements listed below:

IF THE IMPLEMENTATION IS AN ORIGINATOR :

1) generate data streams, for one or more of the document architecture classes

   a) formatted document architecture (FDA),

b) processable document architecture (PDA), or

c) formatted processable document architecture (FPDA)

2) for each document architecture class supported, generate data streams that include only constituents specified in the relevant sections of this implementation agreement,

3) for each constituent of each document architecture class supported, generate data streams that include only those attributes and combinations of attributes permitted by the relevant sections of this implementation agreement, and

4) <u>either</u>

i) for each attribute of each constituent of each document architecture class supported, generate data streams that contain only basic attribute values as specified by the relevant sections of this implementation agreement,

<u>or</u>

ii) for each attribute of each constituent of each document architecture class supported, generate data streams that contain both basic and non-basic attribute values as specified in the relevant sections of this implementation agreement.

IF THE IMPLEMENTATION IS A RECIPIENT :

5) receive data streams, for one or more of the document architecture classes

a) formatted document architecture (FDA),

b) processable document architecture (PDA), or

c) formatted processable document architecture (FPDA)

6) for each document architecture class supported, receive data streams that may include any or all of the constituents specified in the relevant sections of this implementation agreement,

7) for each constituent of each document architecture class supported, receive data streams that include any of the attributes or any combination of attributes

permitted by the relevant sections of this implementation agreement, and

8) *either*

i) for each attribute of each constituent of each document architecture class supported, receive data streams that contain only basic attribute values as specified by the relevant sections of this implementation agreement,

or

ii) for each attribute of each constituent of each document architecture class supported, receive data streams that contain both basic and non-basic attribute values as specified in the relevant sections of this implementation agreement.

Conformance Testing Definitions

Any implementation that can correctly generate data streams in accordance with the conformance requirements listed above in 1), 2), 3) and 4)i, is defined as a "minimal conforming originator."

Any implementation that can correctly receive data streams in accordance with the conformance requirements listed above in 5), 6), 7), and 8)i, is defined as a "minimal conforming recipient."

16.2 PART II - NIST DOCUMENT APPLICATION PROFILE

16.2.1 Characteristics Supported by this DAP

The following sections describe the logical and layout features that can be represented in documents conforming to this document application profile. The features are described in terms that are typical of the user-perceived capabilities of current document processors. The features are grouped into logical features and layout features in order to relate them to their ODA representation.

Documents conforming to this document application profile may contain any or all of the following content architectures:

a) character text,
b) raster graphics,
c) geometric graphics.

16.2.1.1 Logical Characteristics

Document Logical Structure

The logical structure of documents comprise numbered segments (e.g., chapters, sections or numbered paragraphs), passages, paragraphs, figures and footnotes. Numbered segments can be nested and automatic numbering systems are provided for.

The logical structure of a document conforming to this document application profile consists of a hierarchy of logical objects. The following is an example of a generic document logical structure derived from this document application profile:

```
Document
     Passage(s)
          Paragraph
               Text
               Footnote
                    Footnote reference
                    Footnote body
               Text
               Figure
               Text
          Figure
          Numbered Segment
               Number
          Title
          Passage
               Paragraph
               Figure
     Numbered Segment
     ....
```

Document Structure Elements

Document

A document is composed of a sequence of numbered segments or passages each of which is optionally titled and consists of a sequence of paragraphs and/or figures and/or further passages or numbered segments.

Passage

A passage consists of any logical sequence of paragraphs, figures, and/or further passages or numbered segments that can be regarded as an entity for reading or for layout presentation.

For example, separate passages may be provided for:

(a) the contents to be placed on the title page of a report,
(b) the body of the report, and
(c) the contents to be placed in appendices.

A table is a particular case of a passage. A single paragraph or a single figure is a simple case of a passage.

Segment

A segment is a part of a document which has an automatic number which precedes any other contents and which serves to identify the segment uniquely.

The contents of a segment may begin with a segment title starting on the same line as the segment number.

The document originator may define different classes of segments having in common some presentation features and/or some layout features. For example, the document originator may define a class of segments which always begin on a new page, and another class of segments which are laid out using a special left or right margin offset.

An automatically generated segment number consists of either a number or a series of numbers separated by instances of an arbitrary specified character string separator. In the case of a series of numbers, the segment number is equal to the automatically generated segment number (if any) of the enclosing segment followed by a single index number to uniquely identify the segment.

Index numbers are generated sequentially within any numbered segment. The method of numbering may be a combination of the following:

a) Arabic numerals
b) Upper/lower case letters
c) Upper/lower case Roman numerals

Paragraph

A paragraph is a contiguous amount of content in the intended reading order; a paragraph may, for example, contain a mixture of embedded figures, footnote references, and character text.

A paragraph may contain zero, one or more embedded footnote references. Multiple consecutive footnote references, without intervening text, are also permitted.

A paragraph may contain zero, one or more embedded figures and, optionally, figure references. Multiple consecutive figures and/or figure references, without intervening text, are permitted.

A paragraph may comprise a number of character sequences concatenated together, for example if the character sequences were separately derived or generated.

The document originator may define different classes of paragraphs having in common some presentation features and/or some layout features. For example, the document originator may define classes of paragraphs for "abstract", "standard paragraph", and "summary".

Figure

A figure is an amount of geometric graphics or raster graphics content designed to occupy a rectangular area.

One or more paragraphs can be associated with a figure, for example to provide captions or notes.

Footnote

A footnote consists of a footnote reference and a footnote body.

The footnote body is a contiguous amount of text that can be read out of sequence from the paragraph containing a reference to it.

Footnote Reference

A footnote reference may have an automatically generated label or one supplied by the user. (Both types of footnote references may be present.) If the label is automatically generated then the label may be represented by Arabic numerals, upper or lower case Roman numerals, or upper or lower case alphabetic characters.

Automatically generated footnote numbers are incremented sequentially from an initial value which may be set to any non-negative value at the beginning of the document and reset at any segment or passage, as required.

Reference

A general purpose reference mechanism is provided within paragraph. For example, this reference may be used to reference figures, page numbers, chapter numbers, etc. in other parts of the document.

16.2.1.2 Layout characteristics

Document Layout Structure

The page layout structure allows for several pagesets (e.g., for introductory material and annexes in addition to the main text body, which could be in chapters.) The body area of a page may contain multiple columns and areas for graphics. Header and footer contents can also include figures.

The following is an example of a generic document layout structure derived from this document application profile:

- Document
  - Page set
    - Page
      - Header area
      - Body area
        - Single frame
        - Multiple columns
        - Individual frame(s)
        - Mixed set of frames
      - Footer area
      - ....

Document Layout Structure Elements

Document

A document consists of a sequence of one or more page sets.

Page set

The pages within a page set all have the same dimensions and orientation (landscape or portrait) but may differ in layout and/or content of the header and footer areas.

There may be an optional first page of one particular page layout and this may be followed by either of the following:

a) Repeated pages with the same layout
b) Repeated pages designed for alternating recto and verso layout

Page layout

This document application profile supports various page dimensions including ISO A4-A0, North American Letter, North American Legal and the assured reproduction area of the ISO A4, North American Letter, North American Legal, and ISO A3. The default is the common assured reproduction area of ISO A4 which is equivalent to North American Letter.

A page layout consists of:

a) An optional header area that is reserved for header contents,
b) A single body area, and

c) An optional footer area that is reserved for footer contents

Particular header and footer contents are associated with each page layout.

Body Area Layout

The body area may be subdivided into rectangular frames. Thus the layout may consist of any sequence of:

a) Single frame of fixed width, equal or less than body area width, and fixed height or height adjustable to fit contents,

b) Set of multiple column frames of fixed widths per column and fixed height or height adjustable to fit contents,

c) Individual frames with fixed position and dimensions, potentially overlaid, allowing for support of forms, and

d) Mixed set of frames with various properties, e.g., fixed-size figure frame with fixed-sized caption frame beneath and adjustable height text frame beside both

See Figure 2.1 for illustrations.

Frames which have fixed position and dimensions are permitted to overlap.

Header Area Layout

This is a rectangular area above the body area. It may be subdivided into a number of rectangular frames, for example to contain textual information and graphics such as company logos.

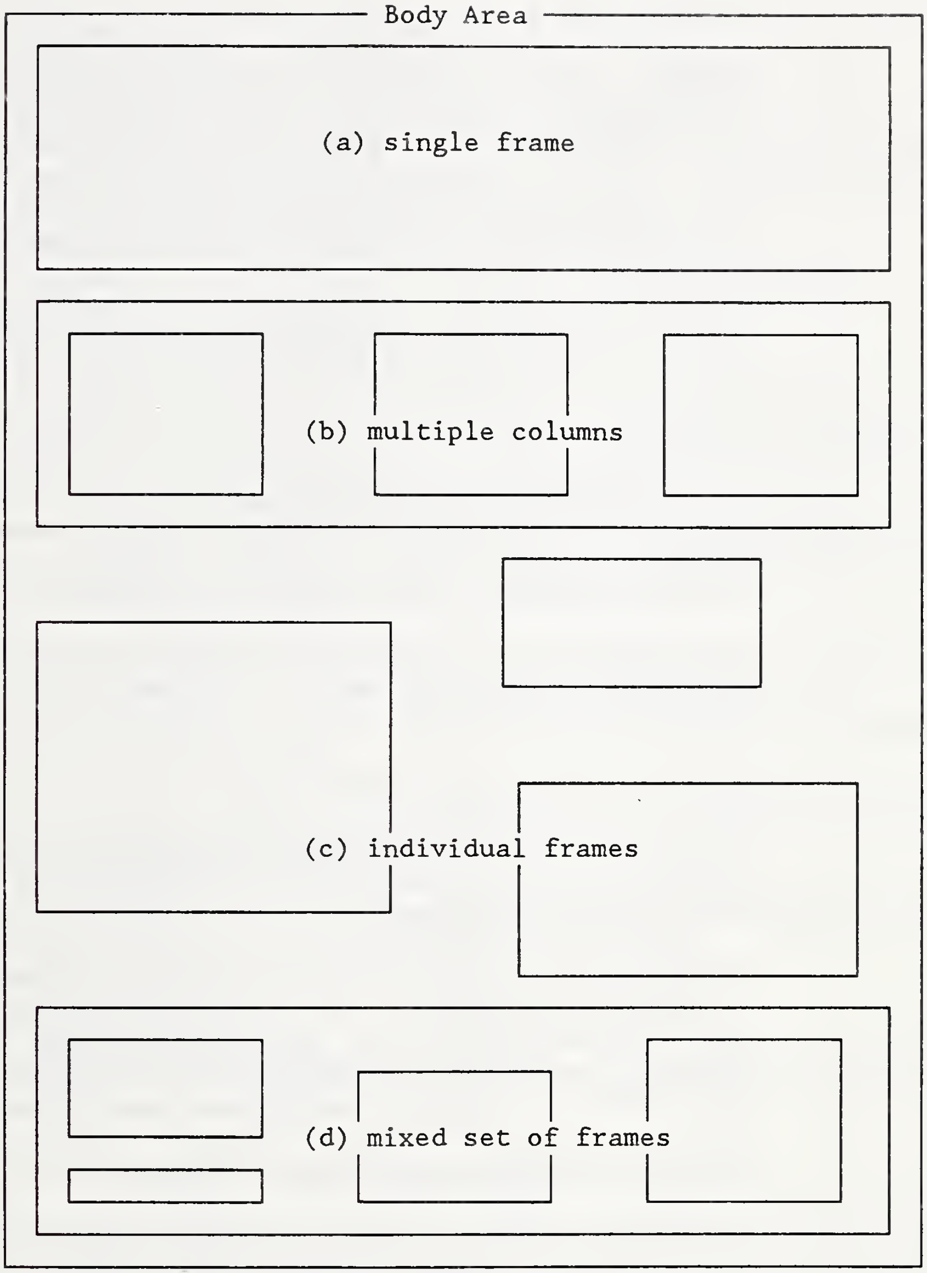


Figure 16.1 Examples of layout within body area

Footer Area Layout

This is a rectangular area below the body area. It may be sub-divided into a number of rectangular frames, for example, to contain textual information and graphics such as company logos.

Header Contents and Footer Contents

Header contents or footer contents may consist of a sequence of paragraphs and/or figures that are constrained to be laid out entirely within the corresponding header or footer area.

One or more automatically generated page numbers may be included anywhere within header contents and/or footer contents.

Header contents or footer contents must not include any footnote or footnote reference or automatic segment numbers.

Page Numbering

An automatically generated page number may occur at any position within header contents or footer contents. Page numbers may represented in Arabic numerals, lower/upper case Roman numerals or lower/upper case letters.

Page numbers are generated sequentially and the sequence can be restarted from any positive integer value at the beginning of any page set.

Layout of Document Logical Contents

The sequence of passages and/or segments is laid out in one or more body areas such that it flows through the sequence of pages in the document.

Controls are needed in order to break the flow of contents at appropriate points. For example, following the passages to be placed on the title page of a document it may be required to control the flow in order to direct subsequent text onto a new page of a different page layout.

Layout of Passage (or Segment) Contents

A passage may be laid out in any of the following ways:

a) As separate passages (see below),
b) Below the previous text within a containing passage, or
c) As a sequence of passages

Layout of passage contents

Controls are available to guide the layout of passages or their subordinate paragraphs and figures.

A passage can be positioned at a fixed position (e.g. the start) of a new body area or in a new frame below the previous contents of a body area.

In case of sets of multiple columns, content generally flows from the bottom of one column of the set to the top of the next column to the right.

Regardless of content type, the various paragraphs and figures in a passage may be laid out within specified frames.

The various methods of subdivision of body areas may be combined with certain frames being designated for flowing text and other frames for particular contents. Thus text may appear to flow around other contents. For example, several figures can be contained within a passage and effect of text flow around the figures and their captions can be produced. See Figure 2.2 for illustration.

A new set of multiple frames can occur beneath a similar set. Thus parallel text (e.g., multilingual) can be synchronized or a table effect can be generated. See Figure 2.3 for illustration.

A variation of the table technique can be used for labelling and annotating paragraphs.

A complete passage can be constrained to be contained in the same body area or frame (by indivisibility).

Layout Controls

The following properties may be specified to control where body area or page breaks occur.

a) New column set (New Layout Object)

> This specifies that the contents should be laid out in the first column (or frame) of a new set of columns (or frames).

b) Unconditional column break (New Layout Object)

> This indicates that the contents must be displayed in the next column (or frame).

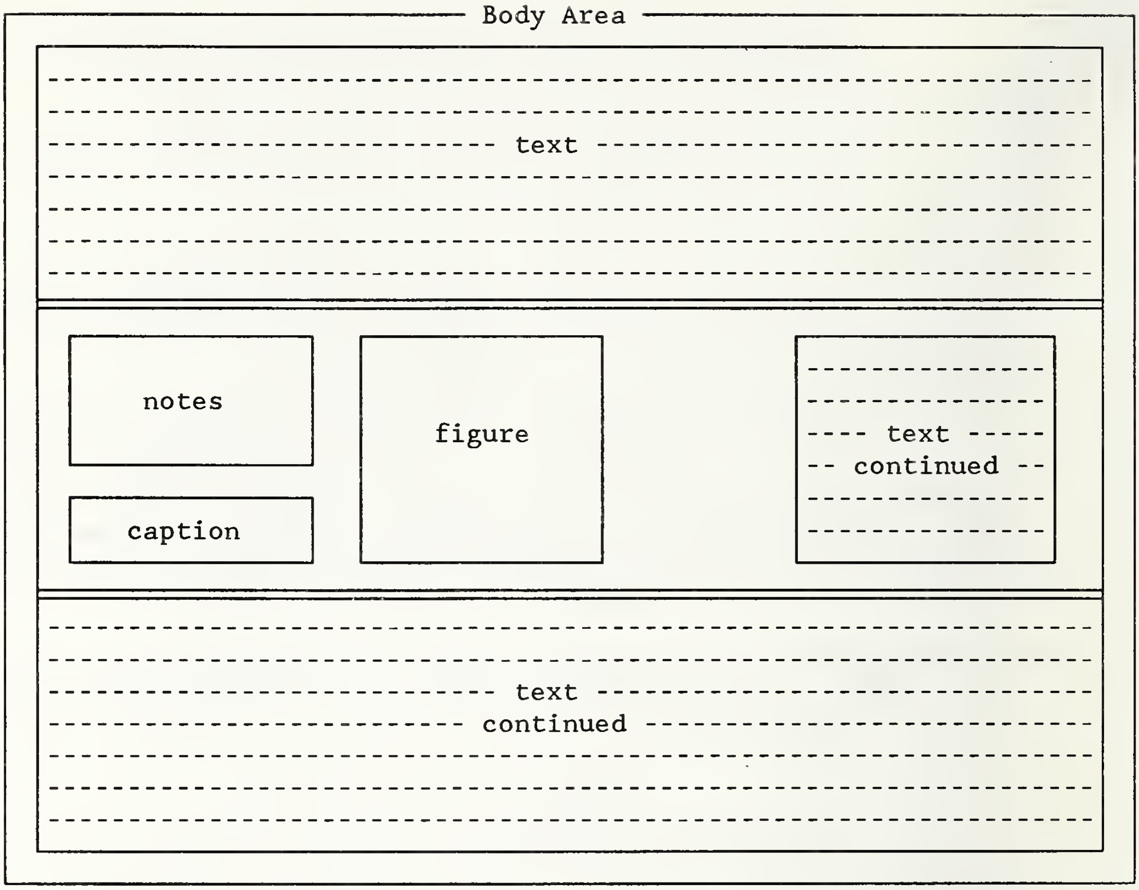


Figure 16.2 Example of text flow around figure

Note: "Caption" and "Notes" contain formatted character content.

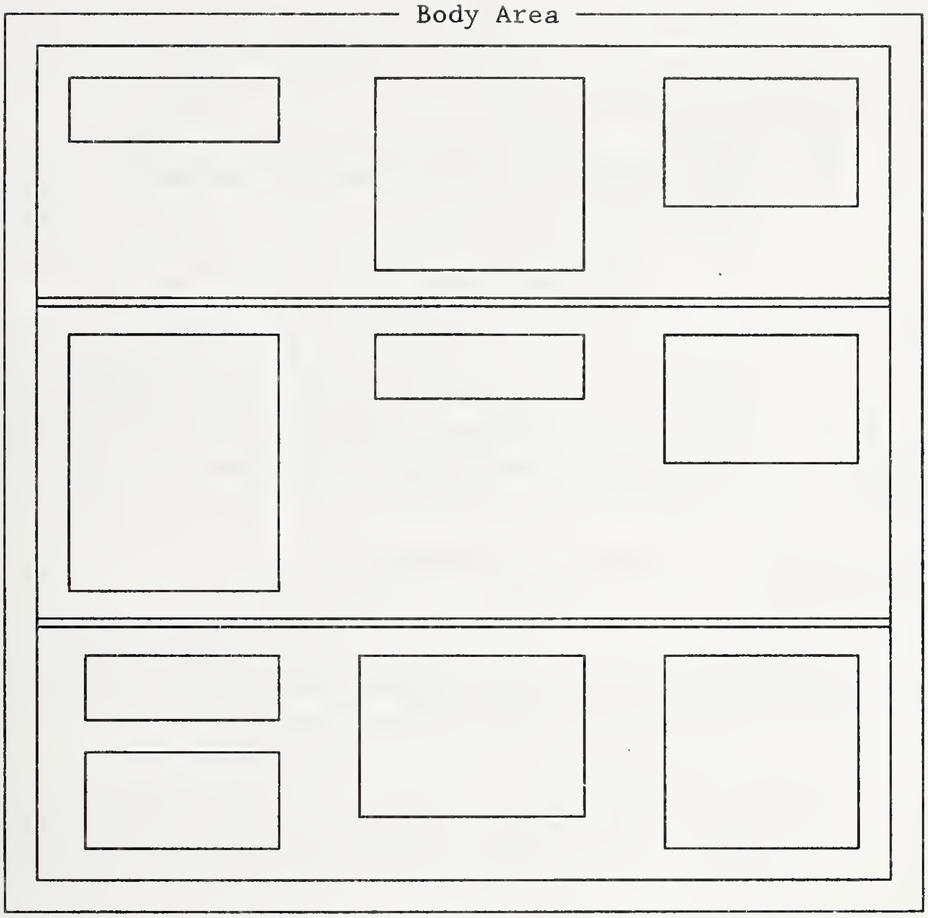


Figure 16.3 Example of synchronized text.

c) Layout Object Class

This indicates that the contents concerned must be displayed in a specified frame, e.g., to control figure positioning.

d) New Page Set (New Layout Object)

This indicates that the contents should be laid out in a new page set.

e) New Page Layout (New Layout Object)

This indicates that the contents should be laid out on a new page of a particular page layout.

f) Unconditional Page Break (New Layout Object)

This indicates that the contents must be displayed in the body area of the next page.

g) Indivisibility

This indicates that a passage (segment, paragraph or figure) must be laid out within a single frame, body area or page set.

h) Same Page/Same Area (Same Layout Object)

This specifies that the start of a passage, numbered segment, paragraph, or figure, for example, must be laid out in the same frame or body area as the end of the previous content (for example, to keep a first paragraph with a title)

Layout of Paragraph Contents

A paragraph may or may not specify its own margins, alignment and tab stops. The indentation of the first line may be different from the remainder of the paragraph. The separation between successive paragraphs can be controlled.

Within a passage the contents of a paragraph may be laid out in two or more frames to allow text to flow around a figure. The figure may or may not be logically associated with that paragraph.

By using the widow and orphan features, layout of paragraphs can also be controlled in order to determine how a paragraph should be divided across two pages.

The orphan size specifies the minimum number of lines of text that must be allocated to the first body area or frame.

The widow size specifies the minimum number of lines of text that must be allocated to the last body area or frame when a paragraph is split over two or more body areas or frames.

Layout of Figure Contents

A figure may occur beneath the previous contents of a body area or frame or can be specified to occupy a particular frame.

Any paragraphs associated with the figure, for example, to provide captions or notes, can be positioned to occupy rectangular areas positioned above, below or beside the figure.

Layout of Footnote Contents

A footnote body may be placed at the bottom of a body area of a page and may or may not be constrained to be entirely in the same body area as the reference to it. If multiple footnotes occur in the same body area the corresponding footnote bodies can be placed in the body area in the same order as the reading order of their references.

16.2.1.3 Content Characteristics

A document may contain any of the following types of content:

- character text utilizing graphic characters as defined in ISO 6937 and ISO 8859 together with character presentation techniques defined in ISO 8613/6,

- raster graphics images encoded according to CCITT Recommendations T.4 and T.6 and in unencoded bitmap form, and

- geometric graphics images in accordance with the minimum capabilities of ISO 8632, each figure consisting of a single picture only.

16.2.2 TECHNICAL SPECIFICATION

16.2.2.1 Summary of Technical Specification

Overview

The NIST DAP, in accordance with ISO 8613, allows documents to be represented in the following forms:

- processable form, which facilitates the revision of a document by a recipient,

- formatted form, which facilities the reproduction of a document as intended by the originator, or

- formatted processable form, which facilitates the reproduction of a document as intended by the originator or facilitates the revision of a document.

Specification of Constituents

This section specifies the required and optional constituents used for the representation of documents that conform to the NIST DAP.

Constituents specified as 'required' must occur in any document that conforms to the NIST DAP. Constituents listed as 'optional' may or may not be present in the document depending upon the requirements of the particular document.

Formatted Form Documents

Required Constituents

- a document profile
- layout object descriptions representing a specific layout structure

Optional Constituents

- layout object class descriptions representing a 'partial' generic layout structure
- presentation styles

Processable Form Documents

Required Constituents

- a document profile
- logical object class descriptions representing a 'complete' generic logical structure
- logical object descriptions representing a specific logical structure

Optional Constituents

- layout object class descriptions representing a 'complete' generic layout structure
- layout styles
- presentation styles

Formatted Processable Form Documents

Required Constituents

- a document profile

- logical object class descriptions representing a 'complete' generic logical structure

- logical object descriptions representing a specific logical structure

- layout object class descriptions representing a 'complete' generic layout structure

- layout object descriptions representing a specific layout structure

Optional Constituents

- layout styles

- presentation styles

16.2.2.2 Notation and Constraints

Introduction

This section presents the notation used in the diagrams and tables which define the permissible structures and attribute values for this document application profile.

There are two sets of diagrams: one describing the logical structure and one describing the layout structure. There are six tables describing attribute value ranges for logical constituents, layout constituents, character content, raster graphics content, geometric graphics content, and the document profile.

Although the layout of these tables is similar, some differences exist. In general, constraints are expressed for constituents by listing an ODA attribute name together with a set of permissible values identified for those attributes required to be present in the interchanged stream and for those additionally permitted in the interchanged stream. For example,

content-architecture-class {cp,cfp}

specifies for a given constituent that the content-architecture-class attribute in this DAP may take one of the values character-processable or character-formatted-processable.

Attributes in the tables specifying values for the character, raster, and geometric content also specify BASIC, NON-BASIC, and DEFAULT values as well as detailing for which content architecture class the attribute may be specified. For example,

```
     select-graphic-rendition BASIC          {0..7,9,10..19,
                                              21..27,29}
                              NON-BASIC      {NONE}
                              DEFAULT        {0}
                              CLASS          {cf,cp,cfp}
```

specifies, for a given content-information attribute, that the select-graphic-rendition control function may be one or more of the basic values representing normal intensity (0), bold (1), faint (2), italicized (3), underlined (4) and so on; that no non-basic values may be specified; that the default is normal intensity (0); and that the control function may be specified for any of the character content architecture classes.

Other than the attributes "dimensions" and "medium-type," there are no non-basic values allowed for attributes in the logical and layout object and object class descriptions. The respective default values, when different from IS 8613, are specified in the document profile attribute "document application profile defaults." For example,

```
     dimensions          {#horizontal{#fixed{9240}}
                         #vertical{#fixed{12400}}}
```

specifies for the document profile attribute dimensions that the default values for the horizontal and vertical are equal to the common assured reproduction area of ISO A4 and North American Letter.

Notation

This section informally describes a meta-language used to define, in terms of a two-level grammar, the NIST notation used in the specification of this document application profile.

The first level provides production rules for the direct specification of attribute values whereas the second level grammar provides productions for the indirect specification of the attributes "bindings", "generator for subordinates" and "content generator".

The notation used in this Implementation Agreement is an extension of the notation defined in Annex A of IS 8613/2.

