NISTIR 4382

WORKING IMPLEMENTATION AGREEMENTS FOR OPEN SYSTEMS INTERCONNECTION PROTOCOLS

Based on the proceedings of the
NIST Workshop for Implementors of OSI
Plenary Assembly Held June 22, 1990
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

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary
NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
John W. Lyons, Director

NIST

NISTIR 4382

WORKING IMPLEMENTATION AGREEMENTS FOR OPEN SYSTEMS INTERCONNECTION PROTOCOLS

Based on the proceedings of the
NIST Workshop for Implementors of OSI
Plenary Assembly Held June 22, 1990
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

June 1990

Issued September 1990

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary
NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
John W. Lyons, Director

Table of Contents

Table of Contents

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 Workshop for Implementors of OSI. 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. In the status sections of each chapter as appropriate, specific functionality may be flagged as being "likely" to become stable at the next workshop.

Only non-stable text is included in this document. Errata to Stable material, as well as new stable functionality, is presented as an aligned edition (in replacement page format) issued at the same time as this document.

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 3 as of June 1990" records mature agreements considered advanced enough for use in product development or procurement reference.

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

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, these two documents give the reader a complete picture of current OSI agreements in this forum.

An implementor should look at the aligned section in the Stable Document to get the true current status of stable material. In this Working Document, all references to the Stable Document are to V3 as of June 1990. Where appropriate, statements related to backward compatibility, interworking considerations, or agreement maintenance are given in this document.

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 ANSI X3T5 or ANSI 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.

<u>FTAM SIG</u>

Scope

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

- in particular to maintain the FTAM Phase 2 and Phase 3 specifications with respect to experiences from implementations and from testing. It is a goal that FTAM Phase 3 will remain backward compatible with FTAM Phase 2.

- to act as Registration Authority for OIW FTAM objects.

- to define further FTAM functionality.

- to conduct liaison with standardization bodies such as ISO SC 21 and ANSI X3T5.5.

- to conduct liaison with and contribute to other bodies working on FTAM harmonization such as the Regional Workshops (EWOS, AOW) and the ISO SGFS to define Functional Standards

  and

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

High priority work items:

- Maintain FTAM Phase 2 and Phase 3 Agreements

- Maintain OIW FTAM object register

- Contribute to development of FTAM ISPs

- Specify use of general Character Set Agreements

- Specify requirements of FTAM to a Directory Service

- Specify use of Filestore Management functions

Low priority work items:

- Specify use of Security functions

- Specify use of Overlapped Access

<u>X.400 (MESSAGE HANDLING SYSTEMS) SIG</u>

Scope of Work:

- To develop Stable MHS Agreements among Vendors and Users for the implementation of interoperable products.

- o To conduct Liaison with Standardization Bodies, such as X3V1 as ANSI TAG to ISO/IEC JTC1 SC18, U. S. CCITT Study Group D for input to Study Group VII/Q18, and U. S. CCITT Study Group A for input to Study Group I.

- o To Actively work with other Regional Bodies, primarily (EWOS, AOW) but including others, to define International Standardized Profiles (ISPs) for CCITT X.400 MHS, and ISO/IEC MOTIS.

- o To Review Abstract Tests for X.400 and MOTIS and provide feedback to appropriate bodies.

Current Work Items:

- o MHS use of X.500 Directory

- o Body Parts / Content Types

- o MHS Security Issures

- o Access Units

- o MHS Registration Issues

- o Maintain 1984 MHS Stable Agreements

- o Contribute to development of MHS ISPs

- o MHS routing.

Future Work Items for Next Year:

- o EDI over X.400 and MOTIS

- o Distribution Lists over X.400 and MOTIS.

<u>LOWER LAYER SIG</u>

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 - such study is to include management objects, security, ISDN user-network interfaces for use in conjunction with OSI network services, routing exchange protocols, etc.

Produce and maintain recommendations for implementation of these layers, o Where necessary, provide input to the relevant standards bodies concerning layers 1-4, in the proper manner, and

Review base standard abstract test suites with the goal of identifying the test cases required for the layer 1-4 Implementation Agreements. Develop test cases for Implementation Agreement functionality not present in the base standard (if any).

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.

OBJECTIVES:

o to develop agreements based on IS/DIS

o to develop/draft NWI proposals for submission to national bodies on areas not covered by existing standards work

o to draft contributions on proposed NWIs

o to register security objects

o to provide consultancy to other SIGs

o to act as a well-focused group

- to propagate security information
- to recommend and coordinate activities.

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.

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

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

VIRTUAL TERMINAL SIG

Scope

To develop agreements concerning implementation and testing of Virtual Terminal systems based on ISO 9040/9041 and their addenda. To monitor the X-window system and potentially develop implementors agreements for OSI compatibility.

Objectives

- o Develop VTE-profiles to support diverse interactive applications and environments.
- o Develop Control Objects which may be referenced and used within VTE-profiles.
- o Register and maintain OIW VT objects.
- o Conduct liaison with standards organizations, other regional workshops and vendor/user groups as necessary.
- o Review and, if necessary, generate abstract test cases for VTE-profiles.
- o Harmonize OIW VTE-profiles with those from other regional workshops.
- o Adopt ISP format for OIW VTE-profiles under development.
- o Migrate existing OIW VTE-Profiles to ISP format.
- o Develop X-windows Implementors' Agreement, if necessary.
- o Register and Maintain OIW X-windows Objects.
- o Adopt ISP Format for OIW X-windows Implementors' Agreements, if necessary.
- o Review and, if necessary, generate abstract test cases for X-windows.

High Priority

- o Maintain stabilized OIW VTE-profiles and Control Objects.
- o Develop fully general TELNET profile in ISP format.
- o Develop Scroll Profile in ISP format.

Low Priority

- o Develop abstract test cases.
- o Develop Page profile.
- o Migrate stable profiles to ISP format - Forms, TELNET, X.3, Transparent.

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, RTSE and CCR.
- 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, RTSE, and CCR.
- 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 EWOS, AOW, and ISO,

- o Develop implementor's agreements that promote the efficiency of protocol implementations.
- o Develop implementor's agreements that promote ease in the verification of interoperability,
- o Develop necessary conformance statements.

NETWORK MANAGEMENT SIG

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

The SIG will:

- o Agree upon NIST Implementors OSI systems management reference model
- o Develop product level specifications for implementations, relating to common services/protocols for exchanging management information between OSI nodes
- o 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
- o 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

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

<u>OFFICE DOCUMENT ARCHITECTURE</u>

Scope

To develop agreements concerning implementation and testing of Office Document Architecture (ODA) systems based on ISO 8613, its addenda and related international standards.

Objectives

- o Develop ODA document application profiles to support a diverse set of applications and environments;
- o Register and maintain ODA document application profiles;
- o Conduct liaison with standards organizations, other groups developing ODA document application profiles, vendor/user groups and testing authorities as necessary;
- o Review and, if necessary, generate abstract test cases for ODA document application profiles;
- o Harmonize OIW ODA document application profiles with those from other international groups; and
- o Participate, as necessary, in the ISO ISP processing of FOD-type profiles.

High Priority

- o Develop and maintain OIW ODA document application profiles;
- o Harmonize OIW ODA document application profiles with other international groups; and
- o Assist in the progression of OIW ODA document application profiles through the ISO ISP process.

Low Priority

- o Develop abstract test cases;
- o Integrate addenda and extensions to the base standard into OIW ODA document application profiles; and
- o Develop awareness of ODA in vendor and user groups.

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.

- o 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

- o Produce TR10000-format OSI TP Profile,

- o Describe TP's use of other profile services: ACSE, CCR, Pres., Dir.,
- o Produce CCR profile covering TP requiremnts,
- o Liaise with other internal and external organizations as required,
- o Communicate with EWOS and AOW to reach goal of an aligned profile, and
- o Act as registration authority for OIW TP objects, as necessary.

MANUFACTURING MESSAGE SPECIFICATION (MMS) SIG

Scope

To create an open forum for discussion and agreements pertaining to MMS and issues related to MMS.

Objectives

- o To produce agreements for implementations of MMS (ISO 9506)
- o To produce implementation agreements for IS implementations which enable existing DIS based implementations (such as specified in the MAP 3.0 specification) with minimal modifications to interoperate with IS implementations.
- o To produce implementation agreements on MMS Companion Standards (as recognized by ISO TC184/SC5/WG2) after those have reached ISO DIS or equivalent status.
- o Develop Conformance requirements
- o Develop recommendations on MMS testing

As Necessary

- o Respond to defect reports as accepted
- o Provide feedback on Addendum material
- o To produce implementation agreements on any ISO DIS (or higher level) or equivalent document defining alternate mappings of MMS to an OSI or other international standards based manufacturing communications architecture such as might be progressed from IEC SE 65
- o Provide input on ISP for MMS when the ISO process for it is defined

High Priority Work Items

- o Define a subset of MMS (ISO-9506) suitable for initial implementations

- Produce a set of implementation agreements appropriate to that initial subset of MMS encompassing the objectives

- Study ISO test methodologies and produce recommendations for MMS test implementations. If necessary, provide input on MMS specific requirements for the ISO test methodologies

- Provide input to ISO on Abstract Test Cases to facilitate conformance and interoperability testing on the initial subset

- Provide input to ISO on the elaboration of service procedures for error conditions and on the relation of the use of specific error codes to these error conditions for the initial subset.

Low Priority Work Items

Study and comment on DP level or equivalent documents relating to MMS activities defined in the objectives

Develop subsequent subsets of MMS

- Produce a set of implementors agreements for the subsequent subsets

- Provide input on Test Cases for the subsequent subsets

- Provide input on errors for the subsequent subsets

- Provide input to ISO on MMS ASE specific management entities.

REMOTE DATABASE ACCESS SIG

Scope:

For all RDA Implementations based on ISO 9579:

- Develop Implementors' agreements;

- Provide input to national and international standards organizations on RDA related standards and profiles;

- Coordinate with other organizations on matters relevant to RDA.

Objectives:

- Use ISO 9579 Generic RDA and the ISO SQL Specialization as a basis for Implementors' Agreements on the RDA SQL ASE and its application contexts;

- Provide input to ANSI and ISO on the specification of an RDA ISP.

High Priority Work Items

1. To develop a work plan for RDA Implementors' Agreements with an associated time schedule, using the following tasks as a basis:

    a. review ULA agreements affecting RDA implementations,

    b. specify limits on encodings in RDA pdus,

    c. specify minimum conformance requirements for RDA implementations,

    d. identify and describe recommended practices in the implementation of RDA services and protocols,

    e. identify implementor defined items in ISO 9075 (SQL) affecting interoperability in an OSI environment,

    f. identify implementor defined items in ISO 9579 (RDA) affecting interoperability,

    g. identify RDA implementation requirements for CCR and TP,

    h. harmonize ULA requirements with SQL requirements with respect to handling of variant character sets in RDA.

Low Priority Work Items

1. Future RDA specializations, if any.

1.7 POINTS OF CONTACT

OSI Workshop -Chairman	Tim Boland	NIST	(301) 975-3608
OSI Workshop - Registration	Brenda Gray	NIST	(301) 975-3664
Directory Services - Vice Chair	You-Bong Weon-Yoon	AT&T Bell Labs	(201) 522-5073
FTAM SIG - Acting Chair	Klaus Truoel	GMD/DFN	49-615-1-875-700
Lower Layers SIG	Fred Burg	AT&T	(201) 949-0919
Manufacturing Message Secification (MMS) SIG	Herbert Falk	SISCO	(313) 774-0070
Network Management SIG - Acting Chair	Paul Brusil	Mitre	(617) 271-7632
ODA SIG	Frank Dawson	IBM	(214) 556-5052
Remote Database Access SIG	Rich Gerhardt	GM	(313) 947-0572
Security SIG	James Galvin	Trusted Info. Systems	(301) 854-6889
Technical Liaison Committee	Einar Stefferud	NMA-Northrop	(714) 841-3711
Transaction Processing SIG	Andrew P. Schwartz	IBM Corporation	(415) 855-4766
Upper Layers SIG	Mark Thomas	AT&T Bell Labs	(201) 522-6671
Virtual Terminal SIG	Cyndi Jung	3COM	(415) 940-7664
X.400 SIG	Barbara Nelson	Retix	(213) 399-1611
MAP	Gary Workman	GM	(313) 947-0599
TOP	Laurie Bride	BCS	(206) 763-5719
Government OSI Profile	Jerry Mulvenna	NIST	(301) 975-3631

1.8 PROFILE CONFORMANCE

This section presents general concepts for profile conformance. These concepts shall be observed when writing Implementation Agreements.

1.8.1 General Principle

Conformance to an OSI Profile (Implementation Agreements, Functional Standards) implies conformance to the referenced Base Standards.

Therefore, a Profile shall not specify any requirement that would contradict or cause non-conformance to the Base Standards to which it refers (see TR 10000-1, clauses 6.1, 6.3.1). The conformance requirements defined in ISO/IEC TR 10000-1 fully apply.

1.8.2 Constraints

Base standards usually provide options for PDUs, parameters, encoding choices, value ranges, etc.

A profile may make specific choices of these options and ranges of values. For the promotion of interoperability, pragmatic constraints or minimum requirements may be imposed (e.g., the limitation of Search operations, selection of encoding choices, value ranges, byte ranges for encoding). These minimum requirements or restrictions shall not contradict the conformance requirements of the respective base standards.

1.8.2.1 Sending/Encoding Entity

In order to promote interworking, reasonable restrictions or minimum requirements may be specified in a profile as described above.

1.8.2.2 Receiving/Decoding Entity

Minimum requirements of receiving/decoding capability for alternatives, permissible values, etc. may be specified in a profile. A profile shall not specify the behavior of a receiving/decoding entity when receiving data which is outside the scope of or excluded by the Profile for senders.

A Profile Conformance Test shall be limited by the scope of the profile specification and shall not probe beyond its boundaries. That means, the capability of a receiver/decoder would be tested only in the range of choices or values which are specified for the sending/encoding entity (i.e., for interworking between systems both being conformant to the Profile).

1.8.3 Classification of Conformance

Conformance requirements of a profile shall be related to conformance requirements of a base standard as written in clause 6.5 and annex C of ISO/IEC TR 10000-1. For the conformance classes, the following terminology shall be used, unless otherwise specified by the base standard or equivalent conformance requirements for a profile as required by the ISO/IEC Technical Committee that is responsible for the base standard:

m	mandatory
o	optional
c	conditional
x	excluded
i	out of scope
-	not applicable

Table of Contents

2 SUB NETWORKS

Editor's Note: All references to Stable Agreements in this Section are to Version 3 dated June 1990.

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

This material is current as of June 22, 1990.

Editor's Note: The FDDI material in particular has been identified as a candidate for stability in September 1990.

2.4 ERRATA

Errata are reflected in replacement pages of Version 3, Stable Document, dated June 1990.

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 There shall be some manual or programmatic means of resetting stations and concentrators to the values specified as defaults herein or in X3.139-1987.

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

4 The default value of T_Req shall be equal to either T_MAX or T_Req_Max[1]. whichever is less.

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

P	SD	FC	DA	SA	Info	FCS	ED	FS

Figure 8-. FDDI FRAME FORMAT.

P: Preamble
- 16 Idle Symbols for Transmitting
- >=6 Idle Symbols for Copying
- >=2 Idle Symbols for Repeating

SD: Starting Delimiter (2 Symbols, JK)
FC: Frame Control (2 Symbols)
DA: Destination Address (12 Symbols)

[1] T_Req_Max is defined inthe Ring Management (RMT) section of Station Management. It is used in the resolution of duplicate addres problems which prevent ring initialization. Stations which have detected that they are duplicates during ring initialization take action to make sure that they loose the Claim process to other stations having a T_Req value less the T_Req_Max. T_Req_Max is specified in SMT to have a value $\geq$ 167.8 millesecond.

SA:	Source Address (12 Symbols)
INFO:	Information Field (=<8956 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 active 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 effective delay contribution component which can result is 1.164 microseconds. This number, not one microsecond, should be used per PHY to compute maximum possible network delay.

2.5.5.3 Physical Layer Media Dependent (PMD, X3.166-1989)

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^{-1}0.

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^{-1}2.

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 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).

ANNEX A

(Refer to the Stable Implementation Agreements Document.)

A.1 Data Link Layer, D-Channel

(Refer to the Stable Implementation Agreements Document.)

A.2 Signaling

(Refer to the Stable Implementation Agreements Document.)

Table of Contents

3 NETWORK LAYER

Editor's Note: All references to Stable Agreements in this Section are to Version 3 dated June 1990.

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

This material is current as of June 22, 1990.

Editor's Note: The priority material (Sections 3.5.1 and 3.11) and the addressing material (Section 3.7) should be examined closely for possible stability in September 1990.

3.4 ERRATA

Errata are reflected in pages of Version 3, Stable Document, dated June 1990.

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:

o (Refer to the Stable Implementations Agreements document).

o 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.[1]

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),

- "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)

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

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)

3.6.4 Interworking

(Refer to the Stable Implementation Agreements Document.)

3.7 ADDRESSING

- Refer to the Stable Implementations Agreements Document

o Within routing domains intending to operate using the IS -IS Intradomain Routing Protocol defined in ISO/IEC JTC 1/SC 6 N4945, it is recommended that the DSP have a binary abstract syntax and that the last nine octets are structured as follows:

2 octets	6 octets	1 octet
AREA	ID	N-Selector

where the AREA field identifies a unique subdomain of the routing domain, the ID field identifies a unique system within an area, and an N-SELECTOR identifies a user of the Network Layer Service.

See the OSI Routing Framework document (ISO/TR 9575) for definitions of the above terms and concepts.

The above recommendation may be applicable in other routing environments.

3.8 ROUTING

3.8.1 ISO 9542 End System to Intermediate System Routing

(Refer to the Stable Implementation Agreements Document.)

10. ISO 8473 PDUs multicast as a result of the Query Configuration function shall use the Network Layer Protocol ID (NLPID) assigned to ISO-8473.

11. An ISO 8473 PDU received as a result of another ES having performed the Query Configuration function shall be processed as follows:

    - If the ISO 8473 PDU is addressed to one of the NSAPs present in the ES, the End System shall process the PDU according to the applicable clauses of ISO 8473 and invoke the Configuration Response Function (clauses 6.6) of ISO 9542

    - If the ISO 8473 PDU is not addressed to one of the NSAPs present in the ES, the End System shall discard the PDU without generating as ISO 8473 Error Report

12. For purposes of address matching and SNPA extraction, the first octet of the option parameter value of an address (clause 7.4.5) or SNPA Mask (clause 7.4.6) shall be aligned with the first octet (AFI) of the encoded trial NSAP Address.

The following items represent proposed solutions to defects in ISO 9542. These solutions are being progressed as defect reports to ISO 9542. These items will be deleted when the corresponding defect report is approved.

> An End System may choose to ignore an RD PDU received for a destination to which the ES has not sent traffic for some period of time. An ES must record redirection information only for those other systems with which it is in active communication.
>
> A holding time value of zero is permitted. When configuration and/or redirection information with a zero holding time is received, prior information shall be replaced, thus causing the system to set its holding timer to zero and discard the corresponding information.
>
> If one or more ISs suggested an ESCT, the minimum of the non-zero suggested values replaces the current value of the ES's CT.

