NISTIR 89-4082

Working Implementation Agreements for Open Systems Interconnection Protocols

Based on the proceedings of the
NIST Workshop for Implementors of OSI
Plenary Assembly Held March 17, 1989
National Institute of Standards and Technology
Gaithersburg, MD 20899

Tim Boland, Editor

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards and Technology
National Computer Systems Laboratory
Gaithersburg, MD 20899

March 1989

Issued May 1989

NISTIR 89-4082

Working Implementation Agreements for Open Systems Interconnection Protocols

Based on the proceedings of the
NIST Workshop for Implementors of OSI
Plenary Assembly Held March 17, 1989
National Institute of Standards and Technology
Gaithersburg, MD 20899

Tim Boland, Editor

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards and Technology
National Computer Systems Laboratory
Gaithersburg, MD 20899

March 1989

Issued May 1989

NOTE: As of 23 August 1988, the National Bureau of Standards (NBS) became the National Institute of Standards and Technology (NIST) when President Reagan signed into law the Omnibus Trade and Competitiveness Act.

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

NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
Raymond G. Kammer, Acting Director

Table of Contents

List of Figures

List of Tables

1. GENERAL INFORMATION

1.1 PURPOSE OF THIS DOCUMENT

This document records working (not stable) implementation specification agreements of OSI protocols among the organizations participating in the NIST/OSI Workshop Series for Implementors of OSI Protocols. This work is not currently considered advanced enough for use in product development or procurement reference. However, it is intended that this work be a basis for future stable agreements. It is possible that any material contained in this document may be declared stable in the future, and the material should be considered in this light.

As each protocol specification is completed (becomes technically stable), it is moved from this working document to the stable companion document as described below.

o The companion document, "Stable Implementation Agreements for Open Systems Interconnection Protocols, Version 2, Edition 1 " records mature agreements considered advanced enough for use in product development or procurement reference. This document is released with a version number.

New text relating to any of the referenced subjects appears first in this working document. In general, new material must reside in this working document for at least one workshop period before being moved into the Stable Document.

Agreements text is either in this Working Document (not yet stable) or in the aligned Stable Document (has been declared stable). It is a goal that the same text not appear in the same position in both documents at once (except for section one).

The benefit of this document is that it gives the reader a glimpse of new functionality, for planning purposes. Together with the aligned, associated stable document plus errata, these two documents give the reader a complete picture of current OSI agreements in this forum.

This Working Document also records technical, alignment, and editorial errata to the companion Stable Document (Version 2, Edition 1); these errata would be described in the "errata" sections of each appropriate chapter. Thus, an implementor should look at the aligned section in the Stable Document plus any errata described in this working document to get the true current status of stable material.

1.2 PURPOSE OF THE WORKSHOP

At the request of industry, the National Institute of Standards and Technology 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

See the aligned section of the Stable Implementation Agreements Document for information.

1.4 USE AND ENDORSEMENT BY OTHER ENTERPRISES

The Workshops are held for those organizations expressing an interest in implementing or procuring OSI protocols and open systems. However, there is no corporate commitment to implementations associated with Workshop participation.

The Agreements in this document were a basis for testing and product demonstrations in the Enterprise Networking Event in Baltimore, MD, June, 1988.

The agreements contained in earlier versions of this document were used for OSI demonstrations at the National Computer Conference in 1984 and at the AUTOFACT conference in 1985.

The agreements from several versions of this document have been adopted for use in implementations running on OSINET.

The MAP/TOP Steering Committee has endorsed these agreements and will "continue the use of the most current, applicable Implementors Workshop Agreements in all releases of the MAP and TOP specifications."

The COS Strategy Forum has "adopted a resolution stating that as a matter of policy COS should select as its sources of Implementation Agreements organizations or forums that are: (1) Broadly open, widely recognized OSI Workshops (NIST/OSI Workshops are first preference) ..."

The implementation specifications from the "Stable Implementation Agreements for Open System Interconnection Protocols" are referenced in Federal Information Processing Standard 146, "Government OSI Profile (GOSIP)."

1.5 RELATIONSHIP OF THE WORKSHOP TO THE NIST LABORATORIES

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. This is work made available to the industry volunteers and to others making valid commitments to organized events and activities such as NCC, AUTOFACT, and OSINET. As soon as this work can be adequately documented, it is placed in the public domain through submission to the National Technical Information Service. Any organization may then obtain the work at nominal charge.

The NIST laboratories bear no other relationship to the Workshop.

1.6 STRUCTURE AND OPERATION OF THE WORKSHOP

1.6.1 Plenary

The main body of the Workshop is a plenary assembly. Any organization may participate. Representation is international. NIST prefers for the business of Workshops to be conducted informally, since there are no corresponding formal commitments within the Workshop by participants to implement the decisions reached. The guidelines followed are: 1) one vote per company or independent division, 2) only companies that regularly attend should vote, 3) only companies that plan to sell or buy a protocol should vote on its implementation decisions, 4) only companies knowledgeable of the issues should vote, and 5) no proxy votes are admissible. Other voting rules are contained in the draft Procedures Manual, Section 2.3.

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, usually during the Workshop. As technical work is completed by a SIG, it is presented to the plenary for disposition. Companies participating in a SIG are expected to participate in the plenary. Voting rules for SIGS are the same as voting rules for the plenary.

Special Interest Groups sometimes correspond with organizations performing related work, such as ANSI committees. Such correspondence should be sent through the plenary to the parent committee, such as ANSC X3T5 or ANSC X3S3. When SIG meetings take place between Workshops, the correspondence from these meetings should be addressed directly to the parent committee and copied to the Workshop plenary.

Following are procedures for cooperative work among Special Interest Groups.

- Any SIG (SIG 1) or individual having issues to discuss with or requirements of another SIG (SIG 2) should bring the matter to the attention of the chairperson of that SIG (SIG 2).

- The SIG 2 chairperson should bring the matter before SIG 2 for action.

- SIG 2 should respond to the concerns or needs of SIG 1 or the individual in a timely manner.

- If the matter cannot be satisfactorily resolved or if the request is outside the charter assigned to SIG 1, then it should be brought before the plenary.

- SIGs are expected to complete work in a timely manner and bring the results before the plenary for disposition. However, the plenary may elect to act on any issue within the scope of the workshop at any time.

Following are the charters of the Special Interest Groups.

FTAM SIG

Scope

- to develop stable FTAM Agreements between vendors and users for the implementation of interoperable products

- in particular to develop the FTAM Phase 2 product-level specifications and maintain these specifications with respect to experiences from implementations and from testing

- to define further FTAM functionality in the Phase 3 specifications. These will contain only extensions of FTAM Phase 2. It is a goal that Phase 3 will be backward compatible with FTAM Phase 2. The set of future work items listed below may be changed by the plenary if the work is more appropriate for other SIGs.

- to conduct liaison with and contribute to other bodies working on FTAM harmonization such as CEN/CENELEC, POSI, and the ISO activities to define Functional Standards

  and

- to conduct liaison with vendor/user groups such as COS, MAP, TOP, and SPAG

High priority work items:

- Complete and maintain FTAM Phase 2 Agreements
- Specify implementation of Error Recovery control procedures, specifically
- Error Recovery and Restart Data Transfer functional units
- Specify Concurrency Control parameter.
- Specify implementation of Character Set ISO 6937
- Specify requirements of FTAM to a Directory Service
- Specify use of Presentation Context Management functional unit.

Low priority work items:

- Add new Document Types/Constraint Sets
- Define use of Access Control
- Specify FADU Locking functional unit
- Specify File Store management (e.g., file directories)
- Specify File Name conventions
- Specify use of Overlapped Access

X.400 (MESSAGE HANDLING SYSTEMS) SIG

Develop product-level specifications for Message Handling Systems using the CCITT X.400 Recommendations.

Develop abstract tests for X.400, as requested by the ad hoc rapporteur for this study question in CCITT. This work is to be submitted by the plenary (after its approval) to the U.S. Department of State as a proposed U.S. contribution to CCITT Study Group VII.

LOWER LAYER SIG

The Lower Layer SIG will study OSI layers 1-4 and produce recommendations for implementations to support the projects undertaken by the workshop and the work of the other SIGs. Both connectionless and connection-oriented modes of operation will be studied. The SIG will accept direction from the plenary for work undertaken and the priority which it is assigned.

The objectives of the Lower Layer SIG are:

- Study OSI layers 1-4 as directed by the plenary,

- Produce and maintain recommendations for implementation of these layers,

- Where necessary, provide input to the relevant standards bodies concerning layers 1-4, in the proper manner, and

- Begin work on the implementation specification of the ISO Network Layer Routing Exchange Protocol prior to the ISO draft achieving DIS status.

The Lower Layer SIG will study both existing and emerging ISDN standards pertaining to user access and user services. The SIG will:

- Develop implementation agreements for user-network interfaces

- Develop conformance requirements

- Conduct Liaison with other standards/interest groups

OSI SECURITY ARCHITECTURE SIG

GOAL: To develop an overall OSI Security Architecture which is consistent with the OSI reference model and which economically satisfies the primary security needs of both the commercial and Government sectors.

APPROACH: To define a security architecture encompassing the security addenda presently being specified at certain OSI layers, the required cryptographic algorithms and related key management functions, and the security management functions which must be performed between the layers and the peer entities defined in the OSI architecture.

DIRECTORY SERVICES SIG

Produce functional implementation agreements based on ISO/CCITT specifications for Directory Services in accordance with the objectives and goals of the plenary.

- Provide a subset for NIST publication which is functional and forward compatible to further work by this Special Interest Group.

- Define stable core functionality which can be implemented in the near term.

VIRTUAL TERMINAL SIG

This Special Interest Group's charter is based upon the implementation of Draft International Standards 9040 and 9041 and their respective addenda, in providing Basic Virtual Terminal Service.

This group will develop agreements for the implementation and testing of the following terminal types.

- o X.29 PAD
- o TELNET
- o Basic Scrolling
- o Basic Paging
- o Basic Forms

UPPER LAYERS SIG

The charter of the Upper Layers SIG is as follows.

- o Develop product level specifications for the implementation of:
    - o Session service and protocol
    - o Presentation service and protocol
    - o ACSE service and protocol
    - o Remote Operations Service Element (ROSE)
    - o Reliable Transfer Service Element (RTSE)

- o In addition, the specifications to be developed by the Upper Layers SIG will address issues that are common to layers 5-7 such as addressing, registration, etc. This SIG will review output and proposals from other SIGs to ensure consistency with international standards regarding Upper Layer Architecture.

- o The specifications developed will be done to support the requirements of all ASE SIGs.

The objectives of the Upper Layers SIG are to:

- o Study OSI Session, Presentation, ACSE, ROSE, and RTSE

- o Incorporate implementor's agreements in the 1988 NBS standing document,

- o Produce and maintain recommendations for implementations of these layers,

- o Where necessary provide input to the relevant standards bodies concerning Session, Presentation, ACSE, ROSE, and RTSE

- o React in a timely manner (i.e., to develop corresponding implementor's agreements) to technical changes in ISO documents.

The following are the guidelines under which the Upper Layers SIG will operate:

- Align implementation agreements with other organizations such as ANSI and ISO,

- Develop implementor's agreements that promote the efficiency of protocols,

- Develop implementor's agreements that promote ease in the verification of interoperability,

- Develop necessary conformance statements.

NETWORK MANAGEMENT SIG

Will use phased workload approach to accommodate volume of emerging OSI management-related standards,

The SIG will:

- Agree upon NBS Implementors OSI systems management reference model

- Develop product level specifications for implementations, relating to common services/protocols for exchanging management information between OSI nodes

- Develop product level specifications for implementations relating to specific management services for exchanging fault management (FM), Security Management (SM), Configuration Management (CM), Accounting Management (AM), and Performance Management (PM) information between OSI nodes

- Initiate and coordinate with appropriate layer SIGs product level specifications of layer-specific management information to support FM, SM, CM, AM, and PM.

As necessary, the SIG will:

- Establish liaisons with various standards bodies

- Provide feedback for additional/enhanced services and protocols for OSI management

OFFICE DOCUMENT ARCHITECTURE

The SIG will:

- develop one or more product level specifications for implementations of ISO/DIS 8613, i.e., the SIG will define one or more Document Application Profiles (DAPs)

- develop requirements for conformance testing of products purporting conformance to the (se) DAP (s)

- specify and describe requirements for services that manage the generation and interpretation of the ODA document representation

- determine preferred relationships between ODA and other document interchange formats

- promote the SIG's agreements (e.g., presentations, product demonstrations, press releases)

As necessary, the SIG will:

- establish liaison with required SIGs (e.g., X.400, FTAM, and Upper Layers SIGs) to seek efficient transfer capability for document interchange based on the ODA SIG agreements

- provide feedback and liaison to groups working on ISO/DIS 8613 related activities

REGISTRATION SIG

The NIST OSI Workshop Registration Authority Special Interest Group (RA SIG) will deal with OSI Registration for the following areas:

A. Registration of NIST OSI Workshop-Specified Objects.

The NIST OSI Workshop RAD SIG will define the procedures for the operation of the NIST Registration Authority (i.e., NIST).

1. Define policies and procedures for the registration of objects defined by the NIST OSI Workshop,

2. Take account of currently existing OSI Workshop registration work,

3. Establish policies for the publication and promulgation of registered objects;

4. Liaise with other OSI Workshop SIGs, appropriate standards bodies (e.g., ANSI) and other appropriate organizations.

B. Support for ANSI (U.S.) Registration activities

Promote the registration of MHS Private and Administrative Management Domain Names, Network-Layer-Addresses, and other Administrative Objects by ANSI or a surrogate appointed by ANSI. If ANSI feels that it cannot serve as the Registration Authority or delegate its authority to another organization, then the NIST OSI Workshop RA SIG should actively support the search for another organization to carry out this work.

This SIG will conduct a self-assessment, three NIST OSI Workshop Plenary Meetings after the Charter is approved, to determine if it has fulfilled its mission. Based on this assessment, the SIG will either be disbanded or continue. This procedure will continue until the SIG is disbanded.

TRANSACTION PROCESSING SIG

The SIG will be the focal point for all work on Transaction Processing within the Workshop. In particular:

1. Define DP/DIS/IS 10026 (TP) Implementation Agreements.

2. Liaise with Upper Layers SIG to define DIS/IS 9805 (CCR) Implementation Agreements to satisfy TP requirements.

3. Liaise with other internal and external organizations as required.

1.7 POINTS OF CONTACT

OSI Workshop - Chairman	Tim Boland	NIST	(301) 975-3608
OSI Workshop - Registration	Brenda Gray	NIST	(301) 975-3664
FTAM SIG	Klaus Truoel	GMD/DFN	49-615-875700
X.400 SIG	Charles Fox	DEC	44-734-854885
Lower Layers SIG	Fred Burg	AT&T	(201) 949-0919
Security SIG	Denny Branstad	NIST	(301) 975-2913
Directory Services SIG	Chris Moore	Wollongong	(415) 962-7160
Virtual Terminal SIG	Cyndi Jung	3COM	(415) 940-7664
Upper Layers SIG	David Chappell	Cray Research	(612) 825-7928
ODA SIG	Frank Dawson	IBM	(214) 556-5052
Network Management	Paul Brusil	Mitre	(617) 271-7632
Technical Liaison Committee	J.J. Cinecoe	Wang	(508) 967-5514
MAP	Gary Workman	GM	(313) 947-0599
TOP	Laurie Bride	BCS	(206) 763-5719
Government OSI Profile	Jerry Mulvenna	NIST	(301) 975-3631
OSINET			
Steering Committee	Jerry Mulvenna	NIST	(301) 975-3631
Technical Committee	Carol Edgar	NIST	(301) 975-3613
SME (MAP/TOP Sponsorship)	Mark Shaw		(313) 271-1500
U.S. Government OSI User's Committee	Jerry Mulvenna	NIST	(301) 975-3631

2. SUB NETWORKS

2.1 INTRODUCTION

(Refer to Stable Implementation Agreements Document)

2.2 SCOPE AND FIELD OF APPLICATION

(Refer to Stable Implementation Agreements Document)

2.3 STATUS

An updated chapter containing all material considered stable as of the close of the March 13-17, 1989 workshop is available. It completely supersedes the material printed in Version 2, Edition 1 of the NIST/OSI Stable Implementors' Agreements dated December, 1988. Copies may be obtained by contacting the LLSIG officers as given below:

Fred Burg - Chairman
AT&T Bell Labs - Room 1M-325
Holmdel, NJ 07733
Phone 201-949-0919

Kathy Fuller - Secretary
Codex Corporation
20 Chabot Boulevard
Mansfield, MA 02048-1193
Phone: 617-364-2000

The major technical change from December 1988 was the movement (addition) of material in Section 2.6.3, #4 into this updated chapter. Other changes are editorial in nature. This updated chapter will be referenced as "Revised Stable Implementation Agreements Document."

2.4 ERRATA

Errata are as described above. Contact above individuals for more details.

2.5 LOCAL AREA NETWORKS

(Refer to Stable Implementation Agreements Document)

2.5.1 IEEE 802.2 Logical Link Control

(Refer to Stable Implementation Agreements Document)

2.5.2 IEEE 802.3 CSMA/CD Access Method

(Refer to Stable Implementation Agreements Document)

2.5.3 IEEE 802.4 Token Bus Access Method

(Refer to Stable Implementation Agreements Document)

2.5.4 IEEE 802.5 Token Ring Access Method

(Refer to Stable Implementation Agreements Document)

2.5.5 Fiber Distributed Data Interface (FDDI)

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

The following are implementation agreements with respect to FDDI MAC.

1 The address length shall be 48 bits.

2 The term "default" is defined to be the value of a parameter in an FDDI station or concentrator as originally supplied by the vendor. Stations need not be reset to the default values by a power off condition, but there shall be some manual or programmatic means of resetting stations and concentrators to the specified default values.

3 The default value of T_Max shall be at least 165 milliseconds and not more than 200 milliseconds.

4 The value of T_Reg shall be equal to T_Max unless set otherwise by the Network Manager or by a concentrator initializing a slave tree to achieve "graceful insertion".

5 All FDDI stations shall receive Info_Fields of 0 to 4478 bytes. The frame is defined as follows:

P	SD	FC	DA	SA	Info	FCS	ED	FS

Figure 2.1 FDDI STATION

P: Preamble (4 Idle Symbols)
SD: Starting Delimiter (2 Symbols, JK)
FC: Frame Control (2 Symbols)
DA: Destination Address (12 Symbols)
SA: Source Address (12 Symbols)
FCS: Frame Check Sequence (8 Symbols)
ED: Ending Delimiter (1 Symbol)
FS: Frame Status (3 Symbols)

6 Stations shall not use restricted token service.

2.5.5.2 Token Ring Physical Level (PHY,X3.148-1988)

The following implementation agreement is with respect to the FDDI PHY specifications.

1 The delay, that is the time between when a station receives a Starting Delimiter (JK symbol pair) until it repeats that Starting Delimiter, when that Starting Delimiter is preceded by a sequence of a Starting Delimiter followed by 50 Idle Symbols shall not exceed:

- one microsecond in a station, and

- one microsecond times the number of ports in a concentrator, in addition to the delays contributed by the slaves of the concentrator.

The measurement method described above allows a consistent repeatable measurement, however it does not measure maximum possible delay. When the delay is one microsecond as measured above, the maximum delay which can result is 1.164 microseconds. This number, not one microsecond, should be used per PHY to compare maximum possible network delay.

2.5.5.3 Physical Layer Media Dependent (PMD, X3.166-198X)

The following implementation agreements are with respect to the FDDI PMD specification.

1 Stations shall repeat all valid packets under all signal conditions specified in Section 5.2, "Active Input Interface", with a bit error rate (BER) of not more than 2.5 x 10^{-10}.

2 Stations shall repeat all valid packets under all signal conditions specified in Section 5.2, "Active Input Interface", except that the Minimum Average Power shall be -29 dBm (2 dB above the specified minimum), with a BER of not more than 10^{-12}.

2.6 X.25 WIDE AREA NETWORKS

2.6.1 Introduction

(Refer to the Stable Implementation Agreements Document).

2.6.2 ISO 7776

(Refer to the Stable Implementation Agreements Document).

2.6.3 ISO 8208

(Refer to the Revised Stable Implementation Agreements Document).

2.7 INTEGRATED SERVICES DIGITAL NETWORKS (ISDN)

2.7.1 Introduction

(Refer to the Stable Implementation Agreements Document).

2.7.2 Implementation Agreements

(Refer to the Stable Implementation Agreements Document).

2.7.2.1 Physical Layer, Basic Access at "U"

(Refer to the Stable Implementation Agreements Document).

2.7.2.2 Physical Layer, Basic Access at S and T

(Refer to the Stable Implementation Agreements Document).

2.7.2.3 Physical Layer, Primary Rate at "U"

(Refer to the Stable Implementation Agreements Document).

2.7.2.4 Data Link Layer, D-Channel

(Refer to the Stable Implementation Agreements Document).

2.7.2.5 Signaling

(Refer to the Stable Implementation Agreements Document).

2.7.2.6 Data Link Layer B-Channel

(Refer to the Stable Implementation Agreements Document).

2.7.2.7 Packet Layer

(Refer to the Stable Implementation Agreements Document).

2.7.3 Rate Adaptation[1]

The following recommendations are made with respect to implementation of Draft T1E1.4/88-071, V.120 ISDN Rate Adaptation Specifications.

1 The preferred method of Information Transfer (V.120 Section 3.5) in Asynchronous Protocol Sensitive mode is Multiple Frame Acknowledged Information Transfer.

2 V.120 terminal adapters should not resend the last I-frame transmitted as a poll upon expiry of timer T200 (although they must respond appropriately if they receive an I-frame poll).

[1] It is recognized that these agreements are not relevant to implementations of OSI. They were originally developed at the request of the NIST NIU Executive Committee and are temporarily included in these agreements until a comparable ISDN Agreements document is available.

3. NETWORK LAYER

3.1 INTRODUCTION

(Refer to the Stable Agreements Document)

3.2 SCOPE AND FIELD OF APPLICATION

(Refer to the Stable Agreements Document)

3.3 STATUS

An updated chapter containing all material considered stable as of the close of the March 13-17, 1989 workshop is available. It completely supersedes the material printed in Version 2, Edition 1 of the NIST/OSI Stable Implementors' Agreements dated December, 1988. Copies may be obtained by contacting the LLSIG officers as given below:

Fred Burg - Chairman
AT&T Bell Labs - Room 1M-325
Holmdel, NJ 07733
Phone 201-949-0919

Kathy Fuller - Secretary
Codex Corporation
20 Chabot Boulevard
Mansfield, MA 02048-1193
Phone: 617-364-2000

The major technical change from December 1988 was the movement (addition) of material in Section 3.9.3 into this updated chapter. Other changes are editorial in nature. This updated chapter will be referenced as "Revised Stable Implemantation Agreements Document."

3.4 ERRATA

Errata are as described above. Contact above individuals for more details.

3.5 CONNECTIONLESS-MODE NETWORK SERVICE (CLNS)

3.5.1 ISO 8473

1. Subsets of the protocol:

(Refer to the Stable Implementation Agreements Document).

2. Mandatory Functions:

(Refer to the Stable Implementation Agreements Document).

3. Optional Functions:

- (Refer to the Stable Implementations Agreements document).

- Intermediate systems implementing priority shall do so as described below. For End system network entities the implementation of priority is optional, but if implemented it shall also be done as described below.

  1 NPDUs shall be scheduled based on the priority functions of ISP 8473. The scheduling algorithm for achieving this priority function is left as a local matter. It is required that the following constraints be met as described below.

    - An NPDU of lower priority shall not overtake an NPDU of higher priority in an intermediate system (i.e. exit an IS ahead of a higher priority NPDU arriving before it).

    - A minimum flow shall be provided for lower priority PDUs.[2]

  2 According to ISO 8473, the priority level is a binary number with a range of 0000 0000 (lowest priority) to 0000 1111 (highest priority level). Within this range, the four abstract values corresponding to the four levels defined in Section 3.11 shall be encoded as follows:

    - "high reserved" priority will be encoded with value 14 (0000 0000 0000 1110),

    - "high" priority will be encoded with value 10 (0000 0000 0000 1010),

[2] The scheduling algorithm by which this is accomplished is for further study.

- "normal" priority will be encoded with value 5 (0000 0000 0000 0101), and

- "low" priority will be encoded with value "zero" (0000 0000 0000 0000)

For a receiving network entity, a value lower than 5 shall be considered as "low"; a value lower than 10 and higher than 5 shall be considered as "normal", and a value lower than 14 and higher than 10 shall be considered as "high".

3 Network entities supporting priority shall process PDUs in which the priority parameter is absent as either "low", "normal", or "high" according to a locally configurable parameter. This is to ensure that NPDUs not containing the priority parameter can be processed by intermediate systems in a defined manner with respect to those which do contain the priority parameter.

4 IEEE 802.4 and IEEE 802.5 local area networks as well as some X.25 networks implementations have the ability to support subnetwork priorities. When available, a subnetwork priority function should be utilized in support of the priority requested of the network layer. The mapping of network layer priority levels onto subnetwork priority levels is a local configuration matter.

3.5.2 Provision of CLNS over Local Area Networks

(Refer to the Stable Agreements Document)

3.5.3 Provision of CLNS over X.25 Subnetworks

(Refer to the Stable Agreements Document)

3.5.4 Provision of CLNS over ISDN

(Refer to the Stable Implementation Agreements document).

3.5.4.1 CLNP Utilizing X.25 Services

(Refer to the Stable Implementations Agreements document).

3.5.5 Provision of CLNS over Point-to-Point Links

(To be based on ISO 8880)

3.6 CONNECTION-MODE NETWORK SERVICE

3.6.1 Mandatory Method of Providing CONS

3.6.1.1 General

(Refer to the Stable Implementation Agreements document).

3.6.1.2 X.25 WAN

(Refer to the Stable Implementation Agreements document).

3.6.1.3 LANs

(Refer to the Stable Implementation Agreements document).

3.6.1.4 ISDN

(Refer to the Stable Implementation Agreements document).

3.6.1.5 PRIORITY

Priority for CONS will be addressed with the implementation of X.25-1988 in a future version of these agreements.

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

(Refer to the Stable Implementation Agreements Document)

3.6.3 Agreements on Protocols

(Refer to the Stable Implementation Agreements Document)

3.6.3.1 ISO 8878

(Refer to the Stable Implementation Agreements Document)

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

(Refer to the Stable Implementation Agreements Document)

3.7 ADDRESSING

(Refer to the Stable Agreements Document)

3.8 ROUTING

3.8.1 End System to Intermediate System Routing

(Refer to the Stable Agreements Document)

3.8.2 Intermediate Systems to Intermediate Systems Routing

(Refer to the Stable Implementation Agreements

3.9 PROCEDURES FOR OSI NETWORK SERVICE/PROTOCOL IDENTIFICATION

3.9.1 General

(Refer to the Stable Implementation Agreements document).

3.9.2 Processing of Protocol Identifiers

(Refer to the Stable Implementation Agreements document).

3.9.2.1 Originating NPDUs

(Refer to the Stable Implementation Agreements document).

3.9.2.2 Destination System Processing

(Refer to the Stable Implementation Agreements document).

3.9.2.3 Further Processing in Originating End System

(Refer to the Stable Implementation Agreements document).

3.9.3 Applicable Protocol Identifiers

(Refer to the Revised Stable 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

(Refer to the Stable Agreements Document)

3.11 USE OF PRIORITY[3]

3.11.1 Introduction

Within the OSI environment, Quality of Service (QoS) parameters are intended to influence the qualitative behavior of the various OSI Layer entities. QoS is described in terms of parameters related to performance, accuracy, and reliability (e.g. delay, throughput, priority, error rate, security, failure probability, and etc.).

QoS covers a broad spectrum of issues. As a first step, these agreements address the efficient sharing of Layer 1, 2, & 3 transmission resources by making use of the priority parameter. To accomplish this, implementation agreements and encodings are provided for Network and Transport Layer protocols. The implication of these agreement for upper lower protocols is limited to the conveyance of priority information in both directions between an application entity and the service boundary for the Transport Layer.

The implementation of priority as defined herein is optional for intermediate systems and end systems, but if implemented shall be as defined in the layer specific agreements (for Network Layer see Section 3.5.1; for Transport Layer see Section 4.5.1.2.6, and for Upper Layers the section will be included at a later date).

3.11.2 Overview

The purpose of the priority parameter, in the context of the lower layers, is to influence the scheduling of the transmission of data on subnetworks, in CONS as well as CLNS environments (end systems as well as intermediate systems). The priority parameter

[3] This section provides initial proposals on the use of priority. The proposal requires further technical review before considering it as having support as an implementation agreement. Refer to the following documents for further technical information:

LLSIG 88-64 LLSIG 88-120 LLSIG 88-122

as defined is to be used by OSI Applications to control the "priority of data". Within the lower layers this translates into a contention for transmission resources, which has a direct impact on performance.

In order to implement practical mechanisms for scheduling the transmission of data units while maintaining the usefulness of priority, the specification of priority levels is limited to four; one corresponding to each of the four service classes:

- low priority
- normal priority
- high priority
- high reserved priority

The high reserved priority level is intended primarily for OSI network management purposes. The three lower priority levels are intended for information exchange by users.

These four priority levels are used, from an applications point of view, in the various communications lower layers (Transport, Network and Data Link) to provide a consistent mapping of "abstract priority levels" in and n-service onto the n-1 service and when available, priority parameter values in the layer protocol. In the upper layers (ASCE, Presentation and Session) local mechanisms are expected to be provided to application layer ASEs with a means for conveying priority information in both directions through the communication upper layers.

For example, this implies that an application request for a high priority service will be conveyed through association/presentation/session and will result in a high priority data transport connection and either high priority data CLNP PDUs (CLNS case) or a high priority data network connection/X.25 virtual call (CONS case).

3.12 CONFORMANCE

(Agreements to be added at a later date)

4. TRANSPORT LAYER

4.1 INTRODUCTION

(Refer to Stable Implementation Agreements Document)

4.2 SCOPE AND FIELD OF APPLICATION

(Refer to the Stable Implementation Agreements document).

4.3 STATUS

An updated chapter containing all material considered stable as of the close of the March 13-17, 1989 workshop is available. It completely supersedes the material printed in Version 2, Edition 1 of the NIST/OSI Stable Implementors' Agreements dated December, 1988. Copies may be obtained by contacting the LLSIG officers as given below:

Fred Burg - Chairman
AT&T Bell Labs - Room 1M-325
Holmdel, NJ 07733
Phone 201-949-0919

Kathy Fuller - Secretary
Codex Corporation
20 Chabot Boulevard
Mansfield, MA 02048-1193
Phone: 617-364-2000

The major technical change from December 1988 was the movement (addition) of material in Section 4.7 into this updated chapter. Other changes are editorial in nature. This updated chapter will be referenced as "Revised Stable Implementation Agreements Document."

4.4 ERRATA

Errata are as described above. Contact above individuals for more details.

4.4.1 ISO/CCITT Defect Reports

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

4.5 PROVISION OF CONNECTION MODE TRANSPORT SERVICES

(Refer to the Stable Implementation Agreements document).

4.5.1 Transport Class 4

4.5.1.1 Transport Class 4 Overview

(Refer to the Stable Implementation Agreements document).

4.5.1.2 Protocol Agreements

4.5.1.2.1 Rules for Negotiation

Implementations shall not send user data in the **DR TPDU**. The disposition of any user data received in a **DR TPDU** is implementation dependent.

(For other rules refer to the Stable Implementation Agreements document).