Grammar Meta-Language

The grammars have a BNF-style that uses the following meta-language symbols:

```
::=          Used to specify that the string of symbols on the right-hand
             side is to be substituted for the non-terminal symbol on the
             left-handside;

->           Used to indicate that the non-terminal symbol on the left-
             hand side evaluates to the string of symbols on the right-
             hand side;

|            Used to separate alternatives;

< >          Used as a pair of symbols to delimit a non-terminal symbol;

<<  >>       Used as a pair of symbols to delimit a nonterminal symbol
             that is specified in the second-level grammar;

--   --      Used as a pair of symbols to delimit a comment string;

{   }        Used as a pair of symbols to delimit a syntactical unit;

[  ]         Used as a pair of symbols to delimit an optional syntactical
             unit;

...          Used following a syntactical unit to indicate that the
             syntactical unit may be repeated;

"  "         Used as a pair of symbols to delimit a terminal symbol;

 +           Used to indicate a string concatenation
```

First level Grammar

```
<attribute-specification>    ::=  <attribute-name>[**] {<conditional-
                                  value>| <attribute-value-range>}

<attribute-name>             ::= -- An attribute name from ISO
                                   8613--

<conditional-value>          ::=CASE {<boolean expression>"{"<simple-
                                  value>"}"}...

<boolean-expression>         ::=[<name><rel-op>] <atomic-value>

<attribute-value-range>      ::= <composite-value> | <simple-value>
```

```
<composite-value>            ::= <single-range> | <set-range> |
                                 <seq-range>

<single-range>               ::= "#" <parameter-name> <attribute-
                                 value-range>

<parameter-name>             ::= --A parameter or sub-parameter name
                                 from ISO 8613--

<set-range>                  ::= "{" <simple-value> [ ","
                                 <simple-value>]..."}"

<seq-range>                  ::="{" {<simple-value>}..."}"

<simple-value>               ::=<constraint>|
                               <keyword>|
                               <atomic-value>|
                               <expr-value>

<constraint>                 ::=<binding-constraint>|
                               <constr-expr-constraint>|
                               <<content-gen-constraint>>|
                               <reference-constraint>

<binding-constraint>         ::="INITIALIZATION" "(" {<<binding-
                                 name>>}... ")"|
                                "MANIPULATION" "(" {<<binding-
                                 name>>}... ")"

<constr-expr-constraint>     ::=<op1> "(" <constr-expr-
                                 constraint>  ")" |
                               <opN> "(" <construction-
                                 list>")"|
                               <DAP-constituent>

<op1>                        ::=<<iter>> | <<poss>> |
                               <<opt>> | <<rep>> | <<optrep>>

<opN>                        ::=<<ser>> | <<any>> | <<set>> |
                               <<cho>> | <<seq>> | <<agg>>

<construction-list>          ::=<constr-expr-constraint> |
                               <constr-expr-constraint> ","
                               <construction-list>


<reference-constraint>       ::= --string literal--

<keyword>                    ::="AS_8613" | "NONE" | "N/A"

<DAP-constituent>            ::=<DAP-constituent-
name>["."<variable>]
```

```
<DAP-constituent-name>      ::=  Logdoc, Passage,
                                 NumberedSegment,Number, Title,
                                 Paragraph, Phrase, FNote, FNBody,
                                 Figure, Text, Reference, Ref, Raster,
                                 Geometric, Hdr-Or-Ftr-Content,
                                 PageNumber, Laydoc, Pageset, Page,
                                 RPage, VPage, Header, Footer, BodyArea,
                                 FrameA, FrameB, FrameC, FrameD, FrameE,
                                 FrameF, FrameG, FrameH, FrameI, FrameJ,
                                 FrameK, Block

<variable>             ::=  --   string literal that distinguishes a
                                 particular instantiation of the DAP
                                 constituent.--

<expr-value>           ::=<function-value>| <expr>

<function-value>       ::={"OBJECT-CLASS-ID-OF" "("{<DAP-
                            constituent>
                            [{"|"<DAP-constituent>}...]}")"}|
                            {"OBJECT-ID-OF" "("{ <DAP-constituent>
                            [{"|" <DAP-constituent>}...]} ")"}

<expr>                 ::=<simple-expr>|<expr><rel-op><simple-
                            expr>|
                            "SIZE"<rel-op><simple-expr>

<simple-expr>          ::=<term>|<simple-expr><add-op><term>

<term>                 ::=<atomic-value>|<term><mult-
                            op><atomic-value>

<atomic-value>         ::=--Any attribute value specified in ISO
                            8613 Part 2,4,6,7 or 8--

<rel-op>               ::= "=" | "<" | "." | "<> | "<=" | ">="      |
                            "=="

<add-op>                    ::= "+" | "-"

<mult-op>                   ::= "*" | "/"

<name>                 ::={[<DAP-constituent>"#"]<attribute-name>
                            ["#"<parameter-name>]...}
```

Keywords

**
Used to denote that if this attribute is specified for a constituent in the generic structure then the corresponding attribute in the specific structure may not be specified. Note that the use of facility is currently under study.

AS_8613
Used to denote that any value may be specified that is permitted in ISO 8613.

NONE
Used to indicate that there are no valid values for this attribute or parameter.

N/A
Used to denote that there is no applicable attribute value

OBJECT_CLASS_ID_OF
Used to specify any object call identifier from the set of instances of a particular constituent constraint.

OBJECT_ID_OF
Used to specify any object identifier from the set of instances of a particular constituent constraint.

STYLE_OF
Used to specify any style identifier from the set of instances of a particular style constraint.

INITIALIZATION
Constrains bindings that identify which binding name value pairs can be initialized for a particular constituent constraint. Note that a table is provided in the second level grammar that defines the allowed initialization for all instances of bindings.

MANIPULATION
Used to identify a binding that provides an expression value. A BNF specification is provided in the second level grammar that defines the allowed expressions for all instances of bindings.

SIBLING
Used to imply an instance of a referenced layout object class which is immediately subordinate to the (same) layout object class that is immediately superior to the layout object class for which a relation is being specified.

RELATION
Defines constraints for location of layout objects.

```
Delimiters
{...}         Used to delimit a syntactical unit.
[...]         Used to delimit optional units.
(...)         Used as a general delimiter.
<...>         Used to delimit a non-terminal symbol.
/*...*/       Used to delimit a comment.
Separators
,      : Used to separate set elements.
|      : Used to separate alternatives.
..     : Used to specify a range of integers or characters.
#      : Used to announce a parameter and attribute value range.
.      : Used to distinguish instances of object classes or objects.

Relational operators
=      : equality.
>      : greater than.
<      : less than.
>=     : greater than or equal to.
<=     : less than or equal to.
<>     : not equal.
==     : equivalent to.

Arithmetic Operators
+      : addition.
-      : subtraction.
*      : multiplication.
/      : division.
```

Second-level Grammar

-- The valid construction expressions for an object class (i.e. the allowable values for the attribute "generator for subordinates") are specified using a meta-construction defined in the first-level grammar. The semantics of a meta-construction are defined such that the result of its evaluation is a construction expression. The result of evaluating a <DAP-constituent> is the identifier of an object class derived from the named constituent constraint. The terminals comprising <op1> and <opN> are called construction operators and are defined in the following second-level grammar. --

-- The operator **opt** evaluates to a construction expression composed of an optional construction factor specifying an object class identifier or construction expression which is derived from the specified construction expression constraint. --

```
<<opt>>        ->   "OPT" {--object class identifier-- |
                           --contained expression--}
```

-- The operator **rep** evaluates to a construction expression composed of an repetitive construction factor specifying an object class identifier or construction expression which is derived from the specified construction expression constraint. --

```
<<rep>>         ->    "REP" {--object class identifier-- |
                             --contained expression--}
```

-- The operator **optrep** evaluates to a construction expression composed of an optional repetitive construction factor specifying an object class identifier or construction expression which is derived from the specified construction expression constraint. --

```
<<optrep>>      ->    "OPTREP" {--object class identifier-- |
                                --contained expression--}
```

-- The operator **cho** evaluates to a construction expression composed of an choice construction factor specifying an object class identifier or construction expression which is derived from the specified construction expression constraint. --

```
<<cho>>         ->    "CHO" {--object class identifier-- |
                             --contained expression--}
```

-- The operator **seq** evaluates to a construction expression composed of an sequence construction factor specifying an object class identifier or construction expression which is derived from the specified construction expression constraint. --

```
<<seq>>         ->    "SEQ" {--object class identifier-- |
                             --contained expression--}
```

-- The operator **agg** evaluates to a construction expression composed of an aggregate construction factor specifying an object class identifier or construction expression which is derived from the specified construction expression constraint. --

```
<<agg>>         ->    "AGG" {--object class identifier-- |
                             --contained expression--}
```

-- The operator **iter** evaluates to either no construction expression or a construction expression which evaluates to a sequence of objects belonging to object classes derived from the specified construction expression constraint. The object classes need be neither identical nor distinct. --

```
<<iter>>  ::=  <null-constr-expr> | <iter-constr-expr>

<null-constr-expr>  ->   --null construction expression--

<iter-constr-expr>  ->   <constr-term> |
                         "SEQ" <term-sequence> |
                         "AGG" <term-sequence> |
                         "CHO" <term-sequence>

<term-sequence>     ::=  <constr-term> |
                         <constr-term> <term-sequence>

<constr-term>       ::=  <constr-factor> |
                         "OPT" <constr-factor> |
                         "REP" <constr-factor> |
                         "OPTREP" <constr-factor>

<constr-factor>     ::=  --object class identifier-- |
                         --contained expression--
```

-- The operator **poss** evaluates to either no construction expression or a construction expression which evaluates to either an empty sequence or an object belonging to a class derived from the specified construction expression constraint. --

```
<<poss>>  ::=  <null-constr-expr> | <poss-constr-expr>

<null-constr-expr>  ->   --null construction expression--

<poss-constr-expr>  ->   <constr-term> |
                         "CHO" <term-sequence>

<term-sequence>     ::=  <constr-term> |
                         <constr-term> <term-sequence>

<constr-term>       ::=  <constr-factor> |
                         "OPT" <constr-factor>

<constr-factor>     ::=  --object class identifier-- |
                         --contained expression--
```

-- The operator **ser** evaluates to either no construction expression or a construction expression which evaluates to a sequence of objects one instance of each belonging to an object class derived from the construction expression constraint specified in the corresponding position in the construction list.--

```
<<ser>>   ::=  <null-constr-expr> | <ser-constr-expr>

<null-constr-expr>  ->   --null construction expression--

<ser-constr-expr>   ->   <constr-term> |
                         "SEQ" <term-sequence> |
                         "CHO" <term-sequence>
```

```
<term-sequence>     ::=  <constr-term> |
                         <constr-term> <term-sequence>

<constr-term>       ::=  <constr-factor>


<constr-factor>     ::=  --object class identifier-- |
                         --contained expression--
```

-- The operator **any** evaluates to either no construction expression or a construction expression which evaluates to an object belonging to an object class derived from any one of the specified construction expression constraints. --

```
<<any>>   ::=  <null-constr-expr> | <any-constr-expr>

<null-constr-expr>  ->   --null construction expression--

<any-constr-expr>   ->   <constr-term> |
                         "CHO" <term-sequence>

<term-sequence>     ::=  <constr-term> |
                         <constr-term> <term-sequence>

<constr-term>       ::=  <constr-factor>

<constr-factor>     ::=  --object class identifier-- |
                         --contained expression--
```

-- The operator **set** evaluates to either no construction expression or a construction expression which evaluates to a sequence of objects; one instance of each belonging to an object class derived from a distinct construction expression constraint within the specified list, in any order. --

```
<<set>>   ::=  <null-constr-expr> | <set-constr-expr>

<null-constr-expr>  ->   --null construction expression--

<set-constr-expr>   ->   <constr-term> |
                         "SEQ" <term-sequence> |
                         "AGG" <term-sequence> |
                         "CHO" <term-sequence>

<term-sequence>     ::=  <constr-term> |
                         <constr-term> <term-sequence>

<constr-term>       ::=  <constr-factor>

<constr-factor>     ::=  --object class identifier-- |
                         --contained expression--
```

-- This document application profile permits bindings to be used for automatic numbering schemes, e.g., page numbers and section numbers. This production describes the conventions to be used. The constituent constraints identify bindings by names which describe the use of each binding. Any number of bindings may be used corresponding to each name. --

```
   <<binding name>>    ::=  <numbers> | "PGnum" |
                            <numberstrings> | <prefixes> |
                            <suffixes>| <separators>

   <numbers>           ::=  "number-" + <n>

   <numberstrings>     ::=  "numberstring-" + <n>

   <prefixes>          ::=  "prefix-" + <n>

   <suffixes>          ::=  "suffix-" + <n>

   <separators>        ::=  "separator-" + <n>

   <n>                 ::=  --any character string from the set of
                            characters: "0".."9"--
```

-- A numberstring binding can be initialized in an object superior to the relevant numbering scheme (e.g., a passage can initialize a numbering scheme for subordinate sections). A number binding can be initialized at each hierarchical level (e.g., section) to start the numbering sequence for subordinates. The prefix, separator and suffix bindings can be initialized at a level above the numbering scheme and can be re-specified at any level within the numbering scheme.

Binding Names	Initial value
number-n	any non-negative number
PGnum	any non-negative number
numberstring-n	zero length string
prefix-n	string literal
suffix-n	string literal
separator-n	string literal

The binding "numberstring" of the numbered object can be used to construct the character string representation of the number. If the numbered objects are all of the same object class, the ORDINAL() numeric function application can be used to create the sequence. If the numbered objects are of different object classes, sequences are generated by incrementing the value of another binding called number-n.--

```
<number-n>        ::=   INC(B_REF(PREC(CURR_OBJ))(number-n))
```

```
-- The "numberstring" binding is referenced by a content generator in a
   subordinate of the numbered object.--

<numberstring-n>       ::=  <hierarchic exprn> | <simple exprn>

<hierarchic exprn>     ::=  B_REF(SUP_OBJ(CURR_OBJ))(numberstring-n) +
B_REF(SUP_OBJ(CURR_OBJ))
                            (separator-n) + <simple exprn>

<simple exprn>         ::=  <string function>
                            (B_REF(CURR_OBJ)(number-n))
                           | <string function> (ORD(CURR_OBJ))
                           | <string literal>

<string function> ::=  MK_STR | U_ALPHA | L_ALPHA |
                       U_ROM | L_ROM

-- There are four possible ways of determining the value for the
   attribute "content generator," as follows: --

<<content-gen-constraint>>  ::= <content-generator-1> |
                                <content-generator-2> |
                                <content-generator-3> |
                                <content-generator-4>

<content-generator-1> ::= <num st1> | <pre st1> + <num st1> |
                          <num st1> + <suf st1> |
                          <pre st1> + <num st1> + <suf st1>

   <num st1>       ::=   B_REF(SUP_OBJ(CURR_OBJ))
                         (numberstring-n)

   <pre st1>       ::=   B_REF(SUP_OBJ(CURR_OBJ))(prefix-n)
                       | <string literal>

   <suf st1>       ::=   B_REF(SUP_OBJ(CURR_OBJ))(suffix-n)
                       | <string literal>

<content-generator-2> ::= <pgnum st2> | <pgpre st2> +
                          <pgnum st2> | <pgnum st2> +
                          <pgsuf st2> | <pgpre st2> +
                          <pgnum st2> + <pgsuf st2>

   <pgpre st2>     ::=  <string literal>

   <pgsuf st2>     ::=  <string literal>

   <pgnum st2>     ::=  MK_STR (<numeric expression2>)
                      | U_ALPHA (<numeric expression2>)
                      | L_ALPHA (<numeric expression2>)
                      | U_ROM   (<numeric expression2>)
                      | L_ROM   (<numeric expression2>)
```

```
<numeric expression2> ::= B_REF(SUP(CURR_INST(<class
                         or type>,CURR_OBJ)))(number | PGnum)

<class or type>     ::= frame | page |
                        OBJECT_ID_OF({FrameA | FrameB |
                              FrameC | FrameD | FrameG |
                              FrameH | FrameI | FrameJ |
                              FrameK | Page | RPage |
                              VPage})

<content-generator-3>    ::=
                    MK_STR (<numeric expression3>)
                    | U_ALPHA (<numeric expression3>)
                    | L_ALPHA (<numeric expression3>)
                    | U_ROM   (<numeric expression3>)
                    | L_ROM   (<numeric expression3>)

<numeric expression3> ::=     B_REF(SUP(CURR_INST(frame,
                              CURR_OBJ)))(PGnum)

<content-generator-4>    ::=  <num st4>|<pre st4> + <num st4>|<num
                              st4> + <suf st4> | <pre st4> + <num st4>
                              + <suf st4>
<num st4>           ::=  B_REF(<any-object>)
                         (numberstring-n)

<pre st4>           ::=  B_REF(<any-object>) (prefix-n) | <string
                         literal>

<suf st4>           ::=  B_REF(<any-object>) (suffix-n) | <string
                         literal>

<any-object>        ::=  OBJECT_ID_OF({Logdoc | Passage
                         |NumberedSegment | Number | Title |Paragraph
                         |Phrase | FNote | FNBody | Figure | Text
                         |Reference | Ref | Raster | Geometric
                         |Common-Content | PageNumber | Laydoc
                         |Pageset | Page | RPage | VPage | Header
                         |Footer | BodyArea | FrameA | FrameB | FrameC
                         |FrameD | FrameE | FrameF | FrameG | FrameH
                         |FrameI | FrameJ | FrameK | Block})
```

16.2.3 LOGICAL STRUCTURE

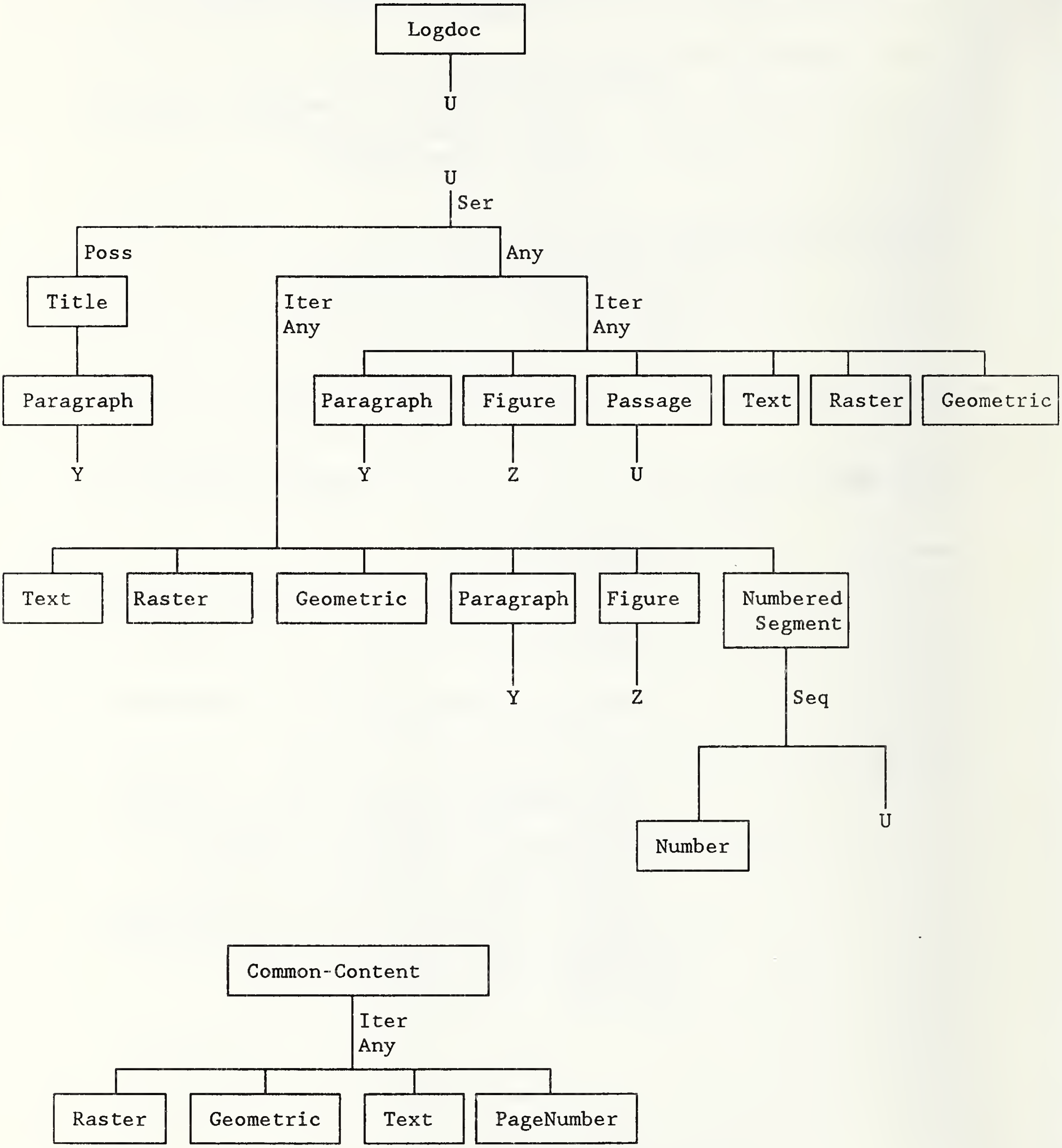


Diagram of Logical Structure (1 of 2)

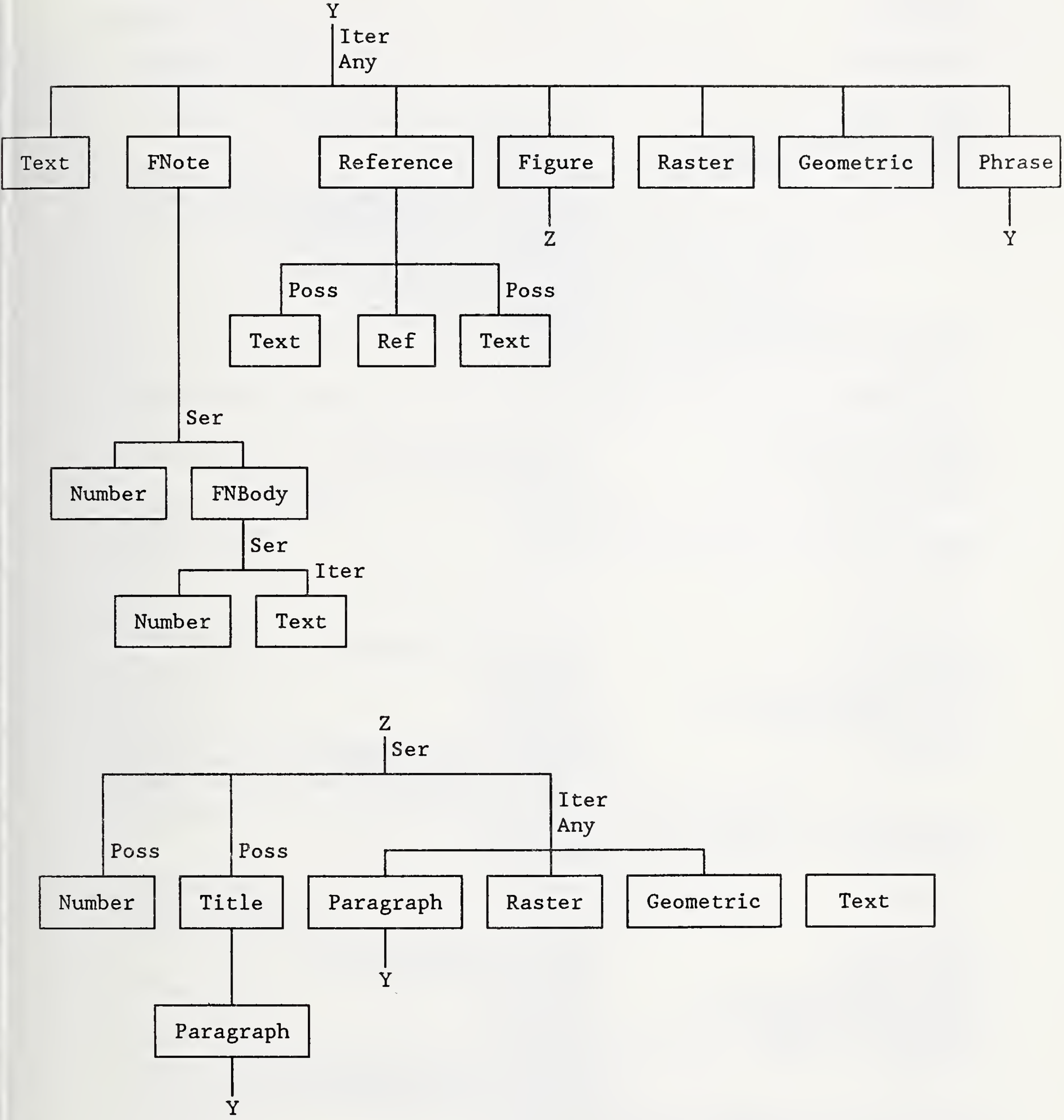


Diagram of Logical Structure (2 of 2)

REQUIRED

--Generic--

Object-Type	{document-logical-root}
Object-Class-Identifier	{AS_8613}
Generator-For-Subordinates	{ser(poss(Title), any(iter(any(Paragraph, Figure,NumberedSegment, Text,Raster,Geometric)), iter(any(Paragraph,Figure, Passage,Text,Raster, Geometric))))}
Application-Comments	{"Logdoc"}

--Specific--

Object-Identifier	{AS_8613}
Object-Class	{OBJECT_CLASS_ID_OF(Logdoc)}
Subordinates	{AS_8613}

PERMITTED

--Generic--

Layout-Style	{STYLE_OF(L-style1)}
Resource	{AS_8613}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Bindings	{INITIALIZATION(number, numberstring,prefix, suffix,separator)}
Default-Value-Lists	{AS_8613}
Protection	{AS_8613}

--Specific--

Object-Type	{document-logical-root}
Layout-Style	{STYLE_OF(L-style1)}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Bindings	{INITIALIZATION(number, numberstring,prefix, suffix,separator)}
Default-Value-Lists	{AS_8613}
Protection	{AS_8613}
Application-Comments	{"Logdoc"}

```
Passage

    REQUIRED

    --Generic--

    Object-Type                    {composite-logical-object}
    Object-Class-Identifier        {AS_8613}
    Generator-For-Subordinates     {ser(poss(Title),
                                    any(iter(any(Paragraph,
                                    Figure,NumberedSegment,
                                    Text,Raster,Geometric),
                                    iter(any(Paragraph,Figure,
                                    Passage,Text,Raster,
                                    Geometric))))}
    Application-Comments           {"Passage"}

    --Specific--

    Object-Identifier              {AS_8613}
    Object-Class                   {OBJECT_CLASS_ID_OF(Passage)}
    Subordinates                   {AS_8613}


    PERMITTED

    --Generic--

    Resource                       {AS_8613}
    User-Readable-Comments         {AS_8613}
    User-Visible-Name              {AS_8613}
    Bindings                       {INITIALIZATION(number,
                                    numberstring,prefix,
                                    suffix,separator)}
    Protection                     {AS_8613}
    Layout-Style                   {STYLE_OF(L-Style3)}
    Default-Value-List             {AS_8613}

    --Specific--

    Object-Type                    {composite-logical-object}
    User-Readable-Comments         {AS_8613}
    User-Visible-Name              {AS_8613}
    Bindings                       {INITIALIZATION(number,
                                    numberstring,prefix,
                                    suffix,separator)}
    Protection                     {AS_8613}
    Layout-Style                   {STYLE_OF(L-Style3)}
    Application-Comments           {"Passage"}
    Default-Value-List             {AS_8613}
```

NumberedSegment

```
REQUIRED

--Generic--

Object-Type                   {composite-logical-object}
Object-Class-Identifier       {AS_8613}
Generator-For-Subordinates    {seq(Number,{ser(poss(Title),
                              any(iter(any(Paragraph,
                              Figure,NumberedSegment,
                              Text,Raster,Geometric)),
                              iter(any(Paragraph,Figure,
                              Passage,Text,Raster,
                              Geometric))))})}
Bindings                      {MANIPULATION(number)}
Application-Comments          {"NumberedSegment"}

--Specific--

Object-Identifier             {AS_8613}
Object-Class                  {OBJECT_CLASS_ID_OF
                               (NumberedSegment)}
Subordinates                  {AS_8613}

PERMITTED

--Generic--

Resource                      {AS_8613}
User-Readable-Comments        {AS_8613}
User-Visible-Name             {AS_8613}
Bindings                      {INITIALIZATION(number,
                               numberstring,prefix,
                               suffix,separator)
                               MANIPULATION(numberstring)}
Protection                    {AS_8613}
Layout-Style                  {STYLE_OF(L-Style3)}
Default-Value-List            {AS_8613}

--Specific--

Object-Type                   {composite-logical-object}
User-Readable-Comments        {AS_8613}
User-Visible-Name             {AS_8613}
Bindings                      {INITIALIZATION(number,
                               numberstring,prefix,
                               suffix,separator)
                               MANIPULATION(number,
                               numberstring)}
```

```
Protection                      {AS_8613}
Layout-Style                    {STYLE_OF(L-Style3)}
Application-Comments            {"NumberedSegment"}
Default-Value-List              {AS_8613}
```

Number

```
    REQUIRED

    --Generic--

    Object-Type                     {basic-logical-object}
    Object-Class-Identifier         {AS_8613}
    Content-Generator               {<content-generator-1>}
    Application-Comments            {"Number"}

    --Specific--

    Object-Identifier               {AS_8613}
    Object-Class                    {OBJECT_CLASS_ID_OF(Number)}

    PERMITTED

    --Generic--

    Resource                        {AS_8613}
    Presentation-Style              {STYLE_OF(P-Style1)}
    Content-Architecture-Class      CASE
                                    cf  {2 8 2 6 0}
                                    cp  {2 8 2 6 1}
                                    cfp {2 8 2 6 2}
    User-Readable-Comments          {AS_8613}
    User-Visible-Name               {AS_8613}
    Protection                      {AS_8613}
    Layout-Style                    {STYLE_OF(L-Style4)}

    --Specific--

    Object-Type                     {basic-logical-object}
    Presentation-Style              {STYLE_OF(P_Style1)}
    Content-Architecture-Class      CASE
                                    cf  {2 8 2 6 0}
                                    cp  {2 8 2 6 1}
                                    cfp {2 8 2 6 2}
    Content-Generator               {<content-generator-1>}
    User-Readable-Comments          {AS_8613}
    User-Visible-Name               {AS_8613}
    Protection                      {AS_8613}
    Layout-Style                    {STYLE_OF(L-Style4)}
    Application-Comments            {"Number"}
```

Title

REQUIRED

--Generic--

Object-Type	{composite-logical-object}
Object-Class-Identifier	{AS_8613}
Generator-for-Subordinates	{Paragraph}
Application-Comments	{"Title"}

--Specific--

Object-Identifier	{AS_8613}
Object-Class	{OBJECT_CLASS_ID_OF(Title)}
Subordinates	{AS_8613}

PERMITTED

--Generic--

Resource	{AS_8613}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Protection	{AS_8613}
Layout-Style	{STYLE_OF(L-Style3)}
Default-Value-List	{AS_8613}