3.8.2 DIS 10030 End System to Intermediate System Routing

The protocol used to provide End System to Intermediate System routing in support of the CONS (refer to section 3.6) shall be DIS 10030.

The following agreements apply to the use of DIS 10030:

1. A management mechanism capable of adding and deleting entries in the Routing Information Base (RIB) of both SNAREs and End Systems is recommended. When using the management mechanism to add as entry it should not be timed out, and the entry should be write protected from alteration by the DIS 10030 protocol.

3.8.3 Intra-Domain Intermediate to Intermediate Systems Routing

The protocol used to provide Intermediate System to Intermediate System routing in support of the CLNS (refer to section 3.5) among systems in a single routing domain shall be DP 10589

The following agreements apply to the use of DP 10589:

1. A management mechanism capable of configuring the Identifier, Characteristic, and Status attributes of the managed objects of clause 11 shall be provided.

3.8.4 Inter-Domain Intermediate Systems to Intermediate Systems Routing

An Administrative Authority shall determine the procedures and policies that govern the exchange of routing information with other routing domains.

Intermediate systems shall provide management mechanisms to configure the required inter-domain routing information.

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 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[2]

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 layer protocols is limited to the conveyance of priority information in both directions between an application entity and the service boundary for the Transport Layer.

[2] 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

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 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)

3.13 ANNEX A

Editor's Note: This material was moved to the Stable Document in June 1990. It was not considered an implementor agreement prior to June 1990.

Table of Contents

4 TRANSPORT LAYER

Editor's Note: All references to Stable Agreements in this Section are to Version 3 dated June 1990.

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

This material is current as of June 22, 1990.

The priority material (clause 4.5.1.2.6) in particular has been identified as a candidate for stability in September 1990.

4.4 ERRATA

Errata are reflected in pages of Version 3, Stable Document, dated June 1990.

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 General Rules

(Refer to the Stable Implementation Agreements Document.)

It is recommended that the capability of request acknowledgements be supported and proposed in CR TPDUs.

If request acknowledgements are supported, then if the implementation delays acknowledgements it shall:

a) request use of request acknowledgements in the CR TPDU

b) accept the use of request acknowledgements in the CC TPDU if it was proposed in the CR TPDU.

It is recommended that implementations send both the preferred and existing TPDU size parameters in the CR TPDU. If a CR TPDU is received with the "preferred" parameter and the preferred maximum TPDU size parameter is supported, the preferred parameter shall be returned in the CR TPDU and the existing TPDU size parameter in the CR TPDU shall be ignored.

It is recommended that inactivity timer values be exchanged during connection establishment. This may be mandatory in the future.

If the "exchange of inacitivity timers" capability is supported, the implementation shall send its minumum inactivity timer in the CR TPDU. If a CR TPDU is received with this timer value and the capability is supported, the responding CC TPDU shall contain the inactivity time.

If the Inactivity time is received and the capability is supported, the following shall be used as an upper bound for w:

$(IR-E_{LR})/N \geq W \qquad N \geq 2$

4.5.1.2.2 Transport Class 4 Service Access Points or Selectors

(Refer to the Stable Implementation Agreements Document.)

4.5.1.2.3 Retransmission Timer

As network load increases, the variability of round-trip delay also increases. In environments where load fluctuates widely, it is therefore useful to estimate the variability of round-trip delay measurements and use this estimate in the calculation of retransmission timer values. An estimate of the variability of round-trip delay measurements can be efficiently calculated as an exponentially weighted average of the differences between round-trip delay measurements and the average round-trip delay. This represents the mean deviation of the round-trip delays, which is a useful approximation of the standard deviation and can be much more efficiently computed. The formula is

$$D < -D + (1 - a)(|S - E| - D)$$

where *D* is the estimate of variability in round-trip delays. *S, E,* and *a* are as defined for the preceding formula. As before the value of *a* must be between 0 and 1 and the choice of a value of $1 - 2^{-N}$ allows for efficient update of the average. The value of *a* for the variability estimation, though, does not need to be the same as that used for the round-trip delay estimate. A smaller value for *a* is useful in the variability estimation to cause a more rapid response to changes in round-trip delays. *D* can then be used to calculating retransmission timer values according to the formula:

$$T1 < - E + AR + kD$$

where *T1* is the retransmission timer value, *E* is the estimated average round-trip delay, *AR* is the value of the acknowledgement timer parameter received from the remote transport service provider during connection establishment, and *k* is a locally administered factor. Since *D* approximates the standard deviation of the round-trip delays, but is greater than or equal to the standard deviation, round-trip delays within *k* standard deviations of the mean would be accounted for by the retransmission timer value (eg. $k = 2$, if round-trip delays were normally distributed, would account for 95% of the variability).

Round-trip time measurements based on acknowledgement of any retransmitted data should not be used to update the round-trip delay estimate or the estimate of variability. Such measurements are not reliable since it is ambiguous which transmission of the data is being acknowledged. One strategy for handling a retransmission timeout is to retransmit the PDU and reset the timer with a value that is twice the previous value. In this case, a new roundtrip delay estimate and estimate of variability should be calculated only when an acknowledgement of data is received where none of the acknowledged data has been retransmitted.

4.5.1.2.4 Keep-Alive Function

(Refer to the Stable Implementation Agreements Document.)

4.5.1.2.5 Congestion Avoidance Policies

(Refer to the Stable Implementation Agreements Document).

4.5.1.2.6 Use of Priority[1]

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.

[1] Refer to clause 3.11 for an overview on the use of priority.

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 connection priority level, except as excluded in item 5b and 5d below[2].

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:[3]

- o "high reserved" priority will be encoded with value "zero" (0000 0000 0000 0000), and
- o "high" priority will be encoded with value 5 (0000 0000 0000 0101),
- o "normal" priority will be encoded with value 10 (0000 0000 0000 1010),
- o "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[4].

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. 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.

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

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

[4] 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.

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 General Rules

It is recommended that implementations send both the preferred and existing TPDU size parameters in the CR TPDU. If a CR TPDU is received with the "preferred" parameter and the preferred maximum TPDU size parameter is supported, the preferred parameter shall be returned in the CR TPDU and the existing TPDU size parameter in the CR TPDU shall be ignored.

4.5.2.2.2 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

(Refer to Stable Implementation Agreements Document.)

4.5.3.1 Transport Class 2 Overview

(Refer to Stable Implementation Agreements Document.)

4.5.3.2 Protocol Agreements

It is recommended that implementations send both the preferred and existing TPDU size parameters in the CR TPDU. If a CR TPDU is received with the "preferred" parameter and the preferred maximum TPDU size parameter is supported, the preferred parameter shall be returned in the CR TPDU and the existing TPDU size parameter in the CR TPDU shall be ignored.

4.6 PROVISION OF CONNECTIONLESS TRANSPORT SERVICE

(Refer to Stable Implementation Agreements Document.)

4.7 TRANSPORT PROTOCOL IDENTIFICATION

(Refer to the Stable Implementation Agreements document.)

Table of Contents

5 UPPER LAYERS

Editor's Note: All references to Stable Agreements in this section are to Version 3 dated June 1990.

5.1 INTRODUCTION

(Refer to Stable Agreements Document)

5.1.1 References

(Refer to Stable Agreements Document)

5.2 SCOPE AND FIELD OF APPLICATION

(Refer to Stable Agreements Document)

5.3 STATUS

This version of the upper layer agreements is under development.

5.4 ERRATA

5.4.1 ISO Defect Solutions

5.4.2 Session Defect Solutions Correcting CCITT X.215 and X.225

(Refer to 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

5.5.3.1 Application Context

(Refer to Stable Agreements Document)

5.5.3.2 AE Title

(Refer to Stable Agreements Document)

5.5.3.3 Result Parameter

If the result parameter of the AARE PDU contains the value accepted, then the result-source-diagnostic parameter shall contain the value null.

5.5.4 ASN.1 Encoding Rules

(Refer to Stable Agreements Document)

5.5.5 Connectionless

(Refer to Stable Agreements Document)

5.6 ROSE

(Refer to Stable Agreements Document)

5.7 RTSE

(Refer to Stable Agreements Document)

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

5.8.3.1 Transfer Syntaxes

(Refer to Stable Agreements Document)

5.8.3.2 Presentation Context Identifier

(Refer to Stable Agreements Document)

5.8.3.3 Default Context

(Refer to Stable Agreements Document)

5.8.3.4 P-Selectors

(Refer to Stable Agreements Document)

5.8.3.5 Provider Abort Parameters

(Refer to Stable Agreements Document)

5.8.3.6 Provider Aborts and Session Version

(Refer to Stable Agreements Document)

5.8.3.7 CPC-Type

(Refer to Stable Agreements Document)

5.8.3.8 Presentation-context-definition-result-list

(Refer to Stable Agreements Document)

5.8.3.9 RS-PPDU

(Refer to Stable Agreements Document)

5.8.4 Presentation ASN.1 Encoding Rules

5.8.4.1 Invalid Encoding

(Refer to Stable Agreements Document)

5.8.5 General

5.8.5.1 Presentation Data Value (PDV)

(Refer to Stable Agreements Document)

5.8.6 Connection Oriented

(Refer to Stable Agreements Document)

5.8.7 Connectionless

(Refer to Stable Agreements Document)

5.9 SESSION

5.9.1 Introduction

(Refer to Stable Agreements Document)

5.9.2 Services

(Refer to Stable Agreements Document)

5.9.3 Protocol Agreements

5.9.3.1 Concatenation

(Refer to Stable Agreements Document)

5.9.3.2 Segmenting

(Refer to Stable Agreements Document)

5.9.3.3 Reuse of Transport Connection

(Refer to Stable Agreements Document)

5.9.3.4 Use of Transport Expedited Data

(Refer to Stable Agreements Document)

5.9.3.5 Use of Session Version Number

(Refer to Stable Agreements Document)

5.9.3.6 Receipt of Invalid SPDUs

(Refer to Stable Agreements Document)

5.9.3.7 Invalid SPM Intersections

(Refer to Stable Agreements Document)

5.9.3.8 S-Selectors

(Refer to Stable Agreements Document)

5.9.4 Connectionless

(Refer to Stable Agreements Document)

5.10 UNIVERSAL ASN.1 ENCODING RULES

5.10.1 TAGS

(Refer to Stable Agreements Document)

5.10.2 Definite Length

(Refer to Stable Agreements Document)

5.10.3 EXTERNAL

(Refer to Stable Agreements Document)

5.10.4 Integer

(Refer to Stable Agreements Document)

5.10.5 String Types

(Refer to Stable Agreements Document)

5.10.6 Bit String

(Refer to Stable Agreements Document)

5.11 CHARACTER SETS

(Refer to part 21 -- a new chapter expressly for character sets.)

5.12 CONFORMANCE

(Refer to Stable Agreements Document)

5.12.1 Specific ASE Requirements

(Refer to Stable Agreements Document)

5.12.1.1 FTAM

5.12.1.1.1 Phase 2

(Refer to Stable Agreements Document)

5.12.1.2 MHS

5.12.1.2.1 Phase 1 (1984 X.400)

(Refer to Stable Agreements Document)

5.12.1.2.2 Phase 2, Protocol P1 (1988 X.400)

(Refer to Stable Agreements Document)

5.12.1.2.3 Phase 2, Protocol P7 (1988 X.400)

(Refer to Stable Agreements Document)

5.12.1.2.4 Phase 2, Protocol P3 (1988 X.400)

(Refer to Stable Agreements Document)

5.12.1.3 DS

5.12.1.3.1 Phase 1

(Refer to Stable Agreements Document)

5.12.1.4 Virtual Terminal

5.12.1.4.1 Phase 1a

(Refer to Stable Agreements Document)

5.12.1.4.2 Phase 1b

(Refer to Stable Agreements Document)

5.12.1.5 MMS

For further study.

5.12.1.6 Transaction Processing

ACSE Requirements:

all

Application Context:

The application context is user-defined.

Presentation Requirements:

Presentation Functional Units:

o kernel

Presentation Contexts:

o At least 3 must be supported if the commit functional unit of TP is not supported.

o At least 4 must be supported if the commit functional unit of TP is supported.

Abstract Syntaxes:

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

Associated Transfer Syntax:

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

o "ISO 10026-TP"

{ joint-iso-ccitt(2) transaction-processing(?) abstract-syntax(2) tp-apdus(1) }

Associated Transfer Syntax:

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

- o If required, "ISO 9804-CCR"
  (TBD)

- o At least one user-defined abstract syntax.

Session Requirements:

Session Functional Units:

- o kernel
- o duplex
- o Others as required by CCR (TBD) if the commit functional unit of TP is supported.

Version Number: 2

Maximum size of User Data parameter field: 10,240

Annex A: Recommended Practices

(Refer to Stable Agreements Document)

Annex B: Object Identifier Register

B.1 Register Index

(Refer to Stable Agreements Document)

B. 2 Object Identifier Descriptions

(Refer to Stable Agreements Document)

Table of Contents

6 Registration Authority Procedures for the OSI Implementors Workshop (OIW)

For current Registration Authority information for Workshop--Defined Objects, consult the aligned chapter of Version 3, Stable Implementation Agreements dated June 1990.

Table of Contents

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 contained in Version 3 dated June 1990.

Table of Contents

List of Figures

List of Tables

8 Message Handling Systems

Editor's Note: A vote was passed in June 1990 to add text ensuring the uniqueness of management domain names in the United States.

8.1 Introduction

See Stable Implementation Agreements, Version 3 dated June 1990.

8.2 Scope

See Stable Implementation Agreements, Version 3 dated June 1990.

8.3 Status

See Stable Implementation Agreements, Version 3 dated June 1990.

8.4 Errata

See Stable Implementation Agreements, Version 3 dated June 1990.

8.5 MT Kernel

See Stable Implementation Agreements, Version 3 dated June 1990.

8.5.1 Introduction

See Stable Implementation Agreements, Version 3 dated June 1990.

8.5.2 Elements of Service

See Stable Implementation Agreements, Version 3 dated June 1990.

8.5.3 MTS Transfer Protocol (P1)

See Stable Implementation Agreements, Version 3 dated June 1990.

8.5.4 MTS - APDU Size

This section is for further study by the NIST X.400 SIG. The following support requirement may be increased in the future.

The following agreements govern the size of MPDUs:

- o All MTAEs must support at least one MPDU of at least two megabyte.
- o The size of the largest MPDU supported by a UAE is a local matter.

8.5.5 1988/84 Interworking Considerations

Editor's Note: References to Section 7 are to the Stable Document.

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:

- o Supplementary Information - will need to be truncated if it exceeds the pragmatic constraint identified in Version 2 of these Agreements (64 octets as opposed to 256 octets in the 1988 MHS standards), and
- o Internal Trace Information - If the 1984-based MTA does not support Internal Trace Information per Section 7.7.3.2, the following description is not applicable. When a 1988-based MTA supports interworking with a 1984-based MTA that generates Internal Trace Information as per Section 7.7.3.3, the 1988-based MTA must support reception of the Internal Trace Information by converting the Internal Trace Information from the form in Section 7.7.3.2 to the form specified in 1988 X.411, as per the following description. When the 1988-based MTA sends to a 1984 MTA, the 1988-based MTA must apply the conversion to 1984, as described below. The Stable NBS Implementors Agreements X.400 (1984) implementors' agreements definition for MTA's Internal Trace Information is different from the X.400 (1988) MTA definition. Consequently, a X.400 (1988) MTA operating in an MD with other MTAs of 1984 vintage, must map the Internal Trace Information to and/or from the 1984 format.

What follows are algorithms for mapping between X.400 (1988) Internal Trace element formats and the NIST IA X.400 (1984) Internal Trace element format.

To avoid potential looping within a MD composed of 1984 and 1988 vintage MTAs, MD administrators are strongly advised to name all MTAs (1984 and 1988 vintages) using only the Printable String characters. In X.400 (1988) the MTA-Name is defined to be named using IA5 String characters where in the IAs for X.400 (1984) MTAs, NBS restricted the MTA-Name to be formed using the Printable String character subset of IA5. If the 1988-based MTA Name uses IA5 characters not in the Printable String subset, that Internal Trace Element should be omitted when converting from 1988 to 1984.

```
For each Internal Trace element in the sequence:
DO
  IF MTA-Name is made up of non-Printable String characters:
    Discard this Internal Trace element;
  ELSE
    {  Discard the GlobalDomainIdentifier;
       Copy the MTAname over;
       Within the MTASuppliedInformation:
         Copy the arrival time over;
         Copy the routing action over;
         IF attempted is present
           {  IF it is a domain:
                Discard it;
              IF it is an MTA:
                Copy it to Previous MTAName;
           }
         IF the additional actions are present:
           {  IF the deferred time is present:
                Copy it over;
              IF the other-actions is present:
                Discard it;
           }
    }
END-DO
```

Figure 8.3. 1988 to 1984 Mapping.

```
Find the [APPLICATION 30] entry in the P1 envelope;
FOR each Internal Trace element:
  DO
    Insert the GlobalDomainIdentifier of this MTA;
    Copy the MTAName over;
    Within the MTASuppliedInfo:
      Copy the arrival time;
      IF the deferred time is present:
        copy it to the additional actions field within the
          1988 Internal Trace information;
      IF the routing action is Relayed or Rerouted:
        copy it over;
      IF the routing action is Recipient-reassigned:
        map to Relayed;
      IF the previous MTAName is present:
        copy it to the MTAName in the attempted field;

  END-DO
```

Figure 8.4. 1984 to 1988 Mapping.

8.6 IPM Kernel

8.6.1 Introduction

See Stable Implementation Agreements Version 3 dated June 1990.

8.7 Message Store

8.7.1 Introduction

See Stable Implementation Agreements, Version 3 dated June 1990.

8.7.2 Scope

See Stable Implementation Agreements, Version 3 dated June 1990.

8.7.3 Elements of Service

See Stable Implementation Agreements, Version 3 dated June 1990.

8.7.4 Attribute Types

See Stable Implementation Agreements, Version 3 dated June 1990.

8.7.5 Pragmatic Constraints for Attribute Types

See Stable Implementation Agreements, Version 3 dated June 1990.

8.7.6 Implementation of the MS with 1984 Systems

See Stable Implementation Agreements, Version 3 dated June 1990.

8.7.7 MS Access Protocol (P7)

See Stable Implementation Agreements, Version 3 dated June 1990.

8.7.8 MTS Access Protocol (P3)

See Stable Implementation Agreements, Version 3 dated June 1990.

8.8 Remote User Agent Support

8.8.1 Introduction

See Stable Implementation Agreements, Version 3 dated June 1990.

8.8.2 Scope

See Stable Implementation Agreements, Version 3 dated June 1990.

8.8.3 Elements of Service

See Stable Implementation Agreements, Version 3 dated June 1990.

8.8.4 MTS Access Protocol (P3)

See Stable Implementation Agreements, Version 3 dated June 1990.