4.5.1.2.2 Transport Class 4 Service Access Points or Selectors

(Refer to Stable Implementation Agreements Document)

4.5.1.2.3 Retransmission Timer

(Refer to Stable Implementation Agreements Document)

4.5.1.2.4 Keep-Alive Function

(Refer to Stable Implementation Agreements Document)

4.5.1.2.5 Congestion Avoidance Policies

(Refer to the Stable Implementation Agreements document).

Mandatory Requirements

1 A maximum size for the "receive credit window", the value of which is locally configurable, should be provided. A "receive credit window" reflects the number of credits sent by a Transport entity for a Transport connection. The maximum size of

the "receive credit window" shall be referred to as WR_1.

2 A maximum size for the "sending credit window", the value of which is locally configurable, shall be provided. A "sending credit window" reflects the number of data TPDUs that a Transport entity is willing to send on a Transport connection. The maximum size of the "sending credit window" shall be referred to as WS_1. As specified in ISO 8073, the "sending credit window"" shall also be less than or equal to the remote "receive credit window" as conveyed in the last CDT field.

3 It is strongly recommended that an implementation use a retransmission timer per Transport connection. If, upon expiration of the retransmission timer, an implementation allows more than "1" TPDU to be transmitted a means to locally adjust the maximum number shall be provided.

4 All implementations shall have the capability of operating without delaying ACKs of data TDPUs received in-sequence (i.e., A_L essentially equals zero). If an implementation optionally chooses to explicitly delay ACKs, a means to locally adjust A_L shall be provided.

Optional Requirements

(Refer to the Stable Implementation Agreements document).

4.5.1.2.6 Use of Priority[4]

For end systems, the implementation of priority is optional, but if implemented, one of the four values defined in Section 3.11 shall always be used in an instance of communications. In other words an explicit priority parameter shall be sent.

Additional requirements of systems implementing priority are defined below.

1 When Transport is implemented over a CLNS Network entity, each data TPDU and corresponding NSDU shall be assigned a priority level derived from the Transport

[4] Refer to Section 3.11 for an overview on the use of priority.

connection priority level, except as excluded in item 5b and 5d below[5].

2 A local mechanism shall be provided to convey priority information to the Network service. If appropriate, simultaneous Transport service request can be managed on a priority basis within the Transport Layer.

3 The four abstract values corresponding to the four levels defined in 3.11 shall be encoded as follows:[6]

- "high reserved" priority will be encoded with value "zero" (0000 0000 0000 0000), and
- "high" priority will be encoded with value 5 (0000 0000 0000 0101),
- "normal" priority will be encoded with value 10 (0000 0000 0000 1010),
- "low" priority will be encoded with value 14 (0000 0000 0000 1110)

4 Other values should be interpreted as follows: a value lower than 5 and higher than 0 shall be interpreted as "high", a value lower than 10 and higher that 5 shall be interpreted as "normal", and a value higher than 10 shall be interpreted as "low".

5 The exchange of priority parameters by Transport entities is performed as described below[7].

a If priority is implemented in the end system, a priority value corresponding to one of the four abstract levels defined in Section 3.11 will be conveyed down to the Transport entity and shall be encoded and sent in the CR TPDU as the priority level "desired" for the Transport connection.

b A receiving Transport entity supporting priority management shall either accept the priority level proposed in the CR TPDU or select a lower level.

[5] The approach to assigning priority to an NSDU is for further study.

[6] This encoding has been chosen to be consistent with ISO 8073, The results is a reverse encoding from that for ISO 8473.

[7] ISO 8073 does not define or support a sound negotiation mechanism at this time; the following process will serve to allow a priority level to be established for a TC.

The CR shall not be rejected solely because of the "desired" priority level. The selected priority level shall be encoded and returned to the calling Transport entity in the CC TPDU. The TC priority is also passed to the local session entity with the T-Connect indication primitive and is eventually conveyed to the ASE, which can reject the association if the priority is unacceptable.

If the receiving Transport entity supports priority but receives a CR TPDU without the priority parameter, it shall associate a default priority level with the Transport connection for the purposes of managing the Transport connections which may be under its control. This default level shall not be encoded and placed in the corresponding CC TPDU and shall not result in any priority information being associated with NSDUs being passed to the Network entity supporting the Transport connection. The default shall be either "low", "normal", or "high" according to the locally configurable parameter.

c A receiving Transport entity not supporting priority management shall ignore the parameter in the CR TPDU.

d When the initiating Transport entity receives the CC TPDU containing the priority parameter, it establishes the priority for the Transport connection based on the level received and conveys this to the session entity with the T-Connect confirm primitive. If the priority parameter does not appear in the CC TPDU, the initiating Transport entity shall assume the remote Transport entity does not support priority and will therefore assign a default priority level to the Transport connection for the purposes of managing the Transport connection with respect to the other simultaneous Transport connections which may be under its control. However, this default shall not result in any priority information being associated with NSDUs being passed to the Network entity supporting the Transport connection. The default shall be either "low", "normal", or "high" according to a locally configurable parameter.

4.5.2 Transport Class 0

(Refer to Stable Implementation Agreements Document)

4.5.2.1 Transport Class 0 Overview

(Refer to Stable Implementation Agreements Document)

4.5.2.2 Protocol Agreements

4.5.2.2.1 Transport Class 0 Service Access Points

(Refer to Stable Implementation Agreements Document)

4.5.2.3 Rules for Negotiation

(Refer to Stable Implementation Agreements Document)

4.5.3 Transport Class 2

4.5.3.1 Transport Class 2 Overview

Transport Class 2 is applicable in OSI end systems which provide the Connection-mode Network Service.

4.5.3.2 Protocol Agreements

Transport Class 2 agreements follow:

- The values of the TS1 and TS2 timers shall be configurable. The recommended timer values are:

  TS1: 60 seconds
  TS2: 60 seconds

- If present, the TSAP-id field in the CR and CC TDPUs 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

- The rules for class negotiation shall be used.

  Negotiation from Class 2 to Class 0 is achieved by indicating Class .0 in the Alternative Protocol Class field of the CR TPDU which proposes Class 2. This is only possible when no other transport connections are assigned to the underlying network connection.

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

Note 1: If Class 0 is indicated in the Alternative Protocol Class field and QoS parameters are conveyed and the responding end system chooses Class 0, then the QoS parameters have been ignored by the responding system.

4.6 PROVISION OF CONNECTIONLESS TRANSPORT SERVICE

Document ISO 8072/Add. 2 is the Transport Service Definition covering Connectionless-mode Transmission. Document ISO 8602 is the Protocol for providing the Connectionless-Mode Transport Service.

4.6.1 Connectionless Transport Overview

When providing the connectionless Transport Service, the protocol shall be implemented as specified in ISO 8602.

4.6.2 Protocol Agreements

The connectionless Transport protocol is a relatively simple protocol providing little opportunity for conflicting interpretations. A few relevant agreements follow.

- The optional elements of procedure for use of CLTS over CONS (i.e., 6.3 of ISO 8602) will not be supported.

- A Unitdata TPDU that is received that contains a protocol error or an unknown destination TSAP ID shall be discarded.

4.6.2.1 Connectionless Transport Service Access Points or Selectors

The TSAP selector field in the UD TPDU 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.

4.7 TRANSPORT PROTOCOL IDENTIFICATION

(Refer to the Revised Stable Implementation Agreements document.)

5. UPPER LAYERS

5.1 INTRODUCTION

This section specifies agreements for the implementation of OSI upper layer protocols, including Session, Presentation, ACSE, ROSE, and RTSE.

5.1.1 References

(Refer to Stable Agreements Document.)

5.2 SCOPE AND FIELD OF APPLICATION

The agreements in this section apply to all ASE agreements in this document, including FTAM, X.400, Directory Services, Virtual Terminal, and OSI Network Management. All upper layer agreements specified in Chapter 5 of the NIST Special Publication "Stable Implementation Agreements for Open Systems Interconnection Protocols" (with errata) are also implicitly included in these agreements.

5.3 STATUS

This version of the upper layer agreements is under development.

5.4 ERRATA

These Errata apply to text in Version 2, Edition 1 Stable Document.

NO. OF ERRATA	TYPE	REFERENCED DOCUMENT	SECTION	DESCRIPTION
UL 3/89-1	Editorial	NIST-SP 500-162	5.9.3.4	Change ":" to "." and capitalize the following "i"
UL 3/89-2	Alignment	NIST-SP 500-162	5.4.1	Add new Session Defect reports to existing text: ISO 8326 defect solutions: 023, 024 ISO 8327 defect solutions: 037, 038
UL 3/89-3	Alignment	NIST-SP 500-162	5.4.2	Replace existing text of 5.4.2 with "Session Defects Correcting CCITT X.215

NO. OF ERRATA	TYPE	REFERENCED DOCUMENT	SECTION	DESCRIPTION
				and X.225 The following approved defect solutions have been integrated into the current revisions of ISO 8326 and ISO 8327, but are not part of CCITT X.215 and X.225 (1984). The defect solutions must be incorporated into CCITT Session to insure conformance with ISO Session: ISO 8326 defect reports: 004, 006, 007, 009, 011, 012, 013, 014, 015, 016, 017, 020. ISO 8327 defect reports: 001, 003, 004, 005, 006, 007, 008, 009, 010, 012, 017, 018, 019, 026, 027, 030, 034, 035.

NO. OF ERRATA	TYPE	REFERENCED DOCUMENT	SECTION	DESCRIPTION
CP 3/89-4	Alignment/ Technical	NIST SP-500-162	5.8.3.9	Delete the Note; Change the final period to a comma, and add "so if more than one transfer syntax is proposed, CPC-type values may appers in that SS-user-data parameter. For a presentation context for which the Basic Encoding Rules are a proposed transfer syntax, all PDVs in the user data parameter of the CP PPDU must be encoded using the Basic Encoding Rules and must be examined by the receiving presentation protocol machine. Following CPC-type values may be examined or ignored at the receiver's option (see ISO 8823, Section 6.2.5.3)."

5.4.1 ISO Defect Reports

(See Stable Agreements Document.)

5.8.7 Connectionless

The connectionless Presentation protocol shall be implemented as specified in ISO 2nd PDAD 9576.

The Transfer-syntax-name component of a PDV-list value shall be present in a UD PPDU if and only if more than one transfer syntax name was proposed for the Presentation context of the Presentation data values. The Transfer-syntax-name component of a PDV-list value shall always be present in a UDC-type. The Transfer-syntax-name component of a PDV-list value shall only appear in the UD PPDU and UDC-type.

5.9 SESSION

5.9.1 Introduction

(Refer to Stable Agreements.)

5.9.2 Services

(Refer to Stable Agreements.)

5.9.3 Protocol Agreements

(Refer to Stable Agreements.)

5.9.4 General

TBD

5.9.5 Connection Oriented

TBD

5.9.6 Connectionless

The connectionless Session protocol shall be implemented as specified in ISO DIS 9548.

5.10 UNIVERSAL ASN.1 ENCODING RULES

5.8 PRESENTATION

5.8.1 Introduction

(Refer to Stable Agreements Document.)

5.8.2 Service

(Refer to Stable Agreements Document.)

5.8.3 Protocol Agreements

(Refer to Stable Agreements Document.)

5.8.4 Presentation ASN.1 Encoding Rules

(Refer to Stable Agreements Document.)

5.8.5 General

5.8.5.1 Presentation Data Value (PDV)

- o A Presentation data value (PDV) is a value of a type in an abstract syntax, e.g., a value of an ASN.1 type.
- o A PDV may contain embedded PDVs in different contexts. A change of context within a PDV is indicated by an EXTERNAL. EXTERNAL implies an embedded PDV.
- o A PDV cannot be split across PDV-lists in fully-encoded user data.
- o Fully encoded data that is a series of PDVs in the same Presentation context should be encoded as one PDV-list.

5.8.6 Connection Oriented

The Transfer-syntax-name component of a PDV-list value shall be present in a CP PPDU if and only if more than one transfer syntax name was proposed for the Presentation context of the Presentation data values. The Transfer-syntax-name component of a PDV-list value shall always be present in a CPC-type. The Transfer-syntax-name component of a PDV-list value shall only appear in the CP PPDU and CPC-type.

5.4.2 Session Defects

(See Stable Agreements Document.)

5.5 ASSOCIATION CONTROL SERVICE ELEMENT

5.5.1 Introduction

(Refer to Stable Agreements Document.)

5.5.2 Services

(Refer to Stable Agreements Document.)

5.5.3 Protocol Agreements

It is the intention of the UL SIG to adopt ACSE defect report 8650/004 when it becomes stable. Values for and usees of AE-titles are outside the scope of the Upper Layer SIG.

5.5.4 ASN.1 Encoding Rules

When the ABRT APDU is used during the connection establishment phase, Presentation layer negotiation is considered to be complete, and the "direct-reference" component of EXTERNAL shall not be present.

5.5.5 Connectionless

The connectionless ACSE protocol shall be implemented as specified in ISO DIS 10035.

5.6 ROSE

TBD

5.7 RTSE

TBD

5.8 PRESENTATION

5.8.1 Introduction

(Refer to Stable Agreements Document.)

5.8.2 Service

(Refer to Stable Agreements Document.)

5.8.3 Protocol Agreements

(Refer to Stable Agreements Document.)

5.8.4 Presentation ASN.1 Encoding Rules

(Refer to Stable Agreements Document.)

5.8.5 General

5.8.5.1 Presentation Data Value (PDV)

- A Presentation data value (PDV) is a value of a type in an abstract syntax, e.g., a value of an ASN.1 type.

- A PDV may contain embedded PDVs in different contexts. A change of context within a PDV is indicated by an EXTERNAL. EXTERNAL implies an embedded PDV.

- A PDV cannot be split across PDV-lists in fully-encoded user data.

- Fully encoded data that is a series of PDVs in the same Presentation context should be encoded as one PDV-list.

5.8.6 Connection Oriented

The Transfer-syntax-name component of a PDV-list value shall be present in a CP PPDU if and only if more than one transfer syntax name was proposed for the Presentation context of the Presentation data values. The Transfer-syntax-name component of a PDV-list value shall always be present in a CPC-type. The Transfer-syntax-name component of a PDV-list value shall only appear in the CP PPDU and CPC-type.

5.8.7 Connectionless

The connectionless Presentation protocol shall be implemented as specified in ISO 2nd PDAD 9576.

The Transfer-syntax-name component of a PDV-list value shall be present in a UD PPDU if and only if more than one transfer syntax name was proposed for the Presentation context of the Presentation data values. The Transfer-syntax-name component of a PDV-list value shall always be present in a UDC-type. The Transfer-syntax-name component of a PDV-list value shall only appear in the UD PPDU and UDC-type.

5.9 SESSION

5.9.1 Introduction

(Refer to Stable Agreements.)

5.9.2 Services

(Refer to Stable Agreements.)

5.9.3 Protocol Agreements

(Refer to Stable Agreements.)

5.9.4 General

TBD

5.9.5 Connection Oriented

TBD

5.9.6 Connectionless

The connectionless Session protocol shall be implemented as specified in ISO DIS 9548.

5.10 UNIVERSAL ASN.1 ENCODING RULES

5.10.1 TAGS

(Refer to Stable Document.)

5.10.2 Definite Length

(Refer to Stable Document.)

5.10.3 External

a. If a data value to be encapsulated in an EXTERNAL type is an instance of a single ASN.1 type encoded according to the Basic Encoding Rules for ASN.1, then the option "single-ASN.1-type" shall be chosen as its encoding.

b. If a data value to be encapsulated in an EXTERNAL type is encoded as an integral number of octets, and case a. does not apply, then the option "octet-aligned" shall be chosen as its encoding.

5.10.4 Integer

o Any incidence of an ASN.1 INTEGER type defined in an abstract syntax describing protocol control information must be encoded so that the length of its contents octets is no more than four octets, unless an explicit NIST agreement to the contrary is made for a specific INTEGER type.

5.10.5 String Types

o The contents octets for a constructed encoding of a BIT STRING, OCTET STRING, or character string value consists of the complete encoding of zero, one, or more data values, and the encoding of these data values must be primitive.

5.10.6 Bit String

o Unless otherwise specified in the abstract syntax definition, each bit named in a BIT STRING type used in that abstract syntax definition shall be explicitly encoded in the associated BIT STRING value, even if it is part of a string of trailing zero bits.

Extra trailing bits beyond the exact number of bits which correspond to the complete list of the named bits specified shall never be encoded. This rule applies to all BIT STRING types unless stated otherwise in the standards.

5.11 CHARACTER SETS

These sections describe Information Processing Character Set policies and agreements of the NIST OSI Workshop. These policies and agreements are based upon ISO Character Set International Standards and CCITT Character Set Recommendations. The Policy section describes agreements on character set practices which the SIGs are expected to implement where the basic standards upon which Implementation Agreements are founded do not specify contrary requirements. The Agreements section records specific Workshop agreements on character sets. The Tutorial Appendix B summarizes the character set practices of each of the SIGs, including all relevant encoding information drawn from the appropriate ISO Registers, ISO Standards, and CCITT Recommendations.

The objectives of this section are to:

- o Collect in one place all relevant character set information for all NIST OSI Workshop agreements and present relevant information from related standards (e.g., ASN.1),
- o Establish policy for future NIST OSI Workshop Agreements,
- o Describe character set conformance requirements,
- o Record NIST OSI Workshop Character Set agreements, and
- o Harmonize the use of character sets in conjunction with other OSI Workshops (e.g., EWOS and AOW).

5.11.1 Policy

Policy is defined to be a set of rules for formulating character set agreements. The SIGs are expected to abide by these policies to the extent possible under the constraints of their relevant standards. Exceptions should be recorded in the Specific Implementor Agreements Requirements section of this chapter.

5.11.1.1 Restrictions on Character Sets

An Application Service Element shall place no restriction on the character sets supported for user data, file contents, body parts, or other information which is passed through without processing (future processing).

5.11.1.2 Character Comparisons

All implementation agreements covering character comparisons and collation shall be recorded in this chapter.

5.11.2 Agreements

5.11.2.1 Encoding

5.11.2.1.1 Overprint, Composite Character

A composite character is defined as a diacritical in combination with an alphabetic as in ISO 6937. A composite character is considered as one character for purposes of comparison and character string computation.

With the exception of non-spacing diacriticals, sequences of graphic characters and control functions which would result in the presentation of two or more graphic characters in a single character position shall not be used, unless special provision has been made, subject to mutual agreement between the interchange parties. So, for example, the sequence "a BACKSPACE '" must be interpreted as three characters rather than as a single character.

5.11.2.1.2 Code Extension Facilities

This section constitutes the prior agreement on code extension required by ISO 2022.

For ASN.1 GeneralString and GraphicString types, the assumed extension facilities are as though the following escape sequences form ISO 2022 have been applied: ESC 2/0 4/3 and ESC 2/0 5/10. These sequences indicate:

- o 8-bit environment,
- o the G0, G1, and G2 graphic sets shall be used,
- o no locking shift functions shall be used, and
- o characters from G2 may be accessed by use of the single-shift 2 control function.

Designation ESCAPE sequences in a data stream are permitted. No Announcers of extension facilities may be used within these ASN.1 string types.

For ASN.1 T.61String ... <to be determined>

5.11.2.2 Comparisons

5.11.2.2.1 Matching Characters

A character value submitted with another character value does not have to be drawn from the same character set. However, the match is restricted to a list of pairs of character set values for which equality or inequality is defined. The result of comparing characters from a pair of character sets not in this list is undefined.

This list shows the pairs of character sets between which matching is defined.

ISO 6937-2	ISO 8859-1

Two characters are said to be equal if and only if their names are identical. The names are recorded in the registration of the character sets in the **International Register of Coded Character Sets to be used with Escape Sequences** and not the character set International Standard or Recommendation. In the case of ISO 6937-2 the composite characters which are formed from a diacritical followed by an alphabetic are not registered. Thus, the following table defines the match in terms of the ISO 6937-2 character name and the corresponding ISO Register name.

ISO 6937 name	**ISO Register Name**

<to be added>

5.11.2.2.2 Caseignore Comparisons

In character comparisons in which case is ignored, the matching rules of the section entitled "Matching Characters" are relaxed in that the characters are equal if their names differ only by one name having SMALL where the other name has CAPITAL.

5.11.2.2.3 Caseignore Comparisons

An agreement on comparison, other than equality or inequality, between characters requires a definition of a collating sequence. Such definitions shall be recorded in this chapter. The NIST OSI Workshop currently has no such agreements in place.

The collating sequence of letters, accented letters and other graphic symbols is not currently defined in an international standard or recommendation.

Preferred collating sequences might vary between countries.

5.11.2.2.4 Comparing Strings

In this section a character string is considered to be a sequence of characters, some of which may be composed of multiple bytes depending upon the character set encodings which are specified. Comparing two character strings gives the same answer independent of each character string's ASN.1 packaging:

- o as constructed or primitive form

- o definite or indefinite length form.

<this section will be further developed>

5.11.2.3 Agreements about Character Set Standards and Recommendations

This section covers agreements about:

- o subrepertoires supported,
- o standardized options selected,
- o component character sets and their registrations in the International Register of Coded Character Sets to be used with Escape Sequences where there is a choice to be made, or the standard does not specify it, and,
- o the designation of component character sets within the ISO 2022 Code Extension Model where there is a choice to be made.

For tutorial purposes, the consequences of these agreements and the constraints of the related character set standards are brought together in Appendix B.

5.11.2.3.1 ISO 8859 Character Sets

Implementations supporting ISO 8859-1 are required to support the following two graphic character sets from the **International Register of Coded Character Sets to be used with Escape Sequences**:

6	ASCII Graphic Character Set in G0
100	Right Hand Part of Latin Alphabet No. 1 in G1

Support of ISO 8859-7 Greek Alphabet is optional as an addition to 8859-1. This option requires the following set from the **International Register of Coded Character Sets to be used with Escape Sequences**:

126	Right Hand Part of the Latin/Greek Alphabet

Within this option, sets 100 and 126 may be designated into G1 and G2 respectively, or into G2 and G1 respectively.

5.11.2.3.2 ISO 6937-2 Character Sets

Implementations supporting ISO 6937-2 are required to support ISO 6937-2 Addendum 1 and one or more of the following subrepertoires as defined in the **International Register of Subrepertoires**.

9	Western European data processing and interchange
3	Text communication in European Languages (Subrepertoire of graphic characters for teletex)

Implementation supporting ISO 6937-2 are required to use the following character sets from the **International Register of Coded Character Sets to be used with Escape Sequences**:

2	International Reference Version of ISO 646 in G0
142	Supplementary set of Latin Alphabetic and non Alphabetic Graphic Characters in G2

The supplementary set shall be designated in G2. For subrepertoires 2 and 5, the supplementary set may be omitted at the discretion of the sending application.

5.11.2.3.3 CCITT T.61

Implementations of CCITT Recommendation T.61 other than X.400-1984 must support the 1988 version.

Support for JIS X0208 is optional. If JIS is supported, it shall be designated into G1. Support for Greek is outside the scope of these agreements. Dynamically Redefinable Character Sets (DRCS) shall not be used.

Support for T.61 as an ASN.1 GeneralString is outside of these agreements. Support of the graphic set components of T.61 as an ASN.1 GraphicString is outside the scope of these agreements.

The supplementary set of Graphic Character (ISO Registration 103) shall be designated in G2 when it is in use. It may be omitted where subsequent characters are drawn only from the basic set, or only from a standardized option.

Use of T.61 except where mandated by standards is outside the scope of these agreements. Exceptions to this rule for specific Application Service Element protocol elements must be documented in the individual chapters.

5.11.2.3.4 JIS 6226

This Japanese set is optionally supported.

Implementations supporting JIS X0208 are required to support the following two graphic sets:

6	ASCII Graphic Character Set in G0
87	Japanese Character Set JIS X0208 in G1

and optionally,

15	Japanese Katakana Character Set JIS (registration pending) in G2

These agreements are subject to verification of final text.

5.11.3 References for Character Set Text

CCITT Recommendation T.61 - 1985, "Character Repertoire and Coded Character Sets for the International Teletex Service", CCITT Red Book, Terminal Equipment and Protocols for Telematic Services, Recommendations of the T Series, International Telecommunications Union, Geneva.

DIS 8859-7 - 1987, "Information processing -- 8-bit single-byte coded graphic character sets -- Part 7: Latin/Greek alphabet", International Organization for Standardization, Geneva.

IS 2022 - 1986, "Information processing -- ISO 7-bit and 8-bit coded character sets -- Code extension techniques", International Organization for Standardization, Geneva.

IS 6429 - 1983, "Information Processing -- ISO 7-bit and 8-bit coded character sets -- Additional control functions for character-imaging devices", International organization for Standardization, Geneva.

IS 646 - 1983, "Information Processing -- ISO 7-bit coded character set for information interchange", International Organization for Standardization, Geneva.

IS 6937/1 - 1983, "Information processing -- Coded character sets for text communication -- Part 1: General introduction", International Organization for Standardization, Geneva.

IS 6937/2 - 1983, "Information processing -- Coded character sets for text communication -- Part 2: Latin alphabetic and non-alphabetic graphic characters", International Organization for Standardization, Geneva.

IS 8859-1 - 1987, "Information processing -- 8-bit single-byte coded graphic character sets -- Part 1: Latin alphabet No. 1", International Organization for Standardization, Geneva.

ISO Character Set Register - 1989, "International Register of Coded Character Sets to be Used With Escape Sequences", European Computer Manufactures Association, Geneva.

5.12 CONFORMANCE

(Refer to Stable Document.)

5.12.1 Specific ASE Requirements

(Refer to Stable Document.)

5.12.1.1 FTAM

(Refer to Stable Document.)

5.12.1.2 MHS

(Refer to Stable Document.)

5.12.1.2.1 Phase 1

(Refer to Stable Document.)

5.12.1.2.2 Phase 2, Protocol P7

(Refer to Stable Document.)

ROSE Requirements:

Operation and association classes are used as per the standard.

RTSE Requirements:

- o TWA
- o normal-mode

ACSE Requirements:

all

The use of AP-TITLE, AE-QUALIFIER, AP-INVOCATION-ID, and AE-INVOCATION-ID are prohibited; however, a receiving entity must be capable of ignoring them (if present) without refusing the connection.

Application Contexts:

- o "MS-access" - mandatory; normal mode
- o "MS-reliable-access" - optional; normal mode

Abstract Syntaxes:

- o "ISO 8650-ACSE1"

Associated Transfer Syntax:

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

Presentation Requirements:

Presentation Functional Units:

- o kernel

Presentation Contexts:

- o 2

Abstract Syntaxes:

- o ?

  Associated Transfer Syntax:

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

Session Requirements:

Session Functional Units:

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

Version Number: 2

Maximum size of User Data parameter field: 10,240

Session Notes:

- o MHS proposes both versions 1 and 2 for pass through mode, but only version 2 for normal mode.

- o 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.

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

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

5.12.1.2.3 Phase 2, Protocol P3

ROSE Requirements:

As per Phase 2, P7.

RTSE Requirements:

?

ACSE Requirements:

As per Phase 2, P7.

Application Contexts:

- "MTS-access" - mandatory
- "MTS-reliable-access" - optional
- "MTS-forced-access" - mandatory
- "MTS-forced-reliable-access" - optional

Presentation Requirements:

As per Phase 2, P7.

Session Requirements:

As per Phase 2, P7.

5.12.1.2.4 Phase 2, Protocol P1

ROSE Requirements:

ROSE is not used.

RTSE Requirements:

- Monologue
- TWA

ACSE Requirements:

As per Phase 2, P7.

Application Contexts:

- "MTS-transfer-protocol-1984" - mandatory
- "MTS-transfer-protocol" - mandatory
- "MTS-transfer" - mandatory

Presentation Requirements:

As per Phase 2, P7.

Session Requirements:

As per Phase 2, P7.

5.12.1.3 DS

(Refer to Stable Document.)

5.12.1.4 Virtual Terminal

(Refer to Stable Document.)

5.12.1.5 Network Management

5.13 REFERENCES

The following documents are referenced in these ongoing NIST agreements on the OSI Upper Layers. Other document references may be found in the Stable Implementation Agreements for OSI Protocols of December, 1988.

5.13.1 ACSE

[A1] Information Processing Systems - Open Systems Interconnection - Connectionless ACSE Protocol to Provide the Connectionless-Mode ACSE Service, ISO DIS 10035: 1989-02-25 (ISO/IEC JTC1/SC21 N 3456).

5.13.2 Session Layer

[S1] Information Processing Systems - Open Systems Interconnection - Session Service Definition: Addendum 3 Covering Connectionless-Mode Session Service, ISO/DAD3 8326: 1989-02-25 (E) (ISO/IEC JTC1/SC21 N 3462).

[S2] Information Processing Systems - Open Systems Interconnection - Connectionless Session Protocol to Provide the Connectionless-Mode Session Service, ISO/DIS 9548: 1989-02-25 (E) (ISO/IEC JTC1/SC21 N 3460).

5.13.3 Presentation Layer

[P1] Information Processing Systems - Open Systems Interconnection - Presentation Service Definition: Draft Addendum 1 Covering Connectionless-Mode Presentation Service, ISO/DAD1 8822: 1989-02-25 (E) (ISO/IEC JTC1/SC21 N 3171).

[P2] Information Processing Systems - Open Systems Interconnection - Connectionless Presentation Protocol to Provide the Connectionless-Mode Presentation Service, ISO/DIS 9576: 1989-02-25 (E) (ISO/IEC JTC1/SC21 N 3172).

6. OBJECT IDENTIFIERS AND OTHER REGISTRATION ISSUES (STABLE)

Editor's Note: For current information on this subject, refer to the aligned section in the Stable Implementation Agreements. New text on this subject will be included here.

ERRATA

NO. OF ERRATA	TYPE	REFERENCED DOCUMENT	SECTION	DESCRIPTION
OI 3/89-1	Editorial	NIST-SP 500-162	First paragraph of Chapter 6	Remove first paragraph (following the Editor's Note) of Chapter 6

7. STABLE MESSAGE HANDLING SYSTEMS

Editor's Note: For current stable MHS agreements, consult the aligned section in the Stable Implementation Agreements document. This section serves as a reference or pointer to stable agreements approved on or before December 16, 1988.

8. MESSAGE HANDLING SYSTEMS

8.1 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 Message Handling Systems (MHS) based on both the CCITT X.400(1988) series of Recommendations and the similar (but not identical) ISO MOTIS standard (see References). The term 'MHS' is used to refer to both sources where a distinction is unnecessary. Similarly, '1984' and '1988' are often used to distinguish between the CCITT X.400(1984) series of Recommendations and the later sources. Figure 8.1 shows the layered structure of this Agreement.

This Implementation Agreement seeks to establish a common specification which is conformant with both CCITT and ISO with a view to:

- o Preventing a proliferation of incompatible communities of MHS systems which are isolated for protocol reasons,
- o Achieving interworking with implementations conforming to the NIST Stable Implementation Agreements for CCITT 1984 X.400-based Message Handling Systems, and
- o Facilitating integration of other OSI-based services (e.g., Directory) within a single real system.

This initial Implementation Agreement is designed to encourage early upgrade of existing 1984-based systems as follows:

- o To add useful 1988 functionality (Message Store, remote UA, etc), and
- o To provide a minimal conformant 1988 MHS as a firm basis for the introduction of further 1988 services and features. Subsequent versions of this Agreement will define such additional 1988 aspects as incremental enhancements.

However, it is <u>not</u> considered that the existing NIST Stable Implementation Agreements for CCITT 1984 X.400-based Message Handling Systems should be withdrawn at this stage and it can be anticipated that X.400(1984) implementations will continue to provide a viable alternative for applications that do <u>not</u> require the additional 1988 functionality for some time.