--Specific--

Object-Type	{composite-logical-object}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Protection	{AS_8613}
Layout-Style	{STYLE_OF(L-Style3)}
Application-Comments	{"Title"}
Default-Value-List	{AS_8613}

Paragraph

REQUIRED

--Generic--

Object-Type	{composite-logical-object}
Object-Class-Identifier	{AS_8613}
Generator-for-Subordinates	{iter(any(Text,FNote, Reference,Figure,Raster, Geometric, Phrase))}
Application-Comments	{"Paragraph"}

--Specific--

Object-Identifier	{AS_8613}
Object-Class	{OBJECT_CLASS_ID_OF (Paragraph)}
Subordinates	{AS_8613}

PERMITTED

--Generic--

Resource	{AS_8613}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Protection	{AS_8613}
Layout-Style	{STYLE_OF(L-Style3)}
Default-Value-List	{AS_8613}

--Specific--

Object-Type	{composite-logical-object}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Protection	{AS_8613}
Layout-Style	{STYLE_OF(L-Style3)}
Application-Comments	{"Paragraph"}
Default-Value-List	{AS_8613}

```
Phrase

   REQUIRED

   --Generic--

   Object-Type                    {composite-logical-object}
   Object-Class-Identifier        {AS_8613}
   Generator-For-Subordinates     {iter(any(Text,FNote, Reference,Figure,
                                  Raster, Geometric, Phrase))}
   Application-Comments           {"Phrase"}

   --Specific--

   Object-Identifier              {AS_8613}
   Object-Class                   {OBJECT_CLASS_ID_OF(PHRASE)}
   Subordinates                   {AS-8613}

   PERMITTED

   --Generic--

   Resource                       {AS_8613}
   User-Readable-Comments         {AS_8613}
   User-Visible-Name              {AS_8613}
   Protection                     {AS_8613}
   Layout-Style                   {STYLE_OF(L-Style3)}
   Default-Value-List             {AS_8613}

   --Specific--

   Object-Type                    {composite-logical-object}
   User-Readable-Comments         {AS_8613}
   User-Visible-Name              {AS-8613}
   Protection                     {AS_8613}
   Layout-Style                   {STYLE_OF(L-Style3)}
   Default-Value-List             {AS_8613}

   Application-Comments           {"Phrase"}
```

FNote

```
    REQUIRED

    --Generic--

    Object-Type                    {composite-logical-object}
    Object-Class-Identifier        {AS_8613}
    Generator-for-Subordinates     {ser(Number,FNBody)}
    Application-Comments           {"FNote"}

    --Specific--

    Object-Identifier              {AS_8613}
    Object-Class                   {OBJECT_CLASS_ID_OF(FNote)}
    Subordinates                   {AS_8613}

    PERMITTED

    --Generic--

    Resource                  {AS_8613}
    User-Readable-Comments    {AS_8613}
    User-Visible-Name         {AS_8613}
    Bindings                  {INITIALIZATION(numberstring,
                               number)
                               MANIPULATION(numberstring,
                               number}
    Protection                {AS_8613}
    Layout-Style              {STYLE_OF(L-Style3)}
    Default-Value-List        {AS_8613}

    --Specific--

    User-Readable-Comments    {AS_8613}
    User-Visible-Name         {AS_8613}
    Bindings                  {INITIALIZATION(numberstring,
                               number)
                               MANIPULATION(numberstring,
                               number}
    Protection                {AS_8613}
    Layout-Style              {STYLE_OF(L-Style3)}
    Application-Comments      {"FNote"}
    Default-Value-List        {AS_8613}
```

FNBody

```
REQUIRED

--Generic--

Object-Type                    {composite-logical-object}
Object-Class-Identifier        {AS_8613}
Generator-for-Subordinates     {ser(Number,(iter(Text)))}
Layout-Style                   {STYLE_OF(L-Style3)}
Application-Comments           {"FNBody"}

--Specific--

Object-Identifier              {AS_8613}
Object-Class                   {OBJECT_CLASS_ID_OF(FNBody)}
Subordinates                   {AS_8613}

PERMITTED

--Generic--

Resource                       {AS_8613}
User-Readable-Comments         {AS_8613}
User-Visible-Name              {AS_8613}
Protection                     {AS_8613}
Deafult-Value-List             {AS_8613}

--Specific--

Object-Type                    {composite-logical-object}
Layout-Style                   {STYLE_OF(L-Style3)}
User-Readable-Comments         {AS_8613}
User-Visible-Name              {AS_8613}
Protection                     {AS_8613}
Application-Comments           {"FNBody"}
Default-Value-List             {AS_8613}
```

Figure

REQUIRED

--Generic--

Object-Type	{composite-logical-object}
Object-Class-Identifier	{AS_8613}
Generator-for-Subordinate	{ser(poss(Number),poss(Title), iter(any(Paragraph,Text, Raster,Geometric)))}
Application-Comments	{"Figure"}

--Specific--

Object-Identifier	{AS_8613}
Object-Class	{OBJECT_CLASS_ID_OF(Figure)}
Subordinates	{AS_8613}

PERMITTED

--Generic--

Resource	{AS_8613}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Bindings	{MANIPULATION(number, numberstring)}
Protection	{AS_8613}
Layout-Style	{STYLE_OF(L-Style3)}
Default-Value-List	{AS_8613}

--Specific--

Object-Type	{composite-logical-object}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Bindings	{MANIPULATION(number, numberstring)}
Protection	{AS_8613}
Layout-Style	{STYLE_OF(L-Style3)}
Application-Comments	{"Figure"}
Default-Value-List	{AS_8613}

```
Text

   REQUIRED

   --Generic--

   Object-Type                    {basic-logical-object}
   Object-Class-Identifier        {AS_8613}
   Application-Comments           {"Text"}

   --Specific--

   Object-Identifier              {AS_8613}
   Object-Class                   {OBJECT_CLASS_ID_OF(Text)}

   PERMITTED

   --Generic--

   Resource                       {AS_8613}
   Presentation-Style             {STYLE_OF(P-Style2)}
   Content-Architecture-Class     CASE
                                  cf   {2 8 2 6 0}
                                  cp   {2 8 2 6 1}
                                  cfp  {2 8 2 6 2}
   User-Readable-Comments         {AS_8613}
   User-Visible-Name              {AS_8613}
   Protection                     {AS_8613}
   Layout-Style                   {STYLE_OF(L-Style5)}
   Content-Portions               {AS_8613}

   --Specific--

   Object-Type                    {basic-logical-object}
   Presentation-Style             {STYLE_OF(P-Style2)}
   Content-Architecture-Class     CASE
                                  cf   {2 8 2 6 0}
                                  cp   {2 8 2 6 1}
                                  cfp  {2 8 2 6 2}
   User-Readable-Comments         {AS_8613}
   User-Visible-Name              {AS_8613}
   Protection                     {AS_8613}
   Layout-Style                   {STYLE_OF(L-Style5)}
   Content-Portions               {AS_8613}
   Application-Comments           {"Text"}
```

Reference

REQUIRED

--Generic--

Object-Type	{composite-logical-object}
Object-Class-Identifier	{AS_8613}
Generator-for-Subordinates	{ser(poss(Text),Ref, poss(Text))}
Application-Comments	{"Reference"}

--Specific--

Object-Identifier	{AS_8613}
Object-Class	{OBJECT_CLASS_ID_OF (Reference)}
Subordinates	{AS_8613}

PERMITTED

--Generic--

Resource	{AS_8613}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Bindings	{MANIPULATION(numberstring, number)}
Protection	{AS_8613}
Layout-Style	{STYLE_OF(L-Style3)}
Default-Value-List	{AS_8613}

--Specific--

Object-Type	{composite-logical-object}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Bindings	{MANIPULATION(numberstring, number)}
Protection	{AS_8613}
Layout-Style	{STYLE_OF(L-Style3)}
Application-Comments	{"Reference"}
Default-Value-List	{AS_8613}

```
Ref

   REQUIRED

   --Generic--

   Object-Type                     {basic-logical-object}
   Object-Class-Identifier         {AS_8613}
   Application-Comments            {"Ref"}

   --Specific--

   Object-Identifier               {AS_8613}
   Object-Class                    {OBJECT_CLASS_ID_OF(Ref)}

   PERMITTED

   --Generic--

   Content-Generator               {<content-generator-4>}
   Content-Portions                {AS_8613}
   Resource                        {AS_8613}
   Presentation-Style              {STYLE_OF(P-Style2)}
   Content-Architecture-Class      CASE
                                   cp   {2 8 2 6 1}
                                   cfp  {2 8 2 6 2}
   User-Readable-Comments          {AS_8613}
   User-Visible-Name               {AS_8613}
   Protection                      {AS_8613}
   Layout-Style                    {STYLE_OF(L-Style5)}

   --Specific--

   Object-Type                     {basic-logical-object}
   Content-Generator               {<content-generator-4>}
   Content-Portions                {AS_8613}
   Presentation-Style              {STYLE_OF(P-Style2)}
   Content-Architecture-Class      CASE
                                   cp   {2 8 2 6 1}
                                   cfp  {2 8 2 6 2}
   User-Readable-Comments          {AS_8613}
   User-Visible-Name               {AS_8613}
   Protection                      {AS_8613}
   Layout-Style                    {STYLE_OF(L-Style5)}
   Application-Comments            {"Ref"}
```

Raster

```
REQUIRED

--Generic--

Object-Type                    {basic-logical-object}
Object-Class-Identifier        {AS_8613}
Application-Comments           {"Raster"}

--Specific--

Object-Identifier              {AS_8613}
Object-Class                   {OBJECT_CLASS_ID_OF(Raster)}

PERMITTED

--Generic--

Content-Portions               {AS_8613}
Content-Architecture-Class     {2 8 2 7 2}    /* rfp */
Resource                       {AS_8613}
Presentation-Style             {STYLE_OF(P-Style3)}
User-Readable-Comments         {AS_8613}
User-Visible-Name              {AS_8613}
Protection                     {AS_8613}
Layout-Style                   {STYLE_OF(L-Style6)}

--Specific--

Object-Type                    {basic-logical-object}
Content-Portions               {AS_8613}
Content-Architecture-Class     {2 8 2 7 2}    /* rfp */
Presentation-Style             {STYLE_OF(P-Style3)}
User-Readable-Comments         {AS_8613}
User-Visible-Name              {AS_8613}
Protection                     {AS_8613}
Layout-Style                   {STYLE_OF(L-Style6)}
Application-Comments           {"Raster"}
```

Geometric

```
   REQUIRED

   --Generic--

   Object-Type                    {basic-logical-object}
   Object-Class-Identifier        {AS_8613}
   Application-Comments           {"Geometric"}

   --Specific--

   Object-Identifier              {AS_8613}
   Object-Class                   {OBJECT_CLASS_ID_OF
                                   (Geometric)}

   PERMITTED

   --Generic--

   Content-Portions               {AS_8613}
   Content-Architecture-Class     {2 8 2 8 0}     /* gfp */
   Resource                       {AS_8613}
   Presentation-Style             {STYLE_OF(P-Style4)}
   User-Readable-Comments         {AS_8613}
   User-Visible-Name              {AS_8613}
   Protection                     {AS_8613}
   Layout-Style                   {STYLE_OF(L-Style6)}

   --Specific--

   Object-Type                    {basic-logical-object}
   Content-Portions               {AS_8613}
   Content-Architecture-Class     {2 8 2 8 0}     /* gfp */
   Presentation-Style             {STYLE_OF(P-Style4)}
   User-Readable-Comments         {AS_8613}
   User-Visible-Name              {AS_8613}
   Protection                     {AS_8613}
   Layout-Style                   {STYLE_OF(L-Style6)}
   Application-Comments           {"Geometric"}
```

Common-Content

REQUIRED

--Generic--

Object-Type	{composite-logical-object}
Object-Class-Identifier	{AS_8613}
Generator-for-Subordinates	{iter(any(Raster,Geometric, Text,PageNumber))}
Application-Comments	{"Common-Content"}

PERMITTED

--Generic--

Resource	{AS_8613}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Protection	{AS_8613}
Default-Value-List	{AS_8613}

```
PageNumber

    REQUIRED

    --Generic--

    Object-Type                     {basic-logical-object}
    Object-Class-Identifier         {AS_8613}
    Content-Generator               {<content-generator-2>}
    Application-Comments            {"PageNumber"}

    PERMITTED

    --Generic--

    Resource                        {AS_8613}
    Presentation-Style              {STYLE_OF(P-Style2)}
    Content-Architecture-Class      {2 8 2 6 1}    /* cp */
    User-Readable-Comments          {AS_8613}
    User-Visible-Name               {AS_8613}
    Protection                      {AS_8613}
    Layout-Style                    {STYLE_OF(L-Style2)}

L-Style1

    REQUIRED

    Layout-Style-Identifier         {AS_8613}
    Layout-Object-Class             {AS_8613}

    PERMITTED

    User-Readable-Comments          {AS_8613}
    User-Visible-Name               {AS_8613}

L-Style2

    REQUIRED

    Layout-Style-Identifier         {AS_8613}

    PERMITTED

    Block-Alignment                 {AS_8613}
    Concatenation                   {AS_8613}
    Indivisibility                  {AS_8613}
    Layout-Category                 {AS_8613}
    Layout-Object-Class             {AS_8613}
    New-Layout-Object               {AS_8613}
    Same-Layout-Object              {AS_8613}
    Offset                          {AS_8613}
```

Separation	{AS_8613}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}

L-Style3

REQUIRED

Layout-Style-Identifier	{AS_8613}

PERMITTED

Indivisibility	{AS_8613}
Layout-Object-Class	{AS_8613}
New-Layout-Object	{AS_8613}
Same-Layout-Object	{AS_8613}
Synchronization	{AS_8613}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}

L-Style4

REQUIRED

Layout-Style-Identifier	{AS_8613}

PERMITTED

Block-Alignment	{AS_8613}
Concatenation	{AS_8613}
Indivisibility	{AS_8613}
Layout-Category	{AS_8613}
Layout-Object-Class	{AS_8613}
New-Layout-Object	{AS_8613}
Offset	{AS_8613}
Same-Layout-Object	{AS_8613}
Separation	{AS_8613}
Synchronisation	{AS_8613}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}

L-Style5

REQUIRED

Layout-Style-Identifier	{AS_8613}

PERMITTED

Block-Alignment	{AS_8613}
Concatenation	{AS_8613}
Fill-Order	{AS_8613}
Indivisibility	{AS_8613}

```
    Layout-Category                {AS_8613}
    Layout-Object-Class            {AS_8613}
    New-Layout-Object              {AS_8613}
    Offset                         {AS_8613}
    Same-Layout-Object             {AS_8613}
    Separation                     {AS_8613}
    Synchronisation                {AS_8613}
    User-Readable-Comments         {AS_8613}
    User-Visible-Name              {AS_8613}


L-Style6

    REQUIRED

    Layout-Style-Identifier        {AS_8613}

    PERMITTED

    Block-Alignment                {AS_8613}
    Indivisibility                 {AS_8613}
    Layout-Category                {AS_8613}
    Layout-Object-Class            {AS_8613}
    New-Layout-Object              {AS_8613}
    Offset                         {AS_8613}
    Same-Layout-Object             {AS_8613}
    Separation                     {AS_8613}
    Synchronisation                {AS_8613}
    User-Readable-Comments         {AS_8613}
    User-Visible-Name              {AS_8613}


P-Style1

    REQUIRED

    Presentation-Style-Identifier {AS_8613}

    PERMITTED

    Border                         {AS_8613}
    Character-Presentation-        /* see section on character
     Attributes                     content */
    Colour                         {AS_8613}
    Transparency                   {AS_8613}
    User-Readable-Comments         {AS_8613}
    User-Visible-Name              {AS_8613}


P-Style2

    REQUIRED
```

```
   Presentation-Style-Identifier {AS_8613}
   Content-Architecture-Class    CASE
                                 cp   {2 8 2 6 1}
                                 cfp  {2 8 2 6 2}

   PERMITTED

   Border                        {AS_8613}
   Character-Presentation-       /* see section on character
    Attributes                    content */
   Colour                        {AS_8613}
   Transparency                  {AS_8613}
   User-Readable-Comments        {AS_8613}
   User-Visible-Name             {AS_8613}


P-Style3

   REQUIRED

   Presentation-Style-Identifier {AS_8613}
   Content-Architecture-Class    {2 8 2 7 2}    /* rfp */

   PERMITTED

   Border                        {AS_8613}
   Colour                        {AS_8613}
   Raster-Graphic-Presentation-  /* see section on raster
    Attributes                    graphics content */
   Transparency                  {AS_8613}
   User-Readable-Comments        {AS_8613}
   User-Visible-Name             {AS_8613}


P-Style4

   REQUIRED

   Presentation-Style-Identifier {AS_8613}
   Content-Architecture-Class    {2 8 2 7 2}    /* gfp */


   PERMITTED

   Border                        {AS_8613}
   Colour                        {AS_8613}
   Geometric-Graphic-Presentation-/* see section on geometric
    Attributes                    content */
   Transparency                  {AS_8613}
   User-Readable-Comments        {AS_8613}
   User-Visible-Name             {AS_8613}
```

16.2.4 LAYOUT STRUCTURE

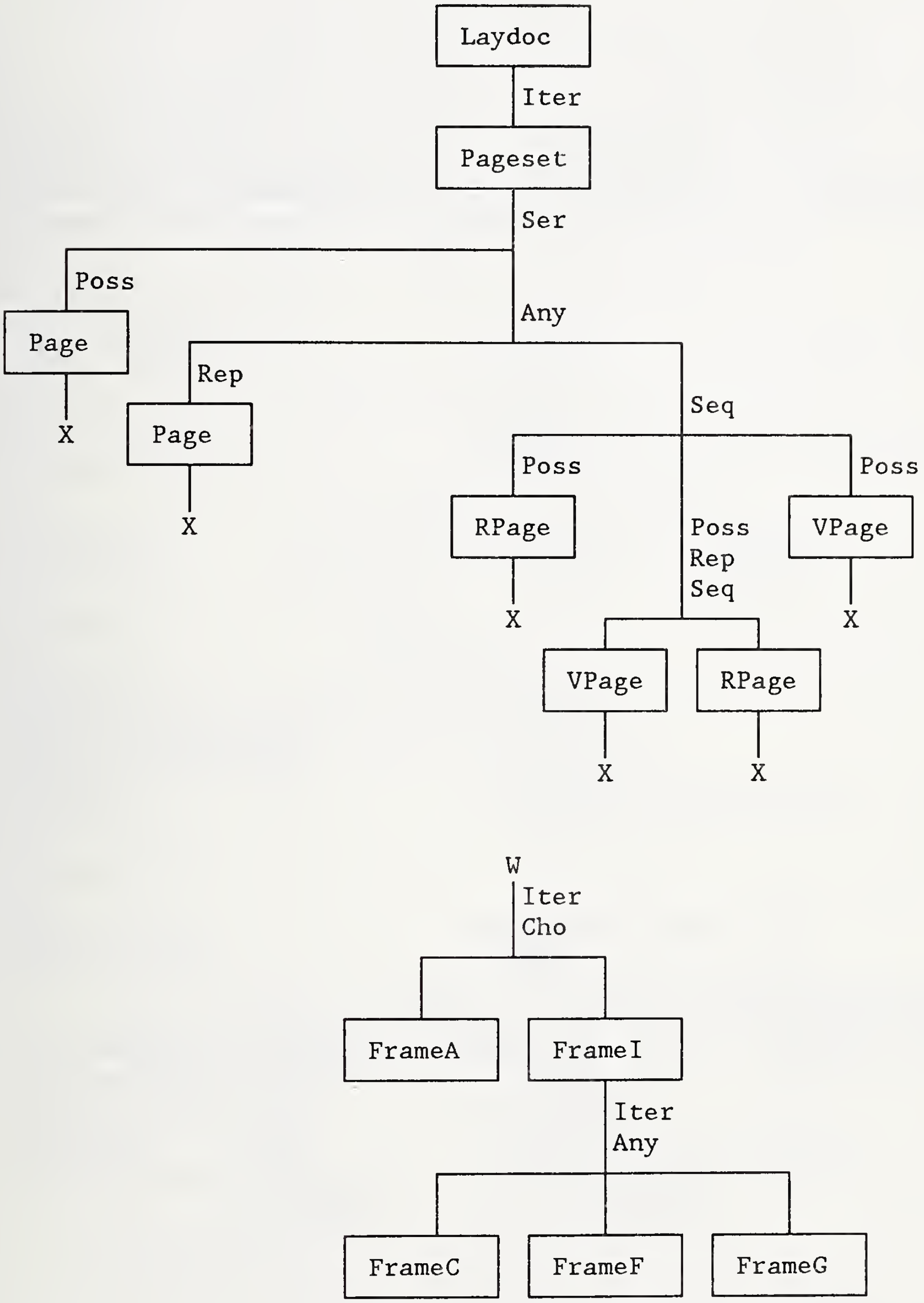


Diagram of Layout Structure (1 of 2)

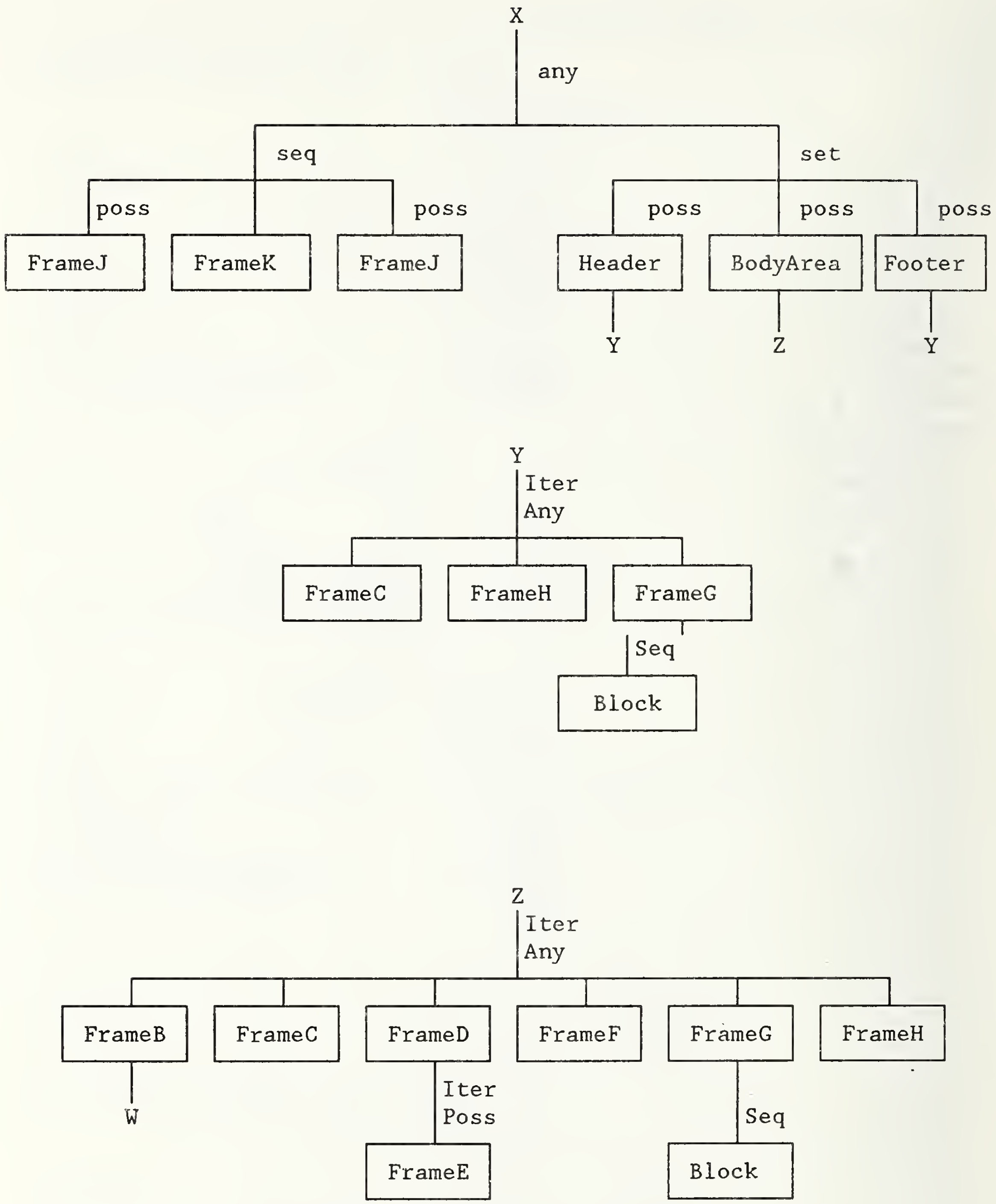


Diagram of Layout Structure (2 of 2)

Laydoc

```
REQUIRED

--Generic--

Object-Type                       {document-layout-root}
Object-Class-Identifier           {AS_8613}
Generator-for-Subordinates        {iter(pageset)}
Application-Comments              {"Laydoc"}

--Specific--

Object-Identifier                 {AS_8613}
Object-Class                      CASE
                                  fda    {N/A}
                                  fpda   {OBJECT_CLASS_ID_OF(Laydoc)}
Subordinates                      {AS_8613}


PERMITTED

--Generic--

Resource                          {AS_8613}
User-Readable-Comments            {AS_8613}
User-Visible-Name                 {AS_8613}
Default-Value-Lists               {AS_8613}
Bindings                          {Initialization (PGnum)}

--Specific--

Object-Type                       {document-layout-root}
Object-Class                      CASE
                             fda          {OBJECT_CLASS_ID_OF(Laydoc)}
                             fpda         {N/A}
User-Readable-Comments            {AS_8613}
User-Visible-Name                 {AS_8613}
Default-Value-Lists               {AS_8613}
Bindings                          {Initialization (PGnum)}
Application-Comments              {"Laydoc"}
```

PageSet

In the expression for "Generator for Subordinates" in this object, there are two instances each of the objects 'Rpage.x' and 'Vpage.y'. The dot notation is used to show that both occurrences of Rpage.x are objects of the type 'Rpage', and if they are used in a generic structure, they must both refer to the same single occurrence of an 'Rpage' object class.