8.9 Naming, Addressing & Routing

8.9.1 Use of O/R Addresses for Routing

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

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

Addressing may be secured according to practices outside the scope of these agreements, such as:

- o manual directories
- o on-line directories, such as X.500
- o ORName address translation algorithms.

8.9.2 Distribution Lists

8.9.2.1 Introduction

This section identifies and specifies the Distribution Lists Functional Group, which covers all issues relating to the performance of distribution list (DL) expansion by an MTA. Other aspects concerned with the use of distribution lists are covered in the MT Kernel and IPM Kernel Functional Groups.

8.9.2.2 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 for the MT Service only, and is only concerned with the performance of DL expansion by an MTA. Such support 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 use of distribution lists is as specified in Section 8.6.

Table 8.13. Distribution Lists: MT Elements of Service

Element of Service	Support
DL Expansion History Indication	M
DL Expansion Prohibited	M
Use of Distribution List	M

8.9.3 MHS Use of Directory

8.9.3.1 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 for use in submitting messages for delivery by the MTS.

The MTS may also use the directory service elements to obtain information, for example, to be used in the routing of messages. This application of the directory service is not defined by the base standards and is therefore not addressed by this Agreement.

8.9.3.2 Functional Configuration

Two MHS functional entities, the IPM UA and MTA, may access the Directory service using the Directory User Agent (DUA). The interface between the UA and DUA, or MTA and DUA is local and not defined. The interaction between the DUA and Directory System Agent (DSA) is specified in Chapter 11. A collocated DUA and DSA is also permitted.

8.9.3.3 Functionality

Some functional usages of directories have been identified for UAs and the MTAs. These are:

UA Specific Functionality:

- o Verify the existence of a Directory Name.
- o Given a partial name, return a list of possibilities.
- o Ability to scan directory entries.
- o Return the O/R Address(es) that correspond to a Directory Name.
- o Determine whether a Directory Name presented denotes a user or a Distribution List.
- o Return the members of a Distribution List.

o Return the capabilities of the entity referred to by a Directory Name.

o Maintenance functions to keep the directory up-to-date, e.g. register and change credentials.

MTA Specific Functionality:

o Authentication.

o Return the O/R Address(es) that correspond to a Directory Name.

o Determine whether a Directory Name presented denotes a user or a Distribution List.

o Return the members of a Distribution List.

o Return the capabilities of the entity referred to by a Directory Name.

o 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, expendability and reliability.

8.9.3.4 Naming and Attributes

Since user-friendliness is of primary importance in a messaging system, the naming conventions used in building the Directory Information Tree (DIT) will impact the ability of a user to make intelligent guesses for Directory Names.

It is recommended that the naming guidelines and DIT structures defined in Annex B of Recommendation X.521/ISO 9594-7 be used as the basis for MHS Directory Names. Annex C of Recommendation X.402/ISO 10021-2 specifies further the MHS specific object classes. The naming for MHS specific object classes are recommended as follows:

(i) the naming for mhs-message-store, mhs-message-transfer-agent, and mhs-user-agent is that of Application Entity in the DIT.

(ii) the naming attribute for mhs-distribution-list is commonName. The organization, organizationalUnit, organizationalRole, organizationalPerson, Locality, or groupOfNames can be immediate superior to entries of object class mhs-distribution-list.

(iii) the naming for mhs-user is that of organizationalPerson, ResidentialPerson, organizationalRole, organizationalUnit, organization, or Locality.

Note: The mhs-user object class is a generic object class which may be used in conjunction with another standard object class for the purpose of adding MHS information attributes, such as ORAddresses, to a Directory entry. The means to associate attributes of a generic object class to an entry (or to different entries) named by a standard object class(es) is by defining a new (un-)registered object class, whose superclass(es) is that of the naming object class(es), and of the generic object class. E.g., to associate mhs-user attributes in the organizationalPerson entry, the new unregistered object class can be defined as shown in Figure 8.9.

```
real-user-entry  ::=  OBJECT CLASS
                      SUBCLASS OF organizationalPerson,
                                  mhs-user
```

Figure 8.9. Example of Unregistered Object Class Definition.

The MHS object classes, attributes, and attribute syntaxes that need to be supported by the Directory are as specified in Annex C of Recommendation X.402/ISO 10021-2.

In addition, the object classes organization, organizationalUnit, organizationalRole, organizationalPerson, locality, groupOfNames, residentialPerson, and country and their attributes and associated syntaxes as defined in X.520 (ISO 9594, Part 6) and X.521 (ISO 9594, Part 7) are required to support the MHS.

8.9.3.5 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 8.14. Use of Directory: MT Elements of Service

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

Table 8.15. Use of Directory: IPM Elements of Service

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

8.10 MHS Management

For further study.

8.11 MHS Security

8.11.1 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.

8.11.2 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 8.16. 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 8.17. 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	*	*

8.12 Specialized Access

8.12.1 Physical Delivery

8.12.1.1 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.

8.12.1.2 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 8.18. Physical Delivery: MT Elements of Service

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

Table 8.19. Physical Delivery: IPM Elements of Service

Element of Service	Origination (IPM UA)	Reception (PDAU)
Additional Physical Rendition	O	O
Basic Physical Rendition	O[1]	M
Counter Collection	M	M
Counter Collection with Advice	O	O
Delivery via Bureaufax Service	O	O
EMS (Express Mail Service)	M	M[2]
Ordinary Mail	O[1]	M
Physical Delivery Notification by MHS	O	O
Physical Delivery Notification by PDS	O	M
Physical Forwarding Allowed	O[1]	M
Physical Forwarding Prohibited	M	M
Registered Mail	O	O
Registered Mail to Addressee in Person	O	O
Request for Forwarding Address	O	O
Special Delivery	M	M[2]
Undeliverable Mail with Return of Physical Message	O[1]	M

Notes:

1) Provided by default (when using a physical delivery address).
2) Must support EMS and/or Special Delivery.

Table 8.20. Physical Delivery O/R Address Attributes

O/R Address Attribute Type	UA Orig	PDAU Recep
administration-domain-name	M	M
country-name	M	M
private-domain-name	M	M
physical-delivery-service-name	O	M
physical-delivery-country-name	M	M
postal-code	M	M
extension-postal-O/R-address-components	O	M
extension-physical-delivery-address-components	O	M
local-postal-attributes	O	M
physical-delivery-office-name	O	M
physical-delivery-office-number	O	M
physical-delivery-organization-name	O	M
physical-delivery-personal-name	O	M
post-office-box-address	O	M
poste-restante-address	O	M
street-address	O	M
unformatted-postal-address	M	M
unique-postal-name	O	M

The handling of Printable Strings and Teletex Strings in O/R address components is for further study.

Table 8.21. Character String Support

Character String	Origination (IPM UA)	Reception (PDAU)
Printable	*	M
Teletex	*	M

8.12.2 Other Access Units

8.12.2.1 Facsimile Access Units

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

8.12.2.2 Telex Access Units

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

8.12.2.3 Teletex Access Units

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

8.13 Conversion

8.13.1 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 MTA.

8.13.2 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 only concerned with the performance of conversion by an MTA. Such support 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 <u>access</u> to conversion facilities is as specified in Section 8.6.

Table 8.22. Conversion: MT Elements of Service

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

8.14 Use of Underlying Layers

8.14.1 MTS Transfer Protocol (P1)

See Stable Implementation Agreements, Version 3 dated June 1990.

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

See Stable Implementation Agreements, Version 3 dated June 1990.

8.15 Error Handling

See Stable Implementation Agreements, Version 3 dated June 1990.

8.15.1 PDU Encoding

8.15.2 Contents

8.15.3 Envelope

8.15.4 Reports

8.15.5 Pragmatic Constraints

If an implementation detects a pragmatic constraint violation, then it may generate an appropriate error indication but is not required to do so.

8.16 Conformance

See Stable Implementation Agreements, Version 3 dated June 1990.

8.17 Annex A: MHS Protocol Specifications

See Stable Implementation Agreements, Version 3 dated June 1990.

8.17.1 MTS Transfer Protocol (P1)

See Stable Implementation Agreements, Version 3 dated June 1990.

8.17.2 Interpersonal Messaging Protocol (P2)

See Stable Implementation Agreements, Version 3 dated June 1990.

8.17.3 MTS Access Protocol (P3)

See Stable Implementation Agreements, Version 3 dated June 1990.

8.17.4 MS Access Protocol (P7)

See Stable Implementation Agreements, Version 3 dated June 1990.

8.17.5 Message Store General Attribute Support

See Stable Implementation Agreements, Version 3 dated June 1990.

8.17.6 Message Store IPM Attribute Support

See Stable Implementation Agreements, Version 3 dated June 1990.

8.18 Annex B: Interpretation of Elements of Service

The objective of this section is to provide clarification, where required, on the functionality of Elements of Service where the MHS standards are unclear or ambiguous. It is not the intent of this section to define how information should be made available or presented to an MHS user, nor is it intended to define how individual vendors should design their products.

The following MHS Elements of Service require further text to be added to their definitions to represent the proposed implementation of these Elements of Service for conformance to this Agreement. Elements of Service which are not referenced in this section are as defined in the MHS base standards.

Reply Request Indication

The reply-recipients and the reply-time may be specified without any explicit reply being requested. This may be interpreted by the recipient as an implicit reply request. Note that for an auto-forwarded message an explicit or implicit reply request may not be meaningful.

Forwarded IP-message Indication

The following use of the original encoded information type in the context of forwarded messages is clarified:

- o The encoded information types of the message being forwarded should be reflected in the new original encoded information types being generated.

- o If forwarding a private message body part, the originator of the forwarded message shall set the original encoded information types in the P1 envelope to Undefined for that body part.

8.19 Annex C: Recommended Practices

See Stable Implementation Agreements, Version 3 dated June 1990.

8.19.1 Printable String

See Stable Implementation Agreements, Version 3 dated June 1990.

8.19.2 Rendition of IA5Text

See Stable Implementation Agreements, Version 3 dated June 1990.

8.19.3 EDI Use of MHS

8.19.3.1 Introduction and Scope

See Stable Implementation Agreements, Version 3 dated June 1990.

8.19.3.2 Model

See Stable Implementation Agreements, Version 3 dated June 1990.

8.19.3.3 Protocol Elements Supported for EDI

See Stable Implementation Agreements, Version 3 dated June 1990.

8.19.3.4 Addressing and Routing

See Stable Implementation Agreements, Version 3 dated June 1990.

8.19.4 ODA Transfer

To ease interworking with 1984 implementations when transferring Office Document Architecture (ODA) documents, the following are recommended for 1988 implementations:

1. Origination UA implementing 1988 profile. The 1988 will generate the ODA according to CCITT Recommendation T.411 Annex E for the destination UA(s) implementing 1988 profile. If the destination UA supports 1984 profile, the approach as described in section 7.12.8 is recommended.

2. Recipient UA implementing 1988 profile. The recipient system will be able to handle the ODA bodypart in P2 (1984) as defined in section 7.12.8 for interworking with 1984 implementation, and will also be able to handle the ODA bodypart as defined in the appropriate base standards.

3. MTA downgrading rules. When transferring an P22 with ODA body part in P22 as described in T.411 to an 1984 MTA, the EITs identified by ODA Object Identifiers are mapped to bits 0 and 10 of the built-in EITs.

If the UA does not register to support P22 or ADA bodypart, a Non-Delivery-Report will be generated as required.

8.20 Annex D: List of ASN.1 Object Identifiers

8.20.1 Content Types

8.20.2 Body Part Types

Table of Contents

9 STABLE FTAM PHASE 2

Editor's Note: For Stable FTAM Phase 2 Agreements, consult the aligned section in the Stable Implementation Agreements Document. This section serves as a reference or pointer to Stable Agreements contained in Version 3 dated June 1990.

Table of Contents

10 ISO FILE TRANSFER, ACCESS AND MANAGEMENT PHASE 3

Editor's Note: For current Stable FTAM Phase 3 Agreements, consult the aligned section in the Stable Implementation Agreements, Version 3 dated June 1990.

Table of Contents

11 DIRECTORY SERVICES PROTOCOLS

11.1 INTRODUCTION

Refer to section 11.1 of Stable Agreements Version 3 as of June 22, 1990.

11.2 SCOPE AND FIELD OF APPLICATION

Refer to section 11.2 of Stable Agreements Version 3 as of June 22, 1990.

11.3 STATUS

Refer to section 11.3 of Stable Agreements Version 3 as of June 22, 1990.

11.4 USE OF THE DIRECTORY

This section will contain introductory text.

11.4.1 MHS

(TBD)

11.4.2 FTAM

(TBD)

11.5 DIRECTORY ASEs AND APPLICATION CONTEXTS

Refer to section 11.5 of Stable Agreements Version 3 as of June 22, 1990.

11.6 SCHEMA

Refer to section 11.6 of Stable Agreements Version 3 as of June 22, 1990.

11.6.1 Support of Structures and Naming Rules

Refer to section 11.6.1 of Stable Agreements Version 3 as of June 22, 1990.

11.6.2 Support of Object Classes and Subclasses

Refer to section 11.6.2 of Stable Agreements Version 3 as of June 22, 1990.

11.6.3 Support of Attribute Types

Refer to section 11.6.3 of Stable Agreements Version 3 as of June 22, 1990.

11.6.4 Support of Attribute Syntaxes

Refer to section 11.6.4 of Stable Agreements Version 3 as of June 22, 1990.

11.6.5 Naming Contexts

Refer to section 11.6.5 of Stable Agreements Version 3 as of June 22, 1990.

11.6.6 Common Profiles

Refer to section 11.6.6 of Stable Agreements Version 3 as of June 22, 1990.

11.6.6.1 OIW Directory Common Application Directory Profile

Refer to section 11.6.6.1 of Stable Agreements Version 3 as of June 22, 1990.

11.6.6.1.1 Standard Application Specific Attributes and Attribute Sets

Refer to section 11.6.6.1.1 of Stable Agreements Version 3 as of June 22, 1990.

11.6.6.1.2 Standard Application Specific Object Classes

Refer to section 11.6.6.1.2 of Stable Agreements Version 3 as of June 22, 1990.

11.6.6.2 OIW Directory Strong Authentication Directory Profile

Refer to section 11.6.6.2 of Stable Agreements Version 3 as of June 22, 1990.

11.6.6.2.1 Other Profiles Supported

Refer to section 11.6.6.2.1 of Stable Agreements Version 3 as of June 22, 1990.

11.6.6.2.2 Standard Application Specific Object Classes

Refer to section 11.6.6.2.2 of Stable Agreements Version 3 as of June 22, 1990.

11.6.7 Restrictions on Object Class Definitions

Refer to section 11.6.7 of Stable Agreements Version 3 as of June 22, 1990.

11.7 PRAGMATIC CONSTRAINTS

Refer to section 11.7 of Stable Agreements Version 3 as of June 22, 1990.

11.7.1 General Constraints

Refer to section 11.7.1 of Stable Agreements Version 3 as of June 22, 1990.

11.7.1.1 Character Sets

Refer to section 11.7.1.1 of Stable Agreements Version 3 as of June 22, 1990.

11.7.1.2 APDU Size Considerations

Refer to section 11.7.1.2 of Stable Agreements Version 3 as of June 22, 1990.

11.7.1.3 Service Control (SC) Considerations

Refer to section 11.7.1.3 of Stable Agreements Version 3 as of June 22, 1990.

11.7.1.4 Priority Service Control

Refer to section 11.7.1.4 of Stable Agreements Version 3 as of June 22, 1990.

11.7.2 Constraints on Operations

Refer to section 11.7.2 of Stable Agreements Version 3 as of June 22, 1990.

11.7.2.1 Filters

Refer to section 11.7.2.1 of Stable Agreements Version 3 as of June 22, 1990.

11.7.2.2 Errors

Refer to section 11.7.2.2 of Stable Agreements Version 3 as of June 22, 1990.

Table 11.1. Working Extension to Stable Agreements Table 11.1

Attribute Type	Content	Constraints	Primary Source	Notes
Destination Indicator	T.61 or Printable String	ub-description 1024	CCITT X.520	
Postal Code	T.61 or Printable String	ub-postal-code 40	CCITT X.520	
Registered Address	Postal Address	ub-postal-line 6 ub-postal-string 30	CCITT X.520	
Search Guide	Criteria	256		
Supported Application Context	Object Identifier	256		

11.7.2.3 Error Reporting – Detection of Search Loop

Refer to section 11.7.2.3 of Stable Agreements Version 3 as of June 22, 1990.

11.7.3 Constraints Relevant to Specific Attribute Types

Refer to section 11.7.3 of Stable Agreements Version 3 as of June 22, 1990.

> **Editor's Note:** Table 11.1 is a working extension to Stable Agreements table 11.1 (reference page 11 – 38 of Stable Agreements Version 3 as of June 22, 1990).

11.8 CONFORMANCE

Refer to section 11.8 of Stable Agreements Version 3 as of June 22, 1990.

11.8.1 DUA Conformance

Refer to section 11.8.1 of Stable Agreements Version 3 as of June 22, 1990.

11.8.2 DSA Conformance

Refer to section 11.8.2 of Stable Agreements Version 3 as of June 22, 1990.

11.8.3 DSA Conformance Classes

Refer to section 11.8.3 of Stable Agreements Version 3 as of June 22, 1990.

11.8.4 Authentication Conformance

Refer to section 11.8.4 of Stable Agreements Version 3 as of June 22, 1990.

11.8.5 Directory Service Conformance

Refer to section 11.8.5 of Stable Agreements Version 3 as of June 22, 1990.

11.8.6 The Directory Access Profile

Refer to section 11.8.6 of Stable Agreements Version 3 as of June 22, 1990.

11.8.7 The Directory System Profile

Refer to section 11.8.7 of Stable Agreements Version 3 as of June 22, 1990.

11.8.8 Digital Signature Protocol Conformance Profile

Refer to section 11.8.8 of Stable Agreements Version 3 as of June 22, 1990.

11.8.9 Strong Authentication Protocol Conformance Profile

Refer to section 11.8.9 of Stable Agreements Version 3 as of June 22, 1990.

11.9 DISTRIBUTED OPERATIONS

Refer to section 11.9 of Stable Agreements Version 3 as of June 22, 1990.

11.9.1 Referrals and Chaining

Refer to section 11.9.1 of Stable Agreements Version 3 as of June 22, 1990.

11.9.2 Trace Information

Refer to section 11.9.2 of Stable Agreements Version 3 as of June 22, 1990.

11.10 UNDERLYING SERVICES

Refer to section 11.10 of Stable Agreements Version 3 as of June 22, 1990.

11.10.1 ROSE

Refer to section 11.10.1 of Stable Agreements Version 3 as of June 22, 1990.

11.10.2 Session

Refer to section 11.10.2 of Stable Agreements Version 3 as of June 22, 1990.

11.10.3 ACSE

Refer to section 11.10.3 of Stable Agreements Version 3 as of June 22, 1990.

11.11 ACCESS CONTROL

Refer to section 11.11 of Stable Agreements Version 3 as of June 22, 1990.

11.12 TEST CONSIDERATIONS

Refer to section 11.12 of Stable Agreements Version 3 as of June 22, 1990.