Interpersonal Messaging System	CCITT X.420	ISO 10021-7
Message Store	CCITT X.413	ISO 10021-5
Message Transfer System	CCITT X.411 CCITT X.419	ISO 10021-4 ISO 10021-6
Remote Operations Service Element	CCITT X.219/229	ISO 9072
Reliable Transfer Service Element	CCITT X.218/228	ISO 9066
Association Control Service Element	CCITT X.217/227	ISO 8649/50
Presentation Layer	CCITT X.216/226	ISO 8822/23
Session Layer	CCITT X.215/225	ISO 8326/27

Figure 8.1 The Layered Structure of this Implementation Agreement

8.2 SCOPE

This Agreement specifies the requirements for MHS implementations based on the 1988 MHS standards (see Figure 8.1 above).

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

(A) PRMD to PRMD,
(B) PRMD to ADMD,
(C) ADMD to ADMD,
(D) MTA to MTA (within a domain, e.g., for MTAs from different vendors),
(E) MTA to remote MS or UA, and
(F) MS to remote UA.

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. In cases E and F, the UA is located remotely from the MTA. Figure 8.2 illustrates the interfaces to which this Agreement applies.

MHS protocols other than the Message Transfer Protocol (P1), the Message Transfer System Access Protocol (P3), the Interpersonal Messaging Protocol (P2), and the Message Store Access Protocol (P7) 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 8.2. Provision for the use of the remaining services defined in the MHS standards is outside the scope of this document.

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.

The 1988 MHS standards cover a wide and diverse range of functional areas, not all of which would be relevant to every implementation.

The initial version of this Agreement will define a minimal conformant MHS implementation which will be capable of interworking with implementations based on the CCITT X.400(1984) Recommendations as defined in Chapter 7 of the NIST Stable Implementation Agreements for OSI Protocols (Version 2 Edition 1, December 1988), and will additionally define the minimum set of requirements which are necessary to provide useful remote UA and/or Message Store services, independent of the level (i.e. 1984 or 1988) of the P1 implementation.

In order to achieve a more precise definition of conformance requirements according to the functionality supported by an implementation (and additionally to facilitate future enhancement of this initial specification), the concept of 'Functional Groups' has been introduced. Figure 8.3 shows the Functional Groups covered by this Agreement and indicates where they are defined in this Chapter. Only the MT and IPM Kernel Functional Groups have to be supported for minimal conformance to this initial Agreement.

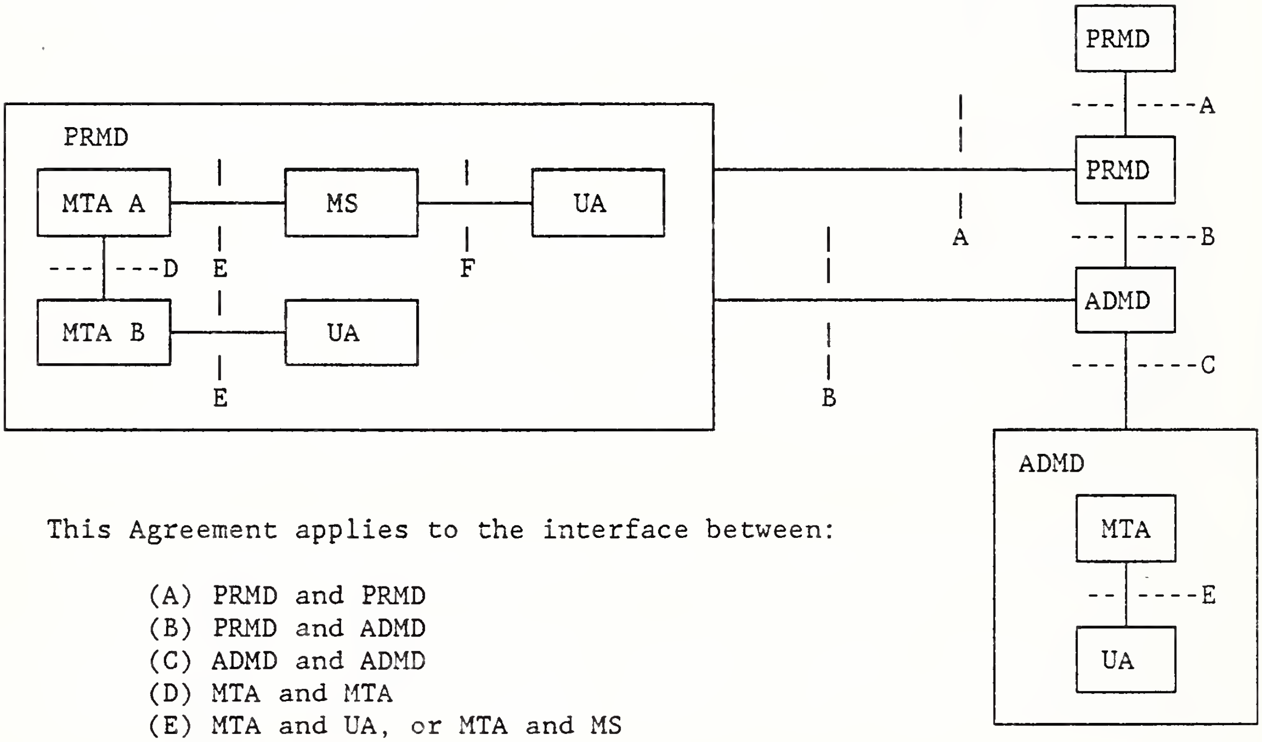
There are two conformance levels defined for the MT Kernel in these Agreements:

- A class 'A' MT Kernel implementation supports transfer (i.e., relaying) only;

- A class 'B' MT kernel implementation supports submission, delivery and transfer (including relaying).
  [**Note**: This does not imply support for the P3 protocol]

In addition, the UAs and MTAs will require access to directory and routing services. Except insofar as they must be capable of providing addressing and routing as described in Section 8.9, these services and associated protocols are not described by this Agreement (see Chapter 11 - Directory Services).

PRMD = Private Management Domain
ADMD = Administration Management Domain



This Agreement applies to the interface between:

(A) PRMD and PRMD
(B) PRMD and ADMD
(C) ADMD and ADMD
(D) MTA and MTA
(E) MTA and UA, or MTA and MS
(F) UA and MS

Figure 8.2 Scenario Definition

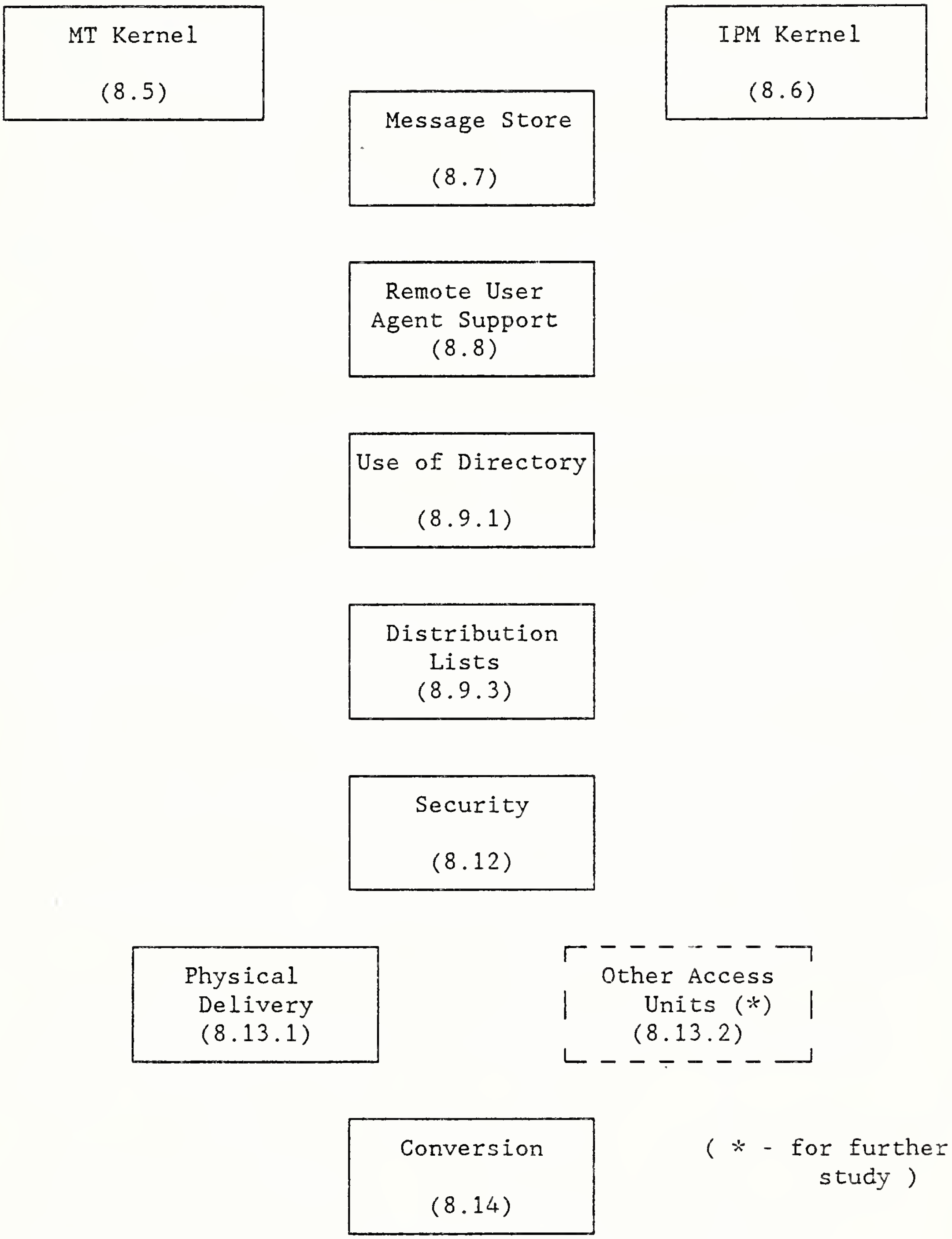


Figure 8.3 MHS Functional Groups

8.3 STATUS

This version of the Implementation Agreements for Message Handling Systems (MHS) is under development. It is based on the CCITT X.400(1988) Recommendations and ISO MOTIS (10021, parts 1-7) standards.

It is intended that the Stable Implementation Agreements will initially include an Agreement which specifies a minimal 1988-based MHS implementation and support for Message Stores and remote User Agents, and which addresses interworking with 1984-based implementations. The remaining features specified in the 1988 standards will be covered in subsequent versions of this Agreement.

8.4 ERRATA

No Errata to Stable material at this time.

8.5 MT KERNEL

8.5.1 Introduction

This section specifies the requirements for a minimal 1988-based MTS implementation (i.e., MTA) which is capable of interworking with 1984-based MTAs. The 'base' MT Service specified in this section does not include:

- Message Store (see 8.7)
- Remote UA (see 8.8)
- Use of Directory Services (see 8.9.1)
- Distribution Lists (see 8.9.3)
- Security (see 8.12)
- Interworking with Physical Delivery systems or Specialized Access (see 8.13)
- Conversion (see 8.14)

Such a minimal 1988-based MTA will have the following capabilities in order to achieve interworking with 1984-based MTAs and to facilitate migration to full 1988 operation:

- It will be protocol-conformant to 1988 P1;

- It will downgrade 1988 P1 to 1984 P1 when relaying to 1984-based MTAs, as specified in Annex B of X.419 (see 8.5.5);

- It will relay the contents of 1988 P1 messages unchanged, even when relaying to 1984-based MTAs;

- It will support both 'normal mode' and 'X.410 mode' protocol stacks (i.e., as required by ISO and CCITT respectively).

8.5.2 Elements of Service

This section specifies the requirements for support of MT Elements of Service by an MTA conforming to the MT Kernel Functional Group of this Agreement.

The classification scheme for support of Elements of Service is as follows:

Mandatory (M) - the Element of Service must be supported and made available to the service user;

Optional (O) - the Element of Service may be supported, but is not required for conformance to this Agreement;

Not Defined/Not Applicable (-) - the Element of Service is not defined by this Agreement or is otherwise not applicable in the particular context;

To Be Determined (*) - the support classification for the Element of Service has yet to be determined (temporary).

The requirements for support of MT Elements of Service for origination and reception and (where relevant) relaying are distinguished. Elements of Service which are new in the 1988 MHS standards are indicated as (1988).

An MTA must support those Basic MT Elements of Service and MT Optional User Facilities defined in clause 19 of X.400(1988) as listed and qualified in Tables 8.1 and 8.2 below.

Table 8.1 MT Kernel : Basic MT Elements of Service

Element of Service	Origination	Reception	Relaying
Access Management	M[1]	M[1]	-
Content Type Indication	M	M	-
Converted Indication	M	M	M
Delivery Time Stamp Indication	-	M	-
Message Identification	M	M	-
Non-delivery Notification	M	M	M
Original Encoded Information Types Indication	M	M	-
Submission Time Stamp Indication	M	M	-
User/UA Capabilities Registration (1988)	-	M[1]	-

Notes: 1) A local matter in the case of co-located UA/MTA and/or MS/MTA configurations.

Table 8.2 MT Kernel : MT Service Optional User Facilities

Element of Service	Origination	Reception	Relaying
Alternate Recipient Allowed	M	M[2]	-
Alternate Recipient Assignment	-	O[2]	-
Conversion Prohibition	M	M	M
Conversion Prohibition in Case of Loss of Information (1988)	O	O	O
Deferred Delivery	M[3]	O	O
Deferred Delivery Cancellation	M	-	-
Delivery Notification	M	M	-
Disclosure of Other Recipients	M	M	M
DL Expansion History Indication	-	M	-
Explicit Conversion	O	O	O
Grade of Delivery Selection	M	M	M
Hold for Delivery	-	M[1]	-
Implicit Conversion	O	O	O
Latest Delivery Designation (1988)	O	O	O
Multi Destination Delivery	M	M	M
Originator Requested Alternate Recipient (1988)	O	O	-
Prevention of Non-delivery Notification	O	-	-
Probe	M	M	M
Redirection Disallowed by Originator (1988)	O	O	-
Redirection of Incoming Messages (1988)	-	O	-
Requested Delivery Method (1988)	M	M	-
Restricted Delivery (1988)	-	O	-
Return of Content	O	O	O

Notes:

1) A local matter in the case of co-located UA/MTA and/or MS/MTA configurations.

2) If Alternate Recipient Assignment is supported on reception, then support of Alternate Recipient Allowed is Mandatory on reception; otherwise, support of Alternate Recipient Allowed is Optional on reception.

3) Support of this MT Element of Service is Mandatory for conformance reasons, but may be performed as a local matter to the originating MTA.

8.5.3 MTS Transfer Protocol (P1)

The requirements for support of MTS Transfer Protocol (P1) elements are detailed in Section 8.17.1 (Appendix A).

Support of MTS Transfer Protocol application contexts by an MTA is classified as follows:

mts-transfer-protocol-1984	Mandatory
mts-transfer-protocol	Mandatory
mts-transfer	Mandatory

Use of the underlying services to support these application contexts is specified in Section 8.15.

8.5.4 Intra Domain Considerations

To be determined.

> **Note:** It has yet to be determined whether this section will be confined to intra-PRMD issues only or will cover all intra-domain implementation considerations.

8.5.5 Downgrading Issues

An MTA conforming to this Agreement will downgrade 1988 P1 to 1984 P1 when relaying to 1984-based MTAs, as specified in Annex B of X.419 with the following additional requirements:

- Supplementary Information - will need to be truncated if it exceeds the pragmatic constraint identified in Version 2 of these Agreements, and

- Internal Trace Information - to be determined.

8.5.6 Error Handling

8.6 IPM KERNEL

8.6.1 Introduction

This section specifies the requirements for a minimal 1988-based IPMS implementation (i.e., UA) which is capable of interworking with 1984-based UAs. The 'base' IPM Service specified in this section does not include:

- Message Store (see 8.7)
- Remote UA (see 8.8)
- Use of Directory Services (see 8.9.1)
- Distribution Lists (see 8.9.3)
- Security (see 8.12)

- Interworking with Physical Delivery systems or Specialized Access (see 8.13)

Such a minimal 1988-based UA will have the following capabilities in order to achieve interworking with 1984-based UAs and to facilitate migration to full 1988 operation:

- It will continue to support content type P2 (encoded as integer 2) on origination and reception;

- It will support receipt of P2 (encoded as integer 22);

- It may originate P2 (22), but the guidelines specified in clause 20.2 of X.420(1988) are to be followed, i.e. the content type shall be encoded as P2 (2) unless 1988 P2 protocol elements are present.

8.6.2 Elements of Service

This section specifies the requirements for support of IPM Elements of Service by a UA conforming to the IPM Kernel Functional Group of this Agreement.

The classification scheme for support of Elements of Service is as defined in Section 8.5.2.

The requirements for support of IPM Elements of Service for origination and reception are distinguished. Elements of Service which are new in the 1988 MHS standards are indicated as (1988).

A UA must support those Basic IPM Elements of Service and IPM Optional User Facilities defined in Clause 19 of X.400(1988) as listed and qualified in Tables 8.3 and 8.4 below.

Table 8.3 IPM Kernel : Basic IPM Elements of Service

Element of Service	Origination	Reception
Access Management	M[1]	M[1]
Content Type Indication	M	M
Converted Indication	-	M
Delivery Time Stamp Indication	-	M
IP-message Identification	M	M
Message Identification	M	M
Non-delivery Notification	M	-
Original Encoded Information Types Indication	M	M
Submission Time Stamp Indication	M	M
Typed Body	M	M
User/UA Capabilities Registration (1988)	-	M[1]

Notes: 1) In the case of a co-located UA/MTA, the method and extent to which this Element of Service is provided is a local matter; it is not necessarily testable in the absence of support for the P3 protocol.

Table 8.4 IPM Kernel : IPM Service Optional User Facilities

Element of Service	Origination	Reception
Alternate Recipient Allowed	O	O/M[2]
Alternate Recipient Assignment	-	O[2]
Authorizing Users Indication	O	M
Auto-forwarded Indication	O	M
Blind Copy Recipient Indication	O	M
Body Part Encryption Indication	O	M
Conversion Prohibition	M	M
Conversion Prohibition in Case of Loss of Information (1988)	O	O
Cross Referencing Indication	O	M
Deferred Delivery	M	-
Deferred Delivery Cancellation	O	-
Delivery Notification	M	-
Disclosure of Other Recipients	O	M
DL Expansion History Indication	-	M
Expiry Date Indication	O	M
Explicit Conversion	O	-
Forwarded IP-message Indication	O	M
Grade of Delivery Selection	M	M
Hold for Delivery	-	O/M[1]
Implicit Conversion	-	O
Importance Indication	O	M
Incomplete Copy Indication (1988)	O	O
Language Indication (1988)	O	M
Latest Delivery Designation (1988)	O	-
Multi Destination Delivery	M	-
Multi-part Body	O	M
Non-receipt Notification Request	O	M
Obsoleting Indication	O	M
Originator Indication	M	M
Originator Requested Alternate Recipient (1988)	O	-
Prevention of Non-delivery Notification	O	-
Primary and Copy Recipients Indication	M	M
Probe	O	-
Receipt Notification Request Indication	O	O
Redirection Disallowed by Originator (1988)	O	-
Redirection of Incoming Messages (1988)	-	O
Reply Request Indication	O	M
Replying IP-message Indication	M	M
Requested Delivery Method (1988)	M	-
Restricted Delivery (1988)	-	O
Return of Content	O	-
Sensitivity Indication	O	M
Subject Indication	M	M

Notes: 1) Mandatory in the case of a remote UA (where the MTA does not support MSs) or a remote UA/MS.

2) If Alternate Recipient Assignment is supported on reception, then support of Alternate Recipient Allowed is Mandatory on reception; otherwise, support of Alternate Recipient Allowed is Optional on reception.

8.6.3 Interpersonal Messaging Protocol (P2)

The requirements for support of Interpersonal Messaging Protocol (P2) elements are detailed in Section 8.17.2 (Appendix A).

8.6.4 Body Part Support

This section specifies the requirements for support of IPM body part types by a UA conforming to this Agreement.

The classification scheme for support of IPM body part types is as defined in Section 8.5.2.

The requirements for support of IPM body part types for origination and reception are distinguished. Body part types which are new in the 1988 MHS standards are indicated as (1988).

A UA must support those IPM body part types defined in Annex C of X.420(1988) as listed and qualified in Table 8.5 below. If an implementation supports a particular body part type for reception, it should also be able to support that body part type for reception if it is part of a forwarded message.

Any body part type that is supported on reception must be supported as integer encoding and as object identifier (externally-defined) encoding.

All body parts with integer-encoded identifiers in the range 0 up to and including 16K-1 are legal and must be relayed. Body part integer-encoded identifiers corresponding to X.121 country codes should be interpreted as described in Note 2 of Figure 8.4. These privately-defined body part types are specified as an interim measure to provide backward compatibility with 1984 MHS implementations. For interworking between UAs based on the 1988 (or later) MHS standards, it is strongly recommended that the externally-defined body part be used instead.

Table 8.5 IPM Kernel : Body Part Types

Body Part Type		Origination	Reception
IA5Text		M	M
Voice		O	O
G3Facsimile		O	O
G4Class1	(TIF0)	O	O
Teletex		O	O
Videotex		O	O
Encrypted		O	O
Message	(ForwardedIPMessage)	O	M
MixedMode	(TIF1)	O	O
BilaterallyDefined	(Unidentified)	O	O
NationallyDefined		O	O
ExternallyDefined (1988)		O	M[1]
PrivatelyDefined	(see Figure 8.4)	O	O

Notes: 1) Any body part type that is supported on reception as integer encoding must also be supported as object identifier encoding.

```
BodyPart ::=  CHOICE {
        ia5-text            [0] IA5TextBodyPart,
                             .
        externally-defined [15] ExternallyDefinedBodyPart,
                             .
                          [234] UKBodyParts,
                             .
                          [310] USABodyParts,
                             .         }
```

Where UKBodyParts and USABodyParts are defined as:

```
SEQUENCE {BodyPartNumber, ANY}
BodyPartNumber ::= INTEGER
```

Note 1) The undefined bit in P1 EncodedInformationTypes must be set when a message contains a privately defined body part. Each UA that expects such body parts should include undefined in the set of deliverable EncodedInformationTypes it registers with the MTA.

Note 2) Body part numbers are interpreted relative to the body part type in which they are used. NIST registers body part numbers for privately-defined formats within the United States.

Figure 8.4 Privately-Defined Body Parts

8.6.5 Error Handling

8.7 MESSAGE STORE

8.7.1 Introduction

This section specifies Agreements for implementation of the Message Store (MS) Functional Group. The MS is responsible for accepting delivery of messages on behalf of a single end-user, and retaining the messages until the end-user's UA is able to retrieve them. Message submission and administration services are provided via "pass-through" to the MTS. Figure 8.5 illustrates the logical relationship of the MS to the UA and MTS.

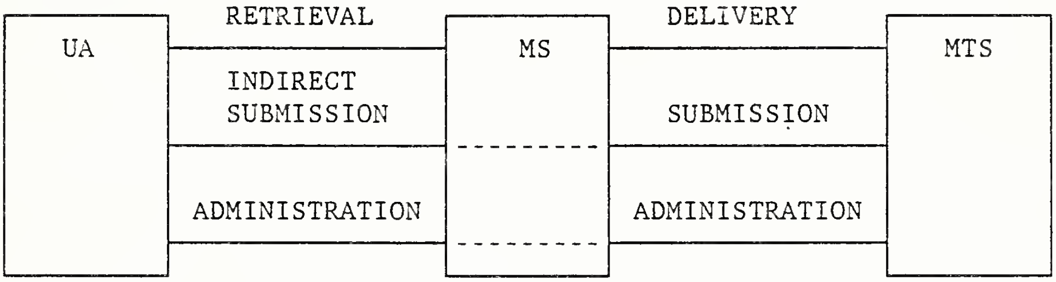


Figure 8.5 Message Store Model

The Agreements in this section specify the Message Store's use of the retrieval, delivery, and administration services. Agreements on submission services are specified in Section 5.8, which describes support for the remote UA. Agreements on the use of message management services defined in ISO 10021-5 are for future study.

The goal of the Agreements in this section is to define the minimal set of features which are necessary to provide useful Message Store services, independent of the MTA implementation version (i.e., 1984 or 1988).

8.7.2 Scope

The scope of the Agreements in this section is depicted in Figure 8.6 below, and is confined to the services and protocols between the boundaries shown (marked with asterisks). Requirements for the UA and MTA are addressed only to the extent that they affect the Message Store and remote User Agent services and protocols. This reflects the additional services required at the UA to support MS access and at the MTA to support a remote MS.

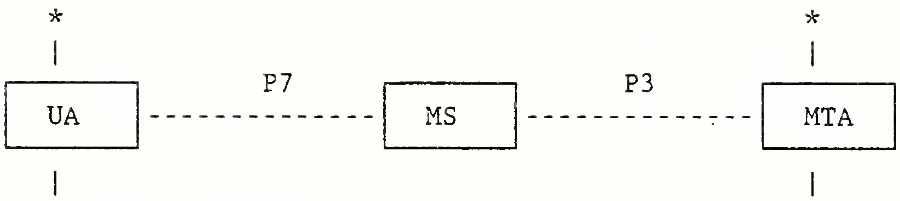


Figure 8.6 Scope of Message Store Agreements

The UA, MS and MTA configuration is not restricted; any of these components may be co-located, although they are depicted as logically separate. In the case of a co-located UA and MS, a proprietary interface may be used instead of P7. In the case of a co-located MS and MTA, a proprietary interface may be used instead of P3.

8.7.3 Elements of Service

This section specifies the requirements for support of Elements of Service to provide a Message Store conforming to the Message Store Functional Group of this Agreement.

The classification scheme for support of Elements of Service is as defined in Section 8.5.2.

Support for Elements of Service is specified both for the Message Store itself and for the User Agent.

Table 8.6 Message Store : Elements of Service

Element of Service	UA	MS
Stored Message Deletion	M	M
Stored Message Fetching	M	M
Stored Message Listing	M	M
Stored Message Summary	M	M
Stored Message Alert	O	O
Stored Message Auto Forward	O	O

8.7.4 Attribute Types

Requirements for support of attributes used in the Message Store are defined in clause 11 of X.413(1988) and in Annex C of X.420(1988).

8.7.5 Pragmatic Constraints for Attribute Types

To be determined.

8.7.6 Implementation of the MS with 1984 Systems

While the Message Store is part of the 1988 MHS standards, implementation of MS services with a 1984 MTA is possible. In order to interoperate with other 1984 MHS systems, implementations with this configuration must adhere to the following guidelines:

- o The UA must generate 1984 P2 PDUs;
- o The UA must identify the content protocol as integer 2 to the MS;
- o The MS must be co-located with the MTA unless 1988 P3 support is provided on the 1984 MTA as well.

To meet these guidelines, the UA may be implemented as follows:

- o The UA could conform to X.420(1984), with 1988 UA extensions for utilizing the MS services;
- o The UA could be a 1988 UA with restrictions on protocol elements generated and by identifying the content type as integer 2 rather than 22. No 1988-specific elements should be generated.

Details of the interface between the 1988 MS and the 1984 MTA when co-located are beyond the scope of these Agreements.

8.7.7 MS Access Protocol (P7)

The requirements for support of MS Access Protocol (P7) elements by an MS and a remote MS-user are detailed in Section 8.17.4 (Appendix A).

The requirements for support of MS Access Protocol (P7) application contexts by an MS and an MS-user are as specified in clauses 6.1 and 10.1 of X.419(1988) (ISO 10021-6) with the additional requirement that an MS-user must at least support the ms-access application context, as follows:

	MS	MS-user
ms-access	Mandatory	Mandatory
ms-reliable-access	Optional	Optional

Use of the underlying services to support these application contexts is specified in Section 8.15.

8.7.8 MTS Access Protocol (P3)

The requirements for support of MTS Access Protocol (P3) elements by an MTA and an MS where the MS is not co-located with the MTA are detailed in Section 8.17.3 (Appendix A).

The requirements for support of MTS Access Protocol (P3) application contexts by an MTA and an MS in such a scenario are as specified in clauses 6.1 and 10.1 of X.419(1988) (ISO 10021-6) with the additional requirement that a remote MS must at least support the mts-access and mts-forced-access application contexts, as follows:

	MTA	MS
mts-access	Mandatory	Mandatory
mts-forced-access	Mandatory	Mandatory
mts-reliable-access	Optional	Optional
mts-forced-reliable-access	Optional	Optional

Use of the underlying services to support these application contexts is specified in Section 8.15.

8.7.9 Error Handling

8.8 REMOTE USER AGENT SUPPORT

8.8.1 Introduction

This section specifies Agreements for implementation of the Remote User Agent Functional Group, i.e. for support of an IPM UA that is not co-located with its MTA. Support of other classes of UA is for further study.

The goal of the Agreements in this section is to define the minimal set of features which are necessary to provide useful remote User Agent services, independent of the MTA implementation version (i.e., 1984 or 1988).

8.8.2 Scope

The scope of the Agreements in this section is depicted in Figure 8.7, and is confined to the services and protocols between the boundaries shown (marked with asterisks). Requirements for the UA and MTA are addressed only to the extent that they affect the remote User Agent services and protocols. Access to a Message Store by a remote User Agent is covered in Section 8.7.

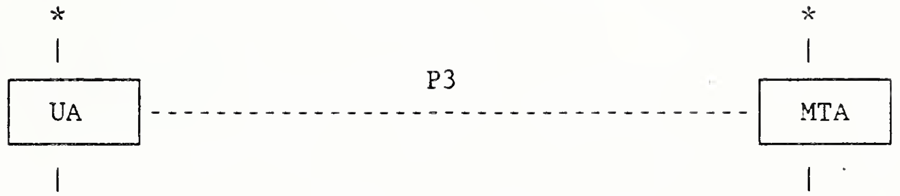


Figure 8.7 Scope of Remote User Agent Agreements

8.8.3 Elements of Service

This section specifies the requirements for support of Elements of Service for conformance to the Remote User Agent Functional Group of this Agreement.

The classification scheme for support of Elements of Service is as defined in Section 8.5.2.

Support for Elements of Service is specified both for the MT Service and for the IPM Service, and is in addition to the support requirements specified in Sections 8.5 and 8.6 if this Functional Group is supported.

Table 8.7 Remote User Agent Support: MT Elements of Service

Element of Service	Origination	Reception
Access Management	M	M
Hold for Delivery	-	M
User Capabilities Registration	-	M

Table 8.8 Remote User Agent Support: IPM Elements of Service

Element of Service	Origination	Reception
Access Management	M	M
Hold for Delivery	-	M
User Capabilities Registration	-	M

8.8.4 MTS Access Protocol (P3)

The requirements for support of MTS Access Protocol (P3) elements by an MTA and an MTS-user (whether UA or UA/MS) where the MTS-user is not co-located with the MTA are detailed in Section 8.17.3 (Appendix A).

The requirements for support of MTS Access Protocol (P3) application contexts by an MTA and an MTS-user in such a scenario are as specified in clauses 6.1 and 10.1 of X.419(1988) (ISO 10021-6) with the additional requirement that a remote MTS-user must at least support the mts-access and mts-forced-access application contexts, as follows:

	MTA	MTS-user
mts-access	Mandatory	Mandatory
mts-forced-access	Mandatory	Mandatory
mts-reliable-access	Optional	Optional
mts-forced-reliable-access	Optional	Optional

Use of the underlying services to support these application contexts is specified in Section 8.15.

8.8.5 Error Handling

8.9 NAMING, ADDRESSING & ROUTING

8.9.1 MHS Use of Directory

8.9.1.1 Introduction

The MHS standards recognize 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 a directory name;

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

o Determine whether the directory 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 possibilities;

o Allow users to scan directory entries;

o Allow users to scan directory entries selectively;

- Return the capabilities of the entity referred to by the directory or O/R name;
- Provide maintenance functions to keep the directory up-to-date.