REQUIRED

--Generic--

```
Object-Type                     {pageset}
Object-Class-Identifier         {AS_8613}
Generator-for-Subordinates      {ser(poss(Page),any(rep(Page),
                                      seq(poss(Rpage.x),
                                     poss(rep(seq(Vpage.y,
                                                   Rpage.x)),
                                        poss(Vpage.y)))))}
Application-Comments            {"PageSet"}
```

--Specific--

```
Object-Identifier               {AS_8613}
Object-Class                    CASE
                           fda       {N/A}
                           fpda      {OBJECT_CLASS_ID_OF(PageSet)}

Subordinates                    {AS_8613}
```

PERMITTED

--Generic--

```
Resource                        {AS_8613}
User-Readable-Comments          {AS_8613}
User-Visible-Name               {AS_8613}
Bindings                        {Initialization (PGnum)}
```

--Specific--

```
Object-Type                     {pageset}
Object-Class                    CASE
                           fda       {OBJECT_CLASS_ID_OF(PageSet)}
                           fpda      {N/A}
User-Readable-Comments          {AS_8613}
User-Visible-Name               {AS_8613}
Bindings                        {Initialization (PGnum)}
Application-Comments            {"PageSet"}
```

Page

```
   REQUIRED

   --Generic--

   Object-Type                    {page}
   Object-Class-Identifier        {AS_8613}
   Generator-for-Subordinates     {any(seq(poss(FrameJ),
                                       FrameK,poss(FrameJ)),
                                       set(poss(Header),
                                       poss(BodyArea),
                                       poss(Footer)))}
   Application-Comments           {"Page"}

   --Specific--

   Object-Identifier              {AS_8613}
   Object-Class                   CASE
                             fda       {N/A}
                             fpda      {OBJECT_CLASS_ID_OF(Page)}
   Subordinates                   {AS_8613}

   PERMITTED

   --Generic--

   Resource                       {AS_8613}
   Dimensions           Basic     {{9240,12400}
                                  /* CARA North American Letter and
                                  ISO A4 portrait */
                                  {9240,12400}
                                  /* ARA North American Letter
                                  portrait and landscape */
                                  {9240,13200}}
                                  /* ARA ISO A4 portrait */
                    Non-Basic     {{9240,12400} {12400,9240}
                                  /* ARA North American Legal
                                  portrait and landscape */
                                  {13200,9240}  /* ARA ISO A4
                                  landscape */
                                  {13200,18480}  {18480,13200}
                                  /* ARA ISO A3 portrait and
                                  landscape */
                                  {18480,26040}  /* ARA ISO A2 */
                                  {26040,36960}  /* ARA ISO A1 */
                                  {36960, 52080}  /* ARA ISO A0 */
                                  {12520,19560} {19560,12520}
                                  /* ARA ANSI B portrait and
                                  landscape */
                                  {10200,13200} /* NPS North
                                  American Letter */
```

```
                                          {10200,16800} /* NPS North
                                          American Legal */
                                          {9920,14030} /* NPS ISO A4 */
                                          {14030,19840} /* NPS ISO A3 */
                                          {13200,20400}}/* NPS ANSI B */
Medium-Type              Basic            {{10200,13200} /* NPS North
                                          American Letter */
                                          {9920,14030}} /* NPS ISO A4 */
                     Non-Basic            {{10200,16800} /* NPS North
                                          American Legal */
                                          {14030,19840} /* NPS ISO A3 */
                                          {13200,20400}}/* NPS ANSI B */
                                          {#Side of Sheet{'unspecified'}} /*
User-Readable-Comments                    {AS_8613}
User-Visible-Name                         {AS_8613}
Transparency                              {AS_8613}
Colour                                    {AS_8613}
Page-Position                             {AS_8613}
Bindings                                  {MANIPULATION (PGnum)}

--Specific--

Object-Type                               {page}
Object-Class                              CASE
                                  fda     {OBJECT_CLASS_ID_OF(Page)}
                                  fpda    {N/A}
Dimensions               Basic            {{9240,12400}
                                          /* CARA North American Letter and
                                          ISO A4 portrait */
                                          {9240,12400}
                                          /* ARA North American Letter
                                          portrait and landscape */
                                          {9240,13200}}
                                          /* ARA ISO A4 portrait */
                     Non-Basic            {{9240,12400} {12400,9240}
                                          /* ARA North American Legal
                                          portrait and landscape */
                                          {13200,9240}  /* ARA ISO A4
                                          landscape */
                                          {13200,18480}  {18480,13200}
                                          /* ARA ISO A3 portrait and
                                          landscape */
                                          {18480,26040}  /* ARA ISO A2 */
                                          {26040,36960}  /* ARA ISO A1 */
                                          {36960, 52080}  /* ARA ISO A0 */
                                          {12520,19560} {19560,12520}
                                          /* ARA ANSI B portrait and
                                          landscape */
                                          {10200,13200} /* NPS North
                                          American Letter */
                                          {10200,16800} /* NPS North
                                          American Legal */
```

		{9920,14030} /* NPS ISO A4 */ {14030,19840} /* NPS ISO A3 */ {13200,20400}}/* NPS ANSI B */
Medium-Type	Basic	{{10200,13200} /* NPS North American Letter */ {9920,14030}} /* NPS ISO A4 */
	Non-Basic	{{10200,16800} /* NPS North American Legal */ {14030,19840} /* NPS ISO A3 */ {13200,20400}}/* NPS ANSI B */
		{#Side of Sheet{'unspecified'}} /*
User-Readable-Comments		{AS_8613}
User-Visible-Name		{AS_8613}
Transparency		{AS_8613}
Page-Position		{AS_8613}
Colour		{AS_8613}
Bindings		{MANIPULATION (PGnum)}
Application-Comments		{"Page"}

RELATIONS

Dimensions == SIBLING(Page(#dimensions));

Note: The relations formulae are used to satisfy the constraint that all pages within an instance of a Pageset must have the same dimensions. Relations apply to both generic and specific structures.

RPage

```
REQUIRED

--Generic--

Object-Type                       {page}
Object-Class-Identifier           {AS_8613}
Generator-for-Subordinates        {any(seq(poss(FrameJ),
                                       FrameK,poss(FrameJ)),
                                       set(poss(Header),
                                       poss(BodyArea),
                                       poss(Footer)))}
Medium-Type           Basic       {{10200,13200} /* NPS North
                                                 American Letter */
                                  {9920,14030}}/* NPS ISO A4 */
                  Non-Basic       {{10200,16800} /* NPS North
                                                 American Legal */
                                  {14030,19840} /* NPS ISO A3 */
                                  {13200,20400}}/* NPS ANSI B */
                                   {#Side Of Sheet{'recto'}}
Application-Comments              {"Rpage"}

--Specific--

Object-Identifier                 {AS_8613}
Object-Class                      CASE
                         fda          {N/A}
                         fpda         {OBJECT_CLASS_ID_OF(RPage)}
Subordinates                      {AS_8613}

PERMITTED

--Generic--

Resource                          {AS_8613}
Dimensions            Basic       {{9240,12400}
                                  /* CARA North American Letter and
                                  ISO A4 portrait */
                                  {9240,12400}
                                  /* ARA North American Letter
                                  portrait and landscape */
                                  {9240,13200}}
                                  /* ARA ISO A4 portrait */
                  Non-Basic       {{9240,12400} {12400,9240}
                                  /* ARA North American Legal
                                  portrait and landscape */
                                  {13200,9240}  /* ARA ISO A4
                                                     landscape */
                                  {13200,18480}  {18480,13200}
                                  /* ARA ISO A3 portrait and
                                  landscape */
```

```
                                 {18480,26040}  /* ARA ISO A2 */
                                 {26040,36960}  /* ARA ISO A1 */
                                 {36960, 52080}  /* ARA ISO A0 */
                                 {12520,19560} {19560,12520}
                                 /* ARA ANSI B portrait and
                                                landscape */
                                 {10200,13200} /* NPS North
                                                American Letter */
                                 {10200,16800} /* NPS North
                                                American Legal */
                                 {9920,14030} /* NPS ISO A4 */
                                 {14030,19840} /* NPS ISO A3 */
                                      {13200,20400}}/* NPS ANSI B */
   Bindings                      {MANIPULATION (PGnum)}
   User-Readable-Comments        {AS_8613}
   User-Visible-Name             {AS_8613}
   Transparency                  {AS_8613}
   Page-Position                 {AS_8613}
   Colour                        {AS_8613}

   --Specific--

   Object-Type                   {page}
   Object-Class                  CASE
                       fda       {OBJECT_CLASS_ID_OF(RPage)}
                       fpda      {N/A}
   Dimensions          Basic     {{9240,12400}
                                 /* CARA North American Letter and
                                 ISO A4 portrait */
                                 {9240,12400}
                                 /* ARA North American Letter
                                 portrait and landscape */
                                 {9240,13200}}
                                 /* ARA ISO A4 portrait */
                   Non-Basic     {{9240,12400} {12400,9240}
                                 /* ARA North American Legal
                                 portrait and landscape */
                                 {13200,9240}  /* ARA ISO A4
                                                     landscape */
                                 {13200,18480}  {18480,13200}
                                 /* ARA ISO A3 portrait and
                                 landscape */
                                 {18480,26040}  /* ARA ISO A2 */
                                 {26040,36960}  /* ARA ISO A1 */
                                 {36960, 52080}  /* ARA ISO A0 */
                                 {12520,19560} {19560,12520}
                                 /* ARA ANSI B portrait and
                                                landscape */
                                 {10200,13200} /* NPS North
                                                American Letter */
                                 {10200,16800} /* NPS North
American Legal */
```

```
                                           {9920,14030} /* NPS ISO A4 */
                                           {14030,19840} /* NPS ISO A3 */
                                           {13200,20400}}/* NPS ANSI B */
Medium-Type            Basic               {{10200,13200} /* NPS North
                                                        American Letter */
                                           {9920,14030}} /* NPS ISO A4 */
                   Non-Basic               {{10200,16800} /* NPS North
                                                        American Legal */
                                           {14030,19840} /* NPS ISO A3 */
                                           {13200,20400}}/* NPS ANSI B */
                                           {#Side Of Sheet{'recto'}}
Bindings                                   {MANIPULATION (PGnum)}
User-Readable-Comments                     {AS_8613}
User-Visible-Name                          {AS_8613}
Transparency                               {AS_8613}
Page-Position                              {AS_8613}
Colour                                     {AS_8613}
Application-Comments                       {"Rpage"}

RELATIONS

Dimensions                  == SIBLING(Vpage(#dimensions));
Dimensions                  == SIBLING(Page(#dimensions));
```

```
VPage

   REQUIRED

   --Generic--

   Object-Type                    {page}
   Object-Class-Identifier        {AS_8613}
   Generator-for-Subordinates     {any(seq(poss(FrameJ,
                                       FrameK,poss(FrameJ)),
                                       set(poss(Header),
                                       poss(BodyArea);
                                       poss(Footer)))}
   Medium-Type          Basic     {{10200,13200} /* NPS North
                                                 American Letter */
                                  {9920,14030}} /* NPS ISO A4 */
                    Non-Basic     {{10200,16800} /* NPS North
                                                 American Legal */
                                  {14030,19840} /* NPS ISO A3 */
                                  {13200,20400}}/* NPS ANSI B */
                                  {#Side Of Sheet{'verso'}}
   Application-Comments           {"Vpage"}

   --Specific--

   Object-Identifier              {AS_8613}
   Object-Class                   CASE
                        fda       {N/A}
                        fpda      {OBJECT_CLASS_ID_OF(VPage)}
   Subordinates                   {AS_8613}

   PERMITTED

   --Generic--

   Resource                       {AS_8613}
   Dimensions           Basic     {{9240,12400}
                                  /* CARA North American Letter and  ISO
A4 portrait */

                                  {9240,12400}
                                  /* ARA North American Letter
                                  portrait and landscape */
                                  {9240,13200}}
                                  /* ARA ISO A4 portrait */
                    Non-Basic     {{9240,12400} {12400,9240}
                                  /* ARA North American Legal
portrait and landscape */

                                  {13200,9240}  /* ARA ISO A4
                                                      landscape */
                                  {13200,18480}  {18480,13200}
                                  /* ARA ISO A3 portrait and
                                  landscape */
```

```
                                          {18480,26040}  /* ARA ISO A2 */
                                          {26040,36960}  /* ARA ISO A1 */
                                          {36960, 52080}  /* ARA ISO A0 */
                                          {12520,19560} {19560,12520}
                                          /* ARA ANSI B portrait and
                                          landscape */
                                          {10200,13200} /* NPS North
                                                         American Letter */
                                          {10200,16800} /* NPS North
                                                         American Legal */
                                          {9920,14030} /* NPS ISO A4 */
                                          {14030,19840} /* NPS ISO A3 */
                                          {13200,20400}}/* NPS ANSI B */
Bindings                                  {MANIPULATION(PGnum)}
User-Readable-Comments                    {AS_8613}
User-Visible-Name                         {AS_8613}
Transparency                              {AS_8613}
Page-Position                             {AS_8613}
Colour                                    {AS_8613}

--Specific--

Object-Type                               {page}
Object-Class                              CASE
                                    fda       {OBJECT_CLASS_ID_OF(VPage)}
                                    fpda      {N/A}
Dimensions                Basic           {{9240,12400}
                                          /* CARA North American Letter and
                                          ISO A4 portrait */
                                          {9240,12400}
                                          /* ARA North American Letter
                                          portrait and landscape */
                                          {9240,13200}}
                                          /* ARA ISO A4 portrait */
                      Non-Basic           {{9240,12400} {12400,9240}
                                          /* ARA North American Legal
                                          portrait and landscape */
                                          {13200,9240}  /* ARA ISO A4
                                                             landscape */.
                                          {13200,18480}  {18480,13200}
                                          /* ARA ISO A3 portrait and
                                          landscape */
                                          {18480,26040}  /* ARA ISO A2 */
                                          {26040,36960}  /* ARA ISO A1 */
                                          {36960, 52080}  /* ARA ISO A0 */
                                          {12520,19560} {19560,12520}
                                          /* ARA ANSI B portrait and
                                                         landscape */
                                          {10200,13200} /* NPS North
                                                         American Letter */
                                          {10200,16800} /* NPS North
                                                         American Legal */
```

```
                                       {9920,14030} /* NPS ISO A4 */
                                       {14030,19840} /* NPS ISO A3 */
                                       {13200,20400}}/* NPS ANSI B */
Medium-Type          Basic             {{10200,13200} /* NPS North
                                                      American Letter */
                                       {9920,14030}} /* NPS ISO A4 */
                     Non-Basic         {{10200,16800} /* NPS North
                                                      American Legal */
                                       {14030,19840} /* NPS ISO A3 */
                                       {13200,20400}}/* NPS ANSI B */
                                       {#Side Of Sheet{'verso'}}
Bindings                               {MANIPULATION(PGnum)}
User-Readable-Comments                 {AS_8613}
User-Visible-Name                      {AS_8613}
Transparency                           {AS_8613}
Page-Position                          {AS_8613}
Colour                                 {AS_8613}
Application-Comments                   {"Vpage"}

RELATIONS

Dimensions                   == SIBLING(Page(#dimensions));
Dimensions                   == SIBLING(Rpage(#dimensions));
```

Header

```
REQUIRED

--Generic--

Object-Type                       {frame}
Object-Class-Identifier           {AS_8613}
Generator-for-Subordinates        {iter(any(FrameH,
                                  FrameC,FrameG))}
Position                          {#fixed{AS_8613}}
Dimensions                        {#horizontal{#fixed{AS_8613}}}
                                  {#vertical{#fixed{AS_8613}}}
Application-Comments              {"Header"}

--Specific--

Object-Identifier                 {AS_8613}
Object-Class                      CASE
                              fda {N/A}
                             fpda {OBJECT_CLASS_ID_OF(Header)}
Subordinates                      {AS_8613}

PERMITTED

--Generic--

Resource                          {AS_8613}
User-Readable-Comments            {AS_8613}
User-Visible-Name                 {AS_8613}
Transparency                      {AS_8613}
Colour                            {AS_8613}
Border                            {AS_8613}
Layout-Path                       {90, 270}

--Specific--

Object-Type                       {frame}
Object-Class                      CASE
                              fda {OBJECT_CLASS_ID_OF(Header)}
                             fpda {N/A}

Position                          {AS_8613}
Dimensions                        {#horizontal{#fixed{AS_8613}}}
                                  {#vertical{#fixed{AS_8613}}}
User-Readable-Comments            {AS_8613}
User-Visible-Name                 {AS_8613}
Transparency                      {AS_8613}
Colour                            {AS_8613}
Border                            {AS_8613}
Layout-Path                       {90,270}
Imaging-Order                     {AS_8613}
```

```
Application-Comments          {"Header"}

RELATIONS

Position#y+Dimensions#y <(SIBLING(BodyArea(#position#y))) ;
Position#y+Dimensions#y <(SIBLING(Footer(#position#y))) ;
```

Note: The relations formulae are used to satisfy the constraint that Header frames, Body frames and Footer frames must not overlap. It is also assumed that the bottom of a header frame must be higher up on the page than the top of either a body frame or a footer frame, and that the bottom of a body frame must be higher up on the page than the top of a footer frame.

Footer

REQUIRED

--Generic--

Object-Type		{frame}
Object-Class-Identifier		{AS_8613}
Generator-for-Subordinates		{iter(any(FrameH, FrameC,FrameG))}
Position		{#fixed{AS_8613}}
Dimensions		{AS_8613}
Application-Comments		{"Footer"}

--Specific--

Object-Identifier		{AS_8613}
Object-Class		CASE
	fda	{N/A}
	fpda	{OBJECT_CLASS_ID_OF(Footer)}
Subordinates		{AS_8613}

PERMITTED

--Generic--

Resource		{AS_8613}
User-Readable-Comments		{AS_8613}
User-Visible-Name		{AS_8613}
Transparency		{AS_8613}
Colour		{AS_8613}
Border		{AS_8613}
Layout-Path		{90,270}

--Specific--

Object-Type		{frame}
Object-Class		CASE
	fda	{OBJECT_CLASS_ID_OF(Footer)}
	fpda	{N/A}
Position		{AS_8613}
Dimensions		{#fixed{AS_8613}}
User-Readable-Comments		{AS_8613}
User-Visible-Name		{AS_8613}
Transparency		{AS_8613}
Colour		{AS_8613}
Border		{AS_8613}
Layout-Path		{90,270}
Imaging-Order		{AS_8613}
Application-Comments		{"Footer"}

RELATIONS

```
Position#y >(SIBLING(Header(#Dimensions#y+position#y))) ;
Position#y > <(SIBLING(BodyArea(Dimensions#y+position#y))) ;
```

BodyArea

```
    REQUIRED

    --Generic--

    Object-Type                        {frame}
    Object-Class-Identifier            {AS_8613}
    Generator-for-Subordinates         {iter(any(FrameB,FrameC,
                                       FrameD,FrameF,
                                       FrameG,FrameH))}
    Position                           {#fixed{AS_8613}}
    Dimensions                         {#horizontal{#fixed{AS_8613}}}
                                       {#vertical{#fixed{AS_8613}}}
    Application-Comments               {"BodyArea"}

    --Specific--

    Object-Identifier                  {AS_8613}
    Object-Class                       CASE
                                  fda  {N/A}
                                  fpda {OBJECT_CLASS_ID_OF(BodyArea)}
    Subordinates                       {AS_8613}

    PERMITTED

    --Generic--

    Resource                           {AS_8613}
    User-Readable-Comments             {AS_8613}
    User-Visible-Name                  {AS_8613}
    Transparency                       {AS_8613}
    Colour                             {AS_8613}
    Border                             {AS_8613}
    Layout-Path                        {90|270}

    --Specific--

    Object-Type                        {frame}
    Object-Class                       CASE
                                  fda  {OBJECT_CLASS_ID_OF(BodyArea)}
                                  fpda {N/A}
    Position                           {AS_8613}
    Dimensions                         {AS_8613}
    User-Readable-Comments             {AS_8613}
    User-Visible-Name                  {AS_8613}
    Transparency                       {AS_8613}
    Colour                             {AS_8613}
    Border                             {AS_8613}
    Layout-Path                        {90|270}
    Imaging-Order                      {AS_8613}
    Application-Comments               {"Body Area"}
```

RELATIONS

```
Position#y >(SIBLING(Header(#Dimensions#y+position#y))) ;
Position#y+Dimensions#y <(SIBLING(Footer(position#y))) ;
```

FrameA

FrameA is a region of the page typically representing a column. The direct subordinates are blocks of content. FrameA is at a fixed or varible position within its superior frame. The dimension orthogonal to the layout path is fixed or variable; in the direction of the layout path, the dimension is the minimum size needed to contain the subordinates.

REQUIRED

--Generic--

Object-Type		{frame}
Object-Class-Identifier		{AS_8613}
Position		{#fixed{AS_8613}| #variable{AS_8613}}

Dimensions		{#horizontal{AS_8613}
		#vertical{AS_8613}|
		{#horizontal{fixed,default}
		#vertical{Rule-B}}

Application-Comments		{"FrameA"}

--Specific--

Object-Identifier		{AS_8613}
Object-Class		CASE
	fda	{N/A}
	fpda	{OBJECT_CLASS_ID_OF(FrameA)}
Subordinates		{OBJECT_ID_OF(Block)}

PERMITTED

--Generic--

Permitted Categories	{AS_8613}
Resource	{AS_8613}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Transparency	{AS_8613}
Colour	{AS_8613}
Border	{AS_8613}

--Specific--

Object-Type		{frame}
Object-Class		CASE
	fda	{OBJECT_CLASS_ID_OF(FrameA)}
	fpda	{N/A}
Position		{AS_8613}
Dimensions		{AS_8613}
Permitted Categories		{AS_8613}

User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Transparency	{AS_8613}
Colour	{AS_8613}
Border	{AS_8613}
Imaging-Order	{AS_8613}
Application-Comments	{"FrameA"}

FrameB

FrameB is a region of the page, typically representing a number of columns. The direct subordinates may only be frames of type A or I. The position is variable (i.e., determined by a rule). The dimension orthogonal to the layout path is maximum for the position; in the direction of the layout path, the dimension is the minimum size needed to contain the subordinates.

REQUIRED

--Generic--

```
Object-Type                      {frame}
Object-Class-Identifier          {AS_8613}
Generator-for-Subordinates       {any(iter(FrameA)iter(FrameI))}
Position                         {#variable{AS_8613}}
Dimensions                       {#horizontal{fixed
                                 {AS_8613},default}
                                 #vertical{Rule-B}}
Application-Comments             {"FrameB"}
```

--Specific--

```
Object-Identifier                {AS_8613}
Object-Class                     CASE
                            fda  {N/A}
                            fpda {OBJECT_CLASS_ID_OF(FrameB)}
Subordinates                     {AS_8613}
```

PERMITTED

--Generic--

```
Resource                         {AS_8613}
User-Readable-Comments           {AS_8613}
User-Visible-Name                {AS_8613}
Transparency                     {AS_8613}
Colour                           {AS_8613}
Border                           {AS_8613}
Layout-Path                      {0| 90| 270}
Balance                          {AS_8613}
```

--Specific--

```
Object-Type                      {frame}
Object-Class                     CASE
                            fda  {OBJECT_CLASS_ID_OF(FrameB)}
                            fpda {N/A}
Position                         {AS_8613}
Dimensions                       {AS_8613}
User-Readable-Comments           {AS_8613}
```

User-Visible-Name	{AS_8613}
Transparency	{AS_8613}
Colour	{AS_8613}
Border	{AS_8613}
Layout-Path	{90|270}
Imaging-Order	{AS_8613}
Balance	{AS_8613}
Application-Comments	{"FrameB"}

FrameC

FrameC is a region of the page, typically representing an area of variable dimension within a column, such as is needed to contain a paragraph of text or a figure. This provides for varying layout on pages (columns or pages) as the document is edited. The direct subordinates are blocks of content. The position is variable (i.e., determined by a rule). The dimension orthogonal to the layout path is maximum for the position; in the direction of the layout path, the dimension is the minimum size needed to contain the subordinate block(s).

REQUIRED

--Generic--

Object-Type		{frame}
Object-Class-Identifier		{AS_8613}
Position		{#variable{AS_8613}}
Dimensions		{#horizontal{fixed {AS_8613},default} #vertical{Rule-B}}
Application-Comments		{"FrameC"}

--Specific--

Object-Identifier		{AS_8613}
Object-Class		CASE
	fda	{N/A}
	fpda	{OBJECT_CLASS_ID_OF(FrameC)}
Subordinates		{OBJECT_ID_OF(Block)}

PERMITTED

--Generic--

Resource		{AS_8613}
User-Readable-Comments		{AS_8613}
User-Visible-Name		{AS_8613}
Transparency		{AS_8613}
Colour		{AS_8613}
Border		{AS_8613}
Permitted-Categories		{AS_8613}

--Specific--

Object-Type		{frame}
Object-Class		CASE
	fda	{OBJECT_CLASS_ID_OF(FrameC)}
	fpda	{N/A}
Position		{AS_8613}
Dimensions		{#horizontal{default}

	#vertical{AS_8613}}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Transparency	{AS_8613}
Colour	{AS_8613}
Border	{AS_8613}
Imaging-Order	{AS_8613}
Application-Comments	{"FrameC"}
Permitted-Categories	{AS_8613}

FrameD

FrameD is a region of the page, typically representing an area of variable dimension within a column, containing a number of items side by side (e.g., text flowing around a picture). The direct subordinates may only be frames of type E. The position is variable (i.e., determined by a rule). The dimension orthogonal to the layout path is maximum for the position; in the direction of the layout path, the dimension is the minimum size needed to contain the first laid out subordinate (e.g., the picture).

REQUIRED

--Generic--

Object-Type		{frame}
Object-Class-Identifier		{AS_8613}
Generator-for-Subordinates		{iter(poss(FrameE))}
Position		{#variable{AS_8613}}

Dimensions		{#horizontal{default}
		#vertical{Rule-A}}

Application-Comments		{"FrameD"}
Layout-Path		{0,180}

--Specific--

Object-Identifier		{AS_8613}
Object-Class		CASE
	fda	{N/A}
	fpda	{OBJECT_CLASS_ID_OF(FrameD)}
Subordinates		{AS_8613}

PERMITTED

--Generic--

Resource	{AS_8613}
User Readable Comments	{AS_8613}
User Visible Name	{AS_8613}
Transparency	{AS_8613}
Colour	{AS_8613}
Border	{AS_8613}
Bindings	{MANIPULATION(PGnum)}

--Specific--

Object-Type		{frame}
Object-Class		CASE
	fda	{OBJECT_CLASS_ID_OF(FrameD)}
	fpda	{N/A}
Position		{AS_8613}
Dimensions		{#horizontal{default}}

	{#vertical{AS_8613}}
User Readable Comments	{AS_8613}
User Visible Name	{AS_8613}
Transparency	{AS_8613}
Colour	{AS_8613}
Border	{AS_8613}
Imaging Order	{AS_8613}
Bindings	{MANIPULATION(PGnum)}
Application Comments	{"FrameD"}

FrameE

FrameE is a region of the page, typically representing some text or a picture, that is side by side with the frames of this type within a superior frame of type D. The direct subordinates are blocks of content. The position is variable (i.e., determined by a rule). The dimension orthogonal to the layout path is the minimum size needed to contain the subordinate block(s). In the direction of the layout path, the dimension is either a fixed dimension or is a computed dimension equal to either the minimum size needed to contain the subordinate block(s) (e.g., picture), or the maximum size for the position (e.g., text).

REQUIRED

--Generic--

Object-Type		{frame}
Object-Class-Identifier		{AS_8613}
Position		{#variable{AS_8613}}

Dimensions		{#horizontal{Rule-B,fixed,default}
		#vertical{Rule-B}}

Permitted-Categories		{AS_8613}
Application-Comments		{"FrameE"}

--Specific--

Object-Identifier		{AS_8613}
Object-Class		CASE
	fda	{N/A}
	fpda	{OBJECT_CLASS_ID_OF(FrameE)}
Subordinates		{OBJECT_ID_OF(Block)}
Permitted-Categories		{AS_8613}

PERMITTED

--Generic--

Resource	{AS_8613}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Transparency	{AS_8613}
Colour	{AS_8613}
Border	{AS_8613}

--Specific--

Object-Type		{frame}
Object-Class		CASE
	fda	{OBJECT_CLASS_ID_OF(FrameE)}
	fpda	{N/A}
Position		{AS_8613}
Dimensions		{AS_8613}

User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Transparency	{AS_8613}
Colour	{AS_8613}
Border	{AS_8613}
Imaging-Order	{AS_8613}
Application-Comments	{"FrameE"}

FrameF

Frame F is a region of the page typically used to contain a footnote. The direct subordinates are blocks of content. The frame is positioned within its superior in reverse order at a fixed position from its superior in the direction orthogonal to the layout path and at a variable position in the direction of the layout path. The dimension orthogonal to the layout path is maximum for the position; in the direction of the layout path, the dimension is the minimum size needed to contain the subordinate blocks.

REQUIRED

--Generic--

Object-Type		{frame}
Object-Class-Identifier		{AS_8613}
Position		{#variable{#fill order{reversed}},#offset{AS_8613}, #separation{AS_8613}, #alignment{AS_8613}}
Dimensions		{#horizontal{default} #vertical{Rule-B}}
Permitted-Categories		{AS_8613}
Application-Comments		{"FrameF"}

--Specific--

Object-Identifier		{AS_8613}
Object-Class		CASE
	fda	{N/A}
	fpda	{OBJECT_CLASS_ID_OF(FrameF)}
Subordinates		{OBJECT-OF(Block)}
Permitted-Categories		{AS_8613}

PERMITTED

--Generic--

Resource	{AS_8613}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Transparency	{AS_8613}
Colour	{AS_8613}
Border	{AS_8613}

--Specific--

Object-Type		{frame}
Object-Class		CASE
	fda	{OBJECT_CLASS_ID_OF(FrameF)}

```
                          fpda {N/A}
Position                      {AS_8613}
Dimensions                    {#horizontal{default}
                              #vertical{AS_8613}}
User-Readable-Comments        {AS_8613}
User-Visible-Name             {AS_8613}
Transparency                  {AS_8613}
Colour                        {AS_8613}
Border                        {AS_8613}
Imaging-Order                 {AS_8613}
Application-Comments          {"FrameF"}
```

FrameG

FrameG is a region of the page, typically representing a set of pieces of content placed at defined positions. This provides for complex, fixed, relatively positioned pieces of content, including overlapping pieces of content (e.g., overlapping pictures or pictures and text). The direct subordinates are blocks of content. The position relative to the superior is fixed; the dimensions are also fixed.