11.12.1 Major Elements of Architecture

Refer to section 11.12.1 of Stable Agreements Version 3 as of June 22, 1990.

11.12.2 Search Operations

Refer to section 11.12.2 of Stable Agreements Version 3 as of June 22, 1990.

11.13 ERRORS

Refer to section 11.13 of Stable Agreements Version 3 as of June 22, 1990.

11.13.1 Permanent vs. Temporary Service Errors

Refer to section 11.13.1 of Stable Agreements Version 3 as of June 22, 1990.

11.13.2 Guidelines for Error Handling

Refer to section 11.13.2 of Stable Agreements Version 3 as of June 22, 1990.

Table 11.2. Working Extension to Stable Agreements Table 11.10

Symptom	Descripton
E_DIT_STRUCTURE	An attempt was made via an add operation to place an object class in a location in the DIT that would violate the DIT structure rules.
E_NONNAMING_ATTRIBUTE	In either an add or ModifyRDN operation, an attribute is included in the last RDN that is not a valid naming attribute according to the DIT structure rules.

11.13.2.1 Introduction

Refer to section 11.13.2.1 of Stable Agreements Version 3 as of June 22, 1990.

11.13.2.2 Symptoms

Refer to section 11.13.2.2 of Stable Agreements Version 3 as of June 22, 1990.

> **Editor's Note:** Table 11.2 is a working extension to Stable Agreements table 11.10 (reference page 11 – 61 to page 11 – 63 of Stable Agreements Version 3 as of June 22, 1990).

11.13.2.3 Situations

Refer to section 11.13.2.3 of Stable Agreements Version 3 as of June 22, 1990.

11.13.2.4 Error Actions

Refer to section 11.13.2.4 of Stable Agreements Version 3 as of June 22, 1990.

> **Editor's Note:** Table 11.3 is a working correction to Stable Agreements table 11.13 (Part 4 of 6, reference page 11 - 69 of Stable Agreements as of June 22, 1990). The correct action under situation ModifyRDN and Symptom E_UNDEFINED_ATT is A(UAT). The current Stable agreements list the action incorrectly as A(OCV).
>
> Table 11.4 is a working extension to Stable Agreements table 11.13 (Part 1 of 6, reference page 11 - 66 of Stable Agreements as of June 22, 1990).
>
> Table 11.5 is a working extension to Stable Agreements table 11.13 (Part 2 of 6, reference page 11 - 67 of Stable Agreements as of June 22, 1990).
>
> Table 11.6 is a working extension to Stable Agreements table 11.13 (Part 3 of 6, reference page 11 - 68 of Stable Agreements as of June 22, 1990).
>
> Table 11.7 is a working extension to Stable Agreements table 11.13 (Part 4 of 6, reference page 11 - 69 of Stable Agreements as of June 22, 1990).
>
> Table 11.8 is a working extension to Stable Agreements table 11.13 (Part 5 of 6, reference page 11 - 70 of Stable Agreements as of June 22, 1990).
>
> Table 11.9 is a working extension to Stable Agreements table 11.13 (Part 6 of 6, reference page 11 - 71 of Stable Agreements as of June 22, 1990).

Table 11.3. Working Correction to Stable Agreements Table 11.13 (Part 4 of 6)

Symptom	Situation				
	Modify-RDN	Remove-Entry	Read	Compare	Trace-Evaluation
E_UNDEFINED_ATT	A(UAT)		A(NSA)(4)	A(NSA)	(7)

Table 11.4. Working Extension to Stable Agreements Table 11.13 (Part 1 of 6)

Symptom	Situation					
	Bind-Local	Bind-Remote-Resolution	Name-Resolution	Add-Entry-Name-Resolution	Add-Entry	Modify-Entry
E_DIT_STRUCTURE					U(NV)	

Table 11.5. Working Extension to Stable Agreements Table 11.13 (Part 2 of 6)

Symptom	Situation					
	Bind-Local	Bind-Remote-Resolution	Name-Resolution	Add-Entry-Name-Resolution	Add-Entry	Modify-Entry
E_NONNAMING_ATTRIBUTE					U(NV)	

Table 11.6. Working Extension to Stable Agreements Table 11.13 (Part 3 of 6)

Symptom	Situation				
	Modify-RDN	Remove-Entry	Read	Compare	Trace-Evaluation
E_DIT_STRUCTURE					

Table 11.7. Working Extension to Stable Agreements Table 11.13 (Part 4 of 6)

Symptom	Situation				
	Modify-RDN	Remove-Entry	Read	Compare	Trace-Evaluation
E_NONNAMING_ATTRIBUTE					

Table 11.8. Working Extension to Stable Agreements Table 11.13 (Part 5 of 6)

Symptom	Situation			
	List (Filter)	Search (Filter)	Search Entry	Abandon
E_DIT_STRUCTURE				

Table 11.9. Working Extension to Stable Agreements Table 11.13 (Part 6 of 6)

Symptom	Situation			
	List (Filter)	Search (Filter)	Search Entry	Abandon
E_NONNAMING_ATTRIBUTE				

11.13.2.5 Reporting

Refer to section 11.13.2.5 of Stable Agreements Version 3 as of June 22, 1990.

11.14 SPECIFIC AUTHENTICATION SCHEMES

Refer to section 11.14 of Stable Agreements Version 3 as of June 22, 1990.

11.14.1 Specific Strong Authentication Schemes

Refer to section 11.14.1 of Stable Agreements Version 3 as of June 22, 1990.

11.14.1.1 ElGamal

Refer to section 11.14.1.1 of Stable Agreements Version 3 as of June 22, 1990.

11.14.1.1.1 References

Refer to section 11.14.1.1.1 of Stable Agreements Version 3 as of June 22, 1990.

11.14.1.1.2 Background

Refer to section 11.14.1.1.2 of Stable Agreements Version 3 as of June 22, 1990.

11.14.1.1.3 Digital Signature

Refer to section 11.14.1.1.3 of Stable Agreements Version 3 as of June 22, 1990.

11.14.1.1.4 Verification

Refer to section 11.14.1.1.4 of Stable Agreements Version 3 as of June 22, 1990.

11.14.1.1.5 Known Constraints on Parameters

Refer to section 11.14.1.1.5 of Stable Agreements Version 3 as of June 22, 1990.

11.14.1.1.6 Note on subjectPublicKey

Refer to section 11.14.1.1.6 of Stable Agreements Version 3 as of June 22, 1990.

11.14.1.2 One–Way Hash Functions

Refer to section 11.14.1.2 of Stable Agreements Version 3 as of June 22, 1990.

11.14.1.2.1 SQUARE–MOD–N Algorithm

Refer to section 11.14.1.2.1 of Stable Agreements Version 3 as of June 22, 1990.

11.14.1.2.2 MD2 Algorithm

Refer to section 11.14.1.2.2 of Stable Agreements Version 3 as of June 22, 1990.

11.14.1.2.3 Study of Other One–Way Hash Functions

The OIW Directory SIG is studying the applicability of alternative one–way hash functions. The most recent development in this area was the announcement by Ralph Merkle that 2–pass SNEFRU has been broken. Its use is therefore discouraged.

11.14.1.2.4 Use of One–Way Hash Functions in Forming Signatures

Refer to section 11.14.1.2.4 of Stable Agreements Version 3 as of June 22, 1990.

11.14.1.3 ASN.1 for Strong Authentication Algorithms

Refer to section 11.14.1.3 of Stable Agreements Version 3 as of June 22, 1990.

> **Editor's Note:** The Stable Agreements Version 3 as of June 22, 1990 contains the following ASN.1.
>
> ```
> elGamal ALGORITHM
>     PARAMETER KeySize
>     ::= {encryptionAlgorithm 1}
> ```
>
> The agreements never explained that the parameter was intended to provide some indication of the level of security (to the extent that the level is a function of the key size). However, recent discussion with several security experts has convenced the Directory SIG that it is not a good idea to judge level of security simply by referring to the key size of a public key cryptographic system. The ASN.1 description has, therefore, been changed to the following.

```
        elGamal ALGORITHM
            PARAMETER NULL
            ::= {encryptionAlgorithm 1}
```

11.14.1.4 Note on the ENCRYPTED MACRO

The value associated with the ENCRYPTED MACRO, as defined in Directory Documents, part 8, clause 8.4 shall be interpreted in the case of ElGamal as being type:

```
    SEQUENCE{ INTEGER, INTEGER }
```

The first integer in the sequence is r (see eq. 5, sec. 11.14.1.1.3 of Stable Agreements as of June 22, 1990). The second integer is s (see eq. 9, sec. 11.14.1.1.3 of Stable Agreements as of June 22, 1990).

11.14.2 Protected Simple Authentication

Refer to section 11.14.2 of Stable Agreements Version 3 as of June 22, 1990.

11.14.3 Simple Authentication

There are two major classes of authentication supported by the Directory (i.e., simple and strong authentication). Simple authentication is based on a password being passed between the two associated entities (DUA–DSA or DSA–DSA). In the case of the DUA–DSA interaction, the password is compared in some way with the password attribute in the user's entry in the Directory. In the case of DSA–DSA interaction, this cannot be done since the DSA object class, as defined in the Directory Documents (Part 7, clause 6.14) does not contain a password attribute.

To facilitate simple authentication between DSAs, a DSA shall have local access to a list of one or more known DSAs, with a copy of each known DSA's password. Maintenance of that information is done through the use of bilateral agreements between DSA administrtors.

11.15 APPENDIX A: MAINTENANCE OF ATTRIBUTE SYNTAXES

Refer to section 11.15 of Stable Agreements Version 3 as of June 22, 1990.

11.15.1 Introduction

Refer to section 11.15.1 of Stable Agreements Version 3 as of June 22, 1990.

11.15.2 General Rules

Refer to section 11.15.2 of Stable Agreements Version 3 as of June 22, 1990.

11.15.3 Checking Algorithms

Refer to section 11.15.3 of Stable Agreements Version 3 as of June 22, 1990.

11.15.3.1 distinguishedNameSyntax

Refer to section 11.15.3.1 of Stable Agreements Version 3 as of June 22, 1990.

11.15.3.2 integerSyntax

Refer to section 11.15.3.2 of Stable Agreements Version 3 as of June 22, 1990.

11.15.3.3 telephoneNumberSyntax

Refer to section 11.15.3.3 of Stable Agreements Version 3 as of June 22, 1990.

11.15.3.4 countryName

Refer to section 11.15.3.4 of Stable Agreements Version 3 as of June 22, 1990.

11.15.3.5 preferredDeliveryMethod

Refer to section 11.15.3.5 of Stable Agreements Version 3 as of June 22, 1990.

11.15.3.6 presentationAddress

Refer to section 11.15.3.6 of Stable Agreements Version 3 as of June 22, 1990.

11.15.4 Matching Algorithms

Refer to section 11.15.4 of Stable Agreements Version 3 as of June 22, 1990.

11.15.4.1 UTCTimeSyntax

Refer to section 11.15.4.1 of Stable Agreements Version 3 as of June 22, 1990.

11.15.4.2 distinguishedNameSyntax

Refer to section 11.15.4.2 of Stable Agreements Version 3 as of June 22, 1990.

11.15.4.3 caseIgnoreListSyntax

Refer to section 11.15.4.3 of Stable Agreements Version 3 as of June 22, 1990.

11.16 APPENDIX B: GLOSSARY

Refer to section 11.16 of Stable Agreements Version 3 as of June 22, 1990.

11.17 APPENDIX C: REQUIREMENTS FOR DISTRIBUTED OPERATIONS

Refer to section 11.17 of Stable Agreements Version 3 as of June 22, 1990.

11.17.1 General Requirements

Refer to section 11.17.1 of Stable Agreements Version 3 as of June 22, 1990.

11.17.2 Protocol Support

Refer to section 11.17.2 of Stable Agreements Version 3 as of June 22, 1990.

11.17.2.1 Usage of ChainingArguments

Refer to section 11.17.2.1 of Stable Agreements Version 3 as of June 22, 1990.

11.17.2.2 Usage of Chainging Results

Refer to section 11.17.2.2 of Stable Agreements Version 3 as of June 22, 1990.

11.18 APPENDIX D: GUIDELINE FOR APPLICATIONS USING THE DIRECTORY

Refer to section 11.18 of Stable Agreements Version 3 as of June 22, 1990.

11.18.1 Tutorial

Refer to section 11.18.1 of Stable Agreements Version 3 as of June 22, 1990.

11.18.1.1 Overview

Refer to section 11.18.1.1 of Stable Agreements Version 3 as of June 22, 1990.

11.18.1.2 Use of the Directory Schema

Refer to section 11.18.1.2 of Stable Agreements Version 3 as of June 22, 1990.

11.18.1.2.1 Use of Existing Object Classes

Refer to section 11.18.1.2.1 of Stable Agreements Version 3 as of June 22, 1990.

11.18.1.2.2 Kinds of Object Classes

Refer to section 11.18.1.2.2 of Stable Agreements Version 3 as of June 22, 1990.

11.18.1.2.3 Use of Unregistered Object Classes

Refer to section 11.18.1.2.3 of Stable Agreements Version 3 as of June 22, 1990.

11.18.1.2.4 Side Effects of Creating Unregistered Object Classes

Refer to section 11.18.1.2.4 of Stable Agreements Version 3 as of June 22, 1990.

11.18.2 Creation of New Object Classes

Refer to section 11.18.2 of Stable Agreements Version 3 as of June 22, 1990.

11.18.2.1 Creation of New Subclasses

Refer to section 11.18.2.1 of Stable Agreements Version 3 as of June 22, 1990.

11.18.2.2 Creation of New Attributes

Refer to section 11.18.2.2 of Stable Agreements Version 3 as of June 22, 1990.

11.18.3 DIT Structure Rules

Refer to section 11.18.3 of Stable Agreements Version 3 as of June 22, 1990.

11.19 APPENDIX E: TEMPLATE FOR AN APPLICATION SPECIFIC PROFILE FOR USE OF THE DIRECTORY

Refer to section 11.19 of Stable Agreements Version 3 as of June 22, 1990.

Table of Contents

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, Version 3 dated June 1990.

Table of Contents

13 Security

13.1 Introduction

13.2 Scope

13.3 Normative References

13.4 Definitions

13.4.1 MHS Elements of Service

Message Origin AuthenticationMT

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 AuthenticationMT

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 SubmissionMT

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 ConfidentialityMT

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 ConfidentialityMT

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 OriginMT

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 SubmissionMT

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 ControlMT

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.

13.5 Symbols and Abbreviations

13.6 Architectures

13.6.1 Introduction

Open Systems Security provides for secure distributed information processing in an environment which is heterogeneous in terms of technology and administration. For example, some environments may require protection from a minimal set of security threats while others require more complete protection.

An objective of the OIW Security SIG is to collaborate with other OIW SIGs in the development of security profiles based upon International Standards and Draft International Standards.*

The architectural objectives include:

a. Development of security profiles in collaboration with other OIW SIGs which support their communication architectures.

b. Agreement on and documentation of the security aspects of current OIW protocols.

c. Ensuring consistency in the use of security services and mechanisms in OIW Implementation Agreements.

Editor's Note: * This refers to the deliverable, stable text and is not to be taken as a constraint on documents to be considered by the group.

13.6.2 General OIW Application Environments

It is useful for the sake of simplification to look at the various OIW groups and to divide them into general categories so that a small set of general security profiles can be applied to similar application environments.

Generalized OIW application environments are given below:

a. Single Application Association (FTAM, VT, MMS, DS)

- Not an application relay
- Association is used to identify the parties in the communication (i.e. no intermediaries)
- Single application over the lifetime of the association
- Data exchanged can use security information which is dependent upon the application association.

b. Store and Forward (MHS)

- All store and forward is done in non-real time (application relay)

- Data exchanged includes complete security information which is not dependent on the application association.

c. Distributed Transactions (TP, RDA)

- Multiple applications over the lifetime of the association are possible
- Features delegation
- ACID properties are mandated (Atomicity, Consistency, Isolation and Durability)
- Need to authenticate at a finer granularity than the association

13.6.3 Security Profiles

This section is organized as follows:

1. Purpose of security profiles
2. Generic threat/security service table
3. Description of generic OIW security profiles

13.6.3.1 Purpose of Security Profiles

Editor's Note: Text TBD. We will further refine the profiles and how they define services and mechanisms in relation to threats.

13.6.3.2 Generic Threat/Security Service Table

Editor's Note: Threat/Security Service table to be developed for:

a. Mapping threats to services will be refined to be a one to many relationship.

b. Detailed threat description.

Editor's Note: Text references to 7498/2 and finer granularity on the threats to be added later.

13.6.3.3 Description of Generic OIW Security Profiles

Profile 0	Null	
Profile 1	Basic	
		Authentication

The rationale for this set of profiles are that it is always an option to not support security at all. However, if security is to be supported, the minimum set of security services provided is authentication. The type of authentication will be a refinement by the specific application environment.
The remaining security services from 7498/2 shown in 13.6.3.2 may or may not be added into application specific security profiles. This is something that needs to be jointly determined between the Security SIG and the other OIW SIG involved.

13.6.4 Guidelines for OIW Application Profile Development

The following guidelines are provided for other OIW SIGs to use in the preliminary development of their own application specific security profile. It is intended that final completion of the security profiles should be done in a joint manner between the Security SIG and the other OIW SIGs.

The basic steps in the guidelines are as follows:

a. Start with the Security SIG Basic Security profile (13.6.3.3).

b. Perform application specific threat analysis. Map the result of this analysis to general security services.

c. Map general security services onto application specific security services (E.G. the threats identified for MHS in X.402 are mapped against MHS specific security services).

d. Editor's Note: Steps d and beyond are TBD. It will require further discussion to decide exactly how the application specific security profile is finally determined, how those profiles can be specified (security context, object identifier?) and how we will specify the mechanisms of choice for the implementation of the profile. Further discussion is needed on Security Policy. This is a priority work item.

Editor's Note: Proposed sections follow below: - text TBD

13.6.l Specification of Security Profile
13.6.l.j Suggested Placement of Security Services
13.6.l.j.k Suggested Mechanisms for Security Services
13.6.l.j.k.l Suggested Placement of Mechanisms

13.7 Key Management

13.8 Lower Layers Security

13.9 Upper Layers Security

13.10 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 13.1 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 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
Msg. Origin Authen.			/1	/			
Report Origin Authen.					/4	/	
Probe Origin Authen.		/		/5			
Proof of Delivery			/2				/
Proof of Submission						/5	
Peer Entity Authen.	/	/		/	/4	/	/
Content Integrity			/1				
Content Confiden.			/1				
Msg. Flow Confiden.			/4				
Msg. Seq. Integrity			/2				
N-Repud. of Origin			/1				
N-Repud. of Submission						/5	
N-Repud. of Delivery			/3				
Access Control	/	/	/1	/	/	/	/

Note: UA: User Agent N-Repud.: Non Repudiation MS: Message Store Authen.: Authentication
. . . . MTA: Message Transfer Agent Confiden.: Confidentiality
. . . . . . Msg.: Message

13.11 Directory Services Security

13.12 Network Management Security

This section outlines an approach to providing security services for OSI Network Management. The goals of this approach are to provide security in a manner that is simple and straightforward to implement, and to avoid any unnecessary computational and managerial overhead. The approach also takes into consideration the need for different levels of securityservices within different network management domains, and the near term requirement for interoperability of network management entities over disparate network types.