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

This section identifies and specifies the Use of Directory Functional Group, which is intended to cover all issues relating to the use by an MHS implementation of Directory Services which conform to the Agreements in Chapter 11.

8.9.1.2 Elements of Service

This section specifies the requirements for support of Elements of Service for conformance to the Use of Directory Functional Group of this Agreement.

The classification scheme for support of Elements of Service is as defined in Section 8.5.2.

Support for Elements of Service is specified both for the MT Service and for the IPM Service.

Table 8.9 Use of Directory : MT Elements of Service

Element of Service	Origination	Reception
Designation of Recipient by Directory Name	M	-

Table 8.10 Use of Directory : IPM Elements of Service

Element of Service	Origination	Reception
Designation of Recipient by Directory Name	M	-

8.9.2 Use of Names & Addresses

It is recognized that these Agreements enable a wide variety of naming and addressing attributes wherein each PRMD may adopt particular routing schemes within its domain.

With the exception of the intra-domain connection agreements, these agreements make no attempt to recommend a standard practice for electronic mail addressing.

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

- manual directories
- on-line directories
- ORName address specifications
- ORName address translation.

Further, each PRMD may adopt naming and addressing schemes wherein the user view may take a form entirely different from the ORName attributes specified in this Agreement, 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 this Agreement. However, mapping between one PRMD user form to another PRMD user form, via the MHS ORName attributes of this Agreement, is outside the scope of this Agreement.

8.9.3 Distribution Lists

8.9.3.1 Introduction

This section identifies and specifies the Distribution Lists Functional Group, which is intended to cover all issues relating to the support of distribution lists by an MHS implementation.

8.9.3.2 Elements of Service

This section specifies the requirements for support of Elements of Service for conformance to the Distribution Lists Functional Group of this Agreement.

The classification scheme for support of Elements of Service is as defined in Section 8.5.2.

Support for Elements of Service is specified both for the MT Service and for the IPM Service.

Table 8.11 Distribution Lists : MT Elements of Service

Element of Service	Origination	Reception
DL Expansion History Indication	*	*
DL Expansion Prohibited	*	*
Use of Distribution List	*	*

Table 8.12 Distribution Lists : IPM Elements of Service

Element of Service	Origination	Reception
DL Expansion History Indication	*	*
DL Expansion Prohibited	*	*
Use of Distribution List	*	*

8.10 CONFORMANCE

8.10.1 Introduction

8.10.2 Configuration Options

MHS implementations may be configured as any single or multiple occurrence or combination of MTA, MS and UA, as illustrated in Figure 8.8. It is not intended to restrict the types of system that may be configured for conformance to these Agreements (although it is equally recognized that not all configuration types may be commercially viable).

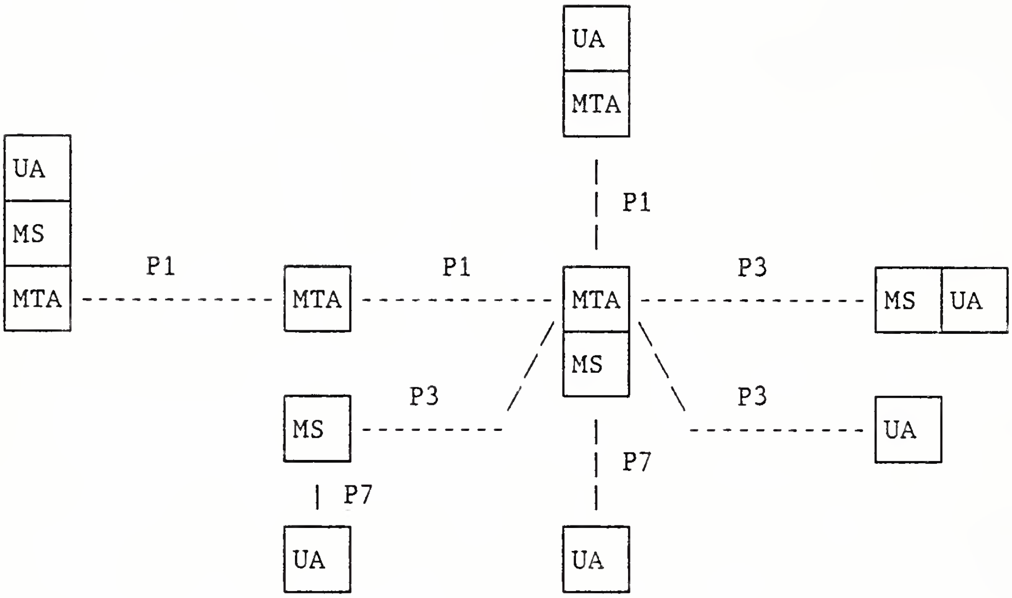


Figure 8.8 Configuration Options

8.10.3 Definition of Conformance

8.10.4 Conformance Requirements

8.11 MHS MANAGEMENT

8.12 MHS SECURITY

8.12.1 Introduction

This section identifies and specifies the MHS Security Functional Group, which is intended to cover all issues relating to provision of secure messaging and secure access management facilities by an MHS implementation.

8.12.2 Elements of Service

This section specifies the requirements for support of Elements of Service for conformance to the MHS Security Functional Group of this Agreement.

The classification scheme for support of Elements of Service is as defined in Section 8.5.2.

Support for Elements of Service is specified both for the MT Service and for the IPM Service (Note: All Elements of Service listed below are 1988).

Table 8.13 MHS Security : MT Elements of Service

Element of Service	Origination	Reception
Content Confidentiality	*	*
Content Integrity	*	*
Message Flow Confidentiality	*	*
Message Origin Authentication	*	*
Message Security Labelling	*	*
Message Sequence Integrity	*	*
Non-repudiation of Delivery	*	*
Non-repudiation of Origin	*	*
Non-repudiation of Submission	*	*
Probe Origin Authentication	*	*
Proof of Delivery	*	*
Proof of Submission	*	*
Report Origin Authentication	*	*
Secure Access Management	*	*

Table 8.14 MHS Security : IPM Elements of Service

Element of Service	Origination	Reception
Content Confidentiality	*	*
Content Integrity	*	*
Message Flow Confidentiality	*	*
Message Origin Authentication	*	*
Message Security Labelling	*	*
Message Sequence Integrity	*	*
Non-repudiation of Delivery	*	*
Non-repudiation of Origin	*	*
Non-repudiation of Submission	*	*
Probe Origin Authentication	*	*
Proof of Delivery	*	*
Proof of Submission	*	*
Report Origin Authentication	*	*
Secure Access Management	*	*

8.13 SPECIALIZED ACCESS

8.13.1 Physical Delivery

8.13.1.1 Introduction

This section identifies and specifies the Physical Delivery Functional Group, which is intended to cover all issues relating to access to physical delivery systems by an MHS implementation.

8.13.1.2 Elements of Service

This section specifies the requirements for support of Elements of Service for conformance to the Physical Delivery Functional Group of this Agreement.

The classification scheme for support of Elements of Service is as defined in Section 8.5.2.

Support for Elements of Service is specified both for the MT Service and for the IPM Service (Note: All Elements of Service listed below are 1988).

Table 8.15 Physical Delivery : MT Elements of Service

Element of Service	Origination	Reception
Additional Physical Rendition	*	*
Basic Physical Rendition	*	*
Counter Collection	*	*
Counter Collection with Advice	*	*
Delivery via Bureaufax Service	*	*
EMS (Express Mail Service)	*	*
Ordinary Mail	*	*
Physical Delivery Notification by MHS	*	*
Physical Delivery Notification by PDS	*	*
Physical Forwarding Allowed	*	*
Physical Forwarding Prohibited	*	*
Registered Mail	*	*
Registered Mail to Addressee in Person	*	*
Request for Forwarding Address	*	*
Special Delivery	*	*
Undeliverable Mail with Return of Physical Message	*	*

Table 8.16 Physical Delivery : IPM Elements of Service

Element of Service	Origination	Reception
Additional Physical Rendition	*	*
Basic Physical Rendition	*	*
Counter Collection	*	*
Counter Collection with Advice	*	*
Delivery via Bureaufax Service	*	*
EMS (Express Mail Service)	*	*
Ordinary Mail	*	*
Physical Delivery Notification by MHS	*	*
Physical Delivery Notification by PDS	*	*
Physical Forwarding Allowed	*	*
Physical Forwarding Prohibited	*	*
Registered Mail	*	*
Registered Mail to Addressee in Person	*	*
Request for Forwarding Address	*	*
Special Delivery	*	*
Undeliverable Mail with Return of Physical Message	*	*

8.13.2 Other Access Units

8.13.2.1 Facsimile Access Units

The possible development of Agreements in this area is for further study.

8.13.2.2 Telex Access Units

It is not currently intended to develop Agreements in this area.

8.13.2.3 Teletex Access Units

It is not currently intended to develop Agreements in this area.

8.14 CONVERSION

8.14.1 Introduction

This section identifies and specifies the Conversion Functional Group, which is intended to cover all issues relating to support of conversion facilities by an MHS implementation.

8.14.2 Elements of Service

This section specifies the requirements for support of Elements of Service for conformance to the Conversion Functional Group of this Agreement.

The classification scheme for support of Elements of Service is as defined in Section 8.5.2.

Support for Elements of Service is specified for the MT Service only, and is in addition to the support requirements specified in Section 8.5 if this Functional Group is supported. Support for IPM Elements of Service for access to conversion facilities is as specified in Section 8.6.

Table 8.17 Conversion : MT Elements of Service

Element of Service	Origination	Reception
Conversion Prohibition in Case of Loss of Information (1988)	*	*
Explicit Conversion	*	*
Implicit Conversion	*	*

8.15 USE OF UNDERLYING LAYERS

8.15.1 MTS Transfer Protocol (P1)

The P1 protocol is mapped onto the Reliable Transfer Service Element (RTSE) either in X.410-1984 mode or in normal mode, as specified in Section 8.5.3. In X.410-1984 mode, the RTSE makes direct use of the services provided by the Session Layer, as specified in Chapter 5 (Upper Layers) of the Stable Implementation Agreements. In normal mode, the RTSE makes use of the services provided by the Association Control Service Element (ACSE) and Presentation Layer, as defined in Chapter 5 (Upper Layers) of these Agreements.

8.15.2 MTS Access Protocol (P3) and MS Access Protocol (P7)

The P3 and P7 protocols make use of the services provided by the Remote Operations Service Element (ROSE), Association Control Service Element (ACSE), Presentation Layer, and, optionally, the Reliable Transfer Service Element (RTSE), as defined in Chapter 5 (Upper Layers) of these Agreements. It is recommended that RTSE be used for recovery purposes when the implementation uses a Transport Class other than 4.

8.16 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.

8.16.1 MPDU Encoding

8.16.2 Contents

8.16.3 Envelope

8.16.4 Reports

8.17 APPENDIX A: MHS PROTOCOL SPECIFICATIONS

The following tables specify the requirements for support of MHS protocol elements for conformance to these Implementation Agreements. It should be noted that the tables specify minimum support for conformance to the relevant Kernel functional groups and where appropriate also specify enhanced support requirements where one or more further functional groups are claimed. All element support is subject to further review and may be upgraded in later versions of these Agreements.

The protocol support classification scheme used in this version of the Agreements is described below, and is very similar to that employed in the existing Stable Implementation Agreements for X.400(1984) and as currently used in the equivalent European work on MHS in EWOS/ETSI. However, it should be noted that the scheme is currently under review both within the NIST X.400 SIG and in the EWOS/ETSI MHS groups and is likely to be revised for later versions of these Agreements.

The classification of support for a protocol element specifies the requirements for implementations conforming to these Implementation Agreements to be able to generate, receive and process that protocol element, as appropriate. The classification of support for each protocol element is relative to that for its containing element. Where subelements within a containing element are not listed, then their support classification shall be assumed to be that of the containing element. Where the range of values to be supported for an element is not specified, then all values defined in the base standard shall be supported.

Mandatory (M) - implementations conforming to these Agreements shall generate this element in all information objects in which, according to the base standards, it shall occur; receiving implementations shall process this element appropriately, and shall regard its absence as a protocol violation unless otherwise specified in the base standards;

Generatable (G) - implementations conforming to these Agreements shall be able to generate this protocol element, but it does not necessarily have to be present in every information object generated (conditions for generation are as specified in the base standards or as otherwise indicated in these Agreements); receiving implementations shall process this element appropriately if it is present;

Supported (H) - implementations conforming to these Agreements may optionally be capable of generating this protocol element, but are not required to do so; receiving implementations shall, however, process this element appropriately if it is present;

Unsupported (X) - implementations conforming to these Agreements may optionally be capable of generating this protocol element, but should not expect any specific action or processing by a receiving implementation except as required to observe criticality indication and any such use is outside the scope of these Agreements; receiving implementations conforming to these Agreements are similarly not required to be able to process this element other than to observe any criticality indication, but must at least be able to relay the semantics of this element where appropriate; the absence of this element should not be assumed by a receiving implementation to convey any significance.

8.17.1 MTS Transfer Protocol (P1)

	Support		Comments/References
	Class B	Class A	
	MT Kernel	MT Kernel	
MTS-APDU			
message	G	H	
envelope	M	M	MessageTransferEnvelope
content	M	M	See P2 - else undefined
probe	G	H	ProbeTransferEnvelope
report	G	G	
envelope	M	M	ReportTransferEnvelope
content	M	M	ReportTransferContent
MessageTransferEnvelope			
message-identifier	M	M	MTSIdentifier
originator-name	M	M	ORName
original-encoded-information-types	G	X	EncodedInformationTypes
content-type	M	M	
built-in	G	X	
external	H	X	
content-identifier	H	X	
priority	G	H	All values
per-message-indicators	G	H	
disclosure-of-recipients	H	H	
implicit-conversion-prohibited	G	H	
alternate-recipient-allowed	G	X	

content-return-request	X	X	
deferred-delivery-time	X	X	
per-domain-bilateral-information	X	X	PerDomainBilateralInfo
trace-information	M	M	TraceInformation
extensions	G	G	ExtensionField
recipient-reassignment-prohibited	X	X	
dl-expansion-prohibited	H	H	
conversion-with-loss-prohibited	H	H	
latest-delivery-time	X	X	See X.411, 14.1.1 note 2
originator-return-address	X	X	
originator-certificate	X	X	
content-confidentiality-algorithm-identifier	X	X	
message-origin-authentication-check	X	X	
message-security-label	X	X	
content-correlator	X	X	
dl-expansion-history	H	H	DLExpansionHistory
internal-trace-information	G	G	InternalTraceInfo
PerRecipientMessageTransfer Fields	M	M	
recipient-name	M	M	ORName
originally-specified-recipient-number	M	M	
per-recipient-indicators	M	M	
explicit-conversion	X	X	
extensions	H	H	ExtensionField
originator-requested-alternate-recipient	X	X	
requested-delivery-method	G	H	
physical-forwarding-prohibited	X	X	
physical-forwarding-address-request	X	X	
physical-delivery-modes	X	X	
registered-mail-type	X	X	
recipient-number-for-advice	X	X	
physical-rendition-attributes	X	X	
physical-delivery-report-request	X	X	
message-token	X	X	
content-integrity-check	X	X	
proof-of-delivery-request	X	X	
redirection-history	H	H	

ProbeTransferEnvelope			
probe-identifier	M	M	MTSIdentifier
originator-name	M	M	ORName
original-encoded-information-types	G	X	EncodedInformationTypes
content-type	M	M	
built-in	G	X	
external	H	X	
content-identifier	H	X	
content-length	G	X	
per-message-indicators	G	H	
disclosure-of-recipients	X	X	
implicit-conversion-prohibited	G	H	
alternate-recipient-allowed	G	X	
content-return-request	X	X	
per-domain-bilateral-information	X	X	PerDomainBilateralInfo
trace-information	M	M	TraceInformation
extensions	G	G	ExtensionField
recipient-reassignment-prohibited	X	X	
dl-expansion-prohibited	H	H	
conversion-with-loss-prohibited	X	X	
originator-certificate	X	X	
message-security-label	X	X	
content-correlator	X	X	
probe-origin-authentication-check	X	X	
dl-expansion-history	H	H	DLExpansionHistory
internal-trace-information	G	G	InternalTraceInfo
PerRecipientProbeTransferFields	M	M	
recipient-name	M	M	ORName
originally-specified-recipient-number	M	M	
per-recipient-indicators	M	M	
explicit-conversion	X	X	
extensions	H	H	ExtensionField
originator-requested-alternate-recipient	X	X	
requested-delivery-method	G	H	
physical-rendition-attributes	X	X	
redirection-history	H	H	

ReportTransferEnvelope			
report-identifier	M	M	MTSIdentifier
report-destination-name	M	M	ORName
trace-information	M	M	TraceInformation
extensions	G	G	ExtensionField
message-security-label	X	X	
originator-and-DL-expansion-history	G	X	OriginatorAndDL ExpansionHistory
reporting-DL-name	X	X	
reporting-MTA-certificate	X	X	
report-origin-authentication-check	X	X	
internal-trace-information	G	G	InternalTraceInfo
ReportTransferContent			
subject-identifier	M	M	MTSIdentifier
subject-intermediate-trace-information	G	G	TraceInformation
original-encoded-information-types	G	G	EncodedInformationTypes
content-type	G	G	
built-in	G	G	
external	G	G	
content-identifier	G	G	
returned-content	H	X	
additional-information	X	X	
extensions	H	H	ExtensionField
content-correlator	H	H	
PerRecipientReportTransferFields	M	M	
actual-recipient-name	M	M	ORName
originally-specified-recipient-number	M	M	
per-recipient-indicators	M	M	
last-trace-information	M	M	
arrival-time	M	M	
converted-encoded-information-types	G	G	EncodedInformationTypes
report	M	M	
delivery	G	X	
message-delivery-time	M	M	
type-of-MTS-user	G	X	All values = H
non-delivery	G	G	
non-delivery-reason-code	M	M	
non-delivery-diagnostic-code	H	H	
originally-intended-recipient-name	G	G	ORName
supplementary-information	X	X	
extensions	G	G	ExtensionField
redirection-history	G	G	RedirectionHistory
physical-forwarding-address	X	X	
recipient-certificate	X	X	
proof-of-delivery	X	X	

Common Data Types

EncodedInformationTypes			
built-in-encoded-information-types	M	M	
non-basic-parameters	X	X	
external-encoded-information-types	H	H	
MTSIdentifier			
global-domain-identifier	M	M	GlobalDomainIdentifier
local-identifier	M	M	
PerDomainBilateralInfo			
country-name	M	M	
administration-domain-name	M	M	DomainName
private-domain-identifier	G	G	DomainName (only encoded as SEQ if both present)
bilateral-information	M	M	
TraceInformation			
TraceInformationElement	G	G	
global-domain-identifier	M	M	GlobalDomainIdentifier
domain-supplied-information	M	M	
arrival-time	M	M	
routing-action	M	M	
relayed	G	G	
rerouted	H	H	
attempted-domain	H	H	GlobalDomainIdentifier
deferred-time	H	H	
converted-encoded-information-types	H	H	EncodedInformationTypes
other-actions	H	H	
redirected	H	H	
dl-operation	H	H	
ExtensionField			
type	M	M	
criticality	H	H	
for-submission	X	X	
for-transfer	G	G	
for-delivery	G	G	
value	M	M	
DLExpansionHistory			
DLExpansion	M	M	
ORAddressAndOptionalDirectoryName	M	M	ORName
dl-expansion-time	M	M	
InternalTraceInfo			

InternalTraceInformationElement	M	M	
global-domain-identifier	M	M	GlobalDomainIdentifier
mta-name	M	M	
mta-supplied-information	M	M	
arrival-time	M	M	
routing-action	M	M	
relayed	G	G	
rerouted	H	H	
attempted			
mta	H	H	
domain	H	H	GlobalDomainIdentifier
deferred-time	H	H	
other-actions	H	H	
redirected	H	H	
dl-operation	H	H	
OriginatorAndDLExpansionHistory			
originator-or-dl-name	M	M	
origination-or-expansion-time	M	M	
RedirectionHistory			
Redirection	M	M	
intended-recipient-name	M	M	
ORAddressAndOptionalDirectory Name	M	M	ORName
redirection-time	M	M	
redirection-reason	M	M	
ORName			
address	M		
standard-attributes	M		
country-name	G		CountryName
administration-domain-name	G		DomainName
network-address	G		
terminal-identifier	G		
private-domain-name	G		DomainName
organization-name	G		
numeric-user-identifier	G		
personal-name	G		
surname	M		
given-name	G		
initials	G		
generation-qualifier	G		
organizational-unit-names	G		
OrganizationUnitName	G		
domain-defined-attributes	G		
DomainDefinedAttribute	G		
type	M		
value	M		

extension-attributes	H	ExtensionAttribute
common-name	H	
teletex-common-name	H	
teletex-organization-name	H	
teletex-personal-name	H	
teletex-organizational-unit-names	H	
teletex-domain-defined-attributes	H	
pds-name	H	
physical-delivery-country-name	H	
postal-code	H	
physical-delivery-office-name	H	
physical-delivery-office-number	H	
extension-OR-address-components	H	
physical-delivery-personal-name	H	
physical-delivery-organization-name	H	
extension-physical-delivery-address-components	H	
unformatted-postal-address	H	
street-address	H	
post-office-box-address	H	
poste-restante-address	H	
unique-postal-name	H	
local-postal-attributes	H	
extended-network-address	H	
terminal-type	H	
directory-name	X	
ExtensionAttribute		
extension-attribute-type	M	
extension-attribute-value	M	
GlobalDomainIdentifier		
country-name	M	CountryName
administration-domain-name	M	DomainName
private-domain-identifier	G	DomainName
CountryName		
x121-dcc-code	H	
iso-3166-alpha2-code	G	
DomainName		
numeric	H	
printable	G	

8.17.2 Interpersonal Messaging Protocol (P2)

	Support		Comments/References
	Minimum	Enhanced	
InformationObject			
ipm	G		IPM
ipn	G		IPN
IPM			
heading	M		
this-IPM	M		IPMIdentifier
originator	G		ORDescriptor
authorizing-users	H		RecipientSpecifier
primary-recipients	G		RecipientSpecifier
copy-recipients	G		RecipientSpecifier
blind-copy-recipients	H		RecipientSpecifier
replied-to-IPM	G		IPMIdentifier
obsoleted-IPMs	H		IPMIdentifier
related-IPMs	H		IPMIdentifier
subject	G		See Note 1
expiry-time	H		
reply-time	H		
reply-recipients	H		ORDescriptor
importance	H		
sensitivity	H		
auto-forwarded	H		
extensions	H		HeadingExtension
incomplete-copy	X		
languages	H		
body	M		BodyPart
IPN			
subject-ipm	M		
ipn-originator	G		ORDescriptor
ipm-preferred-recipient	G		ORDescriptor
conversion-eits	H		EncodedInformationTypes
non-receipt-fields	G		
non-receipt-reason	M		
discard-reason	G		
auto-forward-comment	H		
returned-ipm	X		See Note 2
receipt-fields	H		
receipt-time	M		
acknowledgment-mode	H		
suppl-receipt-info	X		
HeadingExtension			
type	M		
value	M		
IPMIdentifier			
user	H		

user-relative-identifier	M	
ORDescriptor		
formal-name	H	ORName - see Note 3
free-form-name	H	
telephone-number	H	
RecipientSpecifier		
recipient	M	ORDescriptor
notification-requests	H	
reply-requested	H	
BodyPart		
ia5-text	G	
parameters	M	
repertoire	H	Support of ITA2 is for for further study
data	M	
voice	X	
parameters	M	
data	M	
g3-facsimile	X	
parameters	M	
number-of-pages	H	
non-basic-parameters	H	
data	M	
g4-class1	X	
teletex	X	
parameters	M	
number-of-pages	X	
telex-compatible	X	
non-basic-parameters	X	
data	M	
videotex	X	
parameters	M	
syntax	H	
data	M	
encrypted	X	
parameters	M	
data	M	
message	H	
parameters	M	
delivery-time	H	
delivery-envelope	H	See P3 OtherMessage DeliveryFields
data	M	
mixed-mode	X	
bilaterally-defined	X	
nationally-defined	X	
externally-defined	H	
parameters	M	
data	M	

Notes:

1) The ability to generate the maximum size subject is not required.

2) May only be included if specifically requested by the originator.

3) The ORName should be specified wherever possible.

8.17.3 MTS Access Protocol (P3)

8.17.4 MS Access Protocol (P7)

8.18 APPENDIX B: RECOMMENDED PRACTICES

It is not necessary to follow the recommended practices when claiming conformance to this Agreement.

8.18.1 EDI

8.19 APPENDIX C: LIST OF ASN.1 OBJECT IDENTIFIERS

8.19.1 Content Types

8.19.2 Body Part Types

9. STABLE FTAM PHASE 2

Below are recorded Errata to Base Stable FTAM Phase 2 Agreements in Version 2, Edition 1 Stable Document. These Base Stable FTAM Phase 2 Agreements are given in the aligned section in the Stable Implementation Agreements Document, NIST SP 500-162.

ERRATA

NO. OF ERRATA	TYPE	REFERENCED DOCUMENT	SECTION	DESCRIPTION
CP 3/89-1	Editorial	NIST-SP 500-162	Notes to Table 9.1, 9.10, 9.10.1, 9.13, Appendix A, Part 3	Reference to IA5 replaced ISO 646.IRV
CP 3/89-2	Editorial		9.10.1.2	Text including Note 1 clarified, ISO 6429 version 1988 referenced
CP 3/89-3	Editorial		9.5 bullet 7	Agreements on use of AETitle clarified
CP 3/89-4	Editorial		Def. NBS-8, Def. NBS Ordered Flat	FADU identity 'Traversal Number' replaced by 'Node Number'
CP 3/89-5	Editorial		9.3	Note to point to Dec '87 version of FTAM Phase 2
CP 3/89-6	Editorial		Def. NBS-6, Def. NBS-7, Def. NBS-8,	Definition of length parameters completed

NO. OF ERRATA	TYPE	REFERENCED DOCUMENT	SECTION	DESCRIPTION
CP #/89-7	Technical	NIST-SP 500-162	Table 9.7, Note 5	Unconstrained service class outside the scope of the Implementation Profiles
CP 3/89-9	Technical		new section 9.17.9	Parameters filesize, fadu-number may be encoded with up to 8 contents octets
CP 3/89-10	Technical		5.11.1.1.1	Pointer to new 9.17.9 included in 5.11.1.1.1 of Upper Layer Chapter

10. ISO FILE TRANSFER, ACCESS AND MANAGEMENT PHASE 3

Editor's Note: The "NBS" designation shall remain in effect for document types, abstract syntaxes, and constraint sets defined in all FTAM agreements up to 1/1/89. After 1/1/89, any new functionality will reference the "NIST" designation. The editor of this document will add a note explaining the change.

10.1 INTRODUCTION

This section contains Implementors Agreements based on ISO 8571 File Transfer, Access and Management. These Agreements define enhancements to the Stable FTAM Implementation Agreements for OSI Protocols, Version 1, Edition 1, December 1987 (FTAM Phase 2 Agreements, NBS 500-150), including all their subsequent Errata changes as specified in Version 2, Edition 1 (NIST Special Publication 500-162).

Therefore it is assumed that the reader is familiar both with the contents of the base standard ISO 8571 and its underlying layers, and also with the above-mentioned NIST FTAM Phase 2 specifications.

Phase 2 Agreements define six Implementation Profiles which are T1, T2, T3, A1, A2, and M1. In order to avoid ambiguity when referring to these Implementation Profiles the above designations will apply only to Phase 2 functionality, references to Phase 3 enhanced Implementation Profiles will be by the addition of a'.3', i.e. T1.3, T2.3, T3.3, A1.3, A2.3, and M1.3.

10.2 SCOPE AND FIELD OF APPLICATION

These Phase 3 Agreements specify additional functionality to the FTAM Phase 2 Agreements. These additional functions include:

- Further specifications of document types,
- Specification for Restart Data Transfer and Recovery functional units,
- Specification of FADU Locking functional unit, and
- More details on Access Control and Concurrency Control.

All Phase 2 systems are upward compatible to a Phase 3 system and can therefore interwork with it, if the additional functions are negotiated out (e.g. use of Recovery) or not used for the interconnection (e.g. additional features for document types).

10.3 STATUS

These FTAM Phase 3 Agreements are at working paper status, reflecting the results from the FTAM SIG Meeting, March 14-16, 1988. They are expected to become stable by June 1989.

10.4 ERRATA

10.5 CONFORMANCE

In addition to the specific requirements specified in the following subsections, conformance to this Phase 3 specification requires

- conformance to ISO 8571

- conformance to Phase 2 FTAM

10.5.1 Conformance for Access Profiles

The access Profiles A1.3 and A2.3 do not include the requirement for transferring files using the File Transfer service class.

10.6 ASSUMPTIONS

FTAM Phase 3 Agreements specify additional functionality to the Implementation Profiles T1, T2, T3, A1, A2, and M1 as defined in the FTAM Phase 2 Agreements. So all definitions and requirements for these Implementation Profiles apply also to the Phase 3 Agreements.

10.7 FILESTORE AGREEMENTS

10.7.1 Document Types

In addition to the Phase 2 Document Type Agreements the document types FTAM-4 (see ISO 8571-2, Annex-B) and NBS-10, NBS-11, NBS-12 (see Appendix B) are defined for optional support.

Table 10.1 gives the support levels for all document types with respect to the Implementation Profiles.

For FTAM-1, FTAM-2, FTAM-3 and FTAM-4 the supported parameter values for <universal class number> and <string significance> respectively are listed. Other values are outside the scope of these Agreements. No restriction or minimum requirement is defined for the <maximum string length> parameter of these document types.

Table 10.1 Implementation Profiles and Document Types
(a) FTAM-1 Through FTAM-4

Implementation Profile	Document Type	Universal Class Number	String Significance
T1.3, T2.3, T3.3, A1.3, A2.3	FTAM-1	Graphic String (25)	'variable' 'fixed'
		VisibleString (26)	'variable' 'fixed'
		GeneralString (27)	'not-significant'
		IA5String (22)	'not-significant'
T2.3, T3.3, A1.3, A2.3	FTAM-2	GraphicString (25)	'not-significant'
		VisibleString (26)	'not-significant'
		[GeneralString (27)]	'not-significant'
		[IA5String (22)]	'not-significant'
T1.3, T2.3, T.3.3, A1.3, A2.3	FTAM-3	-	'not-significant'
[T2.3], [T3.3], [A1.3], [A2.3]	FTAM-4	-	'not-significant'

Table 10.1 Implementation Profiles and Document Types
(b) NBS-6 Through NBS-11

Implementation Profile	Document Type	Universal Class Number	String Significance
[T2.3], T3.3, [A1.3], A2.3	NBS-6		
[T2.3], T3.3, [A1.3], A2.3	NBS-7		
[T2.3], T3.3 [A1.3], A2.3	NBS-8		
[T1.3], [T2.3], [T3.3]	NBS-9		
[T2.3], [T3.3] [A1.3], [A2.3]	NBS-10		
[T2.3], [T3.3] [A1.3], [A2.3]	NBS-11		

Table 10.1 Implementation Profiles and Document Types
(c) NBS-12

Implementation Profile	Document Type	Universal Class Number	Character-Set Reg. Numbers			String Significance
			C0	G0	G1	
[T2.3], [T3.3] [A1.3], [A2.3]	NBS-12 See Note 3	IA5String	(parameter absent)			'variable' 'fixed'
		GraphicString	(parameter absent)			'variable' 'fixed'
		GraphicString	-	6	100	'variable' 'fixed'
		VisibleString	(parameter absent)			'variable' 'fixed'
		GeneralString	(parameter absent)			'variable' 'fixed'
		GeneralString	1	6	100	'variable' 'fixed'

Notes: 1. Brackets around a Profile designator or a parameter value indicate that the respective document type or parameter value is optionally supported in this Implementation Profile.