REQUIRED

--Generic--

Object-Type		{frame}
Object-Class-Identifier		{AS_8613}
Position		{#fixed{AS_8613}}

Dimensions		{#horizontal{AS_8613}
		#vertical{AS_8613}}

Application-Comments		{"FrameG"}

--Specific--

Object-Identifier		{AS_8613}
Object-Class		CASE
	fda	{N/A}
	fpda	{OBJECT_CLASS_ID_OF(FrameG)}
Subordinates		{OBJECT_ID_OF(Block)}

PERMITTED

--Generic--

Generator-For-Subordinates	{seq(Block)}
Resource	{AS_8613}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Transparency	{AS_8613}
Colour	{AS_8613}
Border	{AS_8613}
Permitted-Categories	{AS_8613}

--Specific--

Object-Type		{frame}
Object-Class		CASE
	fda	{OBJECT_CLASS_ID_OF(FrameG)}
	fpda	{N/A}
Position		{AS_8613}
Dimensions		{AS_8613}
User-Readable-Comments		{AS_8613}
User-Visible-Name		{AS_8613}
Transparency		{AS_8613}

Colour	{AS_8613}
Border	{AS_8613}
Imaging-Order	{AS_8613}
Application-Comments	{"FrameG"}
Permitted-Categories	{AS_8613}

FrameH

FrameH is a region of the page, typically representing a variable piece of logical information in the header or footer area of a page (e.g., current chapter number). The direct subordinates are blocks of content. The frame, in all cases, specifies that its content is derived from the use of the attribute "logical source" referring to a logical object of "header or footer content." The position is fixed or variable (i.e., determined by a rule.) The dimension orthogonal to the layout path is maximum for the position; in the direction of the layout path, the dimension is the minimum size needed to contain the subordinate blocks.

REQUIRED

--Generic--

Object-Type		{frame}
Object-Class-Identifier		{AS_8613}
Position		{#fixed{AS_8613}| #variable {AS_8613}}
Dimensions		{#horizontal{AS_8613} #vertical{AS_8613}| {#horizontal{default} #vertical{Rule-B}}
Logical-Source		{OBJECT_CLASS_ID_OF(Hdr-or-Ftr-Content)}
Application-Comments		{"FrameH"}

--Specific--

Object-Identifier		{AS_8613}
Object-Class		CASE
	fda	{N/A}
	fpda	{OBJECT_CLASS_ID_OF(FrameH)}
Subordinates		{OBJECT_ID_OF(Block)}

PERMITTED

--Generic--

Resource	{AS_8613}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Border	{AS_8613}
Layout-Path	{AS_8613}
Generator-for-Subordinates	{iter(Block)}

--Specific--

Object-Type	{frame}
Object-Class	CASE

	fda	{OBJECT_CLASS_ID_OF(FrameH)}
	fpda	{N/A}
Position		{AS_8613}

Dimensions		{#horizontal{default}
		#vertical{AS_8613}}

User-Readable-Comments		{AS_8613}
User-Visible-Name		{AS_8613}
Border		{AS_8613}
Layout-Path		{AS_8613}
Application-Comments		{"FrameH"}

FrameI

Frame I: A region of the page typically representing a column. The frame may contain any substructure of further frames of any of the types C, F, or G. Blocks are not permitted directly subordinate to FrameI. FrameI is at a fixed position within its superior frame. The dimension orthogonal to the layout path is fixed; in the direction of the layout path, the dimension is the minimum size needed to contain the subordinates.

REQUIRED

--Generic--

Object-Type		{frame}
Object-Class-Identifier		{AS_8613}
Generator-for-Subordinates		{iter(any(FrameC, FrameF,FrameG))}
Position		{#fixed{AS_8613}}
Dimensions		{#horizontal{fixed, default} #vertical{Rule-B}}
Application-Comments		{"FrameI"}

--Specific--

Object-Identifier		{AS_8613}
Object-Class		CASE
	fda	{N/A}
	fpda	{OBJECT_CLASS_ID_OF(FrameI)}
Subordinates		{AS_8613}

PERMITTED

--Generic--

Resource	{AS_8613}
User-Readable-Comments	{AS_8613}
User-Visible-Name	{AS_8613}
Border	{AS_8613}
Transparency	{AS_8613}
Colour	{AS_8613}
Layout-Path	{90,270}

--Specific--

Object-Type		{frame}
Object-Class		CASE
	fda	{OBJECT_CLASS_ID_OF(FrameI)}
	fpda	{N/A}
Position		{AS_8613}
Dimensions		{AS_8613}
User-Readable-Comments		{AS_8613}

User-Visible-Name	{AS_8613}
Border	{AS_8613}
Transparency	{AS_8613}
Colour	{AS_8613}
Layout-Path	{90|270}
Imaging-Order	{AS_8613}
Application-Comments	{"FrameI"}

Frame J

Frame J is a single basic frame to contain header or footer contents. This is provided for compatibility with the CCITT Recommendation T.502.

REQUIRED

--Generic--

Object-Type	{frame}
Object-Class-Identifier	{AS_8613}
Logical-Source	{AS_8613}
Application-Comments	{"Frame J"}

--Specific--

Object-Class	{OBJECT_CLASS_ID_OF(FRAME J)}
Subordinates	{OBJECT_ID_OF(Block)}
Object-Identifier	{AS_8613}

PERMITTED

--Generic--

Position	{#fixed{AS_8613}}
Dimensions	{#fixed{AS_8613}}
User-visible-name	{AS_8613}
User-readable-comments	{AS_8613}

--Specific--

Object-Type	{frame}
Position	{AS_8613}
Dimensions	{#fixed{AS_8613}}
User-visible-name	{AS_8613}
User-readable-comments	{AS_8613}
Application-Comments	{"Frame J"}
Layout-Path	{270}

RELATIONS

FrameJ.2#Position#y	> FrameK.1#Position#y + FrameK.1#Dimensions#vertical

Note: The relations formulae are used to satisfy the constraint that Header frames, Body frames and Footer frames must not overlap. It is also assumed that the bottom of a header frame must be higher up on the page than the top of either a body frame or a footer frame, and that the bottom of a body frame must be higher up on the page than the top of a footer frame.

Frame K

Frame K is a single basic frame to represent the body area of a page. This is provided for compatibility with the CCITT Recommendation T.502.

REQUIRED

--Generic--

Object-Type	{frame}
Object-Class-Identifier	{AS_8613}
Application-Comments	{"Frame K"}

--Specific--

Object-Class	{OBJECT_CLASS_ID_OF(FRAME K)}
Subordinates	{OBJECT_ID_OF(Block)}
Object-Identifier	{AS_8613}

PERMITTED

--Generic--

Position	{#fixed{AS_8613}}
Dimensions	{#fixed{AS_8613}}
User-visible-name	{AS_8613}
User-readable-comments	{AS_8613}

--Specific--

Object-Type	{frame}
Position	{AS_8613}
Dimensions	{#fixed{AS_8613}}
User-visible-name	{AS_8613}
User-readable-comments	{AS_8613}
Application-Comments	{"Frame K"}
Layout-Path	{270}

RELATIONS

FrameK.1#Position#y	> FrameJ.1#Position#y + FrameJ.1#Dimensions#vertical

Note: The relations formulae are used to satisfy the constraint that Header frames, Body frames and Footer frames must not overlap. It is also assumed that the bottom of a header frame must be higher up on the page than the top of either a body frame or a footer frame, and that the bottom of a body frame must be higher up on the page than the top of a footer frame.

Block

REQUIRED

--Generic--

Object-Type		{block}
Object-Class-Identifier		{AS_8613}
Content-Architecture-Class		{AS_8613}
Application-Comments		{"Block"}

--Specific--

Content-Architecture-Class		{AS_8613}
Position		{AS_8613}
Dimensions		{AS_8613}
Object-Identifier		{AS_8613}
Object-Class		CASE
	fda	{N/A}
	fpda	{OBJECT_CLASS_ID_OF(Block)}

PERMITTED

--Generic--

Resource		{AS_8613}
Content-Portions		{AS_8613}
User-Readable-Comments		{AS_8613}
User-Visible-Name		{AS_8613}
Transparency		{AS_8613}
Colour		{AS_8613}
Border		{AS_8613}
Presentation-Style		{AS_8613}

--Specific--

Object-Type		{block}
Object-Class		CASE
	fda	{OBJECT_CLASS_ID_OF(Block)}
	fpda	{N/A}
Position		{AS_8613}
Dimensions		{AS_8613}
Content-Generator		{<content-generator-3>}
Content-Portions		{AS_8613}
User-Readable-Comments		{AS_8613}
User-Visible-Name		{AS_8613}
Transparency		{AS_8613}
Colour		{AS_8613}
Border		{AS_8613}
Presentation-Style		{AS_8613}

Initial-Offset	{AS_8613}
Formatting-Indicator	{AS_8613}
Graphic-Rendition	{AS_8613}
Graphic-Character-Set	{AS_8613}
Application-Comments	{"Block"}

16.2.5 CONTENT-ARCHITECTURE

16.2.5.1 Character-Content-Architecture

For the purposes of conformance, the character content architecture section presents all character content attributes in terms of default, basic and non-basic values. Defaults are presented here where they can be described in simple terms. In three cases where the values are determined algorithmically from other attribute values, reference is made to the description in IS 8613. Basic and non-basic values are supplied for each attribute. In addition, the domain of applicability for each attribute is also defined in terms of classes of content architectures. Particular areas where attributes of the standard are restricted for BASIC systems include character set support and line spacing.

Presentation-Attributes

```
     Alignment                BASIC     {start-aligned|
                                         end-aligned|
                                         centered|
                                         justified}
                              NON-BASIC {NONE}
                              DEFAULT   {start-aligned}
                              CLASS     {cf,cp,cfp}

     Character-Fonts          {
     {Font-Size               BASIC     {NONE}
                              NON-BASIC {ANY}
                              DEFAULT   {NONE}
                              CLASS     {cf,cp,cfp}}
     {Font-Identifier         BASIC     {NONE}
                              NON-BASIC {AS_8613}
                              DEFAULT   {NONE}
                              CLASS     {cf,cp,cfp}}
     }

     Character-Orientation    BASIC     {0|90|180|270}
                              NON-BASIC {NONE}
                              DEFAULT   {0}
                              CLASS     {cf,cp,cfp}

     Character-Path           BASIC     {0|90|180|270}
                              NON-BASIC {NONE}
                              DEFAULT   {0}
                              CLASS     {cf,cp,cfp}
```

```
Character-Spacing         BASIC     {AS_8613}
                          NON-BASIC {NONE}
                          DEFAULT   {120}
                          CLASS     {cf,cp,cfp}
Code-Extension-Announcer  BASIC     {AS_8613}
                          NON-BASIC {NONE}
                          DEFAULT   {AS_8613}
                          CLASS     {cf,cp,cfp}

First-Line-Offset         BASIC     {AS_8613}
                          NON-BASIC {NONE}
                          DEFAULT   {0}
                          CLASS     {cf,cp,cfp}

Formatting-Indicator      BASIC     {yes|no}
                          NON-BASIC {NONE}
                          DEFAULT   {NONE}
                          CLASS     {cf,cfp}


Graphic-Character-Sets    BASIC     {
                          {2/8 46, LSO}|
               /*ISO 6937/2 Primary Set as G0*/
                          {2/14 6/12, LS2R}|
               /*ISO 6937/2 Supplementary Set G2*/
                          {2/8 4/2,LSO}|
               /*ISO 8859/1 Primary Set as G0*/
                          {2/14 4/1, LS2R}|
               /*ISO 8859/1 Supplementary Set as G2*/
                               }
                          NON-BASIC {AS_8613}
                          DEFAULT   {{2/8 4/0, LSO}
                                {2/14 6/12, LS2R}}
                          CLASS     {cf,cp,cfp}

Character-Subrepertoire   BASIC     {
                               1|
                                    /*Full repertoire*/
                               3|
                                    /*Teletex
                                    subrepertoire*/
                               8
                                    /*8859/1
                                    subrepertoire*/
                                    }
                          NON-BASIC {AS_8613}
                          DEFAULT   {8}
                          CLASS     {cf,cp,cfp}
Graphic-Rendition         BASIC     {AS_8613}
                          NON-BASIC {NONE}
                          DEFAULT   {0}
                          CLASS     {cf,cp,cfp}
```

```
Indentation                BASIC      {AS_8613}
                           NON-BASIC  {NONE}
                           DEFAULT    {0}
                           CLASS      {cp,cfp}

Initial-Offset             {
     {Horizontal-Coordinate
                           BASIC      {AS_8613}
                           NON-BASIC  {NONE}
                           DEFAULT    {AS_8613}
                           CLASS      {cf,cfp}}
     {Vertical-Coordinate
                           BASIC      {AS_8613}
                           NON-BASIC  {NONE}
                           DEFAULT    {AS_8613}
                           CLASS      {cf,cfp}}
                           }
Itemization                {
     {Identifier-Alignment
                           BASIC      {no-itemization|
                                       start-aligned|
                                       end-aligned}
                           NON-BASIC  {NONE}
                           DEFAULT    {no-itemization}
                           CLASS      {cf,cp,cfp}}
     {Identifier-Start-Offset
                           BASIC      {AS_8613}
                           NON-BASIC  {NONE}
                           DEFAULT    {AS_8613}
                           CLASS      {cf,cp,cfp}}
     {Identifier-End-Offset
                           BASIC      {AS_8613}
                           NON-BASIC  {NONE}
                           DEFAULT    {0}
                           CLASS      {cf,cp,cfp}}
                           }

Kerning-Offset             {
     {Start-Edge-Offset    BASIC      {AS_8613}
                           NON-BASIC  {NONE}
                           DEFAULT    {0}
                           CLASS      {cf,cp,cfp}}
     {End-Edge-Offset      BASIC      {AS_8613}
                           NON-BASIC  {NONE}
                           DEFAULT    {0}
                           CLASS      {cf,cp,cfp}}
                           }

Line-Layout-Table          {
     {Tab-Reference        BASIC      {AS_8613}
                           NON-BASIC  {NONE}
                           DEFAULT    {NONE}
```

```
                                   CLASS      {cf,cp,cfp}}
     {Tab-Position                 BASIC      {AS_8613}
                                   NON-BASIC  {NONE}
                                   DEFAULT    {NONE}
                                   CLASS      {cf,cp,cfp}}
     {Alignment                    BASIC      {start-aligned|
                                               end-aligned|
                                               centered|
                                               aligned-around}
                                   NON-BASIC  {NONE}
                                   DEFAULT    {start-aligned}
                                   CLASS      {cf,cp,cfp}}
     {Alignment-String             BASIC      {AS_8613}/* Any
                                              single graphic
                                              character */
                                   NON-BASIC  {NONE}
                                   DEFAULT    {NONE}
                                   CLASS      {cf,cp,cfp}}
                                   }

Line-Progression                   BASIC      {90|270}
                                   NON-BASIC  {NONE}
                                   DEFAULT    {270}
                                   CLASS      {cf,cp,cfp}

Line-Spacing                       BASIC {100|150|200|300|400}
                                   NON-BASIC  {NONE}
                                   DEFAULT    {200}
                                   CLASS      {cf,cp,cfp}

Orphan-Size                        BASIC      {AS_8613}
                                   NON-BASIC  {NONE}
                                   DEFAULT    {1}
                                   CLASS      {cp,cfp}

Pairwise-Kerning                   BASIC      {yes|no}
                                   NON-BASIC  {NONE}
                                   DEFAULT    {no}
                                   CLASS      {cf,cp,cf}

Proportional-Line-Spacing
                                   BASIC      {yes|no}
                                   NON-BASIC  {NONE}
                                   DEFAULT    {no}
                                   CLASS      {cp,cfp}

Widow-Size                         BASIC      {AS_8613}
                                   NON-BASIC  {NONE}
                                   DEFAULT    {1}
                                   CLASS      {cp,cfp}
```

```
Control-Functions

Break-Permitted-Here
/*BPH*/                      BASIC     {N/A}
                             NON-BASIC {N/A}
                             DEFAULT   {N/A}
                             CLASS     {cp,cfp}


Carriage-Return              BASIC     {N/A}
/*CR*/                       NON-BASIC {N/A}
                             DEFAULT   {N/A}
                             CLASS     {cf,cp,cfp}

Graphic-Character-Composition
/*GCC*/                      BASIC     {0|1|2}
                             NON-BASIC {NONE}
                             DEFAULT   {0}
                             CLASS     {cf,cp,cfp}

Character-Position-Backward
/*HPB*/                      BASIC     {AS_8613}
                             NON-BASIC {NONE}
                             DEFAULT   {120}
                             CLASS     {cf,cfp}

Character-Position-Relative
/*HPR*/                      BASIC     {AS_8613}
                             NON-BASIC {NONE}
                             DEFAULT   {120}
                             CLASS     {cf,cfp}

No-Justify                   BASIC     {0}
/*JFY*/                      NON-BASIC {NONE}
                             DEFAULT   {0}
                             CLASS     {cf,cfp}


Line-Feed                    BASIC     {N/A}
/*LF*/                       NON-BASIC {N/A}
                             DEFAULT   {N/A}
                             CLASS     {cf,cp,cfp}

No-Break-Here                BASIC     {N/A}
/*NBH*    /                  NON-BASIC {N/A}
                             DEFAULT   {N/A}
                             CLASS     {cp,cfp}

Partial-Line-Down            BASIC     {N/A}
/*PLD*/                      NON-BASIC {N/A}
                             DEFAULT   {N/A}
                             CLASS     {cf,cp,cfp}
```

```
Partial-Line-Up          BASIC     {N/A}
/*PLU*/                  NON-BASIC {N/A}
                         DEFAULT   {N/A}
                         CLASS     {cf,cp,cfp}

Parallel-Texts           BASIC     {0|1|3}
/*PTX*/                  NON-BASIC {NONE}
                         DEFAULT   {0}
                         CLASS     {cp,cfp}

Set-Additional-Character-Separation
/*SACS*/                 BASIC     {AS_8613}
                         NON-BASIC {NONE}
                         DEFAULT   {0}
                         CLASS     {cf,cfp}

Set-Character-Spacing
/*SCS*/                  BASIC     {AS_8613}
                         NON-BASIC {NONE}
                         DEFAULT   {120}
                         CLASS     {cf,cp,cfp}

Select-Graphic-Rendition
/*SGR*/                  BASIC     {0|1|2|3|4|5|6|7|9
                                   |10|11|12|13|14|15
                                   |16|17|18|19|21|22
                                   |23|24|25|27|29}
                         NON-BASIC {NONE}
                         DEFAULT   {0}
                         CLASS     {cf,cp,cfp}

Select-Character-Spacing
/*SHS*/                  BASIC     {0|1|2|3}
                         NON-BASIC {4}
                         DEFAULT   {0}
                         CLASS     {cf,cp,cfp}

Set-Line-Spacing
/*SLS*/                  BASIC     {AS_8613}
                         NON-BASIC {NONE}
                         DEFAULT   {200}
                         CLASS     {cf,cp,cfp}

Start-Of-String          BASIC     {N/A}
/*SOS*/                  NON-BASIC {N/A}
                         DEFAULT   {N/A}
                         CLASS     {cfp}

Space                    BASIC     {N/A}
/*SP*/                   NON-BASIC {N/A}
                         DEFAULT   {N/A}
                         CLASS     {cf,cp,cfp}
```

Set-Reduced-Character-Separation		
/*SRCS*/	BASIC	{AS_8613}
	NON-BASIC	{NONE}
	DEFAULT	{0}
	CLASS	{cf,cp,cfp}
Start-Reverse-String		
/*SRS*/	BASIC	{0|1}
	NON-BASIC	{NONE}
	DEFAULT	{0}
	CLASS	{cf,cp,cfp}
Set-Space-Width	BASIC	{AS_8613}
/*SSW*/	NON-BASIC	{NONE}
	DEFAULT	{NONE}
	CLASS	{cf,cfp}
String-Terminator	BASIC	{N/A}
/*ST*/	NON-BASIC	{N/A}
	DEFAULT	{N/A}
	CLASS	{cfp}
Selective-Tabulation		
/*STAB*/	BASIC	{AS_8613}
	NON-BASIC	{NONE}
	DEFAULT	{NONE}
	CLASS	{cf,cp,cfp}
Substitute-Character		
/*SUB*/	BASIC	{N/A}
	NON-BASIC	{N/A}
	DEFAULT	{N/A}
	CLASS	{cf,cp, cfp}
Select-Line-Spacing		
/*SVS*/	BASIC	{0|1|2|3|4|9}
	NON-BASIC	{NONE}
	DEFAULT	{0}
	CLASS	{cf,cp,cfp}
Line-Position-Backward		
/*VPB*/	BASIC	{AS_8613}
	NON-BASIC	{NONE}
	DEFAULT	{100}
	CLASS	{cf,cp,cfp}
Line-Position-Relative		
/*VPR*/	BASIC	{AS_8613}
	NON-BASIC	{NONE}
	DEFAULT	{100}
	CLASS	{cf,cp,cfp}

Code-Extension-Control

	BASIC	{AS_8613}
	NON-BASIC	{NONE}
	DEFAULT	{NO-DEFAULT}
	CLASS	{cf,cp,cfp}

Content-Portion-Attributes

Type-of-coding	BASIC	{{2 8 3 6 0}}
	NON-BASIC	{NONE}
	DEFAULT	{{2 8 3 6 0}}
	CLASS	{cf,cp,cfp}

16.2.5.2 Raster-Graphics-Content-Architecture

The Raster Graphics Content Architecture permits the inclusion in documents of content portions containing raster graphics which represent a two-dimensional pictorial image in the form of a rectangular two-dimensional array of picture elements (pels). The content architecture is as specified in Part 7 of ISO 8613.

The Raster Graphics Content Architecture defines a formatted processable content architecture. This content architecture class supports Presentation Attributes and Content Portion Attributes. Each attribute comprising this content architecture is specified in subsequent sections in a form that has been defined previously. For each attribute, permissible values are differentiated as BASIC, NON-BASIC and DEFAULT.

Presentation Attributes

These attributes specify constraints and initial conditions relating to the layout and imaging of a raster graphics content portion. This content architecture class supports Shared Presentation Attributes and Logical Presentation Attributes.

Pel-Path	BASIC	{0,90,180,270}
	NON-BASIC	{NONE}
	DEFAULT	{0}
Line-Progression	BASIC	{90,270}
	NON-BASIC	{NONE}
	DEFAULT	{270 }
Pel-Spacing {		
{Length	BASIC	{AS_8613}
	NON-BASIC	{NONE}
	DEFAULT	{4}}

```
               {Pel-Spaces    BASIC     {AS_8613}
                              NON-BASIC {NONE}
                              DEFAULT   {1}}
                         }

     Spacing-Ratio       {
               {Line-Spacing-Value
                              BASIC     {AS_8613}
                              NON-BASIC {NONE}
                              DEFAULT   {1}}
               {Pel-Spacing-Value
                              BASIC     {AS_8613}
                              NON-BASIC {NONE}
                              DEFAULT   {1}}
                         }

     Clipping                 {
               {First-Pair-X-Coordinate
                              BASIC     {AS_8613}
                              NON-BASIC {NONE}
                              DEFAULT   {AS_8613}
               {First-Pair-Y-Coordinate
                              BASIC     {AS_8613}
                              NON-BASIC {NONE}
                              DEFAULT   {AS_8613}
               {Second-Pair-X-Coordinate
                              BASIC     {AS_8613}
                              NON-BASIC {NONE}
                              DEFAULT   {AS_8613}
               {Second-Pair-Y-Coordinate
                              BASIC     {AS_8613}
                              NON-BASIC {NONE}
                              DEFAULT   {AS_8613}
                              }

     Image-Dimensions
                              BASIC     {AS_8613}
                              NON-BASIC {NONE}
                              DEFAULT   {AS_8613}


Content-Portion-Attributes

     Number-of-Pels-Per-Line  BASIC     {AS_8613}
                              NON_BASIC {NONE}
                              DEFAULT   {NONE}

     Number-of-Lines          BASIC     {AS_8613}
                              NON_BASIC {NONE}
                              DEFAULT   {NONE}
```

```
Type-of-coding       BASIC {{28370}|/*CCITT T.6*/
                           {28371}|/*CCITT T.4 1-Dimensional*/
                           {28372}|/*CCITT T.4 2-Dimensional*/
                           {28373}}/*Bitmap*/
               NON-BASIC  {NONE}
               DEFAULT   {{28373}}

     Compression               BASIC     {compressed|
                                           uncompressed}
                               NON-BASIC {NONE}
                               DEFAULT   {compressed}
```

16.2.5.3 Geometric Graphics Content Architecture

The Geometric Graphics Content Architecture permits the inclusion in documents of content portions containing graphics primitives such as lines, markers, filled areas, graphic text, patterns and etc. The content architecture is as specified in part 8 of ISO 8613. It is based on ISO 8632, Computer Graphics Metafile (CGM).

The Geometric Graphics Content Architecture defines a formatted processable content architecture. This content architecture class supports Presentation Attributes, Content Architecture Class Attributes and Content Portion Attributes. Each attribute comprising this content architecture is specified in subsequent sections in a form that has been defined previously. For each attribute, permissible values are differentiated as BASIC, NON-BASIC and DEFAULT.

Presentation Attributes

These attributes specify constraints and initial conditions relating to the layout and imaging of a geometric graphics content portion. This content architecture class supports Presentation Attributes.

The default values for the presentation attributes are specified so as to be consistent with those specified in the Metafile Defaults in Clause 6 of ISO 8632/1. No private values are permitted for any of the presentation attributes.