13.12.1 Threats

For the purpose of discussion, threats are divided into two categories: primary and secondary threats. Primary threats are those considered to be applicable to the full range of network management implementations, while secondary threats are considered to be applicable to the more limited range of highly secure implementations.

The primary threats to be protected against are the following:

a. The masquerading of a manager or agent entity.

b. The fabrication or modification of Common Management Information Protocol (CMIP) data units.

By countering primary threats, disruption of network management services by the casual user can be avoided.

The secondary threats to be protected against are the following:

a. All primary threats.

b. The disclosure of CMIP data units.

c. The replay, reflection, reordering, insertion, or deletion of CMIP data units.

13.12.2 Security Services

13.12.2.1 Basic Security Services

The security services required to counter primary threats are:

a. Peer Entity Authentication

b. Data Origin Authentication

c. Connectionless Integrity

Peer entity authentication is to occur during the establishment of an application association. If the association is successfully established, the underlying security mechanism provides information that is subsequently used in data origin authentication. There the information may be included in or, in some other way, transform the data units of subsequent exchanges so that they can be identified as originating from an authenticate d entity. Both authentication security services are to be provided at the application level of protocol.

Connectionless integrity insures that data units originating from an authenticated source are not modifiable without detection. When combined with a strong data origin authentication mechanism, the ability to fabricate new data units is also countered. Connectionless integrity may be provided at either the application level of protocol or within one of the lower levels of protocol (i.e., transport or network). The former approach is described in this note and the decision of which to employ is left for further study.

13.12.2.2 Enhanced Security Services

The security services required to counter secondary threats are:

a. All basic security services with the possible exception of connectionless integrity.

b. Connectionless confidentiality.

c. Connection integrity with recovery.

Both connectionless confidentiality and connection integrity may be provided at either the application level of protocol or within one of the lower levels of protocol. The latter provision is assumed here. Enhanced security services are not discussed further in this note, but to be issued as a requirement for lower layer protocol and service standards, and according functional standards to be developed.

13.12.3 Security Mechanisms

13.12.3.1 Peer Entity Authentication

Peer Entity AuthenticationIn order to simplify the management aspects associated with various phases of authentication procedures, the authentication scheme proposed is the exact same one used for secure messaging, based on the CCITT X.509 recommendation. The assumption is made that the certification authorities established for messaging would be usable and suitable for network management as well. It is also assumed that certificates will identify the owner, the owner's public key, dates of validity, and be signed by the certification authority, and that successful authentication results in the establishment of a cryptographic association.

The choice of location to employ for conveying authentication information is left for further study. The choices include the association control service element (ACSE) authentication field, and various locations within the ACSE user data field. It is proposed that the ASN.1 definitions of authentication arguments, the procedures for handling those arguments, and their mapping onto ACSE be consistent with other application layer protocols.

13.12.3.2 Connectionless Integrity

In order to identify whether changes to a data unit have occurred it is proposed that an integrity check value (ICV) be computed over the entire data unit and included in the protocol control information for that data unit. The specification and location for conveying this information is left for further study. Because of the envisaged relationship between the underlying mechanisms employed for data origination authentication andconnectionless integrity, they are to be considered jointly.

13.12.3.3 Data Origination Authentication

The proposed security mechanism for data origination authentication is encipherment and intended to protect the ICV computed for connectionless integrity. Successful peer authentication results in the establishment of a cryptographic association between network management entities. The association allows the originator of a data unit to encrypt it or portions of it, and have the peer recipient verify origination through decryption. In order to minimize computational effort, it is proposed that only the integrity check value be enciphered (i.e., a signature) rather than the entire data unit.

This approach implies that data origination authentication information resides with the integrity check value, and that an according ASN.1 definition reflect any requirements of the signing algorithm or choice of algorithm. However, there appears to be no appropriate location in the application layer protocols employed by network management to convey such data origination authentication information. This issue is left for further study.

13.13 Annex A: ISPICS Requirements List

13.14 Annex B: Bibliography

B.1 ISO/IEC JTC1 SC21 N3614 Information Retrieval, Transfer, and Management for OSI

B.2 ISO/IEC DP 9796 Data Cryptographic Techniques

B.3 Secure Data Network System (SDNS): Key Management Profile - Communications Protocol Requirements (SDN-601/NIST IR 90-4262)

B.4 SDNS: Message Security Protocol (SDN-701/NIST IR 90-4250)

B.5 SDNS: Directory (SDN-702/NIST IR 90-4250)

B.6 ISO/IEC JTC1 SC21/WG1 N5002 Security ASE

B.7 Access Control Information Specification (ACIS)

B.8 SDNS: Key Management Protocol - Definition of Services Provided (SDN-902/NIST IR 90-4262)

B.9 SDNS: Key Management Protocol - Specification of the Protocol (SDN-903/NIST IR 90-4262)

B.10 ISO/IEC JTC1 SC21/WG1 N4110 Authentication ASE Exchange

B.11 SDNS: Security Protocol 3 (SDN-301/NIST IR 90-4250)

B.12 SDNS: Security Protocol 4 (SDN-401/NIST IR 90-4250)

B.13 SDNS: Key Management Protocol - SDNS Traffic Key (SDN-906/NIST IR 90-4262)

B.14 ISO/IEC JTC1 SC21/WG1 N5001 Upper Layers Security Model

B.15 ISO/IEC JTC1 SC21/WG1 F29 N5045 Access Control Framework

B.16 ISO/IEC JTC1 SC21/WG1 F30 Authentication Framework

B.17 ISO/IEC JTC1 SC21/WG1 F31 N5047 Integrity Framework

B.18 ISO/IEC JTC1 SC21/WG1 F32 N5046 Non-Repudiation

B.19 ISO/IEC JTC1 SC21/WG4 N3775 Security Audit Trail

B.20 ISO/IEC JTC1 SC21/WG1 N4110 Authentication ASE Exchange

B.21 ISO/IEC JTC1 SC21/WG7 N4022 Key Management Framework

B.22 ISO/IEC JTC1 SC21/WG1 N5048 Confidentiality Framework

B.23 ISO/IEC JTC1 SC21/WG1 N5049 Guide to OSI Security Standards

B.24 ISO/IEC JTC1 SC21/WG1 N5044 Security Framework Overview

13.15 Annex C: Status

13.16 Annex D: Errata

13.17 Annex E: Security Labels

13.18 Annex F: Security-SIG Management Plan

NUMBER	NEXT MILESTONE	DATE
ISO/IEC JTC1 SC21 N3614		
ISO/IEC DP 9796		
SDN-601/NIST IR 90-4262		
SDN-701/NIST IR 90-4250		
SDN-702/NIST IR 90-4250		
ISO/IEC JTC1 SC21/WG1 N5002		
SDN-902/NIST IR 90-4262		
SDN-903/NIST IR 90-4262		
ISO/IEC JTC1 SC21/WG1 N4110		
SDN-301/NIST IR 90-4250		
SDN-401/NIST IR 90-4250		
SDN-906/NIST IR 90-4262		
ISO/IEC JTC1 SC21/WG1 N5001		
ISO/IEC JTC1 SC21/WG1 F29 N5045		
ISO/IEC JTC1 SC21/WG1 F30		
ISO/IEC JTC1 SC21/WG1 F31 N5047		
ISO/IEC JTC1 SC21/WG1 F32 N5046		
ISO/IEC JTC1 SC21/WG4 N3775		
ISO/IEC JTC1 SC21/WG1 N4110		
ISO/IEC JTC1 SC21/WG7 N4022		
ISO/IEC JTC1 SC21/WG1 N5048		

ISO/IEC JTC1 SC21/WG1 N5049

ISO/IEC JTC1 SC21/WG1 N5044

Table of Contents

14 Part 14 - ISO Virtual Terminal Protocol

Editor's Note: References to Stable Agreements in this part refer to Version 3 dated June 1990.

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

See Stable Agreements regarding X.3 profile.

The Page profiles are 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, which include the profiles for Telnet and Transparent operation, are complete and were stabilized in May, 1988. See Stable Agreements.

14.3.2 Status of Phase Ib

The Forms profile of Phase 1b was stabilized in December, 1988. Alignment with EWOS Forms profile was achieved in September, 1989. See Stable Agreements.

14.3.3 Status of Phase II

The Phase II agreements include profiles for Scroll, X.3 and Page operations and will be completed at an unspecified future date, except for X.3, as mentioned below.

The X.3 profile was stabilized in December, 1989. See Stable Agreements.

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

14.4 Errata

14.5 Conformance

See Stable Agreements.

14.6 Protocol

See Stable Agreements.

14.7 OIW Registered Control Objects

14.7.1 Sequenced Application (SA)

See Stable Agreements.

14.7.2 Unsequenced Application (UA)

See Stable Agreements.

14.7.3 Sequenced Terminal (ST)

See Stable Agreements.

14.7.4 Unsequenced Terminal (UT)

See Stable Agreements.

14.7.5 Termination Conditions CO (TC)

This CO is an instance of the standard type TCCO, as defined in ISO 9040. It is initially designed for use with the OIW Scroll VT profile, though as a registered CO it is available for use by other VT profiles.

In addition to the three standardized data elements, it provides a definition and update syntax for further types of Termination Condition. Each additional type is available for use in additional data elements of the CO. The number and type of such additional data elements is defined in the profile using this CO.

14.7.5.1 Entry Number

To be supplied by the Registration Authority.

14.7.5.2 Name of Sponsoring Body

NIST/OSI Workshop for Implementors of OSI, VTSIG.

14.7.5.3 Date

The date of submission of this proposal is September 15, 1989.

14.7.5.4 Identifier

oiw-vt-co-tcco-tc OBJECT IDENTIFIER ::= { oiw-vt-co-tcco tc(0) }

14.7.5.5 Descriptor Value

"OIW VT CO for Termination Conditions"

14.7.5.6 CO VTE-parameters

```
CO-structure    = ,     *(not defined in this registration, see note 1 in 14.7.5.8)*
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 },
*(the following represents possibly multiple invocations of a generic data element type, according to the
value of CO-structure for the instance of this CO. )*
          FOR N=4 to CO-structure
          {
          CO-element-id  = N,     *(acts as integer identifier for the events in this element)*
          CO-category    = "transparent",
          CO-size        =        *(not defined in this registration, see note 2 in 14.7.5.8)* }
```

14.7.5.7 CO Values, Semantic and Update Syntax

The value fields for data elements 1,2 and 3 are defined in ISO 9040.

The value field for each additional data element is defined by the following ASN.1 construct which also defines the update syntax.

```
TermCondList  ::= SEQUENCE OF CHOICE {
              void                  [0] IMPLICIT NULL,
              x3ForwardingCond      [1] IMPLICIT INTEGER,
              stEventList           [2] IMPLICIT Range,
              anySTUpdate           [3] IMPLICIT NULL,
              stEventMasks          [4] IMPLICIT MaskValues,
              dOChars               [5] IMPLICIT DOCharacters }

Range         ::= SEQUENCE OF SEQUENCE {
                                    [1] IMPLICIT LogEvent,
                                    [2] IMPLICIT LogEvent OPTIONAL }
-- each pair represents an interval of values as defined for the value field of
--CO ST, see 14.7.3.7.  The second value in each pair shall not be smaller than
--the first value.  If the second value is omitted, the interval contains only
--the specified first value.

LogEvent      ::= INTEGER
-- values as defined for value field of CO ST, see 14.7.3.7.

MaskValues    ::= SEQUENCE OF SEQUENCE {
              mask                  [1] IMPLICIT LogEvent,
              value                 [2] IMPLICIT LogEvent }

DOCharacters  ::= SEQUENCE OF SEQUENCE {
                                    [1] IMPLICIT Repref,
                                        [2] IMPLICIT INTEGER,
                                        [3] IMPLICIT INTEGER OPTIONAL }

Repref          ::= INTEGER
-- index to the list of repertoires for the Display Object
```

14.7.5.8 Additional Information

Note: The value of CO-structure is defined in the profile to be the number of types of termination conditions available for use within the profile.

Note: The value of CO-size for each additional data element of this CO must be defined within the profile definition which uses those additional termination conditions.

14.7.5.9 Usage

Defined in profile.

14.8 OIW 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 X3 Profile

See Stable Agreements.

14.8.5 Scroll Profile

OIW VTE-Profile Scroll-1989 (r1,r2,...r9)

14.8.5.1 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 bi-directional, forward and backward.

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 minimizing 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".

14.8.5.2 Association Requirements

14.8.5.2.1 Functional Units

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

14.8.5.2.2 Mode

This profile operates in A-mode.

14.8.5.3 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 = "no-constraint",
                y-absolute = "no",
                y-window = profile-argument-r10
        },

erasure-capability = "yes",

*( repertoire-capability is implied by the number of occurrences of profile-argument-r4 )*

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 = 1
        },
erasure capability = "yes",
*( repertoire-capability is implied by the number of occurrences of profile-argument-r4 )*

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 r9 = "TE" THEN
        {
        CO-name                 = TE, *(Termination Event 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-rl,
                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-r9,
            device-control-object = {ST,UT},
            device-control-object = TE
            }
   },

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

14.8.5.4 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 value(s) for the VTE-parameter repertoire-assignment. The value of repertoire-capability is implied by the number of occurrences of this argument. Default is specified by 9040.

r5 - is optional and provides for the negotiation of a value for the VTE-parameter DO-emphasis. The default value is that given in ISO 9040, B.17.3. Refer to ISO 9040 B.17.4 for rules governing the selection of non-default values.

r6 - is optional and provides for the negotiation of value(s) for VTE-parameters foreground-colour-capability and background-colour-capability. Default is 8.

r7 - 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"}.

r8 - 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"}.

r9 - 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.

r10 - is optional and provides for the negotiation of a value for the VTE-parameter y-window of the DOA Display Object. Default is 24.

14.8.5.5 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.

14.8.5.6 Profile Notes

14.8.5.6.1 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).

14.8.5.6.2 Informative Notes

1. This profile models a scrolling device which is capable of scrolling both forwards and backwards. The display pointer may 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. Use of A-mode enables "typed-ahead"into display object DOB, and such updates can be delivered immediately to the peer VT-user, potentially reducing transmission delays. Such delivery will be forced, and marked, by a termination condition or a VT-DELIVER. Type-ahead is at the discretion of the terminal user.

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 device-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.

14.8.5.7 Specific Conformance Requirements

None.

14.9 Annex A

See Stable Agreements.

14.10 Annex B - Clarifications

14.10.1 Defaults

See Stable Agreements.

14.11 Annex C - Object Identifiers

See Stable Agreements for Object Identifiers assigned to objects in the Stable Agreements. Object Identifiers below have been assigned to objects for which work is still in progress.

Profiles defined by OIW VT SIG:

```
oiw-vt-pr-scroll-1989  OBJECT IDENTIFIER ::= { oiw-vt-pr          scroll-1989(3) }
```

Control Objects defined by OIW VT SIG:

```
oiw-vt-co-tcco-tc  OBJECT IDENTIFIER ::=        { oiw-vt-co-tcco  tc(0) }
```

Table of Contents

15 Introduction

Editor's Note: This section has been editorially changed to allow numbers for subelements.

The NIST/OIW Transaction Processing (TP) Sig is developing implementation agreements for the TP model, service and protocol, ISO 10026 (parts 1,2 and 3).

A transaction, as defined in ISO 10026, is a set of related operations characterized by the ACID properties. The ACID properties are:

> *Atomicity*: a property of a set of related operations such that the operations are either all performed, or none of them are performed.
>
> *Consistency*: a property of a set of related operations such that the effect of the operations are performed accurately, correctly, and with validity, with respect to application semantics. Bound data is moved from one consistent state to another consistent state.
>
> *Isolation*: a property of a set of related operations such that the partial results of the operations are not accessible, except by operations of the set.
>
> *Durability*: a property of a completed set of related operations such that all the effects of the operation are not altered by any sort of failure.

15.1 Scope

These agreements will address the following areas:

1. Specification of functional unit profiles:
   A. Kernel
   B. Polarized Control
   C. Shared Control
   D. Handshake
   E. Commit
   F. Unchained Transactions
2. Agreements covering TP services and generation of TP protocol.
3. Agreements covering the use of the following OSI services by TP:
   A. ACSE for association management
   B. CCR for support of provider supported ACID properties
   C. Presentation service
   D. Directory services
4. Agreements with regard to implementation issues not specified in ISO 10026.
5. Statement of requirements to meet conformance to the agreements.
6. Additionally, the following interoperability issues will be addressed:
   A. TP usage by other OSI standards
   B. Application context
   C. Security

15.2 SPECIFICATION OF FUNCTIONAL UNITS

15.2.1 FUNCTIONAL UNITS

Kernel

Polarized Control

Shared Control

Handshake

Commit

Unchained Transactions

15.2.2 COMBINATIONS OF FUNCTIONAL UNITS

Application Transactions

Unchained Provider-supported Transactions

Chained Provider-supported Transactions

15.2.3 TP USE OF OSI SERVICES

15.2.3.1 ACSE - ASSOCIATION MANAGEMENT

15.2.3.2 CCR - PROVIDER ACID PROPERTIES

15.2.3.3 PRESENTATION SERVICES

15.2.3.4 DIRECTORY SERVICES

15.2.3.5 IMPLEMENTATIONS ISSUES NOT SPECIFIED IN ISO 10026

15.2.3.6 APPLICATION CONTEXT

15.2.3.7 SECURITY

15.2.3.8 RECOMMENDED PRACTICES

15.2.4 CONFORMANCE STATEMENT

15.3 OSI TRANSACTION PROCESSING PROTOCOL AGREEMENTS

The tables below detail the requirements included in the NIST OSI TP Implementation Agreement. The tables present the following information:

o Optional and Mandatory PDU fields and their ranges

o Optional and Mandatory ASE service primitive parameters and their ranges

All the tables are written in a PICS-like format. Each row contains a field or parameter followed by the standard's requirements for that item and then NIST's (Implementation Agreement) requirements. For PDU fields and service parameters, additional columns containing a range and notes are included.
Unless otherwise noted, the following column descriptions and keys apply to all tables:

FIELD/PARAMETER: The particular standard-defined field or parameter being described.

STND: The Transaction Processing standard's (ISO 10026) requirements for the item. This field will have one of the following values; their meaning is defined by the international standard.

M: Mandatory
C: Conditional
O: Optional
NU: Not Used

NIST: This implementation agreement's requirements for the item. This field will have one of the following values; their meaning is defined by the implementation agreement.

Y: Supported, this is a mandatory or optional feature in the base standard. Its syntax and semantics shall be implemented as specified in the base standard or the TP agreements by all implementations claiming conformance to the profile. It is not a requirement that the feature shall be used in all instances of communications, unless mandated by the base standard or stated otherwise in the TP agreement. Fully supported attributes will conform to at least the minimum range of values as defined in ISO 10026-3, unless stated otherwise in the TP agreement. Conformant implementations supporting optional features will be able to interoperate with those implementations which do not support the feature. The support of a feature can depend on the support of a class of features to which it belongs, e.g. parameter in a PDU, a PDU in a functional unit.