2. The support level for document types in Implementation Profile M1.3 depends on the T- or A-Implementation Profile, in conjunction with which M1.3 is implemented.

3.

Registration Number	Content	Escape Sequence
1	CO Set of ISO 646	ESC 2/1 4/0
6	ISO 646, USA Version - X3.4 : 1968 (Left-hand part of ISO 8859-1)	ESC 2/8 4/2
100	Right-hand part of Latin Alphabet No. 1, ISO 8859-1, ECMA-94	ESC 2/13 4/1

10.7.2 FADU Identities

In addition to the Phase 2 FADU Identify Agreements the following is specified:

> For the document type NBS-11 used in conjunction with the Transfer service class or the Transfer and Management service class, the support of the FADU identities of 'current', 'next' and 'previous' is outside the scope of these Agreements.

10.7.3 Access Control Attribute

It is the implementor's choice which combinations of fields in an access control element are supported. The ACE combination should be stated in the PICS.

10.8 PROTOCOL AGREEMENTS

10.8.1 Functional Units

For FTAM Phase 3 implementations Recovery and Restart Data Transfer are optionally supported.

FADU locking is optionally supported for Implementation Profiles A1.3 and A2.3.

10.8.2 Implementation Information Parameter

In addition to the Agreements as specified for FTAM Phase 2, Section 9.12 (NIST SP 500-162), the following value is defined

NBS-Phase 3.

10.8.3 F-Check

In order to maximize interoperability, implementations of FTAM service providers should not restrict the amount of data transmitted between successive F-CHECK requests to a single quantity. Variations in the amount of data transmitted between checkpoints may be required to accommodate differences in real end systems supporting FTAM Virtual Filestores and/or in the communications media underlying FTAM associations. It is required that all FTAM implementations are able to receive at least one PSDU between checkpoints.

10.8.4 Error Recovery

Procedures for Class I, II and III errors are defined and supported for FTAM Phase 3 implementations. It is the implementor's choice whether to handle class I errors using F-RESTART PDUs or whether to use the class II error procedure.

10.8.4.1 Docket Handling

When a class III error occurs, the length of time a docket is maintained is determined by the local system. Recovery from a class III error is only possible as long as both end systems maintain the docket.

It is also a local decision how many dockets can be maintained simultaneously.

10.8.4.2 Parameters for Error Recovery

- The semantics of the <FTAM quality of service> parameter is as defined in ISO 8571, including the local knowledge of FERPM.

- No minimum requirement for the <checkpoint window> parameter of the checkpoint size is defined.

- For the <recovery mode> parameter of F-OPEN all three values 'none', 'at-start-of-file' and 'at-any-active-checkpoint' are supported. If recovery mode 'at-start-of-file' is negotiated, no F-CHECK shall be issued. When recovering at the start of the file, the <recovery point> value of 0 shall be used.

Note: This Agreement is because of a deficiency of the standard. All other behaviors would lead to unpredictable results, because text and state tables in 8571-4 are ambiguous.

- It is required that Responders implementing the Restart-data-transfer or the Recovery functional unit must be able to negotiate <recovery mode> parameter to a value other than 'none'.

- For the <diagnostic> parameter of F-CANCEL/F-U-ABORT/F-P-ABORT the term <suggested delay> is supported. The Basic FERPM should wait at least the amount of time as given by the <suggested delay> term before attempting to recover.

10.8.5 Concurrency Control

10.8.5.1 Concurrency Control to whole file

The <concurrency control> parameters of F-SELECT, F-CREATE and F-OPEN with or without the <access control> attribute of Security Group are supported for Initiators and optionally supported for Responders.

If supported by a Responder, details of their possible usage is a local matter and shall be specified in the PICS.

Default values for concurrency control are as specified for FTAM Phase 2 Agreements.

No minimum requirement is defined for <concurrency control> parameter values.

For a first accessor either the specified concurrency locks or the default values are assigned. For a subsequent accessor the access to a file is granted only if this concurrency control requirement, as specified in this concurrency control parameter or given by the default values, can be met. Otherwise the subsequent request shall be rejected.

10.8.5.2 FADU Locking

FADU locking functional unit and the respective <FADU lock> parameters are optionally supported for the Implementation Profiles A1.3 and A2.3.

It is understood that ISO 8571-4 Clause 18.4 also applies to FADU locks; that means that as long as a docket is maintained, FADU locks locking any FADUs recorded in that docket should be maintained.

10.8.6 Create Password

The <create password> parameter for an implementation acting as an Initiator is supported. This parameter is optionally supported for an implementation acting as a Responder.

10.9 Range of Values for Integer-Type Parameter

In addition to the parameters specified for FTAM Phase 2 under the same heading, the parameters

```
F-RECOVER request
          bulk-transfer-number
NBS-AS3
     NBS-Node-Name
          starting-fadu
          fadu-count
```

may be encoded so that the length of its contents octets is no more than eight octets.

10.10 APPENDIX A:

PROFILES REQUIREMENTS LIST FOR FTAM PHASE 3

Full Phase 3 PICS Proforma to be included here.

10.11 APPENDIX B: DOCUMENT TYPES

NBS-10 Random Binary Access Document Type

1. **Entry Number: NBS-10**
2. **Information objects**

Table 10.2 Information objects in NBS-10

document type name	{iso identified-organization icd(9999) organization-code(1) document type(5) random-binary(10)} "NBS-10 random binary access file"
abstract syntax names: a) name of asname1 b) name of asname2 c) name of asname3	{iso identified-organization icd(9999) organization-code(1) abstract- syntax(2) nbs-random-binary(4)} "NBS random binary access file abstract syntax" {iso standard 8571 abstract-syntax(2) ftam-fadu (2)} "FTAM FADU" {iso identified-organization icd(9999) organization-code(1) abstract- syntax(2) nbs-node-name(3)} "NBS random access node name abstract syntax"
transfer syntax names:	{joint-iso-ccitt asn1(1) basic-encoding (1)} "Basic encoding of a single ASN.1 type"
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-random-access(2)} "NBS random access constraint set"
File contents:	Datatype1 ::= a single octet Datatype2 ::= Node-Name --The type to be used for Node-Name is defined in --ISO 8571-FADU --The only Choice for Node-Name is user-coded Datatype3 ::= NBS-Node-Name --As defined by the NBS Node Name Abstract Syntax

3. Scope and field of application

This document type defines the contents of a file for storage, for transfer and access by 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 zero, one or more file access data units each of which consists of one data element. The data element is made up of one octet. The order of these elements is significant. The semantics of the data elements is not specified by this document type.

The document structure takes the form allowed by the FTAM hierarchical file model as constrained by the NBS random access constraint set. The definition for FTAM hierarchical file model appears in 8571-2.

There are no size or length limitations imposed by this definition.

8. Abstract syntactic structure

The abstract syntactic structure of the document is a series of octets.

9. Definition of transfer

9.1. Datatype definition

The presentation data value used for transfer is an ASN.1 OCTET STRING.

Datatype 2 is used to specify the FADU-Identity of "single-name" in the FTAM PDUs specifying FADU-Identity, where "single-name" is defined as an EXTERNAL. The EXTERNAL is defined as Node-Name in the FTAM FADU abstract syntax. The use of Datatype3 is defined in "NBS random access constraint set".

Datatype3 specifies the "user-coded" form of the Node-Name in the FTAM FADU abstract syntax, where "user-coded" is defined as an EXTERNAL. That EXTERNAL is defined by Datatype3. The use of Datatype3 is defined in "NBS random access constraint set".

9.2 Presentation data values

The document is transmitted as a series of presentation data values. Each presentation data value shall consist of the "data" from one or more FADUs concatenated together. The result is one value of the ASN.1 data type OCTET STRING. The "fadu_count" field supplied in the Node-Name specifies the number of FADUs to transfer during a Read operation. The requested FADUs may be transferred as one or more presentation data values.

All values are transmitted in the same (but any) presentation context established to support the abstract syntax name "asname1" declared in Table 10.2.

> Note: Specific carrier standards may impose additional constraints on the presentation context to be used, when the above permits a choice.

Boundaries between P-DATA primitives and between presentation data values are chosen locally by the sending entity at the time of transmission. The boundaries are not preserved when the file is stored and they carry no semantics of the document type. Receivers which support this document type shall accept a document with any of the permitted transfer options.

9.3 Sequence of presentation data values

The sequence of presentation data values is the same as the sequence of Data Units within the file.

10. Transfer syntax

An implementation supporting these document types shall support the transfer syntax generation rules named in Table 10.2 for all presentation data values transferred.

Implementations may optionally support other transfer syntaxes.

11. ASE specific specifications

11.1 Simplification and relaxation

The document type NBS-10 may be simplified to the document type FTAM-3. The resultant document contains the same sequence of data values as would result from accessing the file as an NBS-10 file.

11.2 The READ operation

A READ operation may be applied to a range of FADUs via the FADU Identity of "NodeName". The "starting-fadu" part of the node name specifies the node number of the first FADU; the "fadu-count" specifies the node of consecutive FADUs to be transferred.

A READ operation applied to a range of FADUs that spans beyond the end of file is valid. All available data in the range is transferred. An informative diagnostic (5005) is returned on the F-Data-End Request indicating that the end of file was reached and a portion of the request was satisfied.

11.3 The REPLACE operation

When the REPLACE operation is applied to the root FADU of an NBS-10 document, the transferred data shall be any NBS-10 document.

The REPLACE operation applied to a FADU identity of "traversal number" is used to replace a series of FADUs, starting at the specified position in the file, by the new FADUs being transferred. The number of replaced FADUs is determined by the number of transferred FADUs.

If the replacement spans beyond the end of the existing file, then the additional FADUs are inserted at the end of the file.

11.4 The INSERT operation

When the INSERT operation is applied at the end of file, the transferred data shall be a series of FADUs which would be generated by reading any NBS-10 document type in access context UA.

1. **Entry Number: NBS-11**
2. **Information objects**

Table 10.3 Information Objects in NBS-11

document type name	{iso identified-organization icd (9999) organization-code (1) document type (5) indexed-file-with-unique-keys (11)} "NBS-11 FTAM indexed file with unique keys"

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, NBS-8 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-unique-names (4)} "FTAM ordered flat constraint set with unique names"
file contents: Datatype1 ::= PrimType -- as defined in Annex 9 A, Part 3 of NIST SP 500-162 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 with unique names (see Table 10.3). These definitions appear in ISO 8571-2.

The following additional requirements are specified for the use of the ordered flat constraint set with unique names:

- The FADU identity 'node number' is not required for conformant implementations

- 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 NIST 500-162. 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.

The string-length field of Parameter 1 specifies the length of the value in octets for the INTEGER, BIT STRING and OCTET STRING types. For character-type data elements, the string-length indicates the actual number of characters from the specified character set, not including any escape sequences or overhead from the character encoding.

For floating point numbers, finite form, length-1 and length-2 specify the length in bits of mantissa and exponent, respectively. The length-1 and length-2 values are irrelevant for the other choices of floating point numbers.

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 of NIST 500-162.

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 10.4.

Table 10.4 Datatypes for keys

Key Type	Minimum Range (octets)	Order
ASN.1 INTEGER	(1-2)	increasing numeric value
ASN.1 IA5String	(0-16)	lexical order
ASN.1 GraphicString	(0-16)	lexical order
ASN.1 GeneralString	(0-16)	lexical order
ASN.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 <KeyPosition> parameter.
KeyPosition = 0 implies the key is not part of the data
KeyPosition > 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 10.3, where the PrimType in the datatype is given by the NBS-AS1 definition; or

b) Datatype2 defined in Table 10.3, which is 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 10.2 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-11 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-11 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-11 in key order.

A document of type NBS-11 may be accessed as a document of type NBS-8 (allowed only when reading the file) by specifying document type NBS-8 in the <contents type> parameter in the <F-OPEN REQUEST>.

11.2 Access context selection

A document of type NBS-11 may be accessed in any one of the access contexts defined in the FTAM ordered flat constraint set with unique names. 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-11 document with the same parameter values in access context FA.

A transferred FADU whose name duplicates that of an already existing FADU will cause the <INSERT> operation to fail. The failure shall be signalled by issuing an F-CANGEL Request with a corresponding diagnostic.

11.4 The EXTEND operation

This operation is excluded for the use with this document type.

11.5 The REPLACE operation

When the <REPLACE> operation is applied with FADU Identity 'begin', a transferred FADU whose name duplicates that of a previously transferred FADU will cause the <REPLACE> operation to fail. The failure shall be signalled by issuing an F-CANCEL Request with a corresponding diagnostic.

1. **Entry Number: NBS-12**

2. **Information objects**

Table 10.5 Information objects in NBS-12

<table>
<tr><td>document type name</td><td>{iso identified-organization icd (9999) organization-code (1) document-type (5) simple-text-file (12)
"NBS-12 FTAM simple text file"</td></tr>
<tr><td>abstract syntax names:
a) name for asname1</td><td>{iso identified-organization icd (9999) organization-code (1) abstract-syntax (2) nbs-simple-text (5)}
"NBS simple text abstract syntax"</td></tr>
<tr><td>b) name for asname2</td><td>{iso standard 8571 abstract-syntax(2) ftam-fadu (2)}
"FTAM FADU"</td></tr>
<tr><td>transfer syntax names:</td><td>{joint-iso-ccitt asn1 (1) basic-encoding (1)}
"Basic Encoding of a single ASN.1 type"</td></tr>
<tr><td colspan="2">Parameter Syntax
PARAMETERS ::= SEQUENCE{

universal-class-number [0] IMPLICIT INTEGER,
maximum-string-length [1] IMPLICIT INTEGER,
string-significance [2] IMPLICIT INTEGER {variable (0), fixed (1)},
character-set [3] IMPLICIT OctetString OPTIONAL}</td></tr>
<tr><td>file model</td><td>{iso standard 8571 file-model (3) hierarchical (1)}
"FTAM hierarchical file model"</td></tr>
<tr><td>constraint set</td><td>{iso standard 8571 constraint-set (4) sequential flat(2)}
"FTAM sequential flat constraint set"</td></tr>
<tr><td colspan="2">File contents
Datatype1 ::= NBS Text
--as defined in the NBS Simple Text
--Abstract Syntax registration entry

Datatype2 ::= Node-Descriptor-Data-Element</td></tr>
</table>

3. Scope and field of application

The document type defines the contents of a file for storage, and for transfer and access by 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. In addition, it makes use of the terms character string, graphics character, and format effector as defined in document type registration entry "FTAM-2" in ISO 8571-2.

6. Abbreviations

FTAM	File Transfer, Access and Management

7. Document semantics

This document consists of zero, one or more file access data units, each of which consists of one character string. The order of each of these elements is significant. The semantics of the character strings is not specified by this document type.

The document structure takes any of the forms allowed by the FTAM hierarchical file model as constrained by the sequential flat constraint set. These definitions appear in ISO 8571-2. As additional constraints FADU identity will be limited to the following values:

a) 'begin' and 'end' when using the Transfer or Transfer and Management service classes.

b) 'begin', 'end', 'first', and 'next' when using the Access service class.

Each character string consists of characters from the character set defined by the ASN.1 (ISO 8824) character set type whose universal class number is given by the "universal-class-number" parameter and by the escape sequences contained in the optional "character-set" parameter. If the character set type allows explicit escape sequences, the "character-set" parameter, if present, contains escape sequences which designate and invoke specific character sets. If the "character-set" parameter is not present, character sets are assumed to be designated and invoked as specified in Table 2 in ISO 8825. Character strings shall not contain escape sequences.

ISO 8824 Information Processing Systems-Open Systems Interconnection-Specification of Abstract Syntax Notation 1 (ASN.1).

ISO 8825 Information Processing Systems-Open Systems Interconnection-Basic Encoding Rules for Abstract Syntax Notation One (ASN.1).

ISO 6429 Information Processing-ISO 7-bit and 8-bit coded character sets-Additional control functions for character imaging devices.

There are no size or length limitations imposed by this definition, except those specified here. Each character string is of a length determined by the number of characters given by the "maximum-string-length" parameter.

> **Note:** The length restriction refers to the number of characters from the applicable character set, not to the number of octets in the encoding, nor to the line length in any rendition of the document, where these are different.

The exact significance of the character strings is determined by the "string-significance" parameter. If its value is "variable", the length of the character strings is less than or equal to the length given. If the value is "fixed", the length of each character string is exactly equal to the length given.

If the document is interpreted on a character imaging device (outside the scope of ISO 8571), the interpretation depends on the character set in use.

a) If the character set contains format effectors, they shall be interpreted as defined in ISO 6429; end of string and end of file access data unit are given no formatting significance, and do not contribute to the document semantics;

b) If the character set does not contain format effectors, the end of each character string is interpreted as implying carriage return and line feed formatting actions in any rendition. The end of file access data unit is given no formatting significance beyond that attached to the end of the string in it.

Editor's Note: ISO documents referenced above are:

> ISO 8824-Information Processing Systems-Open Systems Interconnection-Specification of Abstract Syntax Notation One (ASN.1)
>
> ISO 8825-Information Processing Systems-Open Systems Interconnection-Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)

ISO 6429-Information Processing, ISO 7-Bit and 8-Bit Coded Character Sets-Additional Control Functions for Character Imaging Devices

8.Abstract syntactic structure

The abstract syntactic structure of the document is a hierarchically structured file as defined in the ASN.1 modules ISO8571-FADU and ISO 8571 CONTENTS in ISO 8571, in which each of the file contents data elements has the abstract syntactic structure of "NBS Simple Text."

9. Definition of transfer

9.1 Datatype definitions

The file consists of data values which are of either

a) Datatypel defined in Table 10.5, the ASN.1 datatype declared as "NBS-Text" in the NBS Simple Text Abstract Syntax definition. The choice in "NBS-Text" is determined by the universal-class-number parameter; or

b) Datatype2 defined in Table 10.5, the ASN.1 datatype declared as "Data-Element" in the ASN.1 module ISO 8571-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 "Datatypel", carrying one of the character strings of the document. Each character shall be transmitted using one of the character sets identified by the universal-class-number parameter. All values are transmitted in the same (but any) presentation context established to support the abstract syntax name "asnamel" declared in Table 10.5.

b) one value of the ASN.1 datatype "Datatype2". All values are transmitted in the same (but any) presentation context established to support the abstract syntax name "asname2" declared in Table 10.5.

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 P-DATA primitives are chosen locally by the sending at the time of transmission, and carry no semantics of the document type. Receivers which support this document type sahll accept a document with any of the permitted transfer options.

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 character strings 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 these document types shall support the transfer syntax generation rules named in Table 10.5 for all presentation data values transferred.

11. ASE specific specifications

11.1 Simplification and relaxation

11.1.1 Simplification to FTAM-1

This simplification loses information.

The document type NBS-12 may be accessed as a document type FTAM-1. The resultant document contains the same sequence of data values as would result from accessing the structured text file in access context UA. That is, only the presentation data values in the abstract syntax "asname1" are present. If the "character-set" parameter was present before the simplification, its contents will be added to the beginning of each string.

Note: The boundary between file access data units remains a boundary between strings, but any special significance given to it is lost.

11.1.2 Relaxation to FTAM-2

The document type NBS-12 may be relaxed to the document type FTAM-2. If the "character-set" parameter was present before the relaxation, its contents will be added to the beginning of each string.

11.1.3 Character set relaxation

This operation loses explicit information in the document type identification.

A document of type NBS-12 may be relaxed to a different document of type NBS-12 with

- a different "universal-class-number" parameter value,
- a different "character-set" parameter value,
- different values for both of these parameters, or
- no "character-set" parameter value,

if the resultant document type permits all characters from the original document type. If this relaxation involves including format effectors and none were present before the simplification, the characters "carriage return" and "line-feed" shall be added to the end of each string.

Note: If the characters "carriage return" and "line feed" are not part of the format effectors, the formatting action may be represented by "newline", or some other implementation specific choice if there is no representation of "newline" defined.

11.1.4 String length relaxation

This operation loses explicit information in the document type identification.

A document of type NBS-12 may be relaxed to another document type NBS-12 with a larger "maximum-string-length" parameter.

11.2 Access context selection

A document of type NBS-12 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-12 document type with the same parameter values in access context FA.

10.12 APPENDIX C: CONSTRAINT SETS

NBS Random Access Constraint Set

Table 10.6 - Basic Constraints in the NBS Random Access Constraint Set

Constraint set descriptor	NBS Random Access
Constraint set identifier	{iso identified-organization icd(9999) organization-code(1) constraint-set(4) nbs-random-access(2)}
Node names	All names shall be of the same type; the type of the names and an ordering of the names shall be defined when reference is made to the constraint set.
File access actions	Locate, Read, Insert, Erase, Replace
Qualified actions	None
Available access context	UA
Creation state	Root node without an associate data unit
Location after open	Root node
Beginning of file	Root node
End of file	No node selected
Read whole file	Read in access context UA with FADU-Identity of"begin"
Write whole file	Transfer a series of leaf FADUs which would be generated by reading the whole file in access context UA; Perform the transfer with an FADU Identity of "end" and a file access action of "insert", or with an FADU Identity of "begin" and an action of "replace", or with an FADU Identity of "node-number" and an action of "replace". Here "node number" identifies the first FADU in the preorder traversal sequence.

Table 10.7 - Identity Constraints in the NBS Random Access Constraint Set

Action	Begin	End	NodeName	Node Number
Locate				leaf
Read	whole		leaf	
Insert		leaf		
Erase	whole			leaf
Replace	whole			leaf

1. **Field of application**

The NBS Random Access constraint set applies to files which are structured into a sequence of individual FADUs and to which access may be made randomly by NodeName. The structuring of the file into individual FADUs is determined by the NodeName.

2. **Basic constraints**

The basic constraints in the NBS Random Access constraint set are given in Table 10.6.

3. **Structural constraints**

The root node shall not have an associated data unit; all children of the root node shall be leaf nodes and shall 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 the file, with FADU-Identity of "end"; the new node is inserted following all existing nodes in the file. The location following the insert is "end".

Erase: the erase action is 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. Erase with the FADU-Identity of "node number" means truncation of the file.

Replace whole file: the FADU-Identity is "begin" and the complete series of new FADU contents is sent.

Replace new leaves: the FADU-Identity is "node number" and the number of FADUs being replaced is given by the number of FADUs sent.

5. **Identity constraints**

The FADU-Identity associated with the file action shall be one of the identities: begin, end, Node Number and NodeName. The actions with which these identities can be used are given in Table 10.7.

10.13 APPENDIX D: ABSTRACT SYNTAXES

NBS Node Name Abstract Syntax

Abstract Syntax Name
{ iso identified-organization icd (9999) organization-code (1)
abstract-syntax (2) nbs-node-name (3) }

"NBS random access node name abstract syntax"

This is an abstract syntax for the user-coded Node-Name in the FTAM FADU abstract syntax.

```
            NBS-AS3 DEFINITIONS::=

            BEGIN

                NBS-Node-Name::== SEQUENCE

                    {   starting-fadu [0] IMPLICIT INTEGER,
                        fadu-count [1] IMPLICIT INTEGER }
                            --a "fadu-count" of 0 specifies the
--range of FADUs
                            --beginning at "starting-fadu" and      --
ending at "end of file"

            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"

NBS Random Binary Access File Abstract Syntax

Abstract Syntax Name
{ iso identified-organization icd (9999) organization-code (1)
abstract-syntax (2) nbs-random-binary (4) }

"NBS random binary access file abstract syntax"

This is an abstract syntax for the transfer of the file contents for NBS Random binary files.

```
            NBS-AS4 DEFINITIONS::=

            BEGIN
```

```
          NBS-Random Binary ::== OCTET STRING
              --contains one or more presentation data values
              --concatenated together.
              --Each presentation data value is defined as
              --Datatype1 in Table 10.2.

      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"
```

NBS Simple Text Abstract Syntax

Abstract Syntax Name

```
{iso identified-organization icd (9999) organization-code(1)
abstract-syntax (2) nbs-simple-text(5) }
"NBS simple text abstract syntax"

NBS-AS5 DEFINITIONS::==

BEGIN

NBS-Text::= CHOICE {

                IA5String,....--Universal Class 22
                GraphicString,--Universal Class 25
                VisibleString,--Universal Class 26
                GeneralString,--Universal Class 27