Default-Bundle-Representations

```
     Line-Bundle-Representation    {
          {Bundle-Index        BASIC     {1,2,3,4,5}
                               NON-BASIC {NONE}
                               DEFAULT   {N/A}}
          {Line-Type           BASIC     {solid,
                                          dash,
                                    dot,
                                    dash-dot,
                                    dash-dot-dot}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Line-Width
               CASE Line-Width-Specification-Mode
          absolute       BASIC     {.001*max-vdcx,
                                    .001*max-vdcx,
                                    .001*max-vdcx,
                                    .001*max-vdcx,
                                    .001*max-vdcx}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}
          scaled         BASIC      {1.,1.,1.,1.,1.}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Line-Colour CASE Colour-Specification-Mode
          indexed        BASIC     {1,1,1,1,1}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}
          direct         BASIC     {foreground,
                                    foreground,
                                    foreground,
                                    foreground,
                                    foreground}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     }

Marker-Bundle-Representation  {
     {Bundle-Index       BASIC     {1,2,3,4,5}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Marker-Type        BASIC     {dot,
                                    plus,
                                    asterisk,
                                    circle,
                                    cross}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Marker-Size
               CASE Marker-Size-Specification-Mode
          absolute       BASIC     {0.01*max-vdcx,
                                    0.01*max-vdcx,
                                    0.01*max-vdcx,
                                    0.01*max-vdcx,
                                    0.01*max-vdcx}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}
          scaled         BASIC     {1.,1.,1.,1.,1.}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Marker-Colour
               CASE Colour-Specification-Mode
          indexed        BASIC     {1,1,1,1,1}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}
          direct         BASIC     {foreground,
                                    foreground,
                                    foreground,
                                    foreground,
                                    foreground}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     }

Text-Bundle-Representation    {
     {Bundle-Index       BASIC     {1,2}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Text-Font-Index    BASIC     {1,1}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Text-Precision
                         BASIC     {string,character}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Character-Expansion-Factor
                         BASIC     {1.0,0.7}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Character-Spacing
                         BASIC     {0.0,0.0}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}
     {Text-Colour   CASE Colour-Specification-Mode
          indexed        BASIC     {1,1}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}
          direct         BASIC     {foreground,
                                    foreground}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     }

Filled-Area-Bundle-Representation  {
     {Fill-Bundle-Index
                         BASIC     {1,2,3,4,5}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Interior-Style
                         BASIC     {hollow,
                                    hatch,
                                    hatch,
                                    hatch,
                                    hatch}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Fill-Colour   CASE Colour-Specification-Mode
          indexed        BASIC     {1,1,1,1,1}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}
          direct         BASIC     {foreground,
                                    foreground,
                                    foreground,
                                    foreground,
                                    foreground}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Hatch-Index        BASIC     {horizontal,
                                    horizontal,
                                    vertical,
                                    positive-slope,
                                    negative-slope}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}
     {Pattern-Index      BASIC     {1,1,1,1,1}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     }

Edge-Bundle-Representation    {
     {Bundle-Index       BASIC     {1,2,3,4,5}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Edge-Type          BASIC     {solid,
                                    dash,
                                    dot,
                                    dash-dot,
                                    dash-dot-dot}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Edge-Width
               CASE Edge-Width-Specification-Mode
          absolute       BASIC     {0.001*max-vdcx,
                                    0.001*max-vdcx,
                                    0.001*max-vdcx,
                                    0.001*max-vdcx,
                                    0.001*max-vdcx}
                         NON-BASIC {N/A}
                         DEFAULT   {NONE}}
          scaled         BASIC     {1.,1.,1.,1.,1.}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Edge-Colour CASE Colour-Specification-Mode
```

```
          indexed        BASIC     {1,1,1,1,1}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}
          direct         BASIC     {foreground,
                                    foreground,
                                    foreground,
                                    foreground,
                                    foreground}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     }

Default-Pattern-Representation     {
     {Pattern-Table-Index
                         BASIC     {1}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Number-Of-Columns  BASIC     {1}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Number-Of-Rows     BASIC     {1}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Local-Colour-Precision
                         BASIC     {0}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Colour-Array       Case Colour-Specification-Mode
          indexed        BASIC     {1}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
          direct         BASIC     {foreground}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     }

Default-Colour-Representation {
     /* See Note 1 */
     {Start-Index        BASIC     {2}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Colour-List        BASIC     {
                    /* Red */       {255,  0,  0},
                    /* Green */     {  0,255,  0},
                    /* Blue */      {  0,  0,255},
                    /* Yellow */    {255,255,  0},
                    /* Magenta*/    {255,  0,255},
                    /* Cyan */      {  0,255,255},
                    /* Black */     {  0,  0,  0},
                    /* White */     {255,255,255}}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
          }

     /*   Note 1    Colour Table defaults for colour indices
                    0 and 1 are defined to the nominal
                    "background" and nominal "foreground"
                    colours, respectively.
     */


Geometric-Graphics-Encoding-Announcer

          VDC-Type            BASIC     {integer|real}
                              NON-BASIC {NONE}
                              DEFAULT   {integer}

          Integer-Precision   BASIC     {16|32}
                              NON-BASIC {NONE}
                              DEFAULT   {16}

          Real-Precision      BASIC     {{0,9,23}|
                                         {1,16,16}}
                              NON-BASIC {NONE}
                              DEFAULT   {{1,16,16}}

          Index-Precision     BASIC     {8|16}
                              NON-BASIC {NONE}
                              DEFAULT   {16}

          Colour-Precision    BASIC     {8|16}
                              NON-BASIC {NONE}
                              DEFAULT   {8}

          Colour-Index-Precision BASIC  {8|16}
                              NON-BASIC {NONE}
                              DEFAULT   {8}

          Maximum-Colour-Index BASIC    {ANY colour-index}
                              NON-BASIC {NONE}
                              DEFAULT   {63}

          Colour-Value-Extent {
               {Minimum-Colour-Direct
                              BASIC     {ANY colour-direct}
                              NON-BASIC {NONE}
                              DEFAULT   {{0,0,0}}}
               {Maximum-Colour-Direct
                              BASIC     {ANY colour-direct}
                              NON-BASIC {NONE}
                              DEFAULT   {{255,255,255}}}
               }

          Colour-Selection-Mode
```

```
                    BASIC     {indexed|direct}
                    NON-BASIC {NONE}
                    DEFAULT   {indexed}

VDC-Integer-Precision
                    BASIC     {16|32}
                    NON-BASIC {NONE}
                    DEFAULT   {16}

VDC-Real-Precision  BASIC     {{0,9,23}|
                               {1,16,16}}
                    NON-BASIC {NONE}
                    DEFAULT   {{1,16,16}}
Line-Rendition

Line-Width-Specification-Mode
                    BASIC     {absolute|scaled}
                    NON-BASIC {NONE}
                    DEFAULT   {scaled}

Line-Bundle-Index   BASIC     {ANY INDEX}
                    NON-BASIC {NONE}
                    DEFAULT   {1}

Line-Type           BASIC     {solid,
                                dash,
                                dot,
                                dash-dot,
                                dash-dot-dot}
                    NON-BASIC {NONE}
                    DEFAULT   {solid}

Line-Width
     CASE Line-Width-Specification-Mode
          absolute  BASIC     {ANY POSITIVE vdc}
                    NON-BASIC {NONE}
                    DEFAULT   {0.001*max-vdcx}
          scaled    BASIC     {ANY POSITIVE real}
                    NON-BASIC {NONE}
                    DEFAULT   {1.0}

Line-Colour
     CASE Colour-Specification-Mode
          indexed   BASIC     {ANY colour-index}
                    NON-BASIC {NONE}
                    DEFAULT   {1}
          direct    BASIC     {ANY colour-direct}
                    NON-BASIC {NONE}
                    DEFAULT   {foreground}

Line-Aspect-Source-Flags      {
     {Line-Type-ASF BASIC     {bundled|
                                    individual}
                         NON-BASIC {NONE}
                         DEFAULT   {individual}}
          {Line-Width-ASF BASIC    {bundled|
                                    individual}
                         NON-BASIC {NONE}
                         DEFAULT   {individual}}
          {Line-Colour-ASF BASIC   {bundled|
                                    individual}
                         NON-BASIC {NONE}
                         DEFAULT   {individual}}

     Line-Bundle-Specifications
     /* Any bundle consisting of parameters as for
     individual */
                         BASIC     {AS_8613}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}

Marker-Rendition

     Marker-Size-Specification-Mode
                         BASIC     {absolute|scaled}
                         NON-BASIC {NONE}
                         DEFAULT   {scaled}

     Marker-Bundle-Index BASIC     {ANY INDEX}
                         NON-BASIC {NONE}
                         DEFAULT   {1}

     Marker-Type         BASIC     {dot|
                                    plus|
                                    asterisk|
                                    circle|
                                    cross}
                         NON-BASIC {NONE}
                         DEFAULT   {asterisk}

     Marker-Size
          CASE Marker-Size-Specification-Mode
               absolute  BASIC     {ANY POSITIVE vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {0.01*max-vdcx}
               scaled    BASIC     {ANY POSITIVE real}
                         NON-BASIC {NONE}
                         DEFAULT   {1.0}

     Marker-Colour
          CASE Colour-Specification-Mode
               indexed   BASIC     {ANY colour-index}
                         NON-BASIC {NONE}
                         DEFAULT   {1}
```

```
                                 direct    BASIC     {ANY colour-direct}
                                           NON-BASIC {NONE}
                                           DEFAULT   {foreground}

                       Marker-Aspect-Source-Flags         {
                            {Marker-Type-ASF BASIC   {bundled|
                                                      individual}
                                           NON-BASIC {NONE}
                                           DEFAULT   {individual}}
                            {Marker-Size-ASF BASIC   {bundled|
                                                      individual}
                                           NON-BASIC {NONE}
                                           DEFAULT   {individual}}
                       {Marker-Colour-ASF BASIC {bundled|
                                                      individual}
                                           NON-BASIC {NONE}
                                           DEFAULT   {individual}}
                            }

                       Marker-Bundle-Specifications
                       /* Any bundle consisting of parameters as for
individual */
                                           BASIC     {AS_8613}
                                           NON-BASIC {NONE}
                                           DEFAULT   {N/A}

                  Text-Rendition

                       Font-List           BASIC     {AS_8613}
                                           NON-BASIC {NONE}
                                           DEFAULT   {NONE}

                       Character-Set-List  {
                            {Character-Set-Type
                                           BASIC     {94-character-G
                                                         -set|
                                                      96-character-G
                                                         -set}
                                           NON-BASIC {NONE}
                                           DEFAULT   {94-character-G
                                                         -set}}
                            {Designation-Sequence-Tail
                                           BASIC     {4/1|
                                                      4/0|
                                                      4/2}
                                           NON-BASIC {NONE}
                                           DEFAULT   {4/1}}
                            }

                       Character-Coding-Announcer
                                           BASIC     {basic-7-bit|
                                                      basic-8-bit}
```

```
                    NON-BASIC {NONE}
                    DEFAULT   {basic-7-bit}

Text-Bundle-Index   BASIC     {ANY index}
                    NON-BASIC {NONE}
                    DEFAULT   {1}

Text-Font-Index     BASIC     {ANY index}
                    NON-BASIC {NONE}
                    DEFAULT   {NONE}

Text-Precision      BASIC     {string|
                               character|
                               stroke}
                    NON-BASIC {NONE}
                    DEFAULT   {string}

Character-Expansion-Factor
                    BASIC     {ANY POSITIVE real}
                    NON-BASIC {NONE}
                    DEFAULT   {1.0}

Character-Spacing   BASIC     {ANY real}
                    NON-BASIC {NONE}
                    DEFAULT   {0.0}

Text-Colour
     CASE Colour-Specification-Mode
          indexed   BASIC     {ANY colour-index}
                    NON-BASIC {NONE}
                    DEFAULT   {1}
          direct    BASIC     {ANY colour-direct}
                    NON-BASIC {NONE}
                    DEFAULT   {foreground}

Character-Height    CASE      {Scaling-Mode}
          absolute  BASIC     {any positive VDC}
                    NON-BASIC {NONE}
                    DEFAULT   {0.01*max-vdcx}
          scaled    BASIC     {any positive
                              (real)}
                    NON-BASIC {NONE}
                    DEFAULT   {1.0}

Character-Orientation {
     {X-Character-Up-Component
                    BASIC     {ANY vdc}
                    NON-BASIC {NONE}
                    DEFAULT   {0}}
     {Y-Character-Up-Component
                    BASIC     {ANY vdc}
                    NON-BASIC {NONE}
```

```
                            DEFAULT   {1}}
          {X-Character-Base-Component
                            BASIC     {ANY vdc}
                            NON-BASIC {NONE}
                            DEFAULT   {1}}
          {Y-Character-Base-Component
                            BASIC     {ANY vdc}
                            NON-BASIC {NONE}
                            DEFAULT   {0}}
          }

      Text-Path             BASIC     {right|
                                       left|
                                        up|
                                        down}
                            NON-BASIC {NONE}
                            DEFAULT   {right}

      Text-Alignment        {
          {Horizontal-Alignment
                            BASIC     {normal|
                                       left|
                                       center|
                                       right|
                                       continuous}
                            NON-BASIC {NONE}
                            DEFAULT   {normal}}
          {Vertical-Alignment
                            BASIC     {normal|
                                       top|
                                       cap|
                                       half|
                                       base|
                                      bottom|
                                      continuous}
                            NON-BASIC {NONE}
                            DEFAULT   {normal}}
          {Continuous-Horizontal-Alignment
                            BASIC     {ANY real}
                            NON-BASIC {NONE}
                            DEFAULT   {N/A}}
          {Continuous-Vertical-Alignment
                            BASIC {ANY real}
                            NON-BASIC {NONE}
                            DEFAULT   {N/A}}
          }

  Character-Set-Index       BASIC     {ANY index}
                            NON-BASIC {NONE}
                            DEFAULT   {1}

      Alternate-Character-Set-Index
```

```
                              BASIC     {ANY index}
                              NON-BASIC {NONE}
                              DEFAULT   {1}

     Text-Aspect-Source-Flags      {
          {Text-Font-ASF BASIC     {bundled|
                                    individual}
                              NON-BASIC {NONE}
                              DEFAULT   {individual}}
          {Text-Precision-ASF
                              BASIC     {bundled|
                                    individual}
                              NON-BASIC {NONE}
                              DEFAULT   {individual}}
          {Character-Expansion-Factor-ASF
                              BASIC     {bundled|
                                         individual}
                              NON-BASIC {NONE}
                              DEFAULT   {individual}}
          {Character-Spacing-ASF
                              BASIC     {bundled|
                                         individual}
                              NON-BASIC {NONE}
                              DEFAULT   {individual}}
          {Text-Colour-ASF    BASIC     {bundled|
                                         individual}
                              NON-BASIC {NONE}
                              DEFAULT   {individual}}
          }

     Text-Bundle-Specifications
     /* Any bundle consisting of parameters as for
     individual */
                              BASIC     {AS_8613}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}

Fill-Area-Rendition

     Fill-Bundle-Index        BASIC     {ANY index}
                              NON-BASIC {NONE}
                              DEFAULT   {1}

     Interior-Style           BASIC     {hollow|
                                         solid|
                                        pattern|
                                        hatch|
                                        empty}
                              NON-BASIC {NONE}
                              DEFAULT   {hollow}

     Fill-Colour
     CASE Colour-Specification-Mode
          indexed        BASIC     {ANY colour-index}
                         NON-BASIC {NONE}
                         DEFAULT   {1}
          direct         BASIC     {ANY colour-
                                   direct}
                         NON-BASIC {NONE}
                         DEFAULT   {foreground}

Hatch-Index              BASIC     {horizontal|
                                    vertical|
                                   positive-slope|
                                   negative-slope|
                                   vertical-hatch|
                                   cross-hatch}
                         NON-BASIC {NONE}
                         DEFAULT   {horizontal}

Pattern-Index            BASIC     {1|2|3|4|5|6|7|8}
                         NON-BASIC {NONE}
                         DEFAULT   {1}

Fill-Reference-Point     BASIC     {ANY point}
                         NON-BASIC {NONE}
                         DEFAULT   {{0,0}}

Pattern-Size             {
     {X-Component-Height-Vector
                         BASIC     {ANY vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {0}}
     {Y-Component-Height-Vector
                         BASIC     {ANY vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {y-vdcx}}
     {X-Component-Width-Vector
                         BASIC     {ANY vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {x-vdcx}}
     {Y-Component-Width-Vector
                         BASIC     {ANY vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {0}}
     }

Pattern-Table-Representation  {
     {Pattern-Table-Index
                         BASIC     {1|2|3|4|
                                    5|6|7|8}
                         NON-BASIC {NONE}
                         DEFAULT   {N/A}}
     {Number-Of-Columns
```

```
                          BASIC     {1..16}
                          NON-BASIC {NONE}
                          DEFAULT   {1}}
     {Number-Of-Rows
                          BASIC     {1..16}
                          NON-BASIC {NONE}
                          DEFAULT   {1}}
     {Local-Colour-Precision
                          BASIC     {0|1|8|16}
                          NON-BASIC {NONE}
                          DEFAULT   {0}}
     {Colour-Array  CASE Colour-Specification-Mode
          indexed         BASIC     {1..8*1..16*1..16*
                                     ANY colour-index}
                          NON-BASIC {NONE}
                          DEFAULT   {1}
          direct          BASIC     {1..8*1..16*1..16*
                                    ANY colour-direct}
                          NON-BASIC {NONE}
                          DEFAULT   {foreground}}
     }

Fill-Aspect-Source-Flags {
     {Interior-Style-ASF
                          BASIC     {bundled|
                                    individual}
                          NON-BASIC {NONE}
                          DEFAULT   {individual}}
     {Fill-Colour-ASF
                          BASIC     {bundled|
                                    individual}
                          NON-BASIC {NONE}
                          DEFAULT   {individual}}
     {Hatch-Index-ASF
                          BASIC     {bundled|
                                    individual}
                          NON-BASIC {NONE}
                          DEFAULT   {individual}}
     {Pattern-Index-ASF
                          BASIC     {bundled|
                                    individual}
                          NON-BASIC {NONE}
                          DEFAULT   {individual}}
     }

Fill-Bundle-Specifications
/* Any bundle consisting of parameters as for
individual */
                          BASIC     {AS_8613}
                          NON-BASIC {NONE}
                          DEFAULT   {N/A}
```

```
Edge-Rendition

     Edge-Width-Specification-Mode
                              BASIC     {absolute|scaled}
                              NON-BASIC {NONE}
                              DEFAULT   {scaled}

     Edge-Bundle-Index        BASIC     {ANY index}
                              NON-BASIC {NONE}
                              DEFAULT   {1}

     Edge-Visibility          BASIC     {off|on}
                              NON-BASIC {NONE}
                              DEFAULT   {off}

     Edge-Type                BASIC     {solid|
                                         dash|
                                         dot|
                                        dash-dot|
                                        dash-dot-dot}
                              NON-BASIC {NONE}
                              DEFAULT   {solid}

     Edge-Width
          CASE Line-Width-Specification-Mode
               absolute       BASIC     {ANY POSITIVE vdc}
                              NON-BASIC {NONE}
                              DEFAULT   {0.001*max-vdcx}
               scaled         BASIC     {ANY POSITIVE real}
                              NON-BASIC {NONE}
                              DEFAULT   {1.0}

     Edge-Colour
          CASE Colour-Specification-Mode
               indexed        BASIC     {ANY colour-index}
                              NON-BASIC {NONE}
                              DEFAULT   {1}
               direct         BASIC     {ANY colour-direct}
                              NON-BASIC {NONE}
                              DEFAULT   {foreground}

     Edge-Aspect-Source-Flags      {
          {Edge-Type-ASF      BASIC     {bundled|
                                         individual}
                              NON-BASIC {NONE}
                              DEFAULT   {individual}}
          {Edge-Width-ASF     BASIC     {bundled|
                                         individual}
                              NON-BASIC {NONE}
                              DEFAULT   {individual}}
          {Edge-Colour-ASF    BASIC     {bundled|
                                        individual}
                              NON-BASIC {NONE}
                              DEFAULT   {individual}}
          }

     Edge-Bundle-Specifications
     /* Any bundle consisting of parameters as for
     individual */
                              BASIC     {AS_8613}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}

Colour-Representation

     Background-Colour        BASIC     {ANY colour-direct}
                              NON-BASIC {NONE}
                              DEFAULT   {background}

     Colour-Table-Specification    {
          {Start-Index   BASIC     {1..
                                 Maximum-Colour-Index}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
          {Colour-List        BASIC     {ANY Colour-List}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
          }

Transparency-Specification

     Transparency             BASIC     {off|on}
                              NON-BASIC {NONE}
                              DEFAULT   {on}

     Auxiliary-Colour    CASE Colour-Specification-Mode
          indexed             BASIC     {ANY colour-index}
                              NON-BASIC {NONE}
                              DEFAULT   {0}
          direct              BASIC     {ANY colour-direct}
                              NON-BASIC {NONE}
                              DEFAULT   {background}

Transformation-Specification

     VDC-Extent          CASE VDC-Type
          integer
               {{X-Coordinate-First-Point
                              BASIC     {ANY vdc}
                              NON-BASIC {NONE}
                              DEFAULT   {0}}
               {Y-Coordinate-First-Point
                              BASIC     {ANY vdc}
                              NON-BASIC {NONE}
```

```
                         DEFAULT   {0}}
          {X-Coordinate-Second-Point
                         BASIC     {ANY vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {32767}}
          {Y-Coordinate-Second-Point
                         BASIC     {ANY vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {32767}}}
     real
          {{X-Coordinate-First-Point
                         BASIC     {ANY vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {0.0}}
          {Y-Coordinate-First-Point
                         BASIC     {ANY vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {0.0}}
          {X-Coordinate-Second-Point
                         BASIC     {ANY vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {0.9999}}
          {Y-Coordinate-Second-Point
                         BASIC     {ANY vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {0.9999}}
     }

Clip-Rectangle      CASE VDC-Type
     integer
          {{X-Coordinate-First-Point
                         BASIC     {ANY vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {0}}
          {Y-Coordinate-First-Point
                         BASIC     {ANY vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {0}}
          {X-Coordinate-Second-Point
                         BASIC     {ANY vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {32767}}
          {Y-Coordinate-Second-Point
                         BASIC     {ANY vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {32767}}}
     real
          {{X-Coordinate-First-Point
                         BASIC     {ANY vdc}
                         NON-BASIC {NONE}
                         DEFAULT   {0.0}}
          {Y-Coordinate-First-Point
```

```
                                        BASIC     {ANY vdc}
                                        NON-BASIC {NONE}
                                        DEFAULT   {0.0}}
                         {X-Coordinate-Second-Point
                                        BASIC     {ANY vdc}
                                        NON-BASIC {NONE}
                                        DEFAULT   {0.9999}}
                         {Y-Coordinate-Second-Point
                                        BASIC     {ANY vdc}
                                        NON-BASIC {NONE}
                                        DEFAULT   {0.9999}}
              }

     Clip-Indicator                     BASIC     {off|on}
                                        NON-BASIC {NONE}
                                        DEFAULT   {on}

Region-Of-Interest                      BASIC     {rectangle|
                                                  automatic}
                                        NON-BASIC {NONE}
                                        DEFAULT   {automatic}

Picture-Orientation                     BASIC     {0|90|180|270}
                                        NON-BASIC {NONE}
                                        DEFAULT   {0}


Picture-Dimensions                      BASIC     {AS_8613}
                                        NON-BASIC {NONE}
                                        DEFAULT   {automatic}
```

Content Architecture Class Attributes

These attributes identify and describe the content architecture class of a content portion specified within object definitions.

Common-Coding-Attributes

```
Type-Of-Coding                          BASIC     {{2 8 3 8 0 }}
                                        NON-BASIC {NONE}
                                        DEFAULT   {NONE}
```

Content-Information

The content information of a content portion description that conforms to this content architecture is an ASN.1 octet string representing a Computer Graphics Metafile (CGM) conforming to the following constraints.

a) Conform to part 1 of the ISO 8632 standard;

b) Conform to the binary encoding defined in part 3 of the ISO 8632 standard;

c) Consist of a single picture;

d) Conform to the CGM implementation agreement specified in section 6.2 of the Technical and Office Protocols Version 3.0 Recommendation, except as noted with respect to font and colour table support;

e) Generalized Drawing Primitives are ignored;

f) ESCAPE Elements are ignored;

g) External Elements may be ignored.

The following list is a description of the permitted values for the CGM elements.

Delimiter-Elements

No-Op	BASIC	{ANY octet-string}
	NON-BASIC	{NONE}
/* See Note 1 */	DEFAULT	{N/A}
Begin-Metafile	BASIC	{ANY string}
	NON-BASIC	{NONE}
/* See Note 2 */	DEFAULT	{null-string}
End-Metafile	BASIC	{N/A}
	NON-BASIC	{NONE}
	DEFAULT	{N/A}
Begin-Picture	BASIC	{ANY string}
	NON-BASIC	{NONE}
/* See Note 2 */	DEFAULT	{null-string}
Begin-Picture-Body	BASIC	{N/A}
	NON-BASIC	{NONE}
	DEFAULT	{N/A}
End-Picture	BASIC	{N/A}
	NON-BASIC	{NONE}
	DEFAULT	{N/A}

/* Note 1: An arbitrary sequence of n octets. Where n=0,1,..,32767. The sequence of zero or more octets is for padding purposes.

```
Note 2:   Support will be provided for strings with a length
          up to 256 octets, except for data records which
          will support strings with a length up to 32767
          octets.
*/

Metafile-Description-Elements

     Metafile-Version              BASIC     {1}
                                   NON-BASIC {NONE}
                                   DEFAULT   {N/A}

Metafile-Description               BASIC     {ANY string}
                                   NON-BASIC {NONE}
/* See Note 1 */                   DEFAULT   {null-string}
/* See Note 2 */

VDC-Type                           BASIC     {integer|real}
                                   NON-BASIC {NONE}
                                   DEFAULT   {integer}

Integer-Precision                  BASIC     {16}
                                   NON-BASIC {NONE}
                                   DEFAULT   {16}

Real-Precision                     BASIC     {{0,9,23}|
                                              {1,16,16}}
                                   NON-BASIC {NONE}
                                   DEFAULT   {{1,16,16}}

Index-Precision                    BASIC     {16}
                                   NON-BASIC {NONE}
                                   DEFAULT   {16}

Colour-Precision                   BASIC     {8|16}
                                   NON-BASIC {NONE}
                                   DEFAULT   {8}

Colour-Index-Precision             BASIC     {8|16}
                                   NON-BASIC {NONE}
                                   DEFAULT   {8}

Maximum-Colour-Index               BASIC     {ANY colour-index}
                                   NON-BASIC {NONE}
/*See Note 3*/                     DEFAULT   {63}

Colour-Value-Extent
     {MINIMUM-colour-direct
                                   BASIC     {ANY colour-direct}
                                   NON-BASIC {NONE}
                                   DEFAULT   {{0,0,0}}
     }
     {MAXIMUM-colour-direct
                              BASIC     {ANY colour-direct}
                              NON-BASIC {NONE}
                              DEFAULT   {(255,255,255)}
     }

Metafile-Element-List
     {Number-Of-Elements      BASIC     {ANY integer}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}
     }
     {List-Of-Elements        BASIC     {AS_8613}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}
     }

Metafile-Defaults-Replacement
                              BASIC     {AS_8613}
                              NON-BASIC {NONE}
/* See Note 4*/               DEFAULT   {N/A}

Font-List                     BASIC     {NONE}
                              NON-BASIC {AS_8613}
/* See Note 3*/               DEFAULT   {NONE}

Character-Set-List  {
     {Character-Set-Type      BASIC     {94-character-G
                                        -set|
                                        96-character-G
                                        -set}
                              NON-BASIC {NONE}
/* See Note 5 */              DEFAULT   {94-character-G
                                        -set})
     {Designation-Sequence-Tail
                              BASIC     {4/1|
                                          4/0|
                                          4/2}
                              NON-BASIC {NONE}
/* See Note 5 */              DEFAULT   {4/1}})

Character-Coding-Announcer    BASIC {basic-7bit|basic-8bit}
                              NON-BASIC {NONE}
                              DEFAULT   {0}
```

/* **Note 1:** Support will be provided for strings with a length up to 256 octets, except for data records which will support strings with a length up to 32767 octets.

Note 2: The METAFILE DESCRIPTION string parameter will be used to include the sub-string

"NIST/BASIC-1" to label the content as conforming to this agreement. In addition, generators of content conforming to this content architecture are encouraged to include a sub-string that identifies the company and product that produced the CGM.

Note 3: The basic, non-basic or default for this element is different than that specified in the TOP Version 3.0 Recommendation for CGM.

Note 4: The METAFILE DEFAULTS REPLACEMENT element shall not be partitioned. No part of the element will be partitioned. Multiple occurrences of the METAFILE DEFAULTS REPLACEMENT element may be used to avoid the need for partitioning. The METAFILE DEFAULTS REPLACEMENT element must appear in the content portion conforming to this content architecture to establish the defaults for TEXT PRECISION and any other elements that do not assume the defaults specified in ISO 8632 parts 1 and 3.

Note 5: The character set ISO 646, 7-bit Coded Character Set for Information Interchange, is specified with the parameters (0,4/1). The character set ISO 6937/2, Coded Character Sets for Text Communication - Latin Alphabetic and Non-alphabetic Graphic Characters, is specified with the parameters (0,4/0).The character set ISO 8859/1, 8-bit Single Byte Coded Graphic Character Sets-Latin Alphabet No. 1, is specified with the parameters (0,4/2).