O: Optionally supported, is left to the implementation as to whether this feature is supported. If a parameter is optionally supported, then the syntax shall be supported, but it is left to each implementation whether the semantics are supported. The receiver of an unsupported optional parameter which is not subject to negotiation shall, at least, inform the sender by informative diagnostic,and interoperability will not be affected.

NIST RANGE: The allowable range of values for this parameter.

SOURCE: Who supplies data for the parameter. This field will have one of the following values:

TPPM: Transaction Processing Protocol Machine
REQ: Requesting TPSUI

SINK: Who uses the parameter. This field will have one of the following values:

TPPM: Transaction Processing Protocol Machine
IND: Receiving TPSUI
REQ: Requesting TPSUI

NOTES: Any additional comments applying to the parameter.

15.3.1 TP-BEGIN-DIALOGUE-RI

Sending, to begin a dialogue

FIELD	STND	NIST	SOURCE	NIST RANGE	NOTES
Initiating-TPSU-Title	O	O	TPPM	0..2**31-1	
Recipient-TPSU-Title	C	Y	Req	0..2**31-1	
Selected-Functional-Units	C	Y	Req		2
Commit	O	O			
Polarized-Control	O	O			
Handshake	O	O			
Unchained-Transactions	O	O			
Initial-Coordination-Level	C	Y	Req		
Invocation-data	O	O	Req		1
Dialogue/Channel-Identifier	M	Y	TPPM	0..2**31-1	

Sending, to begin a TP channel

FIELD	STND	NIST	SOURCE	NIST RANGE	NOTES
Dialogue/Channel-Identifier	M	Y	TPPM	0..2**31-1	
Channel-utilization	C	Y	TPPM		

Receiving, to begin a dialogue

FIELD	STND	NIST	SINK	NIST RANGE	NOTES
Initiating-TPSU-Title	O	Y	Ind	0..2**31-1	
Recipient-TPSU-Title	C	Y	Ind	0..2**31-1	
Selected-Functional-Units	C	Y	Ind		2
Commit	O	O			
Polarized-Control	O	O			
Handshake	O	O			
Unchained-Transactions	O	O			
Initial-Coordination-Level	C	Y	Ind		
Invocation-data	O	O	Ind		1
Dialogue/Channel-Identifier	M	Y	TPPM	0..2**31-1	

Receiving, to begin a TP channel

FIELD	STND	NIST	SINK	NIST RANGE	NOTES
Dialogue/Channel-Identifier	M	Y	TPPM	0..2**31-1	
Channel-Utilization	C	Y	TPPM		

Note:

1. May need to determine limits on the amount and type of data passed in this manner.
2. See section "Support of Functional Units" for minimum valid combinations of functional units.

15.3.2 TP-BEGIN-DIALOGUE-RC

Sending

FIELD	STND	NIST	SOURCE	NIST RANGE	NOTES
Dialogue/Channel-Identifier	M	Y	TPPM	0..2**31-1	

Receiving

FIELD	STND	NIST	SINK	NIST RANGE	NOTES
Dialogue/Channel-Identifier	M	Y	TPPM	0..2**31-1	

15.3.3 TP-REJECT-RI

Sending

FIELD	STND	NIST	SOURCE	NIST RANGE	NOTES
Type	M	Y	TPPM		
Diagnostic	C	Y			1, 4
User-data	O	O	Req		2, 3
Dialogue/Channel-Identifier	M	Y	TPPM	0..2**31-1	

TP-REJECT-RI, Receiving

FIELD	STND	NIST	SINK	NIST RANGE	NOTES
Type	M	Y	TPPM		
Diagnostic	C	Y			1, 4
User-data	O	O	Req		2, 3
Dialogue/Channel-Identifier	M	Y	TPPM	0..2**31-1	

Note:

1. User/Provider division of values is unclear in standard's ASN.1.
2. May need to determine limits on the amount and type of data passed in this manner.
3. Parameter is present on provider rejects.
4. Parameter is present on user rejects.

15.3.4 TP-BID-RI

No parameters

15.3.5 TP-BID-RC

Sending

FIELD	STND	NIST	SOURCE	NIST RANGE	NOTES
Result	M	Y	TPPM		

Receiving

FIELD	STND	NIST	SINK	NIST RANGE	NOTES
Result	M	Y	TPPM		

15.3.6 TP-END-DIALOGUE-RI

No parameters

15.3.7 TP-U-ERROR-RI

No parameters

15.3.8 TP-U-ERROR-RC

No parameters

15.3.9 TP-P-ERROR-RI

Sending

FIELD	STND	NIST	SOURCE	NIST RANGE	NOTES
Diagnostic	M	Y	TPPM		

Receiving

FIELD	STND	NIST	SINK	NIST RANGE	NOTES
Diagnostic	M	Y	Ind		

15.3.10 TP-ABORT-RI

Sending

FIELD	STND	NIST	SOURCE	NIST RANGE	NOTES
Type	M	Y	TPPM		
Diagnostics	C	Y	TPPM		1, 4
User-data	C	O	Req		2, 3

TP-ABORT-RI, Receiving

FIELD	STND	NIST	SINK	NIST RANGE	NOTES
Type	M	Y	Ind		
Diagnostics	C	Y	Ind		1, 4
User-data	C	O	Ind		2, 3

Note:
1. May want to specify meanings for the reason codes, Permanent and Translent failure.
2. May need to determine limits on the amount and type of data passed in this manner. Text says parm is optional, ASN.1 says mandatory.
3. Parameter is present on provider abort.
4. Parameter is present on user abort.

15.3.11 TP-REQUEST-CONTROL-RI

No parameters

15.3.12 TP-GRANT-CONTROL-RI

No parameters

15.3.13 TP-HANDSHAKE-RI

Sending

FIELD	STND	NIST	SOURCE	NIST RANGE	NOTES
Type	M	Y	TPPM		
Confirmation	C	Y	Req		

TP-HANDSHAKE-RI, *Receiving*

FIELD	STND	NIST	SINK	NIST RANGE	NOTES
Type	M	Y	TPPM		
Confirmation	C	Y	TPPM		1

Note: 1. Parameter is present only on handshake when Shared Control functional unit is active.

15.3.14 TP-HANDSHAKE-RC

No parameters

15.3.15 TP-HANDSHAKE-AND-GRANT-CONTROL-RI

Sending

FIELD	STND	NIST	SOURCE	NIST RANGE	NOTES
Confirmation	M	Y	Req		

Receiving

FIELD	STND	NIST	SINK	NIST RANGE	NOTES
Confirmation	M	Y	TPPM		

15.3.16 TP-HANDSHAKE-AND-GRANT-CONTROL-RC

No parameters

15.3.17 TP-DEFER-RI

Sending

FIELD	STND	NIST	SOURCE	NIST RANGE	NOTES
End-dialogue	O	Y	TPPM		1
Grant-control	O	Y	TPPM		1
Next-Transaction	O	Y	TPPM		1

Receiving

FIELD	STND	NIST	SINK	NIST RANGE	NOTES
End-dialogue	O	Y	TPPM		1
Grant-control	O	Y	TPPM		1
Next-Transaction	O	Y	TPPM		1

Note: **1. The field is mandatory only if required by supported functional units, else it is not used.**

15.3.18 TP-PREPARE-RI

Sending

FIELD	STND	NIST	SOURCE	NIST RANGE	NOTES
Data-permitted	O		Req		

TP-PREPARE-RI, Receiving

FIELD	STND	NIST	SINK	NIST RANGE	NOTES
Data-permitted	O		Ind		

15.3.19 TP-UNCHAIN-RI

No parameters

15.3.20 TP-BEGIN-TRANSACTION-RI

Sending

FIELD	STND	NIST	SOURCE	NIST RANGE	NOTES
Chain	M	Y	TPPM		

Receiving

FIELD	STND	NIST	SINK	NIST RANGE	NOTES
Chain	M	Y	TPPM		

15.3.21 TP-ASSOCIATION-ESTABLISHMENT-RI

Sending

FIELD	STND	NIST	SOURCE	NIST RANGE	NOTES
Protocol Version	M	Y	TPPM		
Contention winner assignment	M	Y	TPPM		
Bid-Mandatory	M	Y	TPPM		

Receiving

FIELD	STND	NIST	SINK	NIST RANGE	NOTES
Protocol Version	M	Y	TPPM		
Contention winner assignment	M	Y	TPPM		
Bid-Mandatory	M	Y	TPPM		

15.3.22 TP-ASSOCIATION-ESTABLISHMENT-RC

Sending

FIELD	STND	NIST	SOURCE	NIST RANGE	NOTES
Protocol Version	M	Y	TPPM		

Receiving

FIELD	STND	NIST	SINK	NIST RANGE	NOTES
Protocol Version	M	Y	TPPM		

15.4 ACSE SERVICE PARAMETERS

This section shows TP's use of ACSE services and parameters.

15.4.1 A-ASSOCIATE

Sending (Request/Response)

PARAMETER	STND	NIST	NIST RANGE	NOTES
Mode	M	Y		
Application Context Name	M	Y		
Calling AP Title	M(A)			
Calling AE Qualifier	M(A)			
Calling AP Invocation Identifier	M			
Calling AE Invocation Identifier	M			
Called AP Title	C(A)			
Called AE Qualifier	C(A)			
Called AP Invocation Identifier	C(B)			
Called AE Invocation Identifier	C(B)			
Responding AP Title	M(A)			
Responding AE Qualifier	M(A)			
Responding AP Invocation Identifier	M(A)			

PARAMETER	STND	NIST	NIST RANGE	NOTES
Responding AE Invocation Identifier	M(A)			
User Information	M	Y		
Result	M	Y		
Diagnostic	O	O		
Calling Presentation Address	M	Y		
Called Presentation Address	M	Y		
Responding Presentation Address	O	O		
Presentation Context Definition List	M	Y		
Presentation Context Definition Result List	O	O		
Default Presentation Context Name	O	NU		
Default Presentation Context Result	O	NU		
Quality of Service	M	Y		
Presentation Requirements	M	Y	Kernel only	
Session Requirements	M	Y	Kernel + Full Duplex + CCR requirements (if used)	

PARAMETER	STND	NIST	NIST RANGE	NOTES
Initial Synchronization point Serial Number	M(A)			
Initial Assignment of Tokens	M(A)			
Session-Connection Identifier	NU	NU		

Note: (A) Only if CCR is used, else parameter is a user option
(B) Parameter becomes mandatory if the association is being established for

A-ASSOCIATE
Receiving (Indication/Confirmation)

PARAMETER	STND	NIST	NIST RANGE	NOTES
Mode	M	Y		
Application Context Name	M	Y		
Calling AP Title	M(A)			
Calling AE Qualifier	M(A)			
Calling AP Invocation Identifier	M			
Calling AE Invocation Identifier	M			
Called AP Title	C(A)			
Called AE Qualifier	C(A)			

PARAMETER	STND	NIST	NIST RANGE	NOTES
Called AP Invocation Identifier	C(B)			
Called AE Invocation Identifier	C(B)			
Responding AP Title	M(A)			
Responding AE Qualifier	M(A)			
Responding AP Invocation Identifier	M(A)			
Responding AE Invocation Identifier	M(A)			
User Information	M	Y		
Result	M	Y		
Result Source	M	Y		
Diagnostic	O	O		
Calling Presentation Address	M	Y		
Called Presentation Address	M	Y		
Responding Presentation Address	O	O		

PARAMETER	STND	NIST	NIST RANGE	NOTES
Presentation Context Definition List	M	Y		
Presentation Context Definition Result List	O	Y		
Default Presentation Context Name	O	NU		
Default Presentation Context Result	O	NU		
Quality of Service	M	Y		
Presentation Requirements	M	Y	Kernel only	
Session Requirements	M	Y	Kernel + Full Duplex + CCR requirements (if used)	
Initial Synchronization Point Serial Number	M(A)			
Initial Assignment of Tokens	M(A)			
Session-Connection Identifier	NU	NU		

Note: (A) Only if CCR is used, else parameter is a user option
(B) Parameter becomes mandatory if the association is being established for recovery purposes (channels)

15.4.2 A-RELEASE

Sending (Request/Response)

PARAMETER	STND	NIST	NIST RANGE	NOTES
Reason	NU	NU		
User information	NU	NU		
Result	M	Y		

Receiving (Indication/Confirmation)

PARAMETER	STND	NIST	NIST RANGE	NOTES
Reason	NU	NU		
User information	NU	NU		
Result	M	Y		

15.4.3 A-ABORT

Sending (Request)

PARAMETER	STND	NIST	NIST RANGE	NOTES
User Information	NU	NU		

Receiving (Indication)

PARAMETER	STND	NIST	NIST RANGE	NOTES
Abort Source	M	Y		
User information	NU	NU		

15.4.4 A-P-ABORT

Receiving (Indication)

PARAMETER	STND	NIST	NIST RANGE	NOTES
Provider Reason	O	O		

15.5 PRESENTATION SERVICE PARAMETERS

This section shows TP's use of Presentation services and parameters.

15.5.1 P-TOKEN-PLEASE

Sending (Request)

PARAMETER	STND	NIST	NIST RANGE	NOTES
Tokens				1
User-data	NU	NU		

Receiving (Indication)

PARAMETER	STND	NIST	NIST RANGE	NOTES
Tokens				1
User-data	NU	NU		

Editor's Note: 1 Why is there an inconsistency in the token parameter P-Token-Please and P-Token-Give.

15.5.2 P-TOKEN-GIVE

Sending (Request)

PARAMETER	STND	NIST	NIST RANGE	NOTES
Tokens	M	Y		

Receiving (Indication)

PARAMETER	STND	NIST	NIST RANGE	NOTES
Tokens	M			

15.5.3 P-DATA

Sending (Request)

PARAMETER	STND	NIST	NIST RANGE	NOTES
User-data	M	Y		

Receiving (Indication)

PARAMETER	STND	NIST	NIST RANGE	NOTES
User-data	M	Y		

15.6 CCR SERVICE PARAMETERS

This section shows TP's use of CCR services and parameters.

15.6.1 C-BEGIN

Sending (Request/Response)

PARAMETER	STND	NIST	NIST RANGE	NOTES
Atomic Action Id.-Master's Name	M	Y		
Atomic Action Id.-Suffix	M	Y		1
Branch Id.-Superior's Name	M	Y		
Branch Id.-Suffix	M	Y		1
User Data	C	Y		

Receiving (Indication/Confirmation)

PARAMETER	STND	NIST	NIST RANGE	NOTES
Atomic Action Id.-Master's Name	M	Y		
Atomic Action Id.-Suffix	M	Y		1
Branch Id.-Superior's Name	M	Y		
Branch Id.-Suffix	M	Y		1
User Data	C	Y		

Note: 1. Must decide which CCR ASN.1 Choice to use

15.6.2 C-PREPARE

Sending (Request)

PARAMETER	STND	NIST	NIST RANGE	NOTES
User-data	C	Y		

Receiving (Indication)

PARAMETER	STND	NIST	NIST RANGE	NOTES
User-data	C	Y		

15.6.3 C-READY

Sending (Request)

PARAMETER	STND	NIST	NIST RANGE	NOTES
User-data	NU	NU		

Receiving (Indication)

PARAMETER	STND	NIST	NIST RANGE	NOTES
User-data	NU	NU		

15.6.4 C-COMMIT

Sending (Request/Response)

PARAMETER	STND	NIST	NIST RANGE	NOTES
User-data	C	Y		

C-COMMIT, Receiving (Indication/Confirmation)

PARAMETER	STND	NIST	NIST RANGE	NOTES
User-data	C	Y		

15.6.5 C-ROLLBACK

Sending (Request/Response)

PARAMETER	STND	NIST	NIST RANGE	NOTES
User-data	C	Y		

Receiving (Indication/Confirmation)

PARAMETER	STND	NIST	NIST RANGE	NOTES
User-data	C	Y		

15.6.6 C-RECOVER

Sending (Request/Response)

PARAMETER	STND	NIST	NIST RANGE	NOTES
Recovery State	M	Y		
Atomic Action Identifier	M	Y		
Branch Identifier	M	Y		
User-data	C	Y		

C-RECOVER, *Receiving (Indication/Confirmation)*

PARAMETER	STND	NIST	NIST RANGE	NOTES
Recovery State	M	Y		
Atomic Action Identifier	M	Y		
Branch Identifier	M	Y		
User-data	C	Y		

Table of Contents

16 OFFICE DOCUMENT ARCHITECTURE

There is international alignment work progressing between the OIW, EWOS, and AOW on the Level 3 DAP (based on Chapter 16 in the Stable Document). As these alignment changes are completed, appropriate changes will be included in a revised Chapter 16. The current intention is to rename Chapter 16 to "Office Document Architecture Level 3 DAP."

The intention is to declare this revised work stable in December 1990.

Table of Contents

17 FUTURE OFFICE DOCUMENT ARCHITECTURE

There is international alignment work progressing between the OIW, EWOS, and AOW on the Level 2 DAP, based on draft text which was in Part 17. As these alignment changes are completed, revised text is provided.

The intent is to make this revised text stable in December 1990.

Previous material in Part 17 has been deleted.

Table of Contents

List of Tables

List of Figures

18 NETWORK MANAGEMENT

Editor's Note: [There is currently no text for subclauses 8, 9, and 10 (described below).]

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

18.1 INTRODUCTION

(Refer to the Stable Implementation Agreements Document.)

18.1.1 References

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

<u>OSI Systems Management References:</u>

[ADDRMVP] ISO/IEC 9596/DAD 2, Common Management Information Protocol Specification: Addendum 2 (Add/Remove Protocol), ISO/IEC JTC1/SC21, 1 February 1990.

[ADDRMVS] ISO/IEC 9595/DAD 2, Common Management Information Service Definition: Addendum 2 (Add/Remove Service), ISO/IEC JTC1/SC21, 1 February 1990.

[ALS] ISO/IEC DIS 9545, Information Processing Systems - Open Systems Interconnection - Application Layer Structure, 15 March 1989.

[AMF] ISO/IEC CD 10164-10, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 10: Accounting Meter Function, ISO/IEC JTC1/SC21 N4958, June 1990.

[AMWD] Information Processing Systems - Open Systems Interconnection - Accounting Management Working Document (Fourth Version), ISO/IEC JTC1/SC21, May 30, 1990.

[ARF] ISO/IEC DIS 10164-4, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 4: Alarm Reporting Function, ISO/IEC JTC1/SC21 N4858, June 1990.

[ARR] ISO/IEC DIS 10164-3, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 3: Attributes for Representing Relationships, ISO/IEC JTC1/SC21 N4857, June 1990.

[CANGETP] ISO/IEC 9596/DAD 1, Common Management Information Protocol Specification: Addendum 1 (CancelGet Protocol), ISO/IEC JTC1/SC21, 1 February 1990.

[CANGETS] ISO/IEC 9595/DAD 1, Common Management Information Service Definition: Addendum 1 (CancelGet Service), ISO/IEC JTC1/SC21, 1 February 1990.