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"
```

11. DIRECTORIES

11.1 INTRODUCTION

Refer to Section 11.1 of Stable Agreements Version 2 Edition 1.

11.2 SCOPE AND FIELD OF APPLICATION

Refer to Section 11.2 of Stable Agreements Version 2 Edition 1.

11.3 STATUS

This version completed December, 1988. Section 11.14.3 was officially moved by the Plenary to be Stable.

11.4 USE OF DIRECTORIES

11.4.1 Introduction

(See Stable Document for current information.)

11.4.2 MHS

(TBD)

11.4.3 FTAM

(TBD)

11.5 DIRECTORY ASEs, APPLICATION CONTEXTS, AND PORTS

Refer to Section 11.5 of Stable Agreements Version 2 Edition 1.

11.6 SCHEMAS

Refer to Section 11.6 of Stable Agreements Version 2 Edition 1.

11.6.1 NAMING CONTEXTS

The root of a naming context must not be an alias entry.

11.7 CLASSIFICATION OF SUPPORT FOR ATTRIBUTE TYPES

Refer to Section 11.7 of Stable Agreements Version 2 Edition 1.

11.8 INTRODUCTION TO PRAGMATIC CONSTRAINTS

Refer to Section 11.8 of Stable Agreements Version 2 Edition 1.

11.9 GENERAL CONSTRAINTS

Refer to Section 11.9 of Stable Agreements Version 2 Edition 1.

11.10 CONSTRAINTS ON OPERATIONS

Refer to Section 11.10 of Stable Agreements Version 2 Edition 1.

11.11 CONSTRAINTS ON ATTRIBUTE TYPES

Refer to Section 11.11 of Stable Agreements Version 2 Edition 1.

11.11.1 Attribute Values

Integer Values

DSAs shall be required to "pass through" encoded integer attribute values of arbitrary length (e.g. when chaining a Directory operation). No Directory component (i.e. DUA or DSA) shall be deemed non-conformant if it encodes integer attribute values of arbitrary length.

Components of the Directory are required to support (for storage and processing), as a minimum, integer attribute values encoded in 4 octets.

11.12 CONFORMANCE

Refer to Section 11.12 of Stable Agreements Version 2 Edition 1.

11.13 DISTRIBUTED OPERATIONS

Refer to Section 11.13 of Stable Agreements Version 2 Edition 1.

11.14 UNDERLYING SERVICES

Refer to Section 11.14 of Stable Agreements Version 2 Edition 1.

11.15 ACCESS CONTROL

Refer to Section 11.15 of Stable Agreements Version 2 Edition 1.

11.16 TEST CONSIDERATIONS

Refer to Section 11.16 of Stable Agreements Version 2 Edition 1.

11.17 ERRORS

Refer to Section 11.17 of Stable Agreements Version 2 Edition 1.

11.18 DSA CHARACTERISTICS

(TBD)

11.19 APPENDIX A: MAINTENANCE OF ATTRIBUTE SYNTAXES

11.19.1 Introduction

Please refer to Appendix A from Stable Agreements Version 2 Edition 1.

11.19.2 General Rules

For description of general rule information, refer to the aligned Section 11.19.2 of the Stable Implementation Agreements.

The following rule is proposed to simplify the handling of attributes:

1) The T.61 string type shall be further constrained to contain no characters other than defined graphic characters and spaces. Character set restrictions shall be specified in Table 11.1.

Table 11.1: Charater Set Restrictions
Upper 4 bits of encoding (hex)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	X	X					X		X	X	X		X	X		
1	X	X							X	X				X		
2	X	X							X	X				X		
3	X	X							X	X				X		
4	X	X							X	X				X		
5	X	X							X	X				X	X	
6	X	X							X	X				X		
7	X	X							X	X				X		
8	X	X							X	X				X		
9	X	X							X	X	X	X		X		
A	X	X							X	X	X	X		X		
B	X	X						X	X	X				X		
C	X	X				X			X	X	X			X		
D	X	X						X	X	X	X			X		
E	X	X				X		X	X	X	X			X		
F	X	X						X	X	X	X			X		X

Notes: 1. Row headings give the lower 4 bits of the encoding in hexadecimal.

2. Entries marked X are illegal T.61 encodings.

Prohibition of the use of and support of recursive distinguished names is for further study.

11.19.3 Checking Algorithms

Please refer to Appendix A from Stable Agreements Version 2 Edition 1.

11.19.4 Matching Algorithms

Please refer to Appendix A from Stable Agreements Version 2 Edition 1.

11.20 APPENDIX B: GLOSSARY

Please refer to Appendix B from Stable Agreements Version 2 Edition 1.

11.21 APPENDIX C: REQUIREMENTS FOR DISTRIBUTED OPERATIONS

Please refer to Appendix C from Stable Agreements Version 2 Edition 1.

11.22 APPENDIX D: REGISTRATION AND USAGE OF OBJECT CLASSES

11.22.1 Introduction

This tutorial material is included because the SIG felt that it was useful clarification (of the Directory documents) to Implementors on matters that could not be deferred. However, implementors should be advised that the material is the subject of change/enhancement in the tandards and lies in an area of substantial instability.

The objective of the tutorial is to clarify how structure rules need to be related to object classes (whether or not a DSA polices structure rules), and the way in which DSAs can administrate entries in relation to the Object Classes which they support.

11.22.2 Primary and Secondary Object Classes

Object classes specify the nature and properties of entries, in terms of the attributes which they must (or may) possess, and also in terms of their possible positions in the DIT and the names that they may have.

Primary object classes define the nature and role of objects, and therefore of the corresponding Directory entries. A Primary object class will normally be associated with a structure rule. Thus, "Country", "Device", "Person" are Primary (although "Person" does not possess a structure rule).

Secondary object classes, by contrast, only qualify Primary object classes, by adding new mandatory or optional attributes. A Secondary Object Class will never be associated with a structure rule. "MHS-User", "Top", "Alias" are Secondary.

The "multiple inheritance" provisions of the Directory Documents enables any particular object (and associated entry) to be defined by zero or more Secondary Object Classes, and by one and just one Primary Object Class. (The rule specifying that there must be just one Primary object class prevents ambiguity in the source of the structure rules.)

Define an Object Class Component as that new information which a particular Object Class adds to the Object Classes of which it is a subset. The Object Class macro is what defines the Object Class Component.

Then, the following rules apply to the derivation of new Object Classes, in accordance with the Directory Documents.

A. Recursive Object Class definitions are forbidden (e.g. an object class may not have itself as a superset).

B. A new Primary Object Class can be derived by the use of superclasses comprising any set of Object Classes if its own Object Class Component defines any structure rules for the Object Class. This allows the derivation of a completely new class of object class, while making use of existing object class definitions.

C. A new Primary Object Class can also be derived by the use ofsuperclasses comprising a single Primary Object Class, and zero, one or more Secondary Object Classes, by inheriting the structure rules associated with the Primary Object Class. This allows the derivation of a related Object Class, and forbids the ambiguity in derivation ofstructure rules that would arise from having more than one Primary superclass.

D. Unregistered Object Classes (i.e. those to which no distinct object identifier is allocated) must always be Primary Object Classes derived in accordance with rule C. That is, the unregistered Object Class Component must not contain structure rules of its own. This prevents the use of unregistered Object Classes which do not obey the structure rules associated with other objects which share the same set of Object Class attribute values.

E. Secondary Object Classes can be derived by the use of superclasses comprising any set of Secondary Object Classes - there can be no structure rules associated with Secondary object Classes.

F. Entries may only be created with an Object Class which is Primary and possesses structure rules. This says that all entries must have structure rules.

11.22.3 Locally Registered Object Classes

A particular DSA is not required to support all Object Classes. It may contain a registry of the object classes which it does support.

The rules above enable the registry to be defined in terms of the locally registered Primary Object Classes which it supports. Each of these can be defined in terms of the single object identifier which represents that Object Class. (Of course, any entry defined with this Object Class contains an attributes whose values include not only the corresponding object identifier, but also the identifiers associated with each of the Object Class's superclasses.)

Associated with each locally registered Primary Object Class could be a list of secondary Object Classes which may be permitted to be used in association with this Primary Object Class. When a new entry is created, its Object Class attributes can then be analysed to determine:

Whether the entry's Object Class attribute is compatible with local registration

The Primary Object Class to which it conforms

The structure rules to which it must conform

The Secondary Object Classes (if any) to which it must conform. Given this analysis, the name and attributes of the entry can be analysed to determine its compatibility with the local registry of Primary Object Classes.

12. STABLE SECURITY AGREEMENTS

Editor's Note: This section points to Stable Security Agreements which are contained in the aligned section of the Stable Implementation Agreements.

13. SECURITY

13.9.1.1 References

13.9.1.2 Definitions

13.9.1.3 Assumptions

13.9.1.4 Motivation

13.9.2 Scope and Field of Application

13.9.3 Specific Security Model

13.9.4 Services Offered

13.9.5 Services Required

13.9.6 Protocols

13.9.7 Management Elements Required/Impacted

13.9.8 Conformance Class Definitions

13.9.9 Conformance Class Specifications

13.9.10 Registration Issues Requirements

13.10 TRANSPORT LAYER

13.10.1 Introduction

13.10.1.1 References

13.10.1.2 Definitions

13.10.1.3 Assumptions

13.10.1.4 Motivation

13.10.2 Scope and Field of Application

13.10.3 Specific Security Model

13.10.4 Services Offered

13.10.5 Services Required

13.12.1.2 Definitions

13.12.1.3 Assumptions

13.12.1.4 Motivation

13.12.2 Scope and Field of Application

13.12.3 Specific Security Model

13.12.4 Services Offered

13.12.5 Services Required

13.12.6 Protocols

13.12.7 Management Elements Required/Impacted

13.12.8 Conformance Class Definitions

13.12.9 Conformance Class Specifications

13.12.10 Registration Issues Requirements

13.13 APPLICATION LAYER

13.13.1 Introduction

13.13.1.1 References

13.13.1.2 Definitions

13.13.1.3 Assumptions

13.13.1.4 Motivation

13.13.2 Scope and Field of Application

13.13.3 Specific Security Model

13.13.4 Services Offered

13.13.4.1 ACSE

13.13.4.2 ROSE

13.15 Message Handling System Security

The following definitions of the elements of security service are based on the 1988 CCITT Recommendations on the Message Handling System (X.400). The fourteen (14) elements of security service are refinements of the five (5) primary security services as defined in IS 7498 Part 2 (Security Architecture). The Implementor's Workshop prepared Table 13.2 that summarizes where in the MHS the element of security service may be performed (the check marks) as stated in the MHS Recommendations. The Special Interest Group in Security (SIG-SEC) then examined each of the 14 elements of security service and placed a priority rating (1-5) next to one of the checkmarks in each row representing the priority that should be given for consideration of standardization and implementation of that element of service. The SIG-SEC reviewed the User Agent (UA) to User Agent peer entities as the first (perhaps preferred) place to implement security and used the check mark in that column if one was present. The SIG-SEC then reviewed the Message Transfer Agent (MTA) to Message Transfer Agent as the second place to implement security if it has not been implemented in the UA-UA protocol. Finally, the interface between the UA and the MTA was investigated for implementing security.

The Implementor's Workshop will be using this table and the set of definitions as a basis upon which future work in MHS security may be performed. The table is and subject to change during future meetings.

Table 13.1 X.400 Relationship between Elements of Security Service and MHS Components

	UA-MS	MS-MTA	UA-UA	UA-MTA	MTA-MTA	MTA-UA	MS-UA
Message Origin Authentication			✓1	✓			
Report Origin Authentication					✓4	✓	
Probe Origin Authentication		✓		✓5			
Proof of Delivery			✓2				✓
Proof of Submission						✓5	
Peer Entity Authentication	✓	✓		✓	✓4	✓	✓
Content Integrity			✓1				
Content Confidentiality			✓1				
Message Flow Confidentiality			✓4				
Message Sequence Integrity			✓2				
Non Repudiation of Origin			✓1				
Non Repudiation of Submission						✓5	
Non repudiation of Delivery			✓3				
Access Control	✓	✓	✓1	✓	✓	✓	✓

UA: User Agent
MS: Message Store
MTA: Message Transfer Agent

13.15.1 Definitions of Elements of Security Service

Message Origin Authentication **MT**

This element of service allows the originator of a message to provide to the recipient(s) of the message, and any MTA through which the message is transferred, a means by which the origin of the message can be authenticated (i.e. a signature). Message Origin Authentication can be provided to the recipient(s) of the message, and any MTA through which the message is transferred, on a per-message basis using an asymmetric encryption technique, or can be provided only to the recipient(s) of the message, on a per-recipient basis either a asymmetric or a symmetric encryption technique.

Report Origin Authentication **MT**

This element of service allows the originator of a message (or probe) to authenticate the origin of a report on the delivery or non-delivery of the subject message (or probe), (a signature). report Origin Authentication is on a per-report basis, and uses an asymmetric encryption technique.

Probe Origin Authentication **MT**

This element of service allows the originator of a probe to provide to any MTA through which the probe is transferred a means to authenticate the origin of the probe (i.e. a signature). Probe Origin Authentication is on a per-probe basis, and uses an asymmetric encryption technique.

Proof of Delivery **MT**

This element of service allows the originator of a message to obtain from the recipient(s) of the message the means to authenticate the identity of the recipient(s) and the delivered message and content. Message recipient authentication is provided to the originator of a message on a per-recipient basis using either symmetric or asymmetric encryption techniques.

Proof of Submission **MT**

This element of service allows the originator of a message to obtain from the MTS the means to authenticate that the message was submitted for delivery to the originally intended recipient. Message submission authentication is provided on a per-recipient basis, and can use symmetric or asymmetric encryption techniques.

Peer Entity Authentication **MT**

This element of service provides confirmation of the identity of the Entity (UA, MTA, MS). It provides confidence at the time of usage only that an entity is not attempting to masquerade as an unauthorized entity.

Content Confidentiality **MT**

This element of service allows the originator of a message to protect the content of the message from disclosure to someone other than the intended recipient(s). Content Confidentiality is on a per message basis, and can use either an asymmetric or a symmetric encryption technique.

Content Integrity **MT**

This element of service allows the originator of the message to provide to the recipient of the message a means by which the recipient can verify that the content of the message has not been modified. Content Integrity is on a per-recipient basis, and can use either an asymmetric or a symmetric encryption technique.

Message Flow Confidentiality **MT**

This element of service allows the originator of the message to protect information which might be derived from observation of the message flow.

Message Sequence Integrity **MT**

This element of service allows the originator of the message to provide to a recipient of the message a means by which the recipient can verify that the sequence of messages from the originator to the recipient has been preserved (without message loss, re-ordering, or replay). Message Sequence Integrity is on a per-recipient basis, and can use either an asymmetric or a symmetric encryption technique.

Non Repudiation of Origin **MT**

This element of service allows the originator of a message to provide the recipient(s) of the message irrevocable proof of the origin of the message. This will protect against any attempt by the originator to subsequently revoke the message or its content. Non Repudiation of Origin is provided to the recipient(s) of a message on a per message basis using asymmetric encryption techniques.

Non Repudiation of Submission **MT**

This element of service allows the originator of a message to obtain irrevocable proof that a message was submitted to the MTS for delivery to the originally specified recipient(s). This will protect against any attempt by the MTS to subsequently deny that the message was submitted for delivery to the originally specified recipient(s). Non Repudiation of Submission is provided to the originator of a message on a per message basis, and uses an asymmetric encryption technique.

Non Repudiation of Delivery **MT**

This element of service allows the originator of a message to obtain from the recipient(s) of the message, irrevocable proof that the message was delivered to the recipient(s). This will protect against any attempt by the recipient(s) to subsequently deny receiving the message or its content. Non Repudiation of Delivery is provided to the originator of a message on a per-recipient basis using asymmetric encryption techniques.

Access Control **MT**

This element of service provides protection against unauthorized use of the resources accessed via MHS. Access decisions are directed by a security policy which may be identity and/or role based.

13.16 DIRECTORY

13.16.1 Introduction

13.16.1.1 References

13.16.1.2 Definitions

13.16.1.3 Assumptions

13.16.1.4 Motivation

13.16.2 Scope and Field of Application

13.16.3 Specific Security Model

13.16.4 Services Offered

13.16.5 Services Required

13.16.6 Protocols

13.16.7 Management Elements Required/Impacted

13.16.8 Conformance Class Definitions

13.16.9 Conformance Class Specifications

13.16.10 Registration Issues Requirements

13.17 VTP

13.17.1 Introduction

13.17.1.1 References

13.17.1.2 Definitions

13.17.1.3 Assumptions

13.17.1.4 Motivation

13.17.2 Scope and Field of Application

13.17.3 Specific Security Model

13.17.4 Services Offered

13.17.5 Services Required

13.17.6 Protocols

13.17.7 Management Elements Required/Impacted

13.17.8 Conformance Class Definitions

13.17.9 Conformance Class Specifications

13.17.10 Registration Issues Requirements

14. ISO VIRTUAL TERMINAL PROTOCOL

14.1 INTRODUCTION

See Stable Agreements.

14.2 SCOPE AND FIELD OF APPLICATION

14.2.1 Phase Ia Agreements

See Stable Agreements

14.2.2 Phase Ib Agreements

See Stable Agreements regarding Forms profile.

The Scroll profile is intended to support line-at-a-time applications and has colour and text attribute capabilities.

14.2.3 Phase II Agreements

The X.3/X.29 PAD profile will support functionality similar to the CCITT recommendations and could be used to implement an X.3/X.29 to ISO-VT gateway.

The Page profile is intended for applications which require page-oriented operation.

14.3 STATUS

These agreements are being done in phases. Below is the current status of each phase.

14.3.1 Status of Phase Ia

The Phase Ia Agreements include the profiles for Telnet and Transparent operation and were completed in May, 1988. See Stable Agreements.

14.3.2 Status of Phase Ib

The Forms profile of Phase 1b was stabilized in December, 1988. See Stable Agreements.

The Scroll profile is not complete.

14.3.3 Status of Phase II

The Phase II agreements will include profiles for X.3/X.29 PAD and Page operations and will be completed at an unspecified future date.

It is intended that Phase II agreements be compatible with Phase I agreements.

14.4 ERRATA

Remove argument - R4 of Scroll Profit Value implied by R5.

Make argument - R2 optional Default = 80.

Scroll Profile, erasure-capability = "yes".

Scroll Profit, remove notes 6-9 (Informative).

Remove Note 1 (Definitive) from Scroll (Moved to an Issues List).

14.5 CONFORMANCE

See Stable Agreements.

14.6 PROTOCOL

See Stable Agreements.

14.7 NIST REGISTERED CONTROL OBJECTS

See Stable Agreements.

14.8 NIST DEFINED VTE-PROFILES

14.8.1 Telnet Profile

See Stable Agreements.

14.8.2 Transparent Profile

See Stable Agreements.

14.8.3 Forms Profile

See Stable Agreements.

14.8.4 Scroll Profile Definition

NIST VTE-Profile Scroll-1988 (r1,r1,...r10)

14.8.4.1 Introduction

This Scrolling A-mode VTE-profile is designed to support line-at-a-time interactions between a terminal and a host system, the type of operation typified by operating system command entry.

Scrolling is unidirectional, forward only.

The profile also provides a facility for switching local echo "on" or "off".

This VTE-Profile supports what is often referred to as "type-ahead", so input from the terminal user is available to the host application as soon as the application is ready for input, thus providing efficiency by eliminating communication delays.

This VTE-profile supports the definition of "input" termination events by the "Application VT-user" so the application can specify what events will cause "input" data to be forwarded to the "Application VT-user".

14.8.4.2 Association Requirements

14.8.4.2.1 Functional Units

This profile has no mandatory Functional Units required to operate.

The Urgent Data Functional Unit is optional, and will be used if available.

14.8.4.2.2 Mode

This profile operates in A-mode.

14.8.4.3 Profile Body

```
Display-objects =
{
    {
```

```
display-object-name = DOA,
DO-access = profile-argument-rl,
dimension = "two",
    x-dimension =
    {
        x-bound = profile-argument-r2,
        x-addressing = "no-constraint",
        x-absolute = "no",
        x-window = x-bound
    },
    y-dimension =
    {
        y-bound = "unbounded",
        y-addressing = "higher only",
        y-absolute = "no",
        y-window = 0
    },

erasure capability = "yes",

Note:   Implied by profile argument r5.

repertoire-assignment = profile-argument-r4,

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
},

{
display-object-name = DOB,
DO-access = opposite of profile-argument-rl,
dimension = "two",
    x-dimension =
    {
        x-bound = profile-argument-r2,
        x-addressing = "no-constraint",
        x-absolute = "no",
        x-window = x-bound
    },
    y-dimension =
    {
        y-bound = "unbounded",
        y-addressing = "higher only",
        y-absolute = "no",
            y-window = 0
        },
    erasure capability = "yes",

    Note:   Implied by profile argument r5.

    repertoire-assignment = profile-argument-r4,

    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
    }
},

Control-objects =
{
    {
    CO-name         = E,    *(standard Echo CO)*
    CO-type-identifier  = vt-b-sco-echo,
    CO-access           = profile-argument-r1,
    CO-priority     = "normal",
    CO-trigger      = "selected",
    CO-category     = "boolean",
    CO-size         = 1
    },

    IF r10 = "TE" THEN
    {
    CO-name         = TE, *(Termination Control CO)*
    CO-type-identifier  = vt-b-sco-tco,
    CO-access           = opposite of profile-argument-r1,
    CO-priority     = "normal",
    CO-trigger      = "selected",
    CO-category     = "integer"
    },

    {
    CO-name         = SA, *(NIST Registered CO)*
    CO-type-identifier  = nist-vt-co-misc-sa,
    CO-access           = profile-argument-r1,
    CO-priority     = "normal",
    CO-trigger      = "not selected",
    CO-category     = "integer",
    CO-size         = 65535
    },
{
CO-name         = UA, *(NIST Registered CO)*
CO-type-identifier  = nist-vt-co-misc-ua,
CO-access           = profile-argument-r1,
CO-priority     = "urgent",
CO-category     = "integer",
CO-size         = 65535
},

{
CO-name         = ST, *(NIST Registered CO)*
CO-type-identifier  = nist-vt-co-misc-st,
CO-access           = opposite of profile-argument-r1,
CO-priority     = "normal",
CO-category     = "integer",
CO-size         = 65535
},

{
CO-name         = UT, *(NIST Registered CO)*
CO-type-identifier  = nist-vt-co-misc-ut,
CO-access           = opposite of profile-argument-r1,
CO-priority     = "urgent",
CO-category     = "integer",
CO-size         = 65535
},

{
CO-name         = TC, *(Termination conditions CO)*
CO-type-identifier  = nist-vt-co-tcco-tc,
CO-structure        = N, *( defined with TCCO)*
CO-access           = profile-argument-r1,
CO-priority     = "normal",
    {
    CO-element-id   = 1, *(termination length)*
     CO-category    = "integer",
     CO-size        = 65535 },
    {
    CO-element-id   = 2, *(time-out mantissa)*
     CO-category    = "integer",
     CO-size        = 65535 },
    {
    CO-element-id   = 3, *(time-out exponent)*
     CO-category    = "integer",
     CO-size        = 65535 },
    {
    CO-element-id   = 4-N, *(from registered TCCO)*
     CO-category    = ???,
     CO-size        = ??? }
The NIST Workshop VT SIG is defining this registered TCCO.
This TCCO is a reference to that registered control object.
    }
}

Device-objects =
{
    {
    device-name = DVA,  *("output" device object)*
    device-default-CO-access = profile-argument-r1,
    device-default-CO-initial-value = 1."true",
    device-display-object = DOA,
    device-minimum-X-array-length = profile-argument-r2,
    device-minimum-Y-array-length = profile-argument-r3,
    device-control-object = {SA,UA}
    },
    {
    device-name = DVB,  *("input" device object)*
    device-default-CO-access = opposite of
                          profile-argument-r1,
    device-default-CO-initial-value = 1."true",
    device-display-object = DOB,
    device-minimum-X-array-length = profile-argument-r2,
    device-control-object = profile-argument-r10,
    device-control-object = {ST,UT}
    }
}