*/

```
Picture-Descriptor-Elements

Scaling-Mode                   {
     {Scaling-Mode             BASIC     {abstract|metric}
                               NON-BASIC {NONE}
                               DEFAULT   {abstract}}
     {Scale-Factor        CASE Scaling-Mode
          abstract             BASIC     {N/A}
                               NON-BASIC {N/A}
                               DEFAULT   {N/A}
          scaled               BASIC     {ANY real}
                               NON-BASIC {NONE}
```

```
/* See Note 1 */              DEFAULT   {25.4}}
    }

Colour-Selection-Mode         BASIC     {indexed|direct}
                              NON-BASIC {NONE}
                              DEFAULT   {indexed}

Line-Width-Specification-Mode
                              BASIC     {absolute|scaled}
                              NON-BASIC {NONE}
                              DEFAULT   {scaled}


Marker-Size-Specification-Mode
                              BASIC     {absolute|scaled}
                              NON-BASIC {NONE}
                              DEFAULT   {scaled}

Edge-Width-Specification-Mode
                              BASIC     {absolute|scaled}
                              NON-BASIC {NONE}
                              DEFAULT   {scaled}

VDC-Extent               CASE VDC-Type
          integer
               {{X-Coordinate-First-Point
                              BASIC     {ANY vdc}
                              NON-BASIC {NONE}
                              DEFAULT   {0}}
               {Y-Coordinate-First-Point
                              BASIC     {ANY vdc}
                              NON-BASIC {NONE}
                              DEFAULT   {0}}
               {X-Coordinate-Second-Point
                              BASIC     {ANY vdc}
                              NON-BASIC {NONE}
                              DEFAULT   {32767}}
               {Y-Coordinate-Second-Point
                              BASIC     {ANY vdc}
                              NON-BASIC {NONE}
                              DEFAULT   {32767}}}
          real
               {{X-Coordinate-First-Point
                              BASIC     {ANY vdc}
                              NON-BASIC {NONE}
                              DEFAULT   {0.0}}
               {Y-Coordinate-First-Point
                              BASIC     {ANY vdc}
                              NON-BASIC {NONE}
                              DEFAULT   {0.0}}
               {X-Coordinate-Second-Point
                              BASIC     {ANY vdc}
                              NON-BASIC {NONE}
                              DEFAULT   {0.9999}}
               {Y-Coordinate-Second-Point
                              BASIC     {ANY vdc}
                              NON-BASIC {NONE}
                              DEFAULT   {0.9999}}
          }

Background-Colour             BASIC     {ANY colour-direct}
                              NON-BASIC {NONE}
                              DEFAULT   {background}

/*   Note 1:   The Scale-Factor parameter of SCALING MODE
               element is always a 32-bit floating point
               value, even when REAL PRECISION has selected
               fixed-point for other real numbers.  It is
               not apparent in ISO 8632 what the precision
               of this floating point parameter is when
               fixed point reals have been selected.  Its
               precision shall be (0,9,23).
/*

Control-Elements

VDC-Integer-Precision         BASIC     {16|32}
                              NON-BASIC {NONE}
                              DEFAULT   {16}

VDC-Real-Precision            BASIC     {{0 9 23}|
                                         {1 16 16}}
                              NON-BASIC {NONE}
                              DEFAULT   {{1 16 16}}

Auxiliary-Colour    CASE Colour-Specification-Mode
          indexed             BASIC     {ANY colour-index}
                              NON-BASIC {NONE}
                              DEFAULT   {0}
          direct              BASIC     {ANY colour-direct}
                              NON-BASIC {NONE}
                              DEFAULT   {background}

Transparency                  BASIC     {off|on}
                              NON-BASIC {NONE}
                              DEFAULT   {on}

Clip-Rectangle      CASE VDC-Type
          integer
               {{X-Coordinate-First-Point
                              BASIC     {ANY vdc}
                              NON-BASIC {NONE}
                              DEFAULT   {0}}
               {Y-Coordinate-First-Point
```

```
                                   BASIC     {ANY vdc}
                                   NON-BASIC {NONE}
                                   DEFAULT   {0}}
                    {X-Coordinate-Second-Point
                                   BASIC     {ANY vdc}
                                   NON-BASIC {NONE}
                                   DEFAULT   {32767}}
                    {Y-Coordinate-Second-Point
                                   BASIC     {ANY vdc}
                                   NON-BASIC {NONE}
                                   DEFAULT   {32767}}}
                                             real
                    {{X-Coordinate-First-Point
                                   BASIC     {ANY vdc}
                                   NON-BASIC {NONE}
                                   DEFAULT   {0.0}}
                    {Y-Coordinate-First-Point
                                   BASIC     {ANY vdc}
                                   NON-BASIC {NONE}
                                   DEFAULT   {0.0}}
                    {X-Coordinate-Second-Point
                                   BASIC     {ANY vdc}
                                   NON-BASIC {NONE}
                                   DEFAULT   {0.9999}}
                    {Y-Coordinate-Second-Point
                                   BASIC     {ANY vdc}
                                   NON-BASIC {NONE}
                                   DEFAULT   {0.9999}}}

     Clip-Indicator                BASIC     {off|on}
                                   NON-BASIC {NONE}
                                   DEFAULT   {on}

Graphical-Primitive-Elements

     Polyline                      BASIC     {ANY point-list}
                                   NON-BASIC {NONE}
     /* See Note 1 */              DEFAULT   {N/A}

     Disjoint-Polyline             BASIC     {ANY point-list}
                                   NON-BASIC {NONE}
     /* See Note 1 */              DEFAULT   {N/A}

     Polymarker                    BASIC     {ANY point-list}
                                   NON-BASIC {NONE}
     /* See Note 1 */              DEFAULT   {N/A}

     Text                     {
          {Text-Position           BASIC     {ANY point}
                                   NON-BASIC {NONE}
                                   DEFAULT   {N/A}}
          {Final-Flag              BASIC     {final|not-final}
```

```
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Text-String             BASIC     {ANY string}
                              NON-BASIC {NONE}
/* See Note 2 */              DEFAULT   {N/A}}
     }

Restricted-Text               {
/* See Note 3 */
     {Delta-Width             BASIC     {ANY vdc}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Delta-Height            BASIC     {ANY vdc}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Text-Position           BASIC     {ANY point}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Final-Flag              BASIC     {final|not-final}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Text-String             BASIC     {ANY string}
                              NON-BASIC {NONE}
     /* See Note 2 */         DEFAULT   {N/A}}
     }

Append-Text              {
     {Final-Flag              BASIC     {final|not-final}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Text-String             BASIC     {ANY string}
                              NON-BASIC {NONE}
     /* See Note 2 */         DEFAULT   {N/A}}
     }

Polygon                       BASIC     {ANY point-list}
                              NON-BASIC {NONE}
     /* See Note 1 */         DEFAULT   {N/A}

Polygon-Set              {SEQUENCE-OF
     {VERTEX                  BASIC     {ANY point}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {EDGE-OUT-FLAG           BASIC     {invisible|
                                         visible|
                                        close-invisible|
                                        close-visible}}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     }

Cell-Array               {
```

```
/* See Note 4 */
     {Corner-P                BASIC     {ANY point}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Corner-Q                BASIC     {ANY point}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Corner-R                BASIC     {ANY point}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Dimension-X             BASIC     {ANY integer}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Dimension-Y             BASIC     {ANY integer}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Local-Colour-Precision
                              BASIC     {0|1|8|16}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Cell-Representation-Mode
                              BASIC     {rle|packed}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Cell-Colour-Values CASE Colour-Representation-Mode
          indexed             BASIC     {ANY list-colour-
                                        index}
                              NON-BASIC {NONE}
/* See Note 4 */              DEFAULT   {N/A}
          direct              BASIC     {ANY list-colour-
                                        direct}
                              NON-BASIC {NONE}
/* See Note 4 */              DEFAULT   {N/A}}
     }

Rectangle                {
     {First-Corner            BASIC     {ANY point}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Second-Corner           BASIC     {ANY point}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     }

Circle                   {
     {Centre                  BASIC     {ANY point}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Radius                  BASIC     {ANY vdc}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     }
```

```
Circular-Arc-3-Point       {
     {Start-Point                 BASIC     {ANY point}
                                  NON-BASIC {NONE}
                                  DEFAULT   {N/A}}
     {Mid-Point                   BASIC     {ANY point}
                                  NON-BASIC {NONE}
                                  DEFAULT   {N/A}}
     {End-Point                   BASIC     {ANY point}
                                  NON-BASIC {NONE}
                                  DEFAULT   {N/A}}
     }
Circular-Arc-3-Point-Close        {
     {Start-Point                 BASIC     {ANY point}
                                  NON-BASIC {NONE}
                                  DEFAULT   {N/A}}
     {Mid-Point                   BASIC     {ANY point}
                                  NON-BASIC {NONE}
                                  DEFAULT   {N/A}}
     {End-Point                   BASIC     {ANY point}
                                  NON-BASIC {NONE}
                                  DEFAULT   {N/A}}
     {Close-Type-Flag             BASIC     {pie-closure|
                                       chord-closure}
                                  NON-BASIC {NONE}
                                  DEFAULT   {N/A}}
     }

Circular-Arc-Centre        {
     {Centre                      BASIC     {ANY point}
                                  NON-BASIC {NONE}
                                  DEFAULT   {N/A}}
     {Start-Delta-X               BASIC     {ANY vdc}
                                  NON-BASIC {NONE}
                                  DEFAULT   {N/A}}
     {Start-Delta-Y               BASIC     {ANY vdc}
                                  NON-BASIC {NONE}
                                  DEFAULT   {N/A}}
     {End-Delta-X                 BASIC     {ANY vdc}
                                  NON-BASIC {NONE}
                                  DEFAULT   {N/A}}
     {End-Delta-Y                 BASIC     {ANY vdc}
                                  NON-BASIC {NONE}
                                  DEFAULT   {N/A}}
     {Radius                      BASIC     {ANY vdc}
                                  NON-BASIC {NONE}
                                  DEFAULT   {N/A}}
     }

Circular-Arc-Centre-Close         {
     {Centre                      BASIC     {ANY point}
                                  NON-BASIC {NONE}
```

```
                                   DEFAULT   {N/A}}
        {Start-Delta-X             BASIC     {ANY vdc}
                                   NON-BASIC {NONE}
                                   DEFAULT   {N/A}}
        {Start-Delta-Y             BASIC     {ANY vdc}
                                   NON-BASIC {NONE}
                                   DEFAULT   {N/A}}
        {End-Delta-X               BASIC     {ANY vdc}
                                   NON-BASIC {NONE}
                                   DEFAULT   {N/A}}
        {End-Delta-Y               BASIC     {ANY vdc}
                                   NON-BASIC {NONE}
                                   DEFAULT   {N/A}}
        {Radius                    BASIC     {ANY vdc}
                                   NON-BASIC {NONE}
                                   DEFAULT   {N/A}}
        {Close-Type-Flag           BASIC     {pie-closure|
                                                chord-closure}
                                   NON-BASIC {NONE}
                                   DEFAULT   {N/A}}
        }

    Ellipse                   {
        {Centre                    BASIC     {ANY point}
                                   NON-BASIC {NONE}
                                   DEFAULT   {N/A}}
        {First-Conjugate-Diameter-End-Point
                                   BASIC     {ANY point}
                                   NON-BASIC {NONE}
                                   DEFAULT   {N/A}}
        {Second-Conjugate-Diameter-End-Point
                                   BASIC     {ANY point}
                                   NON-BASIC {NONE}
                                   DEFAULT   {N/A}}
        }

    Elliptical-Arc            {
        {Centre                    BASIC     {ANY point}
                                   NON-BASIC {NONE}
                                   DEFAULT   {N/A}}
        {First-Conjugate-Diameter-End-Point
                                   BASIC     {ANY point}
                                   NON-BASIC {NONE}
                                   DEFAULT   {N/A}}
        {Second-Conjugate-Diameter-End-Point
                                   BASIC     {ANY point}
                                   NON-BASIC {NONE}
                                   DEFAULT   {N/A}}
        {Start-Delta-X             BASIC     {ANY vdc}
                                   NON-BASIC {NONE}
                                   DEFAULT   {N/A}}
        {Start-Delta-Y             BASIC     {ANY vdc}
```

```
                                       NON-BASIC {NONE}
                                       DEFAULT   {N/A}}
   {End-Delta-X                        BASIC     {ANY vdc}
                                       NON-BASIC {NONE}
                                       DEFAULT   {N/A}}
   {End-Delta-Y                        BASIC     {ANY vdc}
                                       NON-BASIC {NONE}
                                       DEFAULT   {N/A}}

   }

Elliptical-Arc-Close      {
   {Centre                             BASIC     {ANY point}
                                       NON-BASIC {NONE}
                                       DEFAULT   {N/A}}
   {First-Conjugate-Diameter-End-Point
                                       BASIC     {ANY point}
                                       NON-BASIC {NONE}
                                       DEFAULT   {N/A}}
   {Second-Conjugate-Diameter-End-Point
                                       BASIC     {ANY point}
                                       NON-BASIC {NONE}
                                       DEFAULT   {N/A}}
   {Start-Delta-X                      BASIC     {ANY vdc}
                                       NON-BASIC {NONE}
                                       DEFAULT   {N/A}}
   {Start-Delta-Y                      BASIC     {ANY vdc}
                                       NON-BASIC {NONE}
                                       DEFAULT   {N/A}}
   {End-Delta-X                        BASIC     {ANY vdc}
                                       NON-BASIC {NONE}
                                       DEFAULT   {N/A}}
   {End-Delta-Y                        BASIC     {ANY vdc}
                                       NON-BASIC {NONE}
                                       DEFAULT   {N/A}}
   {Close-Type-Flag                    BASIC     {pie-closure|
                                           chord-closure}
                                       NON-BASIC {NONE}
                                       DEFAULT   {N/A}}
   }
```

/* **Note 1:** The basic value for lists of points that can appear in parameters for metafile elements is 1024.

Note 2: The basic value for length of strings in parameters of metafile elements except data records is 256 octets. For data records the basic value for the length of strings is 32767 octets.

Note 3: The complete restricted text string, including appended text, shall be included in a metafile conforming to this agreement. The complete restricted text string shall be scaled isotopically such that the specified aspect ratio for the text is not distorted and the string fits into the text extent parallelogram.

Note 4: The basic value for the number of colour values that can appear in a colour list parameter for the CELL ARRAY element is 1048576. This supports a 1024 x 1024 image.

*/

```
Attribute-Elements

Line-Bundle-Index               BASIC     {ANY index}
                                NON-BASIC {NONE}
                                DEFAULT   {1}

Line-Type                       BASIC     {solid|
                                           dash|
                                          dot|
                                          dash-dot|
                                          dash-dot-dot}
                                NON-BASIC {NONE}
                                DEFAULT   {solid}

Line-Width
      CASE Line-Width-Specification-Mode
   absolute                     BASIC     {ANY POSITIVE vdc}
                                NON-BASIC {NONE}
                                DEFAULT   {0.001*max-vdcx}
   scaled                       BASIC     {ANY POSITIVE real}
                                NON-BASIC {NONE}
                                DEFAULT   {1.0}

Line-Colour
      CASE Colour-Specification-Mode
   indexed                      BASIC     {ANY colour-index}
                                NON-BASIC {NONE}
                                DEFAULT   {1}
   direct                       BASIC     {ANY colour-direct}
                                NON-BASIC {NONE}
                                DEFAULT   {foreground}

Marker-Bundle-Index             BASIC     {ANY index}
                                NON-BASIC {NONE}
                                DEFAULT   {1}

Marker-Type                     BASIC     {dot|
                                           plus|
                                       asterisk|
                                       circle|
                                       cross}
                             NON-BASIC {NONE}
                             DEFAULT   {asterisk}

Marker-Size
     CASE Marker-Size-Specification-Mode
          absolute           BASIC     {ANY POSITIVE vdc}
                             NON-BASIC {NONE}
                             DEFAULT   {0.01*max-vdcx}
          scaled             BASIC     {ANY POSITIVE real}
                             NON-BASIC {NONE}
                             DEFAULT   {1.0}

Marker-Colour
     CASE Colour-Specification-Mode
          indexed            BASIC     {ANY colour-index}
                             NON-BASIC {NONE}
                             DEFAULT   {1}
          direct             BASIC     {ANY colour-direct}
                             NON-BASIC {NONE}
                             DEFAULT   {foreground}

Text-Bundle-Index            BASIC     {ANY index}
                             NON-BASIC {NONE}
                             DEFAULT   {1}

Text-Font-Index              BASIC     {ANY index}
                             NON-BASIC {NONE}
                             DEFAULT   {1}

Text-Precision               BASIC     {string|
                                character|
                               stroke}
                             NON-BASIC {NONE}
                             DEFAULT   {stroke}

Character-Expansion-Factor
                             BASIC     {ANY POSITIVE real}
                             NON-BASIC {NONE}
                             DEFAULT   {1.0}

Character-Spacing            BASIC     {ANY real}
                             NON-BASIC {NONE}
                             DEFAULT   {0.0}

Text-Colour
     CASE Colour-Specification-Mode
          indexed            BASIC     {ANY colour-index}
                             NON-BASIC {NONE}
                             DEFAULT   {1}
          direct               BASIC     {ANY colour-direct}
                               NON-BASIC {NONE}
                               DEFAULT   {foreground}

Character-Height         CASE      Scaling-Mode
          absolute             BASIC     {ANY POSITIVE vdc}
                               NON-BASIC {NONE}
                               DEFAULT   {0.01*max-vdcx}
          scaled               BASIC     {ANY POSITIVE real}
                               NON-BASIC {NONE}
                               DEFAULT   {1.0}
Character-Orientation {
     {X-Character-Up-Component
                               BASIC     {ANY vdc}
                               NON-BASIC {NONE}
                               DEFAULT   {0}}
     {Y-Character-Up-Component
                               BASIC     {ANY vdc}
                               NON-BASIC {NONE}
                               DEFAULT   {1}}
     {X-Character-Base-Component
                               BASIC     {ANY vdc}
                               NON-BASIC {NONE}
                               DEFAULT   {1}}
     {Y-Character-Base-Component
                               BASIC     {ANY vdc}
                               NON-BASIC {NONE}
                               DEFAULT   {0}}
     }

Text-Path                      BASIC     {right|
                                          left|
                                         up|
                                         down}
                               NON-BASIC {NONE}
                               DEFAULT   {right}


Text-Alignment           {
     {Horizontal-Alignment     BASIC     {normal|
                                          left|
                                         center|
                                         right|
                                         continuous}
                               NON-BASIC {NONE}
                               DEFAULT   {normal}}
     {Vertical-Alignment       BASIC     {normal|
                                          top|
                                         cap|
                                         half|
                                         base|
                                         bottom|
                                         continuous}
```

```
                              NON-BASIC {NONE}
                              DEFAULT   {normal}}
     {Continuous-Horizontal-Alignment
                              BASIC     {ANY real}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Continuous-Vertical-Alignment
                              basic {ANY real}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}}

Character-Set-Index           BASIC     {ANY index}
                              NON-BASIC {NONE}
                              DEFAULT   {1}

Alternate-Character-Set-Index
                              BASIC     {ANY index}
                              NON-BASIC {NONE}
                              DEFAULT   {1}

Fill-Bundle-Index             BASIC     {ANY index}
                              NON-BASIC {NONE}
                              DEFAULT   {1}

Interior-Style                BASIC     {hollow|
                                         solid|
                                        pattern|
                                        hatch|
                                        empty}
                              NON-BASIC {NONE}
                              DEFAULT   {hollow}

Fill-Colour
     CASE Colour-Specification-Mode
          indexed             BASIC     {ANY colour-index}
                              NON-BASIC {NONE}
                              DEFAULT   {1}
          direct              BASIC     {ANY colour-direct}
                              NON-BASIC {NONE}
                              DEFAULT   {foreground}

Hatch-Index                   BASIC     {horizontal|
                                         vertical|
                                        positive-slope|
                                        negative-slope|
                                        vertical-hatch|
                                        cross-hatch}
                              NON-BASIC {NONE}
                              DEFAULT   {horizontal}

Pattern-Index                 BASIC     {1..8}
                              NON-BASIC {NONE}
                              DEFAULT   {1}

Edge-Bundle-Index             BASIC     {ANY index}
                              NON-BASIC {NONE}
                              DEFAULT   {1}

Edge-Type                     BASIC     {solid|
                                         dash|
                                        dot|
                                        dash-dot|
                                        dash-dot-dot}
                              NON-BASIC {NONE}
                              DEFAULT   {solid}
Edge-Width
     CASE Line-Width-Specification-Mode
          absolute            BASIC     {ANY POSITIVE vdc}
                              NON-BASIC {NONE}
                              DEFAULT   {0.001*max-vdcx}
          scaled              BASIC     {ANY POSITIVE real}
                              NON-BASIC.{NONE}
                              DEFAULT   {1.0}

Edge-Colour
     CASE Colour-Specification-Mode
          indexed             BASIC     {ANY colour-index}
                              NON-BASIC {NONE}
                              DEFAULT   {1}
          direct              BASIC     {ANY colour-direct}
                              NON-BASIC {NONE}
                              DEFAULT   {foreground}

Edge-Visibility               BASIC     {off|on}
                              NON-BASIC {NONE}
                              DEFAULT   {off}

Fill-Reference-Point          BASIC     {ANY point}
                              NON-BASIC {NONE}
                              DEFAULT   {{0,0}}

Pattern-Table            {
     /* See Note 1 */
     {Pattern-Table-Index
                              BASIC     {1..8}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Number-Of-Columns       BASIC     {1..16}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Number-Of-Rows          BASIC     {1..16}
                              NON-BASIC {NONE}
                              DEFAULT   {N/A}}
     {Local-Colour-Precision  BASIC     {0|1|8|16}
```

```
                                      NON-BASIC {NONE}
                                      DEFAULT   {N/A}}
     {Colour-Array  CASE Colour-Specification-Mode
          indexed                     BASIC     {1..2048*
     /* See Note 2 */                       ANY colour-index}
                                      NON-BASIC {NONE}
                                      DEFAULT   {N/A}}
          direct                      BASIC     {1..2048*
/* See Note 2 */                       ANY colour-direct}
                                      NON-BASIC {NONE}
                                      DEFAULT   {N/A}}
     }
Pattern-Size          {
     {X-Component-Height-Vector
                                      BASIC     {ANY vdc}
                                      NON-BASIC {NONE}
                                      DEFAULT   {0}}
     {Y-Component-Height-Vector
                                      BASIC     {ANY vdc}
                                      NON-BASIC {NONE}
                                      DEFAULT   {y-vdcx}}
     {X-Component-Width-Vector
                                      BASIC     {ANY vdc}
                                      NON-BASIC {NONE}
                                      DEFAULT   {x-vdcx}}
     {Y-Component-Width-Vector
                                      BASIC     {ANY vdc}
                                      NON-BASIC {NONE}
                                      DEFAULT   {0}}
     }

     Colour-Table-Specification    {
          /* See Note 3 */
          {Start-Index                BASIC     {1..
                                           Maximum-Colour-Index}
                                      NON-BASIC {NONE}
                                      DEFAULT   {2}}
     {Colour-List                     BASIC     {ANY colour-list}
          /* See Note 4 */
                                      NON-BASIC {NONE}
                                      DEFAULT   {
                    /* Red */         {255,  0,  0},
                    /* Green */       {  0,255,  0},
                    /* Blue */        {  0,  0,255},
                    /* Yellow */      {255,255,  0},
                    /* Magenta*/      {255,  0,255},
                    /* Cyan */        {  0,255,255},
                    /* Black */       {  0,  0,  0},
                    /* White */       {255,255,255}}
          }

Aspect-Source-Flags        {SEQUENCE-OF
```

```
{ASF-Type                BASIC     {Line-ASF|
                                    Line-Width-ASF|
                                   Line-Colour-ASF|
                                   Marker-Type-ASF|
                                   Marker-Size-ASF|
                                   Marker-Colour-ASF|
                                   Text-Font-Index-
                                        ASF|
                                   Text-Precision-
                                        ASF|
                                   Character-
                                        Expansion-
                                        Factor-ASF|
                                   Character-
                                        Spacing-ASF|
                                   Text-Colour-ASF|
                                   Interior-Style-
                                        ASF|
                                   Fill-Colour-ASF|
                                   Hatch-Index-ASF|
                                   Pattern-Index-ASF|
                                   Edge-Type-ASF|
                                   Edge-Width-ASF|
                                   Edge-Colour-ASF}
                         NON-BASIC {NONE}
                         DEFAULT   {Line-ASF,
                                    Line-Width-ASF,
                                   Line-Colour-ASF,
                                   Marker-Type-ASF,
                                   Marker-Size-ASF,
                                   Marker-Colour-ASF,
                                   Text-Font-Index-
                                        ASF,
                                   Text-Precision-
                                        ASF,
                                   Character-
                                        Expansion-
                                        Factor-ASF,
                                   Character-
                                        Spacing-ASF,
                                   Text-Colour-ASF,
                                   Interior-Style-
                                        ASF,
                                   Fill-Colour-ASF,
                                   Hatch-Index-ASF,
                                   Pattern-Index-ASF,
                                   Edge-Type-ASF,
                                   Edge-Width-ASF,
                                   Edge-Colour-ASF}}
     {ASF-Value          BASIC     {individual|
                                    bundled}
                         NON-BASIC {NONE}
```

```
                                   DEFAULT   {individual}

    /*  Note 1:   The PATTERN TABLE element has an unspecified
                  effect when it appears in a picture
                  subsequent to any graphical primitives. The
                  PATTERN TABLE element shall appear prior to
                  any graphical primitive elements to insure
                  that interpreting systems without dynamic
                  pattern update can render the intended
                  effect.


        Note 2:   The basic value for the number of colour
                  values in a colour array parameter for the
                  PATTERN TABLE element is 2048. This will
                  support 8 patterns of 16 x 16.

        Note 3:   The COLOUR TABLE element has an unspecified
                  effect when it appears in a picture
                  subsequent to any graphical primitives. The
                  COLOUR TABLE element shall appear prior to
                  any graphical primitive elements to insure
                  that interpreting systems without dynamic
                  colour update can render the intended effect.

        Note 4:   The basic value for the number of colour
                  values in a colour list parameter for the
                  COLOUR TABLE element is 61. This will
                  support 63 entry colour table.
*/

External-Elements

    Message                   {
        {Action-Required-Flag
                                   BASIC     {no-action}
                                   NON-BASIC {action}
                                   DEFAULT   {N/A}}
        {Message-String            BASIC     {ANY string}
                                   NON-BASIC {NONE}
    /* See Note 1 */               DEFAULT   {N/A}}
        }

    Application-Data               BASIC     {ANY data-record}
                                   NON-BASIC {NONE}
    /* See Note 1 */               DEFAULT   {N/A}

/*      Note 1:   The basic value for string parameters in
                  metafile elements is 256 octets, except for
                  data records which will support strings with
                  a length up to 32767 octets.

/*
```

16.2.6 DOCUMENT PROFILE

Presence of Document Constituents

PERMITTED

Generic-Layout-Structure	{AS_8613}
Specific-Layout-Structure	{AS_8613}
Generic-Logical-Structure	{AS_8613}
Specific-Logical-Structure	{AS_8613}
Layout-Styles	{AS_8613}
Presentation-Styles	{AS_8613}
External-Document-Class	{AS_8613}
Resource-Document	{AS_8613}
Resources	{AS_8613}

Document Characteristics

REQUIRED

Document-Application-Profile	/* ASN.1 object identifier to be supplied */

```
Document-Application-Profile-Defaults
     {Dimensions {#horizontal{9240}
                  #vertical{12400}}
      Medium-Type {10200,13200,'unspecified'}
      Graphic-Character-Subrepertoire {8}
      Raster-Graphics-Type-of-Coding {2 8 3 7 3}}
```

Document-Architecture-Class	{AS_8613}
Content-Architecture-Class	{AS_8613}
Interchange-Format-Class	{A}
ODA-Version	/* 1988 */

Non-basic Document Characteristics

PERMITTED

Profile-Character-Sets	/* ISO 6937/2 or ISO 8859/1 */
Comments-Character-Sets	/* ISO 6937/2 or ISO 8859/1 */
Alternative-Representation-Character-Sets	/* ISO 6937/2 or ISO 8859/1 */
Presentation-Features	{AS_8613}

Additional Document Characteristics

PERMITTED

Fonts-List	{AS_8613}
Unit-Scaling	{AS_8613}

Document Management Attributes

REQUIRED

Document-Reference	{AS_8613}

PERMITTED

Title	{AS_8613}
Subject	{AS_8613}
Document-Type	{AS_8613}
Abstract	{AS_8613}
Document-Date-and-Time	{AS_8613}
Creation-Date-and-Time	{AS_8613}
Local-Filing-Date-and-Time	{AS_8613}
Expiry-Date-and-Time	{AS_8613}
Start-Date-and-Time	{AS_8613}
Purge-Date-and-Time	{AS_8613}
Release-Date-and-Time	{AS_8613}
Revision-History	{AS_8613}
Organizations	{AS_8613}
Preparers	{AS_8613}
Owners	{AS_8613}
Authors	{AS_8613}
Copyright	{AS_8613}
Status	{AS_8613}
User-Specific-Codes	{AS_8613}
Distribution-List	{AS_8613}
Additional-Information	{AS_8613}
References-to-Other-Documents	{AS_8613}
Superseded-Documents	{AS_8613}
Keywords	{AS_8613}
Local-File-Reference	{AS_8613}
Document-Size	{AS_8613}
Number-of-Pages	{AS_8613}
Languages	{AS_8613}
Authorization	{AS_8613}
Security-Classification	{AS_8613}
Access-Rights	{AS_8613}

16.2.7 DOCUMENT INTERCHANGE FORMAT

The aspects of this Implementation Agreement that are concerned with the Format of the Interchange of documents are defined in this clause. These aspects include the data stream, the interchange data units, and ASN.1 encodings.

Data Stream

The data stream is in accordance with the Office Document Interchange Format Class A, as defined in ISO 8613-5.

The encoding is in accordance with the Basic Encoding Rules for Abstract Syntax Notation One (ASN.1), as defined in ISO 8825.

ASN.1 Generation and Parsing

This clause covers two distinct aspects of ASN.1 generation and parsing. The first aspect covers ASN.1 practices that are mandatory for an implementation to be conforming to this Implementors Agreement. The second aspect covers ASN.1 practices that are recommended by this Implementors Agreement. These recommended practices are not mandatory for conformance, but are recommended solely in the spirit of improving interoperability among different implementations. As such, it is left to the discretion of the implementor to either follow the recommended practices or to ignore them. By following the recommended practices, an implementation can reduce the possibility of failure due to interoperability problems and minimize the ASN.1 representation.

ASN.1 Generation Requirements

There are no additional requirements, beyond ISO 8824 and ISO 8825, imposed on the ASN.1 generation.

ASN.1 Parsing Requirements

There are no additional requirements, beyond ISO 8824 and ISO 8825, imposed on the ASN.1 parsing.

ASN.1 Generation Recommendations

The focus of the ASN.1 generation recommendations is to generate ASN.1 encodings that will allow parsing by the most rudimentary of implementations. These recommendations are described in the following sub-clauses.

Segmenting Strings

ISO 8825 allows Bit Strings, Octet Strings, and Character Set Strings to be encoded in the Primitive form or in the Constructed form. The choice of which form to use is an option of the encoder. Using the constructed form allows a string to be segmented into a sequence of strings. This sequence of strings is then contained in the constructed form of the string. The constructed form is allowed the use of the indefinite form on content length.

This Implementors Agreement recommends that implementations limit the encoding to one level of the constructed form for Bit Strings, Octet Strings, and Character Set Strings.