[CDTC] Information Processing Systems - Open Systems Interconnection - Systems Management - Part Z: Confidence and Diagnostic Test Classes (First Version) ISO/IEC JTC1/SC21 N4957, May 1990.

[CMIP] ISO/IEC 9596-2, Information Processing Systems - Open Systems Interconnection - Management Information Protocol Specification - Part 2: Common Management Information Protocol, 6 December l989.

[CMIS] ISO/IEC 9595-2, Information Processing Systems - Open Systems Interconnection - Management Information Service Definition - Part 2: Common Management Information Service, 6 December 1989.

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

[DMI] ISO/IEC DIS 10165-2, Information Processing Systems - Open Systems Interconnection - Structure of Management Information - Part 2: Definition of Management Information, ISO/IEC JTC1/SC21 N4867, June 1990.

[ERF] ISO/IEC DIS 10164-5, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 5: Event Report Function, ISO/IEC JTC1/SC21 N4860, June 1990.

[FMWD] Information Processing Systems - Open Systems Interconnection - Systems Management - Fault Management Working Document, ISO/IEC JTC1/SC21 N4077, December 1989.

[FRMWK] ISO 7498-4, Information Processing Systems - Open Systems Interconnection - Basic Reference Model - Part 4: Management Framework, 1989.

[GDMO] ISO/IEC DIS 10165-4, Information Processing Systems - Open Systems Interconnection - SMI - Part 4: Guidelines for the Definition of Managed Objects, ISO/IEC JTC1/SC21 N4852, 15 June 1990.

[LCF] ISO/IEC DIS 10164-6, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 6: Log Control Function, ISO/IEC JTC1/SC21 N4862, June 1990.

[MIM] ISO/IEC DIS 10165-1, Information Processing Systems - Open Systems Interconnection - Management Information Services - Structure of Management Information - Part 1: Management Information Model, ISO/IEC JTC1/SC21 N5252, June 1990.

[MSF] ISO/IEC CD 10164-m, Information Processing Systems - Open Systems Interconnection - Systems Management - Part m: Measurement Summarization Function (Second Working Draft), ISO/IEC JTC1/SC21 N4972, July 2, 1990.

[OAAC] ISO/IEC CD 10164-9, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 9: Objects and Attributes for Access Control, ISO/IEC JTC1/SC21 N4956, June 1990.

[OMF] ISO/IEC DIS 10164-1, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 1: Object Management Function, ISO/IEC JTC1/SC21, June 1990.

[OSIMIL] Management Information Library (MIL) - Revision 3.0, OSI MIB Working Group of NMSIG of NIST/OSI Implementors Workshop, 10 March 1990.

[PMWD] Information Processing Systems - Open Systems Interconnection - Performance Management Working Document (Sixth Draft), ISO/IEC JTC1/SC21 N4981, July 4, 1990.

[SARF] ISO/IEC DP 10164-7, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 7: Security Alarm Reporting Function, ISO/IEC JTC1/SC21 N6064, 20 November 1989.

[SATF] ISO/IEC CD 10164-8, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 8: Security Audit Trail Function, ISO/IEC JTC1/SC21 N4955, June 1990.

[SMF] ISO/IEC DIS 10164-2, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 2: State Management Function, ISO/IEC JTC1/SC21, June 1990.

[SMO] ISO/IEC DIS 10040, Information Processing Systems - Open Systems Interconnection - Systems Management Overview, ISO/IEC JTC1/SC21 N4865R, 16 June 1990.

[SMWD] Information Processing Systems - Open Systems Interconnection - Systems Management - OSI Security Management Working Document - 7th Draft, ISO/IEC JTC1/SC21 N4091, 15 November 1989.

[TMF] Information Processing Systems - Open Systems Interconnection - Systems Management - Part Y: Test Management Function, ISO/IEC JTC1/SC21 N4978, June 1990.

[WMF] ISO/IEC CD 10164-11, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 11: Workload Monitoring Function, ISO/IEC JTC1/SC21 N4959, June 28, 1990.

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, 1988.

[PPS] ISO/IEC DIS 8823, Information Processing Systems - Open Systems Interconnection - Connection Oriented Presentation Protocol Specification, ISO/IEC JTC1/SC21 N2336, 5 April 1988.

[PSD] ISO/IEC Final Text of DIS 8822, Information Processing Systems - Open Systems Interconnection - Connection Oriented Presentation Service Definition, ISO/IEC JTC1/SC21 N2335, 5 April 1988.

[ROSEP] ISO/IEC 9072-2 - Information Processing Systems - Text Communications - Remote Operations Part 2: Protocol Specification, 19 September 1989.

[ROSES] ISO/IEC 9072-1, Information Processing Systems - Text Communications - Remote Operations Part 1: Model, Notation and Service Definition, 19 September 1989.

Other References

[MAP30] MAP 3.0 Network Management Specification, August 1988.

Editor's Note: [Clause 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 Part (Part 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 December, 1989. These Phase 1 agreements provide limited interoperable management in a heterogeneous vendor environment. They are the cornerstone 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 clauses 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 1989). Subsequent phases of these agreements (post December 1989) will expand the scope of problems addressed.

Editor's Note: [The following phase definitions and milestones represent the current workplan of the NMSIG. The target dates are the earliest dates at which the milestones could possibly be accomplished and depend (in part) on optimistic assumptions about the progress of relevant standards.

The scope of Phase 1 IA's will be the following:

Management Functions:
Object Management, State Management, Relationship Management, Error Reporting and Event Control

Management Information:
Information Model, Naming, Guidelines and Template for Defining Managed Objects
Management Communication:
CMIS/P, Association Policies, and Services Required

Management Object:
Support Objects required for above and 14 Managed Object Definitions under development by the OSI MIB WG

Conformance Criteria:
TBD depending on the progress of relevant ISO documents.

The milestones for Phase 1 IAs and earliest target dates are:

Milestone A: [12/89]
Freeze the scope of Phase 1 and approve first draft text for Ongoing IAs that cover all of Phase 1 except Managed Objects and Conformance Criteria.

Milestone B: [3/90]
Add the Phase 1 Managed Objects to the Ongoing IAs.

Milestone C: [6/90]

Align the Ongoing IAs pertaining to Phase 1 with ISO DIS documents. Add conformance criteria pertaining to Phase I to the Ongoing IAs.

Milestone D: [9/90]
Progress the Ongoing IAs pertaining to Phase 1 into Stable IAs.

The preliminary milestones and earliest target dates for Phase 2 are:

Milestone E: [3/90]
Define the Scope of Phase 2 IAs.

Milestone F: [9/90]
Freeze the Scope of Phase 2 IAs and approve the first draft text covering all of Phase 2.

It is the intention of the NMSIG to freeze the content of Phase 1 at Milestone A. Only those changes required to align with the ISO DIS's will be made.

It is the intention of the NMSIG to define Phase 2 functionality as a compatible superset of Phase 1.]

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

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

- o Subclause 18.2.1 describes the relationship between these agreements and the evolving international management standards.
- o Subclause 18.2.2.1 provides a brief overview of the management architecture described in the standards documents.
- o Subclause 18.2.2.2 identifies the constraints imposed on Phase 1 of these agreements.
- o Subclause 18.2.2.3 addresses migration strategies regarding subsequent phases of these agreements.
- o Subclause 18.2.2.4 addresses interoperability with systems associated with other management specifications (including MAP/TOP) [MAP30].
- o Subclause 18.2.3 presents an overview of the functionality supported by Phase 1 of these agreements.

Clause 18.3 -- STATUS: This clause describes the current status of these agreements.

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

Clause 18.5 -- MANAGEMENT FUNCTIONS: This clause documents agreements pertaining to the Systems Management Functions. In addition, it identifies agreements pertaining to the use of other application service elements (e.g., the Common Management Information Service Element (CMISE)).

Clause 18.6 -- MANAGEMENT COMMUNICATIONS: This clause 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.

Clause 18.7 -- MANAGEMENT INFORMATION: This clause is based on evolving ISO documents [MIM] and [GDMO], and provides tutorial material and agreements for management information related concepts and modelling techniques. Subclauses introduce the information model, list principles for naming managed objects and attributes, and provide guidelines for defining management information.

Managed object definitions are outside the scope of this clause and are provided in the Management Information Library (MIL). (The MIL is produced by the OSI MIB Working Group, a subgroup of the NMSIG.)

Clause 18.8 -- IMPLEMENTATION PROFILES/CONFORMANCE CLASSES: This clause describes the implementation profiles/conformance classes that are used to categorize management products. At the highest level, products fall into two broad categories: systems that take on a managing system role and systems that take on an agent system role representing managed objects. (Refer to clause 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.]

Clause 18.9 -- CONFORMANCE: For each of the classes identified in clause 18.8, this clause 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:

- o 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.]

Clause 18.10 -- REGISTRATION REQUIREMENTS: This clause 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 clause 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 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 SC 21/WG 4 Meeting in Seoul, June 5 - 6, 1990.]

Table 1: RELEVANT STANDARDS DOCUMENTS AND THE CURRENT SCHEDULES FOR PROGRESSING THESE DOCUMENTS TO IS STATUS

	Target Dates		
	CD	DIS	IS
Management Framework	9/86	6/87	10/88
Systems Management Overview		6/90	7/91
Structure of Management Information			
Part 1: Management Information Model	5/89	6/90	7/91
Part 2: Definition of Management Information		6/90	8/91
Part 4: Guidelines for the Definition of Managed Objects	11/89	6/90	7/91
Common Management Information Service			1/90
Amendment 1: CancelGet		9/89	11/90
Amendment 2: Add/Remove		9/89	11/90
Amendment: Support for Allomorphism	11/90	11/91	11/92
Amendment: Access Control	11/90	11/91	11/92
Common Management Information Protocol			1/90
Amendment 1: CancelGet		9/89	11/90
Amendment 2: Add/Remove		9/89	11/90
Amendment: State Table	7/91	7/92	7/93
Amendment: Support for Allomorphism	11/90	11/91	11/92
Amendment: PICS Proforma	11/90	6/91	6/92
Configuration Management			
Systems Management - Part 1: Object Management Function		6/90	7/91
Systems Management - Part 2: State Management Function		6/90	7/91
Systems Management - Part 3: Relationship Management Function		6/90	7/91
Fault Management			
Systems Management - Part 4: Alarm Reporting Function		6/90	7/91
Systems Management - Part 5: Event Report Management Function		6/90	7/91
Systems Management - Part 6: Log Control Function	11/89	6/90	7/91
Systems Management - Part Z: Confidence and Diagnostic Test Classes	11/90	8/91	8/92
Systems Management - Part Y: Test Management Function	11/90	8/91	8/92

Document	Target Dates		
	CD	DIS	IS
Security Management			
Systems Management - Part 7: Security Alarm Reporting Function	11/89	6/90	7/91
Systems Management - Part 8: Security Audit Trail Function	7/90	4/91	4/92
Systems Management - Part 9: Objects and Attributes for Access Control	7/90	4/91	4/92
Accounting Management			
Systems Management - Part 10: Accounting Metering Function	7/90	4/91	4/92
Performance Management			
Systems Management - Part 11: Workload Monitoring Function	7/90	4/91	4/92
Systems Management - Part B: Measurement Summarization Function	11/90	8/91	8/92
Systems Management - Part X: Software Management Function	7/92	7/93	7/94
Systems Management - Part A: Time Management Function (Representation of Time)	11/91	8/92	8/93

Given the current state of the standards, the ongoing Phase 1 implementors' agreements are based on documents, some of which are not yet at the DIS level. 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 to DIS and IS, the agreements will be aligned.

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 phases 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 clause 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

Editor's Note: [This subclause is tutorial.]

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., clause 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:

- o References [FRMWK] and [SMO] address the general concepts.
- o References [ALS], [CMIS], and [CMIP] address the communications standards.
- o References [MIM], [DMI], [DMI], and [GDMO] pertain to the definition of management information (managed objects).
- o 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], [ARR], [ARF], and [ERF].]

GENERAL CONCEPTS

Viewed abstractly, 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 physically distributed, the components of the information processing application are themselves distributed. These distributed components take the form of management application processes. These distributed application processes may be organized in many ways, as for example, in a hierarchical manner or on a peer-to-peer basis.

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 message specifying management operations on managed objects. Agent processes may also forward messages 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 messages 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:

- o Configuration Management
- o Fault Management
- o Security Management
- o Performance Management
- o 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:

- o A set of functions that support the functionality within the scope of the SMFA.

- o The procedures associated with the provision of each function.

- o The services required to support these procedures.

- o The use of the underlying OSI services to provide the communications needs.

- o The classes of managed objects that the procedures will operate upon in order to provide the functionality defined by the SMFA.

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 clauses 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. No agreements are provided regarding management domains.

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 clause 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 except for the use of access control on association initialization.

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 subclause 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 subclause 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 subclause is to amplify the high level NM requirements to be met by these IAs. In particular, this subclause 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

Part 18 is currently a working draft of the Phase 1 Network Management Implementors Agreements.

Editor's Note: [The intention is to possibly move at least some of this material to stability in 1990. Therefore, the content of this chapter should be closely examined.]

18.4 ERRATA

(None as yet)

18.5 MANAGEMENT FUNCTIONS AND SERVICES

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]). 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 event report function [ERF] 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 (figure 1) depicts the functional hierarchy of SMFs and SMFAs. There are currently seven SMF draft international standards: Object Management [OMF], State Management [SMF], Attributes For Representing Relationships [ARR], Alarm Reporting [ARF], Event Report [ERF], Log Control [LCF], and Security Alarm Reporting [SARF]. These SMFs 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. Committee drafts are currently in progress for the following additional SMFs: Security Audit Trail [SATF], Accounting Metering [AMF], and Workload Monitoring [WMF]. Working drafts are currently in progress for the following additional SMFs: Confidence and Diagnostic Testing (consisting of two documents, one specifying a Test Management Function [TMF], and the other defining related management support objects classes and attributes [CDTC]), and Measurement Summarization [MSF].

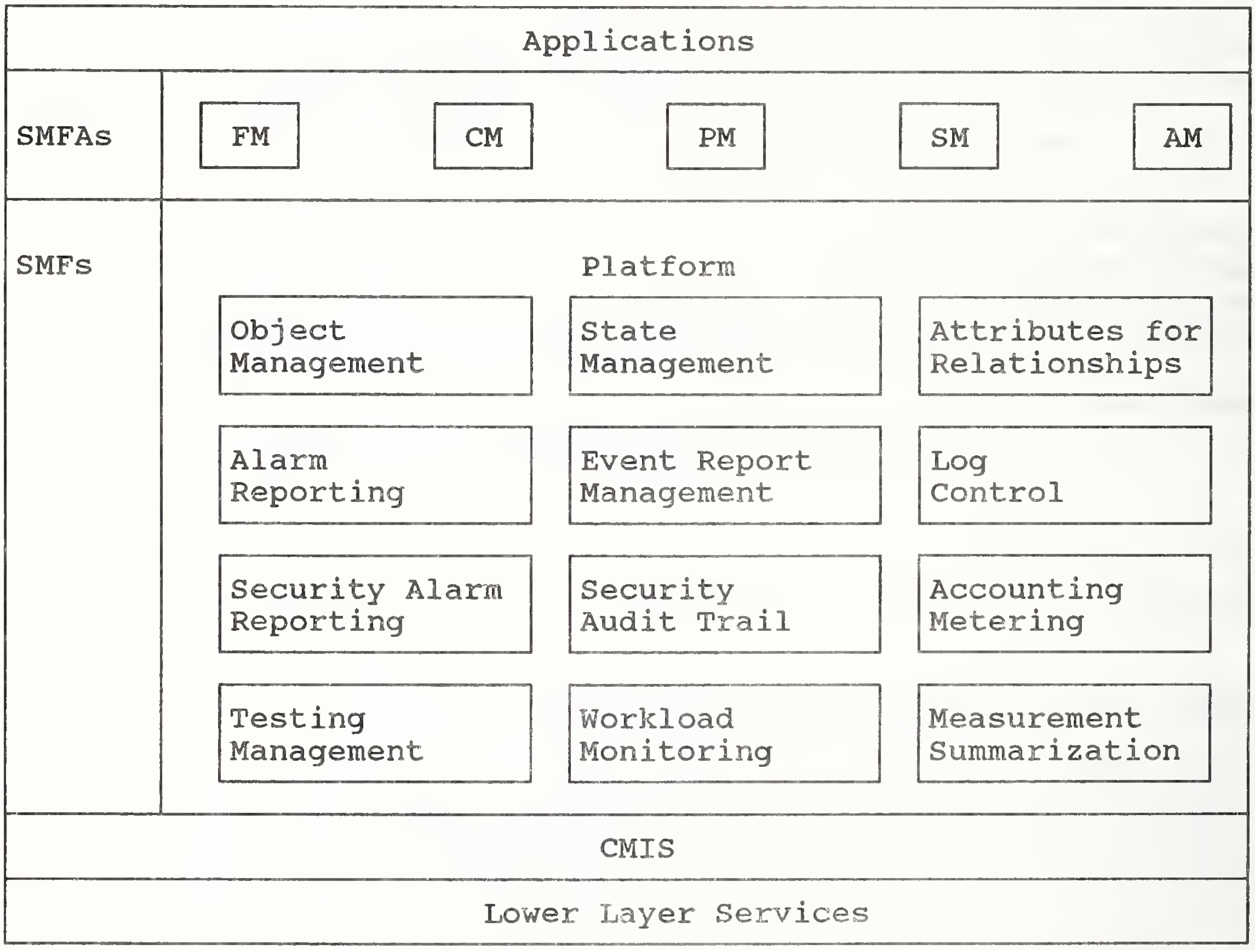


Figure 1: Functional Hierarchy of SMFs and SMFAs.

Editor's Note: [This text represents the output of the June 1990 NMSIG meeting. Changes since the previous version are primarily editorial in nature, including reformatting approved during the March 1990 NMSIG meeting and alignment with the output from the May 1990 ISO SC21/WG4 meeting in Seoul. Seven substantive changes were made, as summarized below. In addition, the following topics must be revisited at the next NMSIG meeting: addition of the Log Control Function [LCF], SMFUs, and Agent/Manager conformance.

1. Use Of Mode Parameter

Existing agreements limiting the support and use of the Mode parameter in all SMF Services mapped to CMIS M-EVENT-REPORT, M-ACTION, and M-SET were modified.

2. Invoke Identifier Values

The previous agreement restricting assignment of values to Invoke Identifier parameters was deleted. (The CMIP agreement on the maximum length of Invoke Identifier parameters remains in clause 18.6.)

3. Seoul Alignment

Many of the changes resulting from Seoul output alignment were editorial. However, some of the changes require further review to determine whether additional agreements are necessary, including the following:

[OMF]	Source Indicator parameter on all notifications
[SMF]	Use of new Status attributes
[ARF]	Use of new Alarm Record MSO
[ERF]	Use of new Event Record MSO Use of new Event Forwarding Discriminator attributes: Usage State Availability Status Backup Address Active Address Scheduling Packages Allomorphic List

4. Backup Object Instance

A new agreement was added on the use of the Alarm Report Backup Object Instance parameter.