type-of-delivery-control = "simple-delivery-control".
```

14.8.4.4 Profile Argument Definitions:

r1 - is mandatory and enables negotiation of which VT-user has update access to display object DOA. It takes values "WACI", "WACA". It implies the asymmetric roles of the VT-users as "Application VT-user" and "Terminal VT-user". If the value for DOA is "WACI", then the association initiator is the "Application VT-user"; if the value of DOA is "WACA", then the association initiator is the "Terminal VT-user". This profile argument is also used to determine which VT-user has access to other VT objects as described above. Reference in the profile definition to "opposite of profile- argument-r1" means that the alternative of the two possible values for profile- argument-r1 is to be used. This argument is identified by the identifier for DO-access for display object DOA.

r2 - is optional and enables negotiation of a value for the VTE-parameter x-bound for the display objects DOA and DOB. It takes an integer value greater than zero.

This argument is identified by the identifier for x-bound for display object DOA. Default is 80.

r3 - is optional and enables the negotiation of a value for the VTE-parameter device-minimum-Y-array-length for device object DVA. It takes an integer value greater than zero; if absent, a device of any length will be satisfactory.

Note: Indicates screen length.

r4 - is optional and provides for the negotiation of a value for VTE-parameter repertoire-capability. Default specified by 9040.

r5 - is optional and provides for the negotiation of value(s) for the VTE-parameter repertoire-assignment. The value of profile-argument-r4 specifies the maximum number of occurrences of this argument. Default is specified by 9040.

r6 - 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.

r7 - is optional and provides for the negotiation of value(s) for VTE-parameters foreground-colour-capability and background-colour-capability. Default is 8.

r8 - is optional and provides for the negotiation of a value for VTE-parameter foreground-colour-assignment. Default is {"white", "black", "red", "cyan", "blue", "yellow", "green", "magenta"}.

r9 - is optional and provides for the negotiation of a value for VTE-parameter background-colour-assignment. Default is {"black", "white", "cyan", "red", "yellow", "blue", "magenta","green"}.

r10 - is optional and enables negotiation of a termination control object. The value for this argument is the value of CO-name for the termination control object, i.e. "TE"; if absent, no termination control is defined.

14.8.4.5 Profile Dependent CO Information

This profile makes use of five NIST registered Control Objects, SA, UA, ST, UT and TCCO. The CO-access in each CO is defined within this profile.

14.8.4.6 Profile Notes

14.8.4.6.1 Definitive Notes

1. Only the first boolean of the default control object contained in each device object is defined. This boolean is defined as the "on/off" switch for the device where the value "true" ="on" and "false" = "off". These values were chosen so the initial value of the boolean, "true", means the device is initially "on" and data to/from the display objects is being mapped to the device.

2. Only one boolean is defined in the standard echo control object, E. The semantics of this boolean is defined such that "false" means "local echo off" and "true" means"local echo on"; these values were chosen so echoing is initially "off" (which would provide security when a password is entered at the start of a terminal session).

14.8.4.6.2 Informative Notes

1. This profile models a scrolling device with scrolling only in the forward direction. The display pointer may not be moved backwards to modify earlier lines. A typical use for this profile is for applications where type-ahead may be advantageous and control over local echo "on"/"off" is required, e.g. the type of application where a conventional teletypewriter device or'teletype-compatible' video device having 'full duplex'capability is often used. Display object DOA referred to above is typically mapped to the display or printing device and display object DOB is typically mapped to the keyboard.

2. Data which is "typed-ahead", as with other data, is delivered to the peer VT-user immediately on detection of a termination condition or a VT-DELIVER due to the use of A-Mode (thus reducing transmission delay).

3. Display object DOB has an unbounded y-dimension so as to provide a blank line for each new line entered.

4. Line-at-a-time forward scrolling is mapped onto an update-window (value zero) which allows NO backward updates to preceding lines (x-arrays). The sevice-minimum-Y-array-length negotiated by profile-argument-r3 can be used to indicate the number of lines (x-arrays) which should remain visible to the human terminal user although specifically NOT available for update.

5. The ability to switch local echo "on" or "off" is always present; the ECHO control object is used for this purpose.

14.8.4.7 Specific Conformance Requirements

None.

14.9 APPENDIX A

See Stable Agreements.

14.10 APPENDIX B - CLARIFICATIONS

14.10.1 Defaults

When a profile argument is not present in either the offer or value list, the default for the corresponding VTE parameter is specified by ISO 9040 or the argument description in the profile.

15. TRANSACTION PROCESSING

Editor's Note: This section is a placeholder for future Transaction Processing (TP) Agreements. The TP Special Interest Group is newly formed and will hold its first regular meeting in March, 1989. Any new text from this group will be inserted here.

16. OFFICE DOCUMENT ARCHITECTURE

Below are given Errata to the Base Stable Office Document Architecture (ODA) Agreements contained in V2.E1 Stable Document. The Base ODA Stable Agreements are in the aligned section of the Stable Implementation Agreements, V2.E1 NIST SP 500-162.

ERRATA

NO. OF ERRATA	TYPE	REFERENCED DOCUMENT	SECTION	DESCRIPTION
ODA 3/89-1	Technical	NIST SP 500-162		Change "content-generator" to REQUIRED rather than PERMITTED on specific in BLOCK layout constituent
ODA 3/89-2	Technical	NIST SP 500-162		Remove recursion of FIGURE in logical structure
ODA 3/89-3	Technical	NIST SP 500-162		Add TITLET to the logical structure
ODA 3/89-4	Technical	NIST SP 500-162		Add PHRASEF to the logical structure
ODA 3/89-5	Technical	NIST SP 500-162		Change bindings on PASSAGE, REFERENCE, NUMBEREDSEGMENT and FIGURE to include both "manipulation" and "initialization"

NO. OF ERRATA	TYPE	REFERENCED DOCUMENT	SECTION	DESCRIPTION
ODA 3/89-6	Technical	NIST SP 500-162		Separate BODYAREA into BODYFRAME1 and BODYFRAME2 in layout structure
ODA 3/89-7	Alignment	NIST SP 500-162		Change notation to that used in ISO 8613 PDAD
ODA 3/89-8	Alignment	NIST SP 500-162		Update conformance statement to align with PAGODA CORE-36
ODA 3/89-9	Alignment	NIST SP 500-162		Align Application-comments with PAGODA CORE-36
ODA 3/89-10	Alignment	NIST SP 500-162		Include ISO 6937/2 minimal and ISO 646 subrepertoires as BASIC values
ODA 3/89-11	Alignment	NIST SP 500-162		Change layout structure to align with PAGODA CORE-36 structure
ODA 3/89-12	Editorial	NIST SP 500-162		Add definitions and references from PAGODA CORE-36
ODA 3/89-13	Editorial	NIST SP 500-162		Add IGS control function

17. FUTURE OFFICE DOCUMENT ARCHITECTURE (ODA)

Editor's Note: This section will contain the new text relating to Office Document Architecture (ODA) Agreements.

18. NETWORK MANAGEMENT

Editor's Note: The notes in this section are meant to be placeholders for future text. They are included here to reflect SIG activity in these areas.

18.1 INTRODUCTION

Within the community of OSI researchers, users, and vendors, there is a recognized need to address the problems of initiating, terminating, monitoring, and controlling communication activities and assisting in their harmonious operation, as well as handling abnormal conditions. The activities that address these problems are collectively called network management.

Network management can then be viewed as the set of operational and administrative mechanisms necessary to:

a. bring up, enroll, and/or alter network resources,

b. keep network resources operational,

c. fine tune these resources and/or plan for their expansion,

d. manage the accounting of their usage, and

e. manage their protection from unauthorized use/tampering.

As such, network management is typically concerned with at least the following five functional areas: configuration management, fault management, performance management, accounting management, and security management. In order to accomplish management, observations about network resource operations and configuration may need to be transferred from network nodes (with management agents) to network managers, or between network managers. Similarly, management commands may need to be disseminated between managers, or from a manager to a network node.

In this section, there are Implementation Agreements (IA's) for providing interoperable OSI management information communication services among OSI systems. Also contained here are agreements on management information, or pointers to other sections of this document where such additional agreements appear.

These agreements pertain to the exchange of management information and management commands between open systems operating in a multivendor environment. Therefore, the goal is to ensure that a management system built by one vendor can manage network objects built by another vendor.

In progressing work on OSI management in the NIST/OSI NMSIG, the OSI management framework specified in ISO 7498/Part 4 (as presented in reference [1]) shall be used as the basis for concepts and terminology relevant (a) to OSI management activities, and (b) to management services supported by OSI management protocols. Thus, these agreements are based on, and employ, protocols developed in accord with the OSI Reference Model. Furthermore, they attempt to eliminate ambiguities in interpretations of management protocol standards and management information standards.

18.1.1 References

The following documents are referenced in the statements of the agreements relating to NIST/OSI network management.

OSI Systems Management References:

[ADDRMVP] ISO/IEC 9596/PDAD 2, Common Management Information Protocol: Add/Remove Protocol, ISO/IEC JTC1/SC21 N3306, January 1989.

[ADDRMVS] ISO/IEC 9595/PDAD 2, Common Management Information Service: Add/Remove Service, ISO/IEC JTC1/SC21 N3305, January 1989.

[ALS] ISO/IEC DIS 9545 (Ballot), Information Processing Systems - Open Systems Interconnection - Application Layer Structure, 15 September 1988.

[AMWD] Information Processing Systems - Open Systems Interconnection - Accounting Management Working Document, ISO/IEC JTC1/SC21 N3314, December 1988.

[CANCETS] ISO/IEC 9595/PDAD 1, Common Management Information Service: CancelGet Service, ISO/IEC JTC1/SC21 N3303, January 1989.

[CANGETP] ISO/IEC 9596/PDAD 1, Common Management Information Protocol: CancelGet Protocol, ISO/IEC JTC1/SC21 N3304, January 1989.

[CMIP] ISO/IEC DIS 9596-2, Information Processing Systems - Open Systems Interconnection - Management Information Protocol Specification - Part 2: Common Management Information Protocol, 22 December 1988.

[CMIS] ISO/IEC DIS 9595-2, Information Processing Systems - Open Systems Interconnection - Management Information Service Definition - Part 2: Common Management Information Service, 22 December 1988.

[CMO] Information Processing Systems - Open Systems Interconnection - Working Draft of the Configuration Management Overview, ISO/IEC JTC1/SC21 N3311, 16 January 1989.

[DMA] ISO/IEC DP 10165-3, Information Processing Systems - Open Systems Interconnection - Structure of Management Information - Part 3: Definitions of Management Attributes, ISO/IEC JTC1/SC21 N3302, January 1989.

[DSO] ISO/IEC DP 10165-2, Information Processing Systems - Open Systems Interconnection - Structure of Management Information - Part 2: Definitions of Support Objects, ISO/IEC JTC1/SC21 N3301, January 1989.

[ERIRF] ISO/IEC DP 10164-4, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 4: Error Reporting and Information Retrieval Function, ISO/IEC JTC1/SC21 N3298, 31 January 1989.

[FMWD] Information Processing Systems - Open Systems Interconnection - Systems Management - Fault Management Working Document, ISO/IEC JTC1/SC21 N3312, January 1989.

[FRMWK] ISO 7498-4 (DIS), Information Processing Systems - Open Systems Interconnection - Basic Reference Model - Part 4: OSI Management Framework - Revision of DIS 7498-4 following Editing Meeting (Sydney), 4 January 1989.

[GDMO] Proposed DP 10165-4, Information Processing Systems - Open Systems Interconnection - SMI - Part 4: Guidelines for the Definition of Managed Objects, ISO/IEC JTC1/SC21 N3437, 10 February 1989.

[LCF] First Working Draft For Systems Management: Log Control Function, ISO/IEC JTC1/SC21 N3309, January 1989.

[MIM] Proposed DP 10165-1, Working Draft for Structure of Management Information - Part 1: Management Information Model, ISO/IEC JTC1/SC21 N3324, January 1989.

[MSC] Proposed DP 10164-5, Information Processing Systems - Open Systems Interconnection - Systems Management - Management Service Control, ISO/IEC JTC1/SC21 N3299, January 1989.

[OMF] ISO/IEC DP 10164-1, Information Processing Systems - Open Systems Interconnection - Systems Management -

Part 1: Object Management Function, ISO/IEC JTC1/SC21 N3295, 31 January 1989.

[OSIMIL] Management Information Library (MIL) - Revision 1.0, OSI MIB Working Group of NMSIG of NIST/OSI Implementors Workshop, March 1989.

[PMWD] Information Processing Systems - Open Systems Interconnection - Performance Management Working Document (Third Draft), ISO/IEC JTC1/SC21 N3313, 18 January 1989.

[RMF] ISO/IEC DP 10164-3, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 3: Relationship Management Function, ISO/IEC JTC1/SC21 N3297, 31 January 1989.

[SMF] ISO/IEC DP 10164-2, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 2: State Management Function, ISO/IEC JTC1/SC21 N3296, 31 January 1989.

[SMO] ISO/DP 10040, Information Processing Systems - Open Systems Interconnection - Systems Management Overview, ISO/IEC JTC1/SC21 N3294, January 1989.

[SMWD] Information Processing Systems - Open Systems Interconnection - Systems Management - Fifth Draft of OSI Security Management Working Document, ISO/IEC JTC1/SC21 N3315, January 1989.

Other OSI References:

[ACSEP] ISO 8650, Information Processing Systems - Open Systems Interconnection -Protocol Specification for the Association Control Service Element (Revised Final Text of DIS 8650), ISO/IEC JTC1/SC21 N2327, 21 April 1988.

[ACSES] ISO 8649, Information Processing Systems - Open Systems Interconnection -Service Definition for the Association Control Service Element (Revised Final Text of DIS 8649), ISO/IEC JTC1/SC21 N2326, 21 April 1988.

[ASN1] ISO 8824, Information Processing Systems - Open System Interconnection - Specification of Abstract Syntax Notation One (ASN.1), 19 May 1987.

[BER] ISO 8825, Information Processing Systems - Open Systems Interconnection -Basic Encoding Rules for Abstract Syntax Notation One (ASN.1), 19 May 1987.

[DIR] ISO 9594 - Information Processing Systems - Open Systems Interconnection - The Directory

[PSD] ISO 8822, Information Processing Systems - Open Systems Interconnection -The Presentation Service Definition.

[ROSEP] ISO 9072-2 - Information Processing Systems - Text Communications - Remote Operations Part 2: Protocol Specification.

[ROSES] ISO 9072-1, Information Processing Systems - Text Communications - Remote Operations Part 1: Model, Notation and Service Definition.

Other References

[MAP30] MAP 3.0 Network Management Specification.

Editor's Note: Section editors whose text cites these references will keep them up-to-date and will provide additional references as needed, e.g., most recent ISO "N" number and date will be provided.

18.2 SCOPE AND FIELD OF APPLICATION

The purpose of this section (Section 18), is to provide implementation agreements that will enable independent vendors to supply customers with a diverse set of networking products that can be managed as part of an integrated environment. Where possible, these agreements are based upon OSI Network Management standards.

Due to the broad scope of the subject, and given that OSI Management standards are still evolving, it is reasonable to assume that a comprehensive set of network management implementors agreements will take a number of years to develop. In order to arrive at an initial set of implementation agreements in a timely fashion, a phased approach has been adopted.

As a first step in this phased approach, the NMSIG has targeted that the initial, Phase 1, interim agreements will be completed by September, 1989. These Phase 1 agreements provide limited interoperable management in a heterogeneous vendor environment. They are the corner stone of our eventual comprehensive inventory of OSI-compatible management agreements. Furthermore, these initial agreements allow the community to gain experience with OSI management standards as they emerge.

The scope of the problem addressed in Phase 1 has been constrained in several ways. The sections below outline the nature of these constraints and thereby serve to clarify the scope and field of application associated with this version of the implementors agreements (December 1988). Subsequent phases of these agreements (post December 1988) will expand the scope of problems addressed.

The following is an outline of the information provided in these agreements (Section 18):

Section 18.2-- SCOPE AND FIELD OF APPLICATION (This section): This section covers several areas. Specifically:

- Section 18.2.1 describes the relationship between these agreements and the evolving international management standards.

- Section 18.2.2.1 provides a brief overview of the management architecture described in the standards documents.

- Section 18.2.2.2 identifies the constraints imposed on Phase 1 of these agreements.

- Section 18.2.2.3 addresses migration strategies regarding subsequent phases of these agreements.

- Section 18.2.2.4 addresses interoperability with systems associated with other management specifications (including MAP/TOP) [21].

- Section 18.2.3 presents an overview of the functionality supported by Phase 1 of these agreements.

Section 18.3 -- STATUS: This section describes the current status of these agreements.

Section 18.4 -- ERRATA: Once this document is incorporated into a version of the Stable Implementation Agreements for Open System Interconnection Protocols, this section will contain corrections to the stable management agreements. In addition, this section documents interim resolutions to defects found in the management standards.

Section 18.5 -- MANAGEMENT FUNCTIONS: This section documents agreements pertaining to the Functions offered by each of the Management Functional Areas. In addition, it identifies agreements pertaining to the use of other application service elements (e.g. the Common Management Information Service Element (CMISE)).

Section 18.6 -- MANAGEMENT COMMUNICATIONS: This section identifies, in detail, the following:

- o Agreements on Association Policies

- o Agreements on the Common Management Information Services (CMIS) offered.

- o Common Management Information Protocol (CMIP) agreements.

- o Agreements pertaining to the services required by CMIP.

Section 18.7 -- MANAGEMENT INFORMATION AGREEMENTS: This section deals with the basic concepts and modeling techniques associated with management information. It provides implementation agreements regarding the naming of managed objects, the Structure of Management Information (SMI) and Generic Definitions of Management Information (GDMI). In addition, this section identifies a list of managed object classes that must be defined to meet the functional goals of these Phase 1 agreements.

Note: This section does NOT provide managed object definitions.

Section 18.8 -- IMPLEMENTATION PROFILES/CONFORMANCE CLASSES: This section describes the implementation profiles/conformance classes that are used to categorize management products. At the highest level, products fall into three broad categories: systems that take on a managing system role, systems that take on an agent system role, and managed objects represented via agent processes. (Refer to Section 18.2.2 for further clarification regarding these categories.) Phase 1 of these agreements defines implementation profiles/conformance classes only for systems that take on an agent system role.

Editor's Note: The NMSIG intends for Phase 1 to ensure that the interface between managing processes and agent processes is adequately specified, thereby enabling the development of interoperable managing processes and agent processes. It is believed that, by identifying implementation profiles/conformance classes only for systems that take on an agent system role, we will also have sufficiently identified the expected behavior of systems that take on a managing system role.

Section 18.9 -- CONFORMANCE: For each of the classes identified in Section 18.8, this section outlines the criteria used to determine whether or not a given product conforms to the class specification that it purports to be. More to the point, in conjunction with Phase 1:

- Systems that take on an agent system role will be tested, via interactions with a test managing system to ensure that they appropriately represent those managed objects that they purport to represent.

Editor's Note: Although systems that take on a managing system role are not to be tested for conformance in Phase 1, it is believed that market presence of conformant systems that take on an agent system role will provide an adequate climate for determining the suitability of systems that take on a managing system role.

Section 18.10 -- REGISTRATION REQUIREMENTS: This section identifies the management entities that must be registered. This includes a listing of those managed objects that must be defined in order to satisfy the functional requirements outlined in the Phase 1 agreements.

In addition, this section describes the mechanisms used to register management entities and the means by which one can obtain information about a registered entity.

18.2.1 Use of Evolving Standards

In general, it is the intent of the NMSIG to base these implementors agreements on existing international management standards.

Editor's Note: Table 18.1 below shows the relevant standards documents and the current schedules for progressing these documents to the IS status. The table describes the work items and associated target dates approved at the Fifth SC 21/WG 4 Meeting in Sydney, November 29 - December 9, 1988. The citations and Reference Section (18.1.1) of this Implementors' Agreement will be updated as soon as possible after receipt of the Sydney documents.

Table 18.1 RELEVANT STANDARDS DOCUMENTS AND THE CURRENT SCHEDULES FOR PROGRESSING THESE DOCUMENTS TO IS STATUS

Document	Target Dates		
	DP	DIS	IS
Management Framework [1]	9/86	6/87	10/88
Systems Management Overview	12/88	8/89	8/90
Structure of Management Information			
Part 1: Management Information Model	5/89	4/90	4/91
Part 2: Definition of Support Management Objects	12/88	4/90	4/91
Part 3: Definition of Management	12/88	4/90	4/91
Attributes	12/88	4/90	4/91
Part 4: Guidelines for Managed Object Definition	10/89	9/90	9/91
Common Management Information Service		9/88	9/89
Addendum 1: CancelGet	12/88	9/89	8/90
Addendum 2: Add/Remove	12/88	9/89	8/90
Common Management Information Protocol		9/88	8/89
Addendum 1: CancelGet	12/88	9/89	8/90
Addendum 2: Add/Remove	12/88	9/89	8/90
Configuration Management			
Systems Management - Part 1: Object Management Function	12/88	7/89	7/90
Systems Management - Part 2: State Management Function	12/88	4/90	4/91
Systems Management - Part 3: Relationship Management Function	12/88	4/90	4/91
Fault Management			
Systems Management - Part 4: Error Reporting and Information Retrieval Function	12/88	4/90	4/91
Systems Management - Part 5: Service Control Function	12/88	4/90	4/91
Systems Management - Part 6: Confidence and Diagnostic Testing Function	10/89	7/90	7/91
Systems Management - Part 7: Log Control Function	10/89	7/90	7/91
Security Management	10/89	7/90	7/91
Accounting Management	10/90	3/92	3/93
Performance Management	10/89	7/90	7/91

Given the current state of the standards, the Phase 1 implementors agreements are based primarily on documents that are in the DP state. In addition, in order to meet the stated objectives of the Phase 1 agreements, some agreements have been formed in advance of the availability of DP's in the relevant areas.

As the relevant standards documents progress from DP to DIS and from DIS to IS, the information contained in the standards will be addressed by these agreements.

Thus subsequent phases of these agreements will incorporate the relevant standards information as the standards become available. In general, the NMSIG will attempt to incorporate information from a standard that has progressed to the DIS or IS state into the subsequent phase of the implementors agreements.

When a defect is found in any of the management related standards, the 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, these agreements will note defect resolutions which have the tentative approval of the appropriate standards committee. These interim resolutions will be recorded in Section 18.4.

Once a defect resolution has been finalized by the appropriate standards body, the agreed upon resolution will be incorporated into the next phase of these implementors agreements. If appropriate, a previous phase that relied on an interim resolution will be examined to determine whether or not errata should be issued to bring the original phase into line with the final resolution.

18.2.2 Management Architecture

18.2.2.1 Systems Management Overview

Reference [SMO] provides an overview of the OSI Systems Management Architecture. What follows is a brief summary of the information contained therein. The material contained here (i.e. Section 18.2.2.1) is tutorial in nature. It is not intended to correct deficiencies that may exist in the standards themselves. This information is primarily intended to serve as an aid to the casual reader of these requirements. For more detail, please refer to the management standards referenced below.

STANDARDS

The OSI System management standards are grouped as follows:

- References [FRMWK] and [SMO] address the general concepts.

- References [ALS], [CMIS], and [CMIP] address the communications standards.

- References [MIM], [DSO], [DMA], and [GDMO] pertain to the definition of management information (managed objects).

- References [CMO], [FMWD], [SMWD], [AMWD], and [PMWD] document functional area standards.

  Editor's Note: Due to reorganization of documents as a result of the December 1988 SC21/WG4 meeting in Sydney, functions have been separated from the management functional areas which originally developed them. The documents which describe these functions include [OMF], [SMF], [RMF], [ERIRF], and [MSC]. This list should be included in the next revision of these agreements.

GENERAL CONCEPTS

In the abstract, a communications environment is made up of a collection of managed objects. Management of the communications environment is viewed as being an information processing application. Management activities are carried out by using the information processing application to manipulate and monitor the managed objects that make up the environment.

Because the environment being managed is distributed, the components of the information processing application are distributed. These distributed components take the form of management application processes. The interactions that take place between management processes are referred to as directives.

Management processes are divided into two categories: managing processes and agent processes. A managing process is that part of a distributed application process that is responsible for carrying out one or more management activities. An agent process is responsible for manipulating and monitoring an associated set of managed objects. A managing process interacts with an agent process to carry out the management activities for which it is responsible.

An agent process performs the management function upon receipt of a directive specifying management operations on managed objects. Agent processes may also forward directives to managing processes to convey information generated by managed objects.

APPLICATION LAYER COMMUNICATIONS

A systems management application entity (SMAE) is that portion of a management process that is responsible for communicating with other management processes (or more specifically, other SMAE's). A SMAE is made up of a collection of cooperating application service elements (ASE's).

The association control service element (ACSE) is used to establish associations with other SMAE's. Once this is done, a systems management application service element (SMASE) is used to exchange information between the associated SMAE's. The SMASE realizes the abstract notion of directives exchanged between management processes.

The SMASE relies on other (standard) ASE's to effect communications. Notably, the services of the common management information service element (CMISE) are used.

Taken as a whole, a SMAE ultimately relies on presentation layer services to communicate.

FUNCTIONAL AREAS

Systems management activities are grouped into five functional areas that are intended to capture the user requirements imposed on management. These functional areas are:

- Configuration Management
- Fault Management
- Security Management
- Performance Management
- Accounting Management

Each of these functional areas is referred to as a Specific Management Functional Area (SMFA). Each SMFA gives rise to a standard that identifies the following:

- A set of functions that support the functionality within the scope of the SMFA.

- The procedures associated with the provision of each function.

- The services required to support these procedures.

- The use of the underlying OSI services to provide the communications needs.

- The classes of managed objects that the procedures will operate upon in order to provide the functionality defined by the SMFA.

MANAGEMENT DOMAINS

Reference [SMO] defines a management domain as follows:

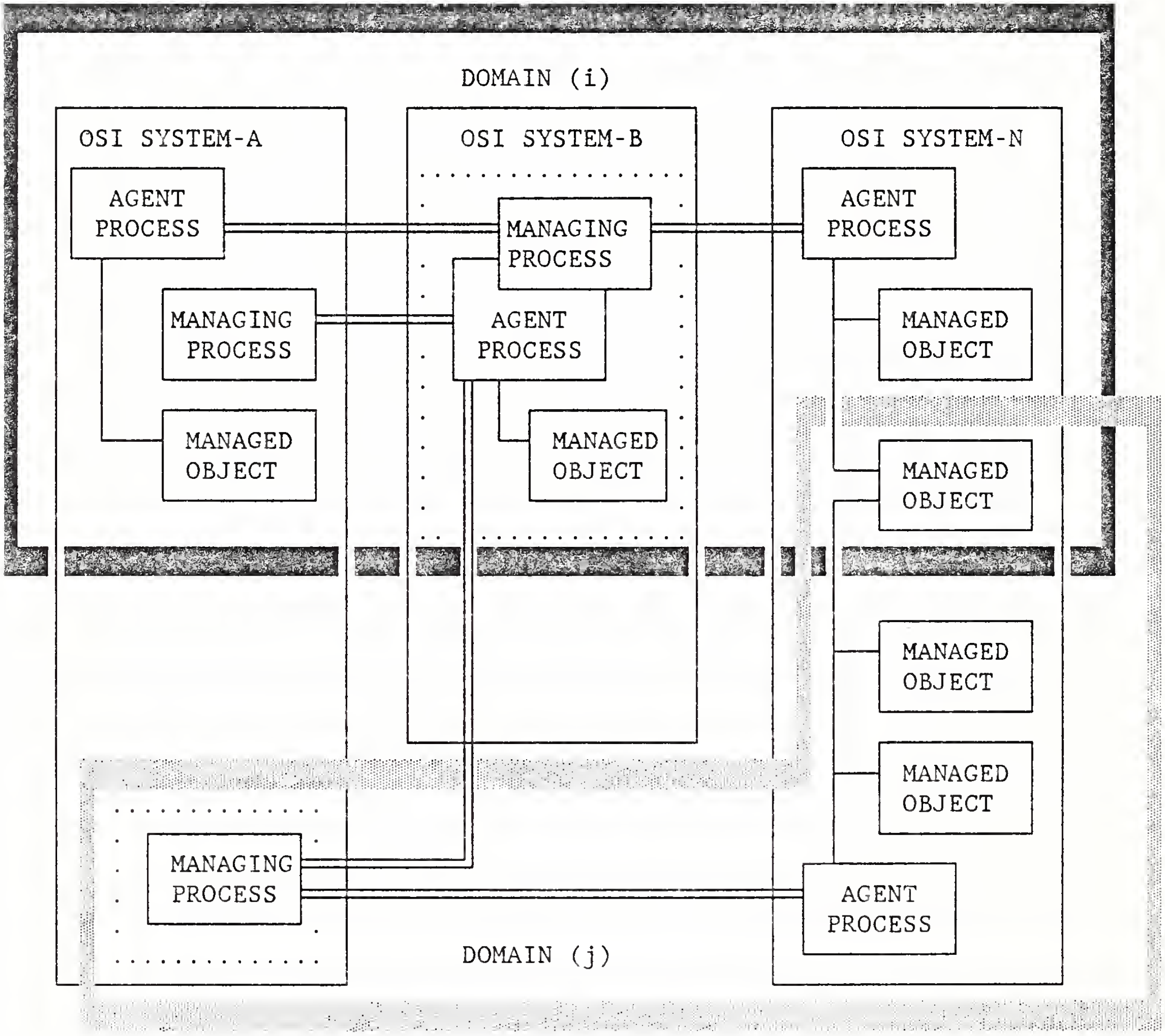
Real open systems may contain managing processes, agent processes, or both. To meet the organizational needs for flexibility, a real OSI Management environment can be partitioned into a number of management domains. For example, management domains can be created in accordance with administrative boundaries.

A management domain is a collection of one or more distributed management processes and their associated managed objects (see Figure 18.1). A real open system can be part of one or more management domains. A single managed object can participate in more than one management domain.

ADMINISTRATION OF MANAGEMENT DOMAINS

The administration of a management domain implies creation, modification, and maintenance of:

- managed objects represented in the MIB:
- relationships among managing and agent processes of distributed management applications;
- relationships among agent processes and managed objects and processes of the distributed management applications.



LEGEND: ═══ MANAGEMENT PROTOCOLS

⋯ REPRESENTS THE DOMAIN'S ADMINISTRATIVE AUTHORITY

Figure 18.1 Concept of Management Domains

The administration of a management domain is carried out by an administrative authority that may be an Administration (a public telecommunications Administration or other organization offering communication services) or a private organization. The organization concerned may or may not elect to make use of these implementation agreements to govern interactions between management processes which are wholly within a management domain.

18.2.2.2 Constraints/Assumptions for Phase 1

The focus of the Phase 1 agreements is to enable a managing process provided by one vendor to interoperate with an agent process provided by a different vendor for the purpose of performing limited management on a set of managed objects. Specifically, these agreements focus on the managing process/agent process interface and the techniques used to define managed objects. These agreements do not address (nor constrain) the mechanisms used by agent processes to manipulate managed objects. Nor should these agreements inhibit our ability to provide post-Phase 1 agreements that meet the long term goals associated with the area of network management.

In order to accomplish this goal in a timely fashion, several simplifying constraints have been imposed on these agreements. These constraints are summarized below.

1. These agreements support only a limited set of functionality. Refer to Sections 18.2.3 and 18.5 for a description of the functionality supported by these agreements.

2. No agreements are provided in support of managing process to managing process communications.

3. Agreements regarding managing process to agent process interactions were (will be) formed without regard to management domains.

   Editor's Note: It is worth noting that the management domains were the subject of much discussion within the NMSIG. It was felt that the definition was unclear, and that the impact of supporting this concept within management products was even less clear. As a result, we have no reason to expect that Phase 1 products will adequately support the needs associated with this concept.

4. All communications supported by these agreements rely on the use of the following application service elements: the association control service element (ACSE), the common management information service element (CMISE), Remote Operations Service Element (ROSE), and the system management application service element (SMASE) identified in Section 18.6.

5. All communications between managing processes/agent processes are based on connection-oriented presentation services.

6. These agreements do not rely on the use of Directory Services.

7. No agreements regarding the security of management are provided.

Editor's Note: The NMSIG has requested, via a liaison statement, that the Security SIG suggest appropriate security agreements to address this area. In the absence of input from the Security SIG, it should be noted that individual management products may implement proprietary security policies that do not interfere with interoperability. For example, a given managing process or agent process may decide to refuse an A-Associate request based on the calling presentation address and some locally defined criteria.

8. It is assumed that every managed object instance will be associated with exactly one agent process. This agent process is responsible for acting as the agent for the managed object with regard to all interactions with the managing systems.

18.2.2.3 Migration to Future Phases

Editor's Note: This section will document the migration plans with regard to ensuring that Phase N products can interact with Phase 1 products.

18.2.2.4 Relationship to Other Management Specifications

Editor's Note: This section will describe the degree to which implementations that conform to these agreements will interoperate with implementations that conform to the other management specifications (including MAP/TOP).

18.2.3 Management Scenarios

Editor's Note: The intent of this section is to amplify the high level NM requirements to be met by these IAs. In particular, this section will provide a high level view of the functionality supported by Phase 1 of these agreements. Based on these scenarios, one should be able to determine the scope of managed object classes that are required to satisfy these scenarios.

18.3 STATUS

Section 18 is currently a working draft of the Phase 1 Network Management Implementors Agreements.

18.4 ERRATA

(None as yet)

18.5 MANAGEMENT FUNCTIONS AND SERVICES

Editor's Note: To aid the casual reader, parts of this section have been written in a tutorial fashion, explaining unclear or obscure areas in the base standards. This material will be deleted when transition to the Stable Agreements Document occurs. The remaining material contains agreements relative to the base standards or to areas deemed important for interoperability but not contained in the base standards.

Editor's Note: Tutorial Material. ISO has partitioned network management into five Specific Management Functional Areas (SMFAs) as a convenience for developing requirements particular to configuration management (CM), fault management (FM), performance management (PM), security management (SM), and accounting management (AM). These requirements are specified in five separate SMFA standards ([CMO], [FMWD], [SMWD], [AMWD], and

[PMWD]). Due to reorganization of documents as a result of the December 1988 SC21/WG4 meeting in Sydney, functions have been separated from the management functional areas which originally developed them. The documents which describe these functions include [OMF], [SMF], [RMF], [ERIRF], and [MSC].

Since the SMFAs have overlapping requirements, management functions and management information applicable to one SMFA are often applicable to other SMFAs. Therefore, the SMFAs point to separate standards that contain the management functions needed to satisfy particular requirements.

This set of management functions is referred to as the System Management Functions (SMFs). They provide a generic platform of common network management capabilities available to any management application. For example, the management services control function [MSC] may be used to report events to satisfy FM, PM, AM, and SM requirements. The log control function [LCF] may be used to satisfy both FM and SM requirements.

The following schematic depicts the functional hierarchy of SMFs and SMFAs. There are seven common SMFs. They provide much of the network management capabilities needed by CM and FM. When additional requirements are identified in other SMFAs, additional SMFs may be developed.