For example, if of type OCTET STRING, the value '432E436F6D6273'H can be encoded in the primitive form as:

Octet String	Length	Contents
04_{16}	07_{16}	$432E436F6D6273_{16}$

The same value may be encoded in the constructed from as:

Octet String	Length	Contents
24_{16}	80_{16}	

Octet String	Length	Contents
04_{16}	02_{16}	$432E_{16}$
04_{16}	05_{16}	$436F6D6273_{16}$
EOC	Length	
00_{16}	00_{16}	

The same value encoded using two levels of constructed form is not recommended by this Implementors Agreement. An example of an encoding containing two levels of construction is:

Octet String	Length	Contents
24_{16}	80_{16}	

Octet String	Length	Contents
24_{16}	04_{16}	

Octet String	Length	Contents
04_{16}	02_{16}	$432E_{16}$
04_{16}	05_{16}	$436F6D6273_{16}$
EOC	Length	
00_{16}	00_{16}	

Length Expression

ISO 8825 allows the content length of an encoding that could be expressed using the short form to also be expressed using the long form. For example, a length of one could be expressed in the short form as 00000001_2 or in the long form as 10000001_2 00000001_2. CCITT Recommendation X.409 (1984) does not allow the same liberty in expressing the length of the encoding length. Implementations using these X.409 rules could present interoperability constraints.

This Implementors Agreement recommends that implementations generate content lengths only in their most economical form.

Ordering of Set Members

ISO 8824 defines sets to be unordered lists of values. It is the generator's option to select an order for the values of the set. Since this ordering is unpredictable from one implementation to the next, it is recommended that generators order the values in a set according to the order in which the members appear in the definition of the set. The intent of this recommendation is to reduce the possible interoperability problems associated with the unpredictable ordering of members in a set.

ASN.1 Parsing Recommendations

The overall intent of these parsing recommendations is to allow a high tolerance in the representation of the ASN.1 syntax without jeopardizing the semantics of the information being conveyed. Each of these tolerances is described in a following sub-clause.

Segmented Strings

The ASN.1 generation restriction on segmenting strings is a recommendation of this Implementors Agreement and is not a requirement of ISO 8825. Therefore, it is recommended that implementations accept string encodings which have been segmented into more than one level of the constructed form.

Encoding of Application Comments

ISO 8613/5 defines the encoding of the attribute Application Comments as an octet string. This Implementation Agreement requires that the encoding within that octet string be in accordance with the ASN.1 syntax specified in the module definition below.

```
     NISTDAPSpecification

     DEFINITION                              ::= BEGIN

     EXPORTS Application-Comments-Encoding;
```

```
Application-Comments-Encoding       ::= SEQUENCE   {
     Constraint-name                [0] IMPLICIT
                                         PrintableString}
                                    END
```

16.2.8 Relationship to Other DAPS

EWOS

There are three Document Application Profiles (DAPs) being defined by the European Workshop on Open Systems (EWOS) ODA Expert Group. These are called Q/111, Q/112, and Q/113.

Q/113 is expected to be equivalent to the NIST DAP.

CCITT

There are three DAPs defined by CCITT:

* T.501 - Document Application Profile MM for the Interchange of Formatted Mixed Mode Documents,

* T.502 - Document Application Profile PM1 for the Interchange of Processable Form Documents, and

* T.503 - Document Application Profile for the Interchange of Group 4 Facsimile Documents.

It is intended that the NIST DAP will be compatible with the CCITT T.502 DAP.

TOP

The NIST DAP will be presented to the Technical and Office Protocol (TOP) Group as a suggested replacement for the TOP Version 3.0 ODA Application Profile.

The content information of a Geometric Content Architecture content portion description defined according to this NIST DAP conforms to the CGM section of the TOP Version 3.0 Recommendation, without any ESCAPE elements and Grouped Drawing Primitives.

INTAP

The Interoperability Technology Association for Information Processing (INTAP) Experts Group on ODA/ODIF is developing two sets of profiles, AE.111n-J and AE.112n-J of small and medium complexity, respectively. The AE.111n-J set includes the AE.1111-J, AE.1114-J, AE.1115-J, and AE.1116-J profiles. The AE.112n-J set includes the AE.1121-J, AE.1124-J, AE.1125-J, and AE.1126-J profiles. AE.1126-J is expected to be a functional equivalent to the NIST DAP.

17. FUTURE OFFICE DOCUMENT ARCHITECTURE

Editor's Note: This section is a placeholder for future stable ODA (Office Document Architecture) implementation agreements. Consult the aligned section of the Ongoing Document for the latest status of this work. As this work becomes stable, it will be moved into this section.

18. FUTURE NETWORK MANAGEMENT

Editor's Note: This section is a placeholder for future stable Network Management Agreements. Consult the aligned section of the Ongoing Agreements Document for the latest text on this subject. As this material is declared stable, it will be moved into this section.

19. REFERENCES

Selected references are grouped by organization publishing the documents and by Reference Model layer to aid relating standards to the OSI Basic Reference Model and to aid relating equivalent standards published by different standards organizations.

19.1 CCITT

Network Layer

CCITT Recommendation X.25 - 1980, Interface Between Data Terminal Equipment (DTE) and Data Circuit-Terminating Equipment (DCE) for Terminals Operating in the Packet Mode on Public Data Networks.

CCITT Recommendation X.25 - 1984, Interface Between Data Terminal Equipment (DTE) and Data Circuit-Terminating Equipment (DCE) for Terminals Operating in the Packet Mode on Public Data Networks.

Transport Layer

CCITT Recommendation X.214, (Red Book, 1984), Transport Service Definition for Open Systems Interconnection for CCITT Applications.

CCITT Recommendation X.224, (Red Book, 1984), Transport Protocol Profile for Open Systems Interconnection for CCITT Applications.

Session Layer

CCITT Recommendation X.215, (Red Book, 1984), Session Service Definition for Open Systems Interconnection for CCITT Applications.

CCITT Recommendation X.225, (Red Book, 1984), Session Protocol Profile for Open Systems Interconnection for CCITT Applications.

Presentation

CCITT Recommendation T.50, (Red Book, 1984), International Alphabet No. 5.

Application Layer -- MHS

CCITT Recommendation X.400, (Red Book, 1984), Message Handling Systems: System Model-Service Elements.

CCITT Recommendation X.401, (Red Book, 1984), Message Handling Systems: Basic Service Elements and Optional User Facilities.CCITT Recommendation X.408, (Red Book, 1984), Message Handling Systems: Encoded Information Type Conversion Rules.

CCITT Recommendation X.409, (Red Book, 1984), Message Handling Systems: Presentation Transfer Syntax and Notation.

CCITT Recommendation X.410, (Red Book, 1984), Message Handling Systems: Remote Operations and Reliable Transfer Server.

CCITT Recommendation X.411, (Red Book, 1984), Message Handling Systems: Message Transfer Layer.

CCITT Recommendation X.420, (Red Book, 1984), Message Handling Systems: Interpersonal Messaging User Agent Layer.

CCITT Recommendation X.430, (Red Book, 1984), Message Handling Systems: Access Protocol for Teletex Terminals.

Application Layer - ODA

CCITT Recommendations

T.6 - Facsimile Coding Schemes and Coding Control Functions for Group 4 Facsimile Apparatus 1984

T.4 - Standardization of Group 3 Facsimile Apparatus for Document Transmission 1984

T.411 - Open Document Architecture (ODA) and Interchange Format - Introduction and General Principles 1988

T.412 - Open Document Architecture (ODA) and Interchange Format - Document Structures 1988

T.414 - Open Document Architecture (ODA) and Interchange Format - Document Profile 1988

T.415 - Open Document Architecture (ODA) and Interchange Format - Document Interchange Format (ODIF) 1988

T.416 - Open Document Architecture (ODA) and Interchange Format - Character Content Architectures 1988

T.417 - Open Document Architecture (ODA) and Interchange Format - Raster Graphics Content Architectures 1988

T.418 - Open Document Architecture (ODA) and Interchange Format - Geometric Graphics Content Architectures 1988

T.501 - Document Application Profile MM for the Interchange of Formatted Mixed Mode Documents 1988

T.502 - Document Application Profile PM1 for the Interchange of Processable Form Documents 1988

T.503 - Document Application Profile for the Interchange of Group 4 Facsimile Documents 1988

CCITT documents may be obtained from:

International Telecommunications Union
Place des Nations, CH 1211,
Geneva 20 SWITZERLAND

19.2 ISO

Status of ISO work can be determined by the reference number; working drafts are referenced by committee and number; e.g., TC 97/SC 6 Nxxxx.

Standards are cited by either ISO xxxx or IS xxxx; DIS and DPs are cited in similar form. Note: ISO TC 97 is now called ISO/IEC JTC1.

Information Processing Systems - Open Systems Interconnection - Basic Reference Model. ISO/IS 7498. First Edition - Oct. 15, 1984. Ref. No. ISO 7498-1984(E).

OSI Basic Reference Model - Part 2: Security Architecture. ISO/DIS 7498-2 TC 97/SC 21/N1895. Project 97.21.18. May 1987.

OSI Basic Reference Model - Part 3: Naming and Addressing. ISO/DIS 7498-3, ISO/TC 97/ SC 21 N2141. May, 1987.

Data Interchange - Structure for the identification of organizations. ISO 6523. 1984-02-01.

Physical Layer

Information Processing Systems - Local Area Networks - Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) and Physical Layer Specification ISO/DIS 8802/3

Information Processing Systems - Local Area Networks - Part 4: Token-Passing Bus Access Method and Physical Layer Specification, ISO/DIS 8802/4

ISO 8802-5 Final Text of ISO/DIS 8802-5: Info proc systems - Local Area Nets -Part 5: Token ring access method and physical layer specification, ISO/TC 97/SC 6 N4477,1987

Data Link Layer

ISO 8802-2 Text for ISO/DIS 8802-2.2, Logical Link Control, ISO/TC 97/SC 6 N4609, 1987

Information Processing Systems - Data Communications - High-Level Data Link Control Procedures - Description of the X.25 LAPB-compatible DTE Data Link Procedures, IS 7776.

Network Layer

Information Processing Systems - Open Systems Interconnection - Network Service Definition, IS 8348.

Information Processing Systems - Open Systems Interconnection - Addendum to the Network Service Definition Covering Connectionless Data Transmission, IS 8348/AD1.

Information Processing Systems - Open Systems Interconnection - Addendum to the Network Service Definition Covering Network Layer Addressing, IS 8348/AD2.

Information Processing Systems - Open Systems Interconnection - Internal Organization of the Network Layer, DIS 8648.

Information Processing Systems - Open Systems Interconnection - Protocol for Providing the Connectionless Network Service, IS 8473.

Information Processing Systems - Open Systems Interconnection -Addendumto IS 8473 - Provision of the Underlying Service Assumed by ISO 8473, ISO TC 97/SC 6 N 3453.

Information Processing Systems - Open Systems Interconnection, Working Draft, End System to Intermediate System Routing Exchange Protocol for use in Conjunction with ISO 8473 ISO/DIS 9542 TC 97/SC 6 N 4053.

Information Processing Systems - Open Systems Interconnection - Data Communications - X.25 Packet Level Protocol for Data Terminal equipment, IS 8208.

Information Processing Systems - Open Systems Interconnection - Data Communications - Use of X.25 to provide the OSI Connection-mode Network Service DIS 8878

Transport Layer

Information Processing Systems - Open Systems Interconnection - Transport Service Definition, IS 8072.

Information Processing Systems - Open Systems Interconnection - Transport Protocol Specification, IS 8072, 1984.

Session Layer

Information Processing Systems - Open Systems Interconnection - Basic Connection Oriented Session Service Definition, ISO 8326: 1987 (E).

Information Processing Systems - Open Systems Interconnection - Basic Connection Oriented Session Protocol Specification, ISO 8327 : 1987 (E).

Information Processing Systems - Open Systems Interconnection - Basic Connection Oriented Session Service Definition-AD 2 to ISO 8326 to Incorporate Unlimited User Data, ISO/IEC JTC1/SC21 N2494.

Information Processing Systems - Open Systems Interconnection - Basic Connection Oriented Session Protocol Specification - AD 2 to ISO 8327 to Incorporate Unlimited User Data, ISO/IEC JTC1/SC21 N2495.

Presentation Layer

Information Processing Systems - Open Systems Interconnection - Connection-Oriented Presentation Service Definition, ISO 8822: 1988 (ISO/IEC JTC1/SC21 N2335).

Information Processing Systems - Open Systems Interconnection - Connection Oriented Presentation Protocol Specification, ISO 8823: 1988 (ISO/IEC JTC1/SC21 N2336).

Interim Revised Text of ISO 8825/PDAD1, ASN.1 Extensions (ISO/IEC JTC1/SC21 N2342).

Interim Revised Text of ISO 8824/PDAD1, ASN.1 Extensions (ISO/IEC JTC1/SC21 N2341).

Information Processing Systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1), ISO 8824: 1987 (E).

Information Processing Systems - Open Systems Interconnection - Specification of Basic Encoding Rules for Abstract Syntax Notation (ASN.1), ISO 8825: 1987 (E).

7-bit Coded Character Set for Information Processing Interchange, ISO 646.

Information Interchange - Representation of Local Time Differentials, ISO 3307.

Application Layer

Information Processing Systems - Open Systems Interconnection - Application Layer Structure, ISO/DP 9545, ISO/TC97/SC21/N1743. July 24, 1987. Revised November 1987.

Application Layer -- FTAM

Information Processing Systems - Open Systems Interconnection - File Transfer, Access and Management Part I: General Introduction, ISO 8571-1: 1987 (E) (ISO/TC97/SC21 N 2331).

Information Processing Systems - Open Systems Interconnection - File Transfer, Access and Management Part 2: Virtual Filestore Definition, ISO 8571-2: 1987 (E) (ISO/TC97/SC21 N 2332).

Information Processing Systems - Open Systems Interconnection - File Transfer, Access and Management Part 3: The File Service Definition, ISO 8571-3: 1987 (E) (ISO/TC97/SC21 N 2333).

Information Processing Systems - Open Systems Interconnection - File Transfer, Access and Management Part 4: File Protocol Specification, ISO 8571-4: 1987 (E) (ISO/TC97/SC21 N 2334).

Application Layer -- ASE/ACSE

Information Processing Systems - Open Systems Interconnection - Service Definition for the Association Control Service Element, ISO 8649: 1987 (E) (ISO/IEC JTC1/SC21 N2326)

Information Processing Systems - Open Systems Interconnection - Protocol Specification for the Association Control Service Element, ISO 8650: 1987 (E) (ISO/IEC JTC1/SC21 N2327)

Application Layer -- VTP

Information Processing Systems - Open Systems Interconnection - Virtual Terminal Service - Basic Class, IS 9040.

Information Processing Systems - Open Systems Interconnection - Virtual Terminal Protocol - Basic Class, IS 9041.

Application Process -- Office Document Interchange -- ODA

ISO 8613/1 - Information processing : Text and Office Systems; Office Document Architecture (ODA) and Interchange Format Part 1: Introduction and General Principles

ISO 8613/2 - Information processing : Text and Office Systems; Office Document Architecture (ODA) and Interchange Format Part 2: Document Structures

ISO 8613/4 - Information processing : Text and Office Systems; Office Document Architecture (ODA) and Interchange Format Part 4: Document Profile

ISO 8613/5 - Information processing : Text and Office Systems; Office Document Architecture (ODA) and Interchange Format Part 5: Office Document Interchange Format

ISO 8613/6 - Information processing : Text and Office Systems; Office Document Architecture (ODA) and Interchange Format Part 6: Character Content Architectures

ISO 8613/7 - Information processing : Text and Office Systems; Office Document Architecture (ODA) and Interchange Format Part 7: Raster Graphics Content Architectures

ISO 8613/8 - Information processing : Text and Office Systems; Office Document Architecture (ODA) and Interchange Format Part 8: Geometric Graphics Content Architectures

Application Process -- Computer Graphics -- CGM/GKS

Information Processing Systems - Computer Graphics - Metafile (CGM) for the Storage and Transfer of Picture Description Information, Part 1; Functional Specification, IS 8632/1

Information Processing Systems - Computer Graphics - Metafile (CGM) for the Storage and Transfer of Picture Description Information, Part 2; Character Encoding, IS 8632/2

Information Processing Systems - Computer Graphics - Metafile (CGM) for the Storage and Transfer of Picture Description Information, Part 3; Binary Encoding, IS 8632/3.

Information Processing Systems - Computer Graphics - Metafile (CGM) for the Storage and Transfer of Picture Description Information, Part 4; Clear Text Encoding, IS 8632/4.

Information Processing Systems - Font and Character Information Interchange, IS 9541.

Information Processing Systems - 8-Bit Single Byte Coded Graphic Character Sets, Part 1; Latin Alphabet Part 1, IS 8859/1.

Information Processing Systems - Computer Graphics Functional Specification of the Graphical Kernel System (GKS), IS 7942.

Information Processing Systems - Computer Graphics - Graphical Kernel System for Three Dimensions (GKS-3D), Functional Description, DIS 8805.

Information Processing Systems - Computer Graphics - Programmers Hierarchical Interactive Graphics System (PHIGS), DP 9592.

Information Processing Systems - Computer Graphics - Interfacing Techniques for Dialogues with Graphical Devices (CGI),ISO TC 97/SC 21 N 1179.

Application Layer -- Directory Services

The Directory--Overview of Concepts, Models, and Services (CCITT Recommendation X.500, ISO 9594)

The Directory--Information Framework (CCITT Recommendation X.501, ISO 9594)

The Directory--Access and System Services Definition (CCITT Recommendation X.511, ISO 9594)

The Directory--Procedures For Distributed Operation (CCITT Recommendation X.518, ISO 9594)

The Directory--Access and System Protocols Specification (CCITT Recommendation X.519, ISO 9594)

The Directory--Selected Attribute Types (CCITT Recommendation X.520, ISO 9594)

The Directory--Selected Object Classes (CCITT Recommendation X.521, ISO 9594)

The Directory--Authentication Framework (CCITT Recommendation X.509, ISO 9594)

Remote Operations-Part 1: Model, Notation and Service Definition (CCITT Recommendation X.219, ISO 9072 Version 5)

Remote Operations-Part 2: Protocol Specification (CCITT Recommendation X.229, ISO 9072 Version 5)

Association Control-Service Definition (CCITT Recommendation X217, ISO 8649)

Association Control-Protocol Definition (CCITT Recommendation X.217, ISO 8650)

Note: ISO 9594, 9072 are preferred texts (over the CCITT counterparts) and are to be taken as of Gloucester, Nov. 1987.

Additional ISO References

ISO 8859-1 - Information Processing - 8-bit Single-byte Coded Graphic Character Sets - Part 1: Latin Alphabet No. 1

ISO 8859-2 - Information Processing - 8-bit Single-byte Coded Graphic Character Sets - Part 2: Latin Alphabet No. 2

DIS 8859-3 - Information Processing - 8-bit Single-Byte Coded Graphic Character Sets - Part 3: Latin Alphabet No. 3

ISO 6937-1 - Information Processing - Coded Character Sets for Text Communication - Part 1: General Introduction

ISO 6937-2 - Information Processing - Coded Character Sets for Text Communication - Part 2: Latin Alphabetic and Non-Alphabetic Graphic Characters

ISO 2022 - Information Processing - ISO 7-Bit and 8-Bit Coded Character Sets - Code Extension Techniques

ISO documents may be obtained from:

Frances E. Schrotter
ANSI
ISO TC 97/SC 6 Secretariat
1430 Broadway
New York, NY. 10018

19.3 Additional References

INTAP-S007-03 - Multi-media (ODA/ODIF) Functional Standard Detail Description (AE.111n-J), Small Function Set Profiles of Document Interchange Format with Layout Information

INTAP-S007-04 - Multi-media (ODA/ODIF) Functional Standard Detail Description (AE.112n-J), Medium Function Set Profiles of Document Interchange Format with Layout Information

19.4 IEEE

Physical Layer

IEEE Standard for Local Area Networks: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) and Physical Layer Specification, ANSI/IEEE Standard 802.3 1985, Institute of Electrical and Electronics Engineers, 345 East 47th St., New York, NY. 10017, 1985.

IEEE Standard for Local Area Networks: Supplements to Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specification ANSI.IEEE Standard 802.3a, b, c, e -1988

Supplement a, MAU and Baseband Medium Specification, Type 10BASE2 (Section 10)

Supplement a, Broadband MAU & Medium Specification, Type 10BROAD36 (Section 11)

Supplement c, Repeater Set and Repeater Unit Specification for Use with 10BASE5 and 10 BASE2 Networks

Supplement e, Physical Signaling, Medium Attachment, and Baseband Medium Specification, Type 10BASE5

IEEE Standard for Local Area Networks: Token-Passing Bus Access Method and Physical Layer Specification, ANSI/IEEE Standard , 802.4

Draft 1987, Institute of Electrical and Electronics Engineers, 345 East 47th St., New York, NY. 10017, 1985.

IEEE Standard for Local Area Networks: Token-Ring Access Method, ANSI/IEEE Standard 802.5- 1986, Institute of Electrical and Electronics Engineers, 345 East 47th St., New York, NY. 10017, 1985.

Data Link Layer

IEEE Standard for Local Area Networks: Logical Link Control, ANSI/IEEE Standard 802.2 - 1987, Institute of Electrical and Electronics Engineers, 345 East 47th St., New York, NY. 10017, 1985.

19.5 NBS

Local Area Networks: Baseband Carrier Sense Multiple Access with Collision Detection Access Method and Physical Layer Profiles and Link Layer Protocol, FIPS 107, NTIS, U.S. Department of Commerce, 5285 Port Royal Road, Springfield, VA. 22161.

Interface Between Data Terminal Equipment (DTE) and DataCircuit-Terminating Equipment (DCE) For Operation With Packet-Switched Data Communications Networks, FIPS 100, NTIS, U.S. Department of Commerce, 5285 Port Royal Road, Springfield, VA. 22161.

Implementation Agreements for Open Systems Interconnection Protocols: NBS Workshop for Implementors of Open Systems Interconnection, National Bureau of Standards, NBSIR 86-3385-6, Robert Rosenthal, Editor, Revised July 1987.

Implementation Agreements Among Participants of OSINET, National Bureau of Standards, Institute for Computer Sciences and Technology, NBSIR 86-3478, 1987.

U. S. Government Open Systems Interconnection Profile (GOSIP), National Bureau of Standards, Institute for Computer Sciences and Technology, 1987.

NBS documents may be obtained from:

NTIS
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA. 22161.
or
National Bureau of Standards
Institute for Computer Sciences and Technology
Gaithersburg, MD. 20899

19.6 MAP

Manufacturing Automation Protocol, General Motors Corporation,Manufacturing Engineering and Development, Advanced Product and Manufacturing Engineering Staff (APMES), APMES A/MD-39, GM Technical Center, Warren, MI. 48090-9040.

19.7 TOP

Technical and Office Protocols Specification Version 3.0, MAP/TOP Users Group, Attention TOP 3.0 Document. One SME Drive, P.O. Box 930, Dearborn Mi. 48121.

19.8 CEN/CENELEC

FTAM Draft Functional Standard A/111 (prENV 41 204) Simple File Transfer-Unstructrued, 1987-12-01.

ENV 41201 "Private Message Handling Systems"

ENV 41202 "Public Message Handling Systems"

19.9 SPAG

FTAM Draft Profile A/112 Positional File Transfer - Flat, 1987-120-7.

FTAM Draft Profile A/122 Positional File Access - Flat, 1987-12-07.

FTAM Draft Profile A/13 File Management, 1987-12-07.

NBS-114A (REV. 2-80)

U.S. DEPT. OF COMM. **BIBLIOGRAPHIC DATA SHEET** *(See instructions)*	1. PUBLICATION OR REPORT NO. NIST/SP-500/162	2. Performing Organ. Report No.	3. Publication Date February 1989

4. TITLE AND SUBTITLE

STABLE IMPLEMENTATION AGREEMENTS FOR OPEN SYSTEMS INTERCONNECTION PROTOCOLS
VERSION 2 EDITION 1, DECEMBER 1988

5. AUTHOR(S)

Tim Boland, Editor

6. PERFORMING ORGANIZATION *(If joint or other than NBS, see instructions)*

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
(formerly NATIONAL BUREAU OF STANDARDS)
U.S. DEPARTMENT OF COMMERCE
GAITHERSBURG, MD 20899

7. Contract/Grant No.

8. Type of Report & Period Covered

Final

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS *(Street, City, State, ZIP)*

SAME AS ITEM #6 ABOVE.

10. SUPPLEMENTARY NOTES

Library of Congress Catalog Card No. 89-600726
Supersedes NBS/SP-500/150 (January 1988)

□ 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)*

This document records current Stable Agreements for Open Systems Interconnection Protocols among the organizations participating in the NIST/OSI Workshop Series for Implementors of OSI Protocols.

12. KEY WORDS *(Six to twelve entries; alphabetical order; capitalize only proper names; and separate key words by semicolons)*

NIST/OSI Workshop; network protocols; open systems interconnection; OSINET; testing protocols

13. AVAILABILITY

xx □ Unlimited

□ For Official Distribution. Do Not Release to NTIS

xx □ 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

508

15. Price

ANNOUNCEMENT OF NEW PUBLICATIONS ON COMPUTER SYSTEMS TECHNOLOGY

Superintendent of Documents,
Government Printing Office,
Washington, DC 20402

Dear Sir:

Please add my name to the announcement list of new publications to be issued in the series: National Institute of Standards and Technology Special Publication 500-.

Name ____

Company ____

Address ____

City __________________ State ______________ Zip Code ___________

(Notification key N-503)

☆ U.S. GOVERNMENT PRINTING OFFICE: 1989—2 4 2 -2 0 0 / 0 5 0 0 8

NIST Technical Publications

Periodical

Journal of Research of the National Institute of Standards and Technology—Reports NIST research and development in those disciplines of the physical and engineering sciences in which the Institute is active. These include physics, chemistry, engineering, mathematics, and computer sciences. Papers cover a broad range of subjects, with major emphasis on measurement methodology and the basic technology underlying standardization. Also included from time to time are survey articles on topics closely related to the Institute's technical and scientific programs. Issued six times a year.

Nonperiodicals

Monographs—Major contributions to the technical literature on various subjects related to the Institute's scientific and technical activities.

Handbooks—Recommended codes of engineering and industrial practice (including safety codes) developed in cooperation with interested industries, professional organizations, and regulatory bodies.

Special Publications—Include proceedings of conferences sponsored by NIST, NIST annual reports, and other special publications appropriate to this grouping such as wall charts, pocket cards, and bibliographies.

Applied Mathematics Series—Mathematical tables, manuals, and studies of special interest to physicists, engineers, chemists, biologists, mathematicians, computer programmers, and others engaged in scientific and technical work.

National Standard Reference Data Series—Provides quantitative data on the physical and chemical properties of materials, compiled from the world's literature and critically evaluated. Developed under a worldwide program coordinated by NIST under the authority of the National Standard Data Act (Public Law 90-396). NOTE: The Journal of Physical and Chemical Reference Data (JPCRD) is published quarterly for NIST by the American Chemical Society (ACS) and the American Institute of Physics (AIP). Subscriptions, reprints, and supplements are available from ACS, 1155 Sixteenth St., NW., Washington, DC 20056.

Building Science Series—Disseminates technical information developed at the Institute on building materials, components, systems, and whole structures. The series presents research results, test methods, and performance criteria related to the structural and environmental functions and the durability and safety characteristics of building elements and systems.

Technical Notes—Studies or reports which are complete in themselves but restrictive in their treatment of a subject. Analogous to monographs but not so comprehensive in scope or definitive in treatment of the subject area. Often serve as a vehicle for final reports of work performed at NIST under the sponsorship of other government agencies.

Voluntary Product Standards—Developed under procedures published by the Department of Commerce in Part 10, Title 15, of the Code of Federal Regulations. The standards establish nationally recognized requirements for products, and provide all concerned interests with a basis for common understanding of the characteristics of the products. NIST administers this program as a supplement to the activities of the private sector standardizing organizations.

Consumer Information Series—Practical information, based on NIST research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.

Order the **above** *NIST publications from: Superintendent of Documents, Government Printing Office, Washington, DC 20402.*

Order the **following** *NIST publications—FIPS and NISTIRs—from the National Technical Information Service, Springfield, VA 22161.*

Federal Information Processing Standards Publications (FIPS PUB)—Publications in this series collectively constitute the Federal Information Processing Standards Register. The Register serves as the official source of information in the Federal Government regarding standards issued by NIST pursuant to the Federal Property and Administrative Services Act of 1949 as amended, Public Law 89-306 (79 Stat. 1127), and as implemented by Executive Order 11717 (38 FR 12315, dated May 11, 1973) and Part 6 of Title 15 CFR (Code of Federal Regulations).

NIST Interagency Reports (NISTIR)—A special series of interim or final reports on work performed by NIST for outside sponsors (both government and non-government). In general, initial distribution is handled by the sponsor; public distribution is by the National Technical Information Service, Springfield, VA 22161, in paper copy or microfiche form.

U.S. Department of Commerce
National Institute of Standards and Technology
(formerly National Bureau of Standards)
Gaithersburg, MD 20899