```
                              Applications
                                   |    various requirements result in
                                   |    various grouping of functional
                                   |    management area
                 +--------+--------+---------+--------+
                 |        |        |         |        |
===================================================================
         |
         |    +----+    +----+   +----+   +----+   +----+
  SMFAs  |    | FM |    | CM |   | PM |   | SM |   | AM |
         |    +----+    +----+   +----+   +----+   +----+
         |      |         |        |        |        |
===================================================================
SMFs     |                     PLATFORM
         |  +-------------+  +----------------------+ +-----------+
         |  |Event Control|  |Service Access Control| |Log Control|
         |  +-------------+  +----------------------+ +-----------+
         |
         |  +---------------+ +-----------------+  +------------+
         |  |Error Reporting| |Error Information|  |Relationship|
         |  +---------------+ |    Retrieval    |  | management |
         |                    +-----------------+  +------------+
         |
         |  +----------------+   +-----------------+ +--------------+
         |  |State Management|   |Object Management| | Confidence & |
         |  +----------------+   +-----------------+ |  Diagnostic  |
         |                                           |     Test     |
         |                                           +--------------+
         |                     (etc ....)
==================================================================
         |
         |                      CMIS
         |
================================================================
         |            Lower Layer Services
```

The following System Management Functions are undergoing standardization:

(1) Object Management Function [OMF]

(2) State Management Function [SMF]

(3) Relationship Management Function [RMF]

(4) Error Reporting and Information Retrieval Functions

[ERIRF]:

a. Error Reporting Function

b. Information Retrieval Function

(5) Management Service Control Functions [MSC]:

a. Event Control Function

b. Service Access Control Function

(6) Event Log Control Function [LCF]

(7) Confidence and Diagnostic Test Function [FMWD].

For the NIST NMSIG Phase 1 network management agreements, it is agreed that only the first six functions will be supported. For each supported System Management Function (Sections 18.5.1-18.5.6, below), agreements pertinent to the accompanying management communication services are given.

18.5.1 Object Management Function

Editor's Note: Tutorial Material. This System Management Function provides the management of Objects in an Open System Environment. In this environment, a managed object (MO) can be identified as a recognizable unit of a data processing resource or a data communications resource that can be remotely managed through the use of OSI management communication ervices (Section 18.6). An MO may be a physical item of equipment, a software component, or a combination of such. Each MO has a set of management information associated with it and an MO identifier by which the set of management information can be manipulated through the use of the OSI management communications services.

The NMSIG Phase 1 network management agreements support all the operations and services in the object management standard [OMF], i.e.,

o Object creation operation
o Object deletion operation
o Object renaming operation
o Attribute reading operation
o Attribute changing operation
o Object listing operation
o Enrol Object Service
o Deenrol Object Service
o Reenrol Object Service

- o Attribute Change Event Report Service
- o Add Value Event Report Service
- o Remove Value Event Report Service

For the operations listed above, the associated managed object classes must be registered in the OSI Management Information Library (MIL) [OSIMIL]. Otherwise, attempted use of any of these operations must result in a <noSuchObjectClass> error returned by the performing CMISE-service-user.

For the services listed above, the Event Reporting Control Model (Section 18.5.5) applies.

Editor's Note: Tutorial Material. The ISO standard emphasizes the differences between an operation and a service. The distinction is that an operation will use a CMIS service directly without introducing a new ASN.1 definition for the CMIS service, whereas a service does introduce a new ASN.1 definition. The following sections describe the operations and services.

Note to the Editor: Please provide a definitive reference that substantiates the above assertion.

18.5.1.1 Object Creation Operation Agreements:

Editor's Note: Tutorial Material. The Object Creation operation is used by a managing system to ask a managed system to create an instance of a managed object in the managed system.

The attribute values of the created object must be obtained, in the following order, from:

(1) any default values defined for the managed object class in the Management Information Library (elsewhere in this chapter) [OSIMIL] and therefore known to the Agent,

(2) a reference object instance, where provided by the invoker,

(3) values explicitly provided by the invoker in the <attributeList>.

The following agreements and clarifications pertinent to Section 8.1 of the base standard [OMF] and regarding the semantics of the confirmed CMIS M-CREATE service (Section 8.3.4 in [CMIS]) are supported by the Phase 1 network management IAs. All CMIS parameters are mandatory, except where noted below.

CMIS M-CREATE request parameters:

<invokeIdentifier>

<managedObjectClass>

<managedObjectInstance> (1) If this parameter is used in the request, it will identify the instance of the object class of the instance to be created by providing the DiscriminatorID and names of Superiors possibly.

(2) Otherwise, the performing CMISE-service-user will assign a value to this identification of this instance.

The managed object definition will specify whether the manager or agent will provide the <managedObjectInstance> value. This means that for a given object class either (1) must always be used or (2) must always be used (refer to Section 6.1.5.2.1 of [MIM]).

<accessControl> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<referenceObjectInstance> When this parameter is used by the invoking CMISE-service-user, it must specify an existing object instance of the same class as the object being created.

<attributeList> This parameter must provide the attribute(s) and their initial value(s) for the object instance if they are neither provided as defaults in the object definition nor obtained from the reference object. Otherwise, a CMIS error of <invalidAttributeValue> will be

returned (Section 8.3.4.1.8 of [CMIS]).

Editor's Note: If an error code of <missingAttributeValue> is defined in the standard in the future, it will be adopted here.

Editor's Note: The standards as written do not show any way (via the ATTRIBUTE macro) to define a default value for an attribute. We are assuming that it is possible to define such default values, however, it is not required that this be done for EVERY attribute.

<u>CMIS M-CREATE response parameters:</u>

<invokeIdentifier>

<managedObjectClass>

<managedObjectInstance> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<attributeList> This parameter specifies all of the created object attributes and values.

Editor's Note: It is anticipated that Section 18.6 of this chapter will define this in common for all M-CREATE's, at which time, the text here can refer to that section directly.

<currentTime> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

Editor's Note: Can any manager other than the manager that created the object manage this new object?

Over which association(s) can this new object be managed?

- o the current association?
- o other extant associations?
- o new associations?

This issue is to be determined as part of the general association policy.

Note that there is a more general problem which applies to access rights and ownership of the created objects. Maybe the protocol section should set the policy for the CMIS M-CREATE service?

18.5.1.2 Object Deletion Operation Agreements:

Editor's Note: Tutorial Material. The Object Deletion operation is used by a managing system to ask a managed system to delete an instance of a managed object in the managed system.

The following agreements and clarifications pertinent to Section 8.3 of the base standard [OMF] and regarding the semantics of the confirmed CMIS M-DELETE service (Section 8.3.5 in [CMIS]) are supported by the Phase 1 network management IAs. All CMIS parameters are mandatory, except where noted below.

CMIS M-DELETE request parameters:

<invokeIdentifier>

<baseManagedObjectClass> (1) If scoping is used for multiple object selection, this parameter identifies the managed object class that is to be used as the starting point for the selection of

managed objects on which the filter is to be applied.

(2) If scoping is used to select the base object only, this parameter identifies the class of the object instance to be deleted.

Editor's Note: <n> level delete is to be discussed further.

<baseManagedObjectInstance> (1) If scoping is used for multiple object selection, this parameter identifies the instance of the managed object that is to be used as the starting point for the selection of managed objects defined by <scope> on which the filter is to be applied.

(2) If scoping is not used to select the base object only, this parameter identifies the DiscriminatorID, and the name(s) of any superior(s) of the object instance to be deleted.

Editor's Note: <n> level delete is to be discussed further.

<accessControl> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<synchronization> <BestEffort> is required.

<scope> This parameter defines the level(s) relative to the base managed object from which objects will be deleted. This is used for deleting multiple object instances. It will be set to <baseObject> if single object selection is used, or set to <n> to specify the depth of the search, or specify the whole subtree.

Editor's Note: <n> level delete is to be discussed further.

<filter>

CMIS M-DELETE response parameters:

<invokeIdentifier>

<linkedIdentifier>

<managedObjectClass> <managed Object Instance>	Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to these parameters.
<currentTime>	Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

18.5.1.3 Object Renaming Operation Agreements:

Editor's Note: Tutorial Material. The Object Renaming operation is used by a managing system to ask a managed system to rename an instance of a managed object in the managed system.

Editor's Note: This section is very controversial. We do not feel that we have a clear understanding of what an OBJECT NAME is. The standard seems to imply that the OBJECT NAME is the distinguishing attribute defined in the object definition. If this is so, it is a <readonly> attribute, and cannot be changed by a CMIS M-SET service. The group feels that it is more appropriate to use a specific CMIS M-ACTION service to carry out this specific operation. The group will submit comments, in this regard, to ISO by the March 1989 ANSI meeting.

The following section aligns with the current standard and may change.

Editor's Note: It is anticipated that this service will have side effects, especially with regard to associations where objects existed with old names, regarding operations with the objects under old names, and regarding discriminator object changes at the managed object's systems as well as the destination system.

The following agreements and clarifications pertinent to Section 8.5 of the base standard [OMF] and regarding the semantics of the confirmed CMIS M-SET service (Section 8.3.2 in [CMIS]) are supported by the Phase 1 network management IAs. All CMIS parameters are mandatory, except where noted below.

CMIS M-SET request parameters:

<invokeIdentifier>

<mode> This parameter will be set to <confirmed>.

<baseManagedObjectClass> Since use of scoping in this service to select multiple objects is not permitted, this parameter identifies the object class of the object instance to be renamed.

<baseManagedObjectInstance> Since use of scoping in this service to select multiple objects is not permitted, this parameter identifies the instance name, and the name(s) of any Superior(s), of the object instance to be renamed.

<accessControl> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<synchronization> <bestEffort> is required.

<scope> <baseObject> is required.

<filter> Use of this parameter is not permitted.

<attributeList>
This parameter will contain only the <distinguishedName> attribute id and new value for the managed object instance. If the attribute is not provided, this service will be rejected via a CMIS ERROR <SetListError> with error code <NoSuchAttributeId>.

CMIS M-SET response parameters:

<invokeIdentifier>

<linkedIdentifier>
This parameter is not used.

<managedObjectClass>
<managedObjectInstance>
Refer to Section 18.6 (ManagementCommunications) of this chapter for agreements pertaining to these parameters.

<attributeList>
This parameter, provided by the managed system, returns the list of <distinguishedName> attribute ids and new values for the managed object instance.

If an error occurs in the retrieval process, a CMIS ERROR <GetListError> will be reported. (Section 8.3.1.1.14 of [CMIS])

<currentTime>
Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

18.5.1.4 Attribute Reading Operation Agreements:

Editor's Note: Tutorial Material. The Attribute Reading operation is used by a managing system to ask a managed system to return the specified attribute values for an instance of a managed object in the managed system.

The following agreements and clarifications pertinent to Section 8.8 of the base standard [OMF] and regarding the semantics of the confirmed CMIS M-GET service (Section 8.3.1 in [CMIS]) are supported by the Phase 1 network management IAs. All CMIS parameters are mandatory, except where noted below.

CMIS M-GET request parameters:

<invokeIdentifier>

<baseManagedObjectClass>

<baseManagedObjectInstance>

<accessControl> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<synchronization> <bestEffort> is required.

<scope>
<filter>

<attributeIdList> This parameter list will contain the list of attributes to be retrieved. If the list is not provided, all attributes will be retrieved.

CMIS M-GET response parameters:

<invokeIdentifier>

<linkedIdentifier>

<managedObjectClass>
<managedObjectInstance> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to these parameters.

<currentTime> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<attributeList> This parameter, provided by the managed system, returns

the list of requested attributes and their values.

If an error occurs in the retrieval process, a CMIS ERROR <GetListError> will be reported. The list will include all requested attributes, and for each attribute there will be chosen either the attribute value (choice of Tag [1]) for the successful retrieval of an attribute, or an attributeIdError (choice of Tag [0]) for the failure case. Refer to Section 8.3.1.1.14 in [CMIS] for more information.

18.5.1.5 Attribute Changing Operation Agreements:

Editor's Note: Tutorial Material. The Attribute Changing operation is used by a managing system to ask a managed system to change the values of one or more specified attributes for a managed object instance in the managed system.

The following agreements and clarifications pertinent to Section 8.9 of the base standard [OMF] and regarding the semantics of the confirmed CMIS M-SET service (Section 8.3.2 in [CMIS]) are supported by the Phase 1 network management IAs. All CMIS parameters are mandatory, except where noted below.

CMIS M-SET request parameters:

<invokeIdentifier>

<baseManagedObjectClass>

<baseManagedObjectInstance>

<accessControl> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<synchronization> <bestEffort> is required.

<scope>

<filter>

<attributeList> This parameter will contain the list of attributes whose values are to be modified.

CMIS M-SET response parameters:

<invokeIdentifier>

<linkedIdentifier>

<managedObjectClass>
<managedObjectInstance> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to these parameters.

<currentTime> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<attributeList> This parameter, provided by the managed system, returns the list of modified attributes and values.

If an error occurs in the process, a CMIS ERROR <SetListError> will be reported. The list will include all attributes requested for modification, and for each one, choose either an <attribute> (choice of Tag [1]) for the successful modification of an attribute, or an <attributeError> (choice of Tag [0]) for the failure case. Refer to (Section 8.3.2.1.14 in [CMIS]) for more information.

18.5.1.6 Object Listing Operation Agreements:

Editor's Note: Tutorial Material. The Object Listing operation is used by a managing system to ask a managed system to retrieve the names of a defined set of managed objects in the managed system. Other attributes can also be retrieved by specifying the attribute names in the request.

The following agreements and clarifications pertinent to Section 8.7 of the base standard [OMF] and regarding the semantics of the confirmed CMIS M-GET service (Section 8.3.1 in [CMIS]) are supported by the Phase 1 network management IAs. All CMIS parameters are mandatory, except where noted below.

Editor's Note: This section is controversial because we must again work with the problematic definition of an OBJECT NAME. Comments will be submitted to the ANSI meeting in March 1989.

The following section assumes that the OBJECT NAME is the same as the <distinguishedName> attribute.

CMIS M-GET request parameters:

<invokeIdentifier>

<baseManagedObjectClass>

<baseManagedObjectInstance>

<accessControl> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<synchronization> <bestEffort> is required.

<scope>

<filter>

<attributeIdList> (1) If this parameter is used, the list will include at least the <distinguishedName> attribute.

(2) If the list is not provided, all attributes

including the <distinguishedName> will be retrieved.

CMIS M-GET response parameters:

<invokeIdentifier>

<linkedIdentifier>

<managedObjectClass>
<managedObjectInstance>
Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to these parameters.

<currentTime>
Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<attributeList>
This parameter, provided by the managed system, returns the attribute ids and values for the specified attributes (including the object name(s) of the requested managed object's <distinguishedName>).

If an error occurs in the retrieval process, a CMIS ERROR <GetListError> will be reported. (Section 8.3.1.1.14 in [CMIS])

18.5.1.7 Object Management Services Agreements

Editor's Note: Tutorial Material. Each of the Object Management Services uses an unconfirmed M-EVENT-REPORT CMIS service (Section 8.3.1 in [CMIS]) to convey its information.

The Event Reporting Model (see Section 18.5.5 in this chapter and [ERIRF], [MSC], [DSO]) defines the following procedure: The agent receives notifications from the appropriate managed objects and causes these potential event reports to be checked against all Event Forwarding Discriminators. The result of this sieve process will yield zero, one or more event reports to be transmitted to the destination systems (according to the attributes of the relevant discriminators) according to the services defined in the subsequent sub-sections. One discriminator may cause the sending of multiple event reports, if the multi-valued attribute ManagementUserIdentification contains multiple AEtitles. Additionally, multiple discriminators may filter the same potential event reports and hence generate multiple event reports.

Editor's Note: Some of the text in this paragraph should be moved to the discussion of the Event Reporting Model in 18.5.4, while retaining some here.]

The following agreements and clarifications pertinent to Sections 8.2, 8.4, 8.6, 8.10, 8.11, and 8.12 of the base standard [OMF] and regarding the semantics of the CMIS M-EVENT-REPORT parameters are supported by the Phase 1 network management agreements for all the Object Management Services Sections 8.5.1.7.1 through 8.5.1.7.6, below):

<invokeIdentifier>	
<mode>	This parameter is set to <unconfirmed>.
<managedObjectClass> <managedObjectInstance>	Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to these parameters.

18.5.1.7.1 Enrol Object Service

Editor's Note: Tutorial Material. The Enrol Object Service is used by the managed system to request the reporting of a creation event of a new managed object instance to a managing system.

In addition to the agreements and clarifications in Section 18.5.1.7, the following agreements and clarifications pertinent to Section 8.2 of the base standard [OMF] and regarding the semantics of the CMIS M-EVENT-REPORT parameters are supported by the Phase 1 network management agreements:

CMIS M-EVENT-REPORT request parameters:

<eventType> This parameter identifies the <enrolObject> Event whose object identifier is defined in [OMF].

<eventTime> This parameter specifies the time when the new instance was created.

<eventArgument> This parameter is not used for this service.

18.5.1.7.2 Deenrol Object Service:

Editor's Note: Tutorial Material. The Deenrol Object Service is used by the managed system to report the deletion of a managed object instance to a managing system.

In addition to the agreements and clarifications in Section 18.5.1.7, the following agreements and clarifications pertinent to Section 8.4 of the base standard [OMF] and regarding the semantics of the CMIS M-EVENT-REPORT parameters are supported by the Phase 1 network management agreements:

<eventType> This parameter identifies the <deenrolObject> Event whose object identifier is defined in [OMF].

<eventTime> This parameter specifies the time when the object instance was deleted.

<eventArgument> This parameter is not used for this service.

18.5.1.7.3 Reenrol Object Service:

Editor's Note: Tutorial Material. The Reenrol Object Service is used by the managed system to report the renaming of a managed object instance to a managing system.

In addition to the agreements and clarifications in Section 18.5.1.7, the following agreements and clarifications pertinent to Section 8.6 of the base standard [OMF] and regarding the semantics of the CMIS M-EVENT-REPORT parameters are supported by the Phase 1 network management agreements:

<eventType> This parameter identifies the <reenrolObject> Event whose object identifier is defined in [OMF].

<eventTime> This parameter specifies the time when the object instance was renamed.

<eventArgument> This parameter will contain the tuple <objectClass, oldObjectInstance, newObjectInstance) for the newly renamed managed object instance (Section 9 of [OMF]).

18.5.1.7.4 Attribute Change Event Report Service:

Editor's Note: Tutorial Material. The Attribute Change Event Report Service is used by the managed system to report an attribute change event to the managing system. The attribute change event indicates a change in the value(s) of one or more attributes of a managed object.

In addition to the agreements and clarifications in Section 18.5.1.7, the following agreements and clarifications pertinent to Section 8.10 of the base standard [OMF] and regarding the semantics of the CMIS M-EVENT-REPORT parameters are supported by the Phase 1 network management agreements:

<eventType> This parameter identifies the <attributeChange> Event whose object identifier is defined in [OMF].

<eventTime> This parameter specifies the time when the attribute value was changed in the object instance.

<eventArgument> This parameter will contain the tuple <attributeId, oldAttributeValue, newAttributeValue> (Section 9 in [OMF]).

18.5.1.7.5 Add Value Event Report Service:

Editor's Note: Tutorial Material. The Add Value Event Report Service is used by the managed system to report the addition of a value to a multi-valued attribute of a managed object at an open system.

Editor's Note: This service presumes the existence of the PDAD2 to CMIS/P

In addition to the agreements and clarifications in Section 18.5.1.7, the following agreements and clarifications pertinent to Section 8.11 of the base standard [OMF] and regarding the semantics of the CMIS M-EVENT-REPORT parameters are supported by the Phase 1 network management agreements:

<eventType> This parameter identifies the <addValue> Event whose object identifier is defined in [OMF].

<eventTime> This parameter specifies the time when the object instance whose attribute value was added to the object instance.

<eventArgument> This parameter will contain the tuple <attributeId, newAttributeValue>, where <newAttributeValue> is the attribute value just added. (Section 9 of [OMF]).

18.5.1.7.6 Remove Value Event Report Service:

Editor's Note: Tutorial Material. The Remove Value Event Report Service is used by the managed system to report the removal of a value from a multi-valued attribute of a managed object at an open system.

Editor's note: This section presumes the existence of the PDAD2 to CMIS/P

In addition to the agreements and clarifications in Section 18.5.1.7, the following agreements and clarifications pertinent to Section 8.12 of the base standard [OMF] and regarding the semantics of the CMIS M-EVENT-REPORT parameters are supported by the Phase 1 network management agreements:

<eventType>	This parameter identifies the <removeValue> Event whose object identifier is defined in [OMF].
<eventTime>	This parameter specifies the time when the attribute value was deleted from the object instance..
<eventArgument>	This parameter will contain the tuple <attributeId, oldAttributeValue>, where <oldAttributeValue> is the attribute value just deleted. (Section 9 of [OMF]).

18.5.2 State Management Function

Editor's Note: Tutorial Material. The State Management Function provides for the examination, setting and notification of changes in the management state of existing managed objects. The managed state of a managed object represents its instantaneous condition of availability and operability from the point of view of configuration management. The managed state consists of (1) operational state, and (2) administrative state.

A list of the possible combinations of the operational and administrative states is given in (Table 1, Section 7.2, [SMF]). The purpose of this list is to control the availability of a managed object, and to make visible information about the general availability of a managed object.

The Phase 1 network management agreements support the two operations and one service defined in the base standard (Section 8 of [SMF]), i.e.,

- o State reading operation
- o State changing operation
- o State change reporting service.

For the State change reporting Service, the Event Reporting Control Model (Section 18.5.5) applies.

18.5.2.1 State Reading Operation Agreements:

Editor's Note: Tutorial Material. The state reading operation enables the managing system to request the managed system to return the values of the configuration state attributes which include the operational and/or administrative state(s) of one or more instances of managed object(s).

The following agreements and clarifications pertinent to Section 8.1 of the base standard [SMF] and regarding the semantics of CMIS M-GET service (Section 8.3.1 in [CMIS]) are supported by the Phase 1 network management IAs. All CMIS parameters are mandatory, except where noted below.

CMIS M-GET request parameters:

<invokeIdentifier>

<baseManagedObjectClass>

<baseManagedObjectClass>

<accessControl> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<synchronization> <bestEffort> is required.

<scope>

<filter>

<attributeIdList> This parameter list will include the list of state attribute(s) (<operational state>, <administrative state>) which the managing system would like to obtain. If the list is not provided, all attributes including the

state attributes will be retrieved.

CMIS M-GET response paramaters:

<invokeIdentifier>

<linkedIdentifier>

<managedObjectClass>
<managedObjectInstance>
Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to these parameters.

<currentTime>
Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<attributeList>
This parameter, provided by the managed system, returns the list of requested state attributes and their values.

If an error occurs in the retrieval process, a CMIS ERROR <GetListError> will be reported. (Section 8.3.1.1.14 in [CMIS])

18.5.2.2 State Changing Operation Agreements:

Editor's Note: Tutorial Material. The state changing operation enables the managing system to request the managed system to change the value of the administrative state attribute of one or more instances of a managed object(s).

The following agreements and clarifications pertinent to Section 8.2 of the base standard [SMF] and regarding the semantics of CMIS M-SET service (Section 8.3.2 in [CMIS]) are supported by the Phase 1 network management IAs. All CMIS parameters are mandatory, except where noted below.

CMIS M-SET request parameters:

<invokeIdentifier>

<baseManagedObjectClass>

<baseManagedObjectInstance>

<accessControl>
Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<synchronization>
<bestEffort> is required.

<scope>

<filter>

<attributeList>
This parameter will include the state attribute (<administrativeState>) and its desired new value.

CMIS M-SET response parameters:

<invokeIdentifier>

<linkedIdentifier>

<managedObjectClass>
<managedObjectInstance>
Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to these parameters.

<currentTime>
Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<attributeList>
This parameter, provided by the managed system, returns the attribute ids and values for the specified attributes (including the modified state attribute).

If an error occurs in the process, a CMIS ERROR <SetListError> will be reported. (Section 8.3.2.1.14 in [CMIS])

18.5.2.3 State Change Reporting Service Agreements:

Editor's Note: Tutorial Material. The state change reporting service enables the managed system to report the change of a state attribute (i.e. either the operational state or administrative state) of a managed object to a managing system.

The following agreements and clarifications pertinent to Section 8.3 of the base standard [SMF] and regarding the semantics of CMIS M-EVENT-REPORT service (Section 8.2.1 in [CMIS]) are supported by the Phase 1 network management IAs. All CMIS parameters are mandatory, except where noted below.

<invokeIdentifier>	
<mode>	This parameter is set to <unconfirmed>.
<managedObjectClass> <managedObjectInstance>	Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to these parameters.
<eventType>	This parameter identifies the <stateChange> Event whose object identifier is defined in [DMA]
<eventTime>	This parameter specifies the time when the object instance state attribute value was changed.
<eventArgument>	This parameter will contain the tuple <oldConfigurationState, newConfigurationState> for the newly changed state object instance [DMA].

18.5.3 Relationship Management Function

18.5.3.1 Relationship Management Model:

18.5.3.2 Relationship Management using the INDIRECT MODEL:

18.5.3.2.1 Relationship creation:

18.5.3.2.2 Relationship deletion:

18.5.3.2.3 Relationship changing:

18.5.3.2.4 Relationship listing:

18.5.3.2.5 Related object listing:

18.5.3.2.6 Relationship creation reporting Service:

18.5.3.2.7 Relationship deletion reporting Service:

18.5.3.2.8 Relationship change reporting Service:

18.5.3.3 Relationship Management using the DIRECT MODEL:

18.5.4 Error Reporting and Information Retrieval Functions:

18.5.4.1 Error Reporting Function:

18.5.4.1.1 Error Reporting Model:

18.5.4.1.2 Support Managed Object:

18.5.4.1.3 Error Reporting Service:

18.5.4.2 Information Retrieval Function:

18.5.4.2.1 Information Retrieval Service:

18.5.5 Management Service Control Functions:

Editor's Note: Tutorial Material. There are two control functions in this category to provide the ability to specify criteria under which event reporting and access control for requested operations can be controlled. The two functions are: (1) Event Reporting Control Function, and (2) Service Access Control Function.

Unless there is an a priori agreement between the managing and the managed systems, the two controlling functions are mandatory. However, a priori agreements are usually a private matter between managing and managed systems, and, as such, are not subject to standardization, and, therefore, lie outside the scope of this document.

The NMSIG Phase 1 network management agreements support only the Event Reporting Control Function. The Service Access Control Function is for further study.

18.5.5.1 Event Reporting Control Function:

Editor's Note: Tutorial Material. The Event Reporting Control function provides services by which event reporting can be distributed and controlled. Event distribution means the selection of chosen events to be reported to some designated system(s) or process(es) within some selected time period. These selections are done by a filtering process using the "DiscriminatorConstruct" attribute of the "Event Forwarding Discriminator" object. Event Reporting Control is the ability to initiate, terminate, suspend, or resume event reporting through the manipulation of an Event Forwarding Discriminator object specified in Section 18.5.5.1.1. In addition, Event Reporting Control can further alter event distribution behavior by changing the distribution attributes in an Event Forwarding

Discriminator object (DiscriminatorConstruct, BeginTime and EndTime etc...).

The following sections contain the NMSIG Phase 1 network management agreements pertaining to the Event Reporting Control Model [RMF], the Support Managed Object to facilitate the Event Reporting Control Function [RMF], and the following services (defined in [RMF]):

- Initiate event reporting service
- Terminate event reporting service
- Suspend event reporting service
- Resume event reporting service
- Modify event forwarding discriminator attributes service
- Retrieve event forwarding discriminator attributes service.

18.5.5.1.1 Event Reporting Control Model Agreements:

The Event Reporting Control function is based on the following assumptions, pictured below:

(1) There is (at least) one managed object capable of generating notifications.

(2) There exists a conceptual event detection and processing function which receives the local notifications and forms potential event reports.

(3) There exist Event Forwarding Discriminator objects which are used for determining whether potential event reports can become real event reports which are then emitted from the open system.

(4) There exists a conceptual process which guides all potential event reports to all Event Forwarding Discriminators for evaluation.

(5) There exists a conceptual process which evaluates the potential event reports using the Event Forwarding Discriminator attributes (DiscriminatorConstruct, BeginTime, EndTime, Destination ...) to determine whether the potential event reports are to be reported to the specified destination system(s).

```
                               Event Forwarding Discriminator
                                   Control Functions
          +--------------+     (Initiate, Terminate, Suspend,
  +---------------+ M.O. |      Resume, Update, etc...)
  | Managed Object|------+                |  |
  +---------------+  |                    |  |
         |           |                    |  |
         | Notifications                  |  |
         |           |                    |  |
         |           |                    |  |
+--------|-----------|--------------------|  |---------+
| Agent  v           v                    v  v         |
|      +----------------+            +--------------+  |
|      | Event Detection|----------> | Event Fwding |-------->
|      |       and      | potential  | Discriminator|  | Event
|      |    Processing  |  Events    |  Processing  |  |Reports
|      |                |----------> |              |-------->
|      +----------------+            +--------------+  |
|                                                      |
+------------------------------------------------------+
```

18.5.5.1.2 Support Managed Object - Event Forwarding Discriminator Agreements

Editor's Note: Tutorial Material. The Event Report Discriminator is a management service control discriminator which is a managed object providing for specification of criteria relevant to selecting events of interest to be reported to other open systems. The criteria must be satisfied by potential event reports related to managed objects before the event report is forwarded to a particular destination. That destination is also specified by the discriminator and is the address of a remote managing process.

One Event Report Discriminator is defined for each destination system to which the event reports are to be sent.

Editor's Note: Tutorial Material. The Event Forwarding Discriminator has the following attributes:

(1) DiscriminatorID: This attribute uniquely identifies the discriminator.

(2) DiscriminatorConstruct: This attribute specifies the conditions which define when an event report should be generated after an event occurs. Each event which occurs in an event generating system has to be evaluated for passing the filter construct. Only those events that pass (match) the filter will result in an event report being sent to the destination system(s).

(3) ManagementUserIdentification: This attribute identifies the systems on whose behalf the event report is performed. This usually indicates the managing system.

Editor's note: Should the Phase 1 network management IA's limit this to containing only a single system at a time? This would mean we would not require use of PDAD2 for CMIS/P.

(4) Discriminator State: This attribute specifies the state in which the Event Report Discriminator object is to be created. The Discriminator object may be created in a "locked" or "unlocked" state.

(5) Begin Time: This attribute identifies the beginning time of a 24 hour interval during which the event report service is active.

(6) End Time: This attribute identifies the ending time of a 24 hour interval during which the event report service is available.

An example: If Begin Time = 8:00 AM and End Time = 5 PM, then event reports will only be sent between the hours of 8:00 AM through 5:00 PM on the basis of this discriminator.

18.5.5.1.3 Initiate Event Reporting Service Agreements:

Note to the Editor: Tutorial material in all subsequent sections needs to be scanned for scenario information and that material should be provided to the scenario section editor.

Editor's Note: Tutorial Material. A user at a managing system may desire that particular events generated at an event generating system be reported to a destination system. To achieve this, the user, from the managing system, will need to create Event Report Discriminator objects for those particular events with the proper parameters at the event generating system.

Each Event Report Discriminator object must include an EventReportConstruct which specifies the desired filtering conditions under which the designated event should be reported to the destination system.

A managing system must issue a single M-CREATE CMIS service request to an event generating system to create a single Event Report Discriminator. Multiple discriminators require multiple M-CREATE CMIS service requests.

Editor's Note: Once the discriminator object is created, is there an implicit assumption that the newly created object forms part of the context implied by the current association context? Can the discriminator object be managed by applications using other associations other than the one over which the CMIS M-CREATE request was issued, or do they need to reassociate? This issue will be determined during the association policy discussions.

Note to the editor: In the following sections, mention is made of the Event Report Discriminator, the Event Report Forwarding Discriminator, and the Discriminator. Are the terms synonymous? If so, please use one term always. If not, please define the terms.

The following agreements and clarifications pertinent to Section 8.1 of the base standard [MSC] and regarding the semantics of the confirmed CMIS M-CREATE service (Section 8.3.4 of [CMIS]) are supported by the Phase 1 network management IAs. All CMIS parameters are mandatory, except where noted below.

<u>CMIS M-CREATE request parameters:</u>

<invokeIdentifier>

<managedObjectClass> The parameter value will always be the <discriminator> class, which is the corresponding class for an Event Report Forwarding Discriminator. This parameter must be included in the request.

<managedObjectInstance> (1) If this parameter is used in the request, it will identify the instance of the discriminator class by providing the DiscriminatorID and names of any superiors.

(2) Otherwise, the performing CMISE-service-user will assign a value to identify the instance.

Editor's Note: Should we agree on using (1) always in the request?

Note to the Editor: Incorporate comments from the Object Creation section, later on.

<accessControl> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<referenceObjectInstance> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<attributeList> This field refers to the Event Report Forwarding Discriminator object

attributes (Section 18.5.5.1.2.1 of this chapter). Any attributes provided by the CMIS-service-user will be used to initialise the corresponding attributes for the newly created instance.

The <discriminatorState> attribute is set to "unlocked" by default.

CMIS M-CREATE response parameters:

<invokeIdentifier>

<managedObjectClass> Same as request

<managedObjectInstance> This parameter is always returned by the response to indicate the instance name of the newly created object.

<attributeList> This parameter specifies ALL the object attributes and values for the NEWLY created Event Report Forwarding Discriminator.

<currentTime> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

18.5.5.1.4 Terminate Event Reporting Service Agreements:

Editor's Note: Tutorial Material. A user in a managing system can use this service to turn off the reporting of events from a specific event generating system.

To achieve that, the user will delete the discriminator object(s) of the unwanted event(s) on the event generating system. The absence of such a discriminator will not stop the generation of potential error reports caused by the managed object, it simply disables event reporting of the particular potential events from the event generating system.

A managing system must issue a single M-DELETE CMIS service request to the event generating system to delete exactly one Event Report Discriminator. Multiple M-DELETE CMIS service requests are needed to delete multiple discriminators.

The following agreements and clarifications pertinent to Section 8.2 of the base standard [MSC] and regarding the semantics of the confirmed CMIS M-DELETE service (Section 8.3.5 of [CMIS]) are supported by the Phase 1 network management IAs. All CMIS parameters are mandatory, except where noted below.

CMIS M-DELETE request parameters:

<invokeIdentifier>

<baseManagedObjectClass>

<baseManagedObjectInstance>

<accessControl> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<synchronization> <BestEffort> is required.

<scope>

<filter>

CMIS M-DELETE response parameters:

<invokeIdentifier>

<linkedIdentifier>

<managedObjectClass>
<managedObjectInstance> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to these parameters.

<currentTime> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

18.5.5.1.5 Suspend Event Reporting Service Agreements:

Editor's Note: Tutorial Material. This service temporarily stops event reports from being sent from the event generating system to the destination system, yet retains the ability to resume the reporting if desired.

To suspend event reporting, a managing system must issue an M-SET CMIS service request to the event generating system to change the value of the <DiscriminatorState> attribute to "locked".

When the <DiscriminatorState> attribute is "locked", any events that would normally occur for this discriminator are discarded and NOT queued up for later transmission.

The following agreements and clarifications pertinent to Section 8.3 of the base standard [MSC] and regarding the semantics of the confirmed CMIS M-SET service (Section 8.3.2 of [CMIS]) are supported by the Phase 1 network management IAs. All CMIS parameters are mandatory, except where noted below.

CMIS M-SET request parameters:

<invokeIdentifier>

<mode> This parameter will be set to <confirmed>.

<baseManagedObjectClass>

<baseManagedObjectInstance>

<accessControl> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<synchronization> <bestEffort> is required.

<scope>

<filter>

<attributeList> This parameter will include the Event Forwarding Discriminator attribute <discriminatorState> with the value of the attribute to be to "locked". (See Section 18.5.5.1.2 of this chapter)

CMIS M-SET response parameters:

<invokeIdentifier>

<linkedIdentifier>

<managedObjectClass>
<managedObjectInstance> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to these parameters.

<currentTime> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

18.5.5.1.6 Resume Event Reporting Service Agreements:

Editor's Note: Tutorial Material. This service enables event reporting for particular types of events, thereby permitting events to be sent from a specific event generating system to a specific destination system.

To resume event reporting, the managing system must issue an M-SET CMIS service request to an event generating system to change the <discriminatorState> attribute to <Unlocked>.

The following agreements and clarifications pertinent to Section 8.4 of the base standard [MSC] and regarding the semantics of the confirmed CMIS M-SET service (Section 8.3.2 of [CMIS]) are supported by the Phase 1 network management IAs. All CMIS parameters are mandatory and are as specified in Section 18.5.5.1.5, with the following difference.

<attributeList>	This parameter will contain the Event Forwarding Discriminator attribute <discriminatorState>. (See Section 18.5.5.1.2 of this chapter). The value of the attribute will be set to "unlocked".

18.5.5.1.7 Modify Event Forwarding Discriminator Attributes Service Agreements:

Editor's Note: Tutorial Material. A managing system can change the conditions of event reporting for some selected events by changing the values of the Event Forwarding Discriminator attributes which are used in the processing associated with event distribution and control. For example, the user may want to move/modify the reporting of a specific type of event to a different destination system, or change the frequency of the event reporting. To achieve such results, a managing system will need to modify the value of the <managementUserIdentification> and/or <DiscriminatorConstruct> attributes to reflect the new needs. This service can be used for locked or unlocked Event Forwarding Discriminator objects.

To change attributes of one specific Event Forwarding Discriminator in one specific event generating system, a managing system must issue a single M-SET CMIS service request to the event generating system. Changes to multiple discriminators in a single event generating system require multiple M-SET CMIS service requests.

The following agreements and clarifications pertinent to Section 8.5 of the base standard [MSC] and regarding the semantics of the confirmed CMIS M-SET service (Section 8.3.2 of [CMIS]) are supported by the Phase 1 network management IAs. All CMIS parameters are mandatory, except where noted below.

CMIS M-SET request parameters:

<invokeIdentifier>

<mode> This parameter will be set to <confirmed>.

<baseManagedObjectClass>

<baseManagedObjectInstance>

<accessControl> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<synchronization> <bestEffort> is required.

<scope>

<filter>

<attributeList> This parameter will specify the Event Forwarding Discriminator attributes to be modified. The modifiable attributes are:
<DiscriminatorConstruct>,
<Management User Identification>,
<Discriminator State>,
<Begin Time>, <End Time>.

Editor's note: This parameter is going to be replaced by the <modificationList> parameter, once PDAD2 for CMIS/P is adopted.

CMIS M-SET response parameters:

<invokeIdentifier>

<linkedIdentifier>

<managedObjectClass>
<managedObjectInstance> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to these parameters.

<attributeList> This parameter will specify the Event Forwarding Discriminator attributes that were modified.

<currentTime> Refer to Section 18.6 (Management Communications) of

this chapter for agreements pertaining to this parameter.

18.5.5.1.8 Retrieve Event Forwarding Discriminator Attributes Service Agreements:

To examine the Event Reporting Discriminator parameters associated with a specific event, a managing system must issue an M-GET CMIS service request to an event generating system to retrieve the values of specific discriminator attributes.

The following agreements and clarifications pertinent to Section 8.5 of the base standard [MSC] and regarding the semantics of the confirmed CMIS M-GET service (Section 8.3.1 of [CMIS]) are supported by the Phase 1 network management IAs. All CMIS parameters are mandatory, except where noted below.

CMIS M-GET request parameters:

<invokeIdentifier>

<baseManagedObjectClass>

<baseManagedObjectInstance>

<accessControl> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

<synchronization> <bestEffort> is required.

<scope>

<filter>

<attributeIdList> This parameter will specify the Event Forwarding Discriminator attributes to be retrieved. The readable attributes are:
<DiscriminatorId>,
<DiscriminatorConstruct>,
<Management User

Identification>,

<Discriminator State>,
<Begin Time>, <End Time>.

Default gets all attributes.

CMIS M-GET response parameters:

<invokeIdentifier>

<linkedIdentifier>

<managedObjectClass>
<managedObjectInstance> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to these parameters.

<attributeList> This parameter will specify the retrieved Event Forwarding Discriminator attributes.

<currentTime> Refer to Section 18.6 (Management Communications) of this chapter for agreements pertaining to this parameter.

18.5.5.2 Service Access Control Function:

Editor's note: This section is for future study.

18.5.6 Event Logging Control Function:

18.5.6.1 Event Logging Model:

18.5.6.2 Support Managed Object:

18.5.6.2.1 Log Discriminator:

18.5.6.2.2 LOG:

18.5.6.3 Log Control Services:

18.5.6.3.1 Initiate Event Logging Service:

18.5.6.3.2 Terminate Event Logging Service:

18.5.6.3.3 Suspend Event Logging Service:

18.5.6.3.4 Resume Event Logging Service:

18.5.6.3.5 Modify Event Logging Parameters Service:

18.5.6.3.6 Event Log Parameters Retrieval Service:

19. REFERENCES

Editor's Note: In this document, references are maintained in the individual sections as appropriate. Additional references for all of the subject covered in this document may be found in the aligned references section of the Stable Implementation Document.

READER RESPONSE FORM

Please retain my name for the next mailing of the NIST/OSI Implementors Workshop.

NAME: ____

ADDRESS: ____

____

____

PHONE NO.: ____

Mail this page to:

National Institute of Standards and Technology
NIST Workshop for Implementors of OSI
Brenda Gray, Registrar
Building 225, Mail Stop B-217
Gaithersburg, MD 20899

NBS-114A (REV. 2-8C)

U.S. DEPT. OF COMM. **BIBLIOGRAPHIC DATA SHEET** *(See instructions)*	1. PUBLICATION OR REPORT NO. NISTIR-89/4082	2. Performing Organ. Report No.	3. Publication Date May 1989

4. TITLE AND SUBTITLE

WORKING IMPLEMENTATION AGREEMENTS FOR OPEN SYSTEMS INTERCONNECTION PROTOCOLS

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)*

10. SUPPLEMENTARY NOTES

☐ Document describes a computer program; SF-185, FIPS Software Summary, is attached.

11. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here)*

This document records current agreements on implementation details for Open Systems Interconnection Protocols among the organizations participating in the NIST/OSI Workshop Series for Implementors of OSI Protocols. These decisions are documented to facilitate organizations in their understanding of the status of agreements. This is a standing document that is updated after each workshop (about 4 times a year).

12. KEY WORDS *(Six to twelve entries; alphabetical order; capitalize only proper names; and separate key words by semicolons)*

NIST/OSI Workshop; local area networks; network protocols; Open Systems Interconnection; OSINET; testing protocols

13. AVAILABILITY

☒ Unlimited

☐ For Official Distribution. Do Not Release to NTIS

☒ Order From Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402.

☐ Order From National Technical Information Service (NTIS), Springfield, VA. 22161

14. NO. OF PRINTED PAGES

233

15. Price

☆ U.S. GOVERNMENT PRINTING OFFICE 1989— 2 3 5 - 8 0 2   0 0 2 7 4