5. Discriminator Construct

A new agreement was added on the value used to represent an "all-pass" Discriminator Construct attribute of the Event Forwarding Discriminator.

6. Use of Scoped ERF-SET

The previous agreement restricting use of scoped ERF-SET was deleted in the process of merging this obsolete SMF Service with the obsolete ERF-Suspend/Resume SMF Service. (The output from Seoul merges these two SMF Services into a single, unified EFD Modification, Suspension, and Resumption SMF Service.)

7. Use of Scoped M-DELETE

The previous agreement prohibiting the use of scoped M-DELETE was relaxed so that use of this feature is now permitted beyond the scope of these agreements.

8. Multiple EFD's With Same Dest Address

The previous agreement prohibiting the creation of multiple Event Forwarding Discriminators with the same Destination Address was deleted. (This "agreement" was probably too strong an interpretation of previously ambiguous text.)

9. Minimal Filter Complexity

The previous agreement on minimal filter complexity was extended to apply to the Discriminator Construct attribute whenever it appears as a parameter in SMF services.]

18.5.1 General Agreements

18.5.1.1 Current Scope Of SMF Agreements

The following System Management Functions are undergoing standardization and have been registered as draft international standards:

(1) Object Management Function [OMF]

(2) State Management Function [SMF]

(3) Attributes For Representing Relationships [ARR]

(4) Alarm Reporting Function [ARF]

(5) Event Report Function [ERF]

(6) Log Control Function [LCF]

(7) Security Alarm Reporting Function [SARF]

These agreements currently address System Management Functions 1 through 5. Although it is anticipated that these agreements will be expanded in the future, all other system management functions (6 through n) are currently beyond the scope of these agreements.

18.5.1.2 Conventions Used In SMF Agreements

Each System Management Function defines a set of services referred to in this document as "SMF services". Agreements pertinent to SMF services are provided in the following subclauses. Each subclause contains a series of tables, as follows.

1. For each System Management Function, a table lists the SMF services encompassed by the function, whether each SMF service is currently within the scope of these agreements, and related management support objects (if any). Although a management support object may be **related** to a SMF service, it may or may not be **required** to provide the SMF service.

2. For each SMF service, an informative table (extracted from base standards) defines the mapping between System Management Function and CMIS service parameters. These tables are explanatory and do not contain agreements.

3. For each SMF service, a normative table references text agreements which constrain the usage and/or value of the associated service parameters. Text agreements defined elsewhere in this document are referenced by clause number. The lack of a reference signifies no agreement beyond the base standard.

These tables include codes which specify parameter usage for request, indication, response, and confirmation service primitives. These codes, defined in subclause 1.8.3 of these agreements (Classification of Conformance), ISO/IEC TR 10000-1 (Framework and Taxonomy of ISPs), and ISO/ETC TR-8509 (Service Conventions), are repeated here for reader convenience:

M	Mandatory
O	Optional
C(p)	If Condition p exists, then parameter is mandatory; otherwise, the parameter is not applicable.
X	Excluded
I	Out Of Scope
-	Not Applicable
(=)	The value of the parameter is identical to the corresponding parameter in the interaction described by the preceding related service primitive.
U	The use of the parameter is a service-user option.

In addition, the convention "A>B" is used in normative tables to indicate both the usage specified by the base standard (A) and the additional constraint imposed by these agreements (B). This convention is intended to call attention to agreements which modify the usage of a service parameter.

Unless otherwise noted, conditional parameters (C) shall be present according to the conditions defined in [CMIS] and the referenced System Management Function base standard.

18.5.1.3 General Agreements Referenced By Many SMF Services

The following general agreements pertain to some or all of the System Management Function services defined throughout clause 18.5. Normative tables for each SMF service reference these general agreements where applicable. These agreements do not apply to SMF services and parameters which do not reference them.

18.5.1.3.1 Use of Scoped M-DELETE with Multiple Object Selection

In order to avoid unanticipated side effects, the CMIS M-DELETE service shall be used with the Scope parameter set to 'base object only' (i.e., this operation may be used only to delete a single managed object instance). Of course, it is a straightforward programming exercise to delete multiple objects, and the intent is to avoid unintentional deletion of large numbers of objects. However, use of any other Scope parameter value in the CMIS M-DELETE service is permitted beyond the scope of these agreements. If scoped M-DELETE is not supported by a performing implementation, an attempt to delete more than one object via a single operation shall fail, and the error 'Invalid Scope' shall be returned.

18.5.1.3.2 Minimal Filter Complexity

If an implementation supports Multiple Object Selection, then it shall minimally support AND and OR with a set of two filter conditions (which shall not themselves be AND or OR), and NOT. In addition, the implementation shall support the filter conditions Equality, GreaterOrEqual, LessOrEqual, and Present. This means that a conforming implementation is not required to support compounds (AND or OR) with more than two items, and is not required to support the Substring filter condition. Additional filter items and conditions are beyond the scope of these agreements.

18.5.1.3.3 Mode Parameter Usage

All SMF Services mapped to CMIS M-EVENT-REPORT, M-ACTION, and M-SET require support of both confirmed and unconfirmed Modes. The user of the SMF Service may elect to use either Mode, based on guidance supplied by the managed object class definition.

18.5.1.3.4 Managed Object Class Response Parameter Usage

The Managed Object Class SMF service parameter shall be supplied on all non-empty responses, even those that reference the base managed object. That is, this parameter is defined to be conditional (C(e)): not applicable if the response is empty; mandatory otherwise. An empty response is defined as an RO-RESULT that does not contain a Result parameter.

18.5.1.3.5 Managed Object Instance Response Parameter Usage

The Managed Object Instance SMF service parameter shall be supplied on all non-empty responses, even those that reference the base managed object. That is, this parameter is defined to be conditional (C(e)): not applicable if the response is empty; mandatory otherwise.

18.5.2 Object Management Function Agreements

18.5.2.1 Introduction

This subclause provides agreements pertinent to the Object Management Function defined by [OMF].

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 an abstraction of a data processing resource or a data communications resource that can be remotely managed through the use of OSI management communication services. 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.

Unless a System Management Function describes a specific SMF service, all other managed object operations and notifications shall map onto the pass-through SMF services defined by [OMF]. These pass-through SMF services map directly onto the corresponding CMIS services.

The Object Management Function does not rely upon any specific management support objects or attributes. The Object Management Function makes use of the following notification types defined in [DMI]:

objectCreation,
objectDeletion,
objectNameChange, and
attributeValueChange.

18.5.2.2 General Agreements

These agreements address the following SMF services defined by the object management standard [OMF]:

Table 2: Scope of Agreements Relating to SMF Services Defined by the Object Management Standard [OMF]

Object Management SMF Service	Within Scope Of Agreements	Related Management Support Objects
Object Creation Reporting	Yes	Event Forwarding Discriminator
Object Delection Reporting	Yes	Event Forwarding Discriminator
Object Name Change Reporting	No	Event Forwarding Discriminator

Object Management SMF Service	Within Scope Of Agreements	Related Management Support Objects
Attribute Value Change Reporting	Yes	Event Forwarding Discriminator
PT-Create	Yes	
PT-Delete	Yes	
PT-Action	Yes	
PT-Set	Yes	
PT-Get	Yes	
PT-Event	Yes	Event Forwarding Discriminator

18.5.2.3 Object Creation Reporting

18.5.2.3.1 Introduction

The Object Creation Reporting SMF service is used by the managed system to report creation of a new managed object instance to a managing system.

The following informative table defines the mapping between System Management Function Object Creation Reporting and CMIS M-EVENT-REPORT service parameters. This tutorial information has been extracted from sections 9.1.1 and 11.2.1 of [OMF] and section 8.2.1 of [CMIS].

Table 3: **Mapping Between System Management Function Object Creation Reporting and CMIS M-EVENT-REPORT Service Parameters**

SMF Object Creation Reporting Parameter	Req	Rsp	CMIS M-EVENT-REPORT Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Mode	M	-			
Managed Object Class	M	U			
Managed Object Instance	M	U			
Object Creation	M	C(=)	Event Type		
Event Time	U	-			

SMF Object Creation Reporting Parameter	Req	Rsp	CMIS M-EVENT-REPORT Parameter	Req	Rsp
Create Information					
Source Indicator	U	-	Event Information		
Additional Create Information	U	-	Event Information		
Current Time	-	U			
Event Reply	-	C			
Errors	-	C			

18.5.2.3.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the Object Creation Reporting SMF service defined by section 9.1.1 of [OMF]. Relevant CMIS agreements defined in subclause 18.6.3.1 are repeated here for completeness.

Table 4: Agreements On Parameter Usage Pertinent to the Object Creation Reporting SMF Service

SMF Object Creation Reporting Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		18.6.4
Mode	M	-	18.5.1.3.3	
Managed Object Class	M	U>C(e)	18.5.1.3.4	18.6.2.6
Managed Object Instance	M	U>C(e)	18.5.1.3.5	18.6.2.1
Object Creation	M	C(=)		18.6.2.6
Event Time	U	-		18.6.2.3
Create Information				
Source Indicator	U	-		18.6.3.1.1
Additional Create Information	U	-		18.6.3.1.1
Current Time	-	U		18.6.2.3
Event Reply	-	C		
Errors	-	C		18.6.4.4

18.5.2.4 Object Deletion Reporting

18.5.2.4.1 Introduction

The Object Deletion Reporting SMF service is used by the managed system to report the deletion of a managed object instance to a managing system.

The following informative table defines the mapping between System Management Function Object Deletion Reporting and CMIS M-EVENT-REPORT service parameters. This tutorial information has been extracted from sections 9.1.2 and 11.2.2 of [OMF] and section 8.2.1 of [CMIS].

Table 5: Mapping Between System Management Function Object Deletion Reporting and CMIS M-EVENT-REPORT Service Parameters

SMF Object Deletion Reporting Parameter	Req	Rsp	CMIS M-EVENT-REPORT Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Mode	M	-			
Managed Object Class	M	U			
Managed Object Instance	M	U			
Object Deletion	M	C(=)	Event Type		
Event Time	U	-			
Delete Information					
Source Indicator	U	-	Event Information		
Additional Delete Information	U	-	Event Information		
Current Time	-	U			
Event Reply	-	C			
Errors	-	C			

18.5.2.4.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the Object Deletion Reporting SMF service defined by section 9.1.2 of [OMF]. Relevant CMIS agreements defined in subclause 18.6.3.1 are repeated here for completeness.

Table 6: Agreements On Parameter Usage Pertinent to the Object Deletion Reporting SMF Service

SMF Object Deletion Reporting Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		18.6.4
Mode	M	-	18.5.1.3.3	
Managed Object Class	M	U>C(e)	18.5.1.3.4	18.6.2.6
Managed Object Instance	M	U>C(e)	18.5.1.3.5	18.6.2.1
Object Deletion	M	C(=)		18.6.2.6
Event Time	U	-		18.6.2.3
Delete Information				
Source Indicator	U	-		18.6.3.1.1
Additional Delete Information	U	-		18.6.3.1.1
Current Time	-	U		18.6.2.3
Event Reply	-	C		
Errors	-	C		18.6.4.4

18.5.2.5 Object Name Change Reporting

The Object Name Change Reporting SMF service is used by the managed system to report the renaming of a managed object instance to a managing system.

Use of the Object Name Change Reporting SMF service is beyond the scope of these agreements.

18.5.2.6 Attribute Value Change Reporting

18.5.2.6.1 Introduction

The Attribute Value Change Reporting SMF service is used by the managed system to report an attribute value change event to the managing system. The attribute value change event indicates a change in the value(s) of one or more attributes of a managed object instance.

The following informative table defines the mapping between System Management Function Attribute Value Change Reporting and CMIS M-EVENT-REPORT service parameters. This tutorial information has been extracted from sections 9.1.4 and 11.2.4 of [OMF] and section 8.2.1 of [CMIS].

Table 7: Mapping Between System Management Function Attribute Value Change Reporting and CMIS M-EVENT-REPORT Service Parameters

SMF Attr Value Change Report Parameter	Req	Rsp	CMIS M-EVENT-REPORT Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Mode	M	-			
Managed Object Class	M	U			
Managed Object Instance	M	U			
Attribute Value Change	M	C(=)	Event Type		
Event Time	U	-			
Attribute Change Information					
Attribute Change Definition					
Attribute ID	M	-	Event Information	U	
Old Attribute Value	U	-	Event Information		
New Attribute Value	M	-	Event Information	U	
Source Indicator	U	-	Event Information		
Additional Change Information	U	-	Event Information		
Current Time	-	U			
Event Reply	-	C			
Errors	-	C			

18.5.2.6.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the Attribute Value Change Reporting SMF service defined by section 9.1.4 of [OMF]. Relevant CMIS agreements defined in subclause 18.6.3.1 are repeated here for completeness.

Table 8: **Agreements On Parameter Usage Pertinent to the Attribute Value Change Reporting SMF Service**

SMF Attr Value Change Report Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		18.6.4
Mode	M	-	18.5.1.3.3	
Managed Object Class	M	U>C(e)	18.5.1.3.4	18.6.2.6
Managed Object Instance	M	U>C(e)	18.5.1.3.5	18.6.2.1
Attribute Value Change	M	C(=)		18.6.2.6
Event Time	U	-		18.6.2.3
Attribute Change Information				
Attribute Change Definition				
Attribute ID	M	-		18.6.3.1.1
Old Attr Value	U>M	-	[1]	18.6.3.1.1
New Attribute Value	M	-		18.6.3.1.1
Source Indicator	U	-		18.6.3.1.1
Additional Change Information	U	-		18.6.3.1.1
Current Time	-	U		18.6.2.3
Event Reply	-	C		
Errors	-	C		18.6.4.4

[1] Old Attribute Value shall be included in ALL requests.

18.5.2.7 PT-Create

18.5.2.7.1 Introduction

The PT-Create SMF service is used by a managing system to ask a managed system to create an instance of a managed object in the managed system.

The following informative table defines the mapping between System Management Function PT-Create and CMIS M-CREATE service parameters. This tutorial information has been extracted from section 9.1.5 of [OMF] and section 8.3.4 of [CMIS].

Table 9: Mapping Between System Management Function PT-Create and CMIS M-CREATE Service Parameters

SMF PT-Create Parameter	Req	Rsp	CMIS M-CREATE Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Managed Object Class	M	C			
Managed Object Instance	U	C			
Support Object Instance	U	-			
Access Control	U	-			
Reference Object Instance	U	-			
Attribute List	U	C			
Current Time	-	U			
Errors	-	C			

18.5.2.7.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the PT-Create SMF service defined by section 9.1.5 of [OMF]. Relevant CMIS agreements defined in subclause 18.6.3.5 are repeated here for completeness.

Table 10: Agreements On Parameter Usage Pertinent to the PT-Create SMF Service

SMF PT-Create Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		18.6.4
Managed Object Class	M	C(e)	18.5.1.3.4	18.6.2.6
Managed Object Instance	U	C(e)	18.5.1.3.5	18.6.2.1, 18.6.3.5.1
Support Object Instance	U	-		18.6.2.1
Access Control	U	-		18.6.2.4
Reference Object Instance	U	-		18.6.2.1

SMF PT-Create Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Attribute List	U	C	[1]	18.6.2.6, 18.6.3.5.2
Current Time	-	U		18.6.2.3
Errors	-	C		18.6.4.4, 18.6.3.5.2

[1] This parameter shall be included in ALL success confirmations.

18.5.2.8 PT-Delete

18.5.2.8.1 Introduction

The PT-Delete SMF service 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 informative table defines the mapping between System Management Function PT-Delete and CMIS M-DELETE service parameters. This tutorial information has been extracted from section 9.1.6 of [OMF] and section 8.3.5 of [CMIS].

Table 11: Mapping Between System Management Function PT-Delete and CMIS M-DELETE Service Parameters

SMF PT-Delete Parameter	Req	Rsp	CMIS M-DELETE Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Linked Id	-	C			
Base Object Class	M	-			
Base Object Instance	M	-			
Scope	U	-			
Filter	U	-			
Access Control	U	-			
Synchronization	U	-			

SMF PT-Delete Parameter	Req	Rsp	CMIS M-DELETE Parameter	Req	Rsp
Managed Object Class	-	C			
Managed Object Instance	-	C			
Current Time	-	U			
Errors	-	C			

18.5.2.8.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the PT-Delete SMF service defined by section 9.1.6 of [OMF]. Relevant CMIS agreements defined in subclause 18.6.3.6 are repeated here for completeness.

Table 12: Agreements On Parameter Usage Pertinent to the PT-Delete SMF Service

SMF PT-Delete Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		18.6.4
Linked Id	-	C		18.6.4
Base Object Class	M	-		18.6.2.6
Base Object Instance	M	-		18.6.2.1
Scope	U	-	18.5.1.3.1	18.6.2.2.1
Filter	U	-	18.5.1.3.2	18.6.2.2.2
Access Control	U	-		18.6.2.4
Synchronization	U	-		18.6.2.2.3
Managed Object Class	-	C(e)	18.5.1.3.4	18.6.2.6
Managed Object Instance	-	C(e)	18.5.1.3.5	18.6.2.1
Current Time	-	U		18.6.2.3
Errors	-	C	18.5.1.3.1	18.6.3.6.1, 18.6.4.4

18.5.2.9 PT-Set

18.5.2.9.1 Introduction

The PT-Set SMF service is used by a managing system to ask a managed system to modify the values of one or more specified attributes for a managed object instance in the managed system.

The following informative table defines the mapping between System Management Function PT-Set and CMIS M-SET service parameters. This tutorial information has been extracted from section 9.1.8 of [OMF] and section 8.3.2 of [CMIS].

Table 13: Mapping Between System Management Function PT-Set and CMIS M-SET Service Parameters

SMF PT-Set Parameter	Req	Rsp	CMIS M-SET Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Linked Id	-	C			
Mode	M	-			
Base Object Class	M	-			
Base Object Instance	M	-			
Scope	U	-			
Filter	U	-			
Access Control	U	-			
Synchronization	U	-			
Managed Object Class	-	C			
Managed Object Instance	-	C			
Modification List	M	-			
Attribute List	-	U			
Current Time	-	U			
Errors	-	C			

18.5.2.9.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the PT-Set SMF service defined by section 9.1.8 of [OMF]. Relevant CMIS agreements defined in subclause 18.6.3.3 are repeated here for completeness.

Table 14: Agreements On Parameter Usage Pertinent to the PT-Set SMF Service

SMF PT-Set Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		18.6.4
Linked Id	-	C		18.6.4
Mode	M	-	18.5.1.3.3	
Base Object Class	M	-		18.6.2.6
Base Object Instance	M	-		18.6.2.1
Scope	U	-		18.6.2.2.1
Filter	U	-	18.5.1.3.2	18.6.2.2.2
Access Control	U	-		18.6.2.4
Synchronization	U	-		18.6.2.2.3
Managed Object Class	-	C(e)	18.5.1.3.4	18.6.2.6
Managed Object Instance	-	C(e)	18.5.1.3.5	18.6.2.1
Modification List	M	-		18.6.2.6, 18.6.3.3.1, 18.6.3.3.3, 18.6.3.3.4
Attribute List	-	U		18.6.2.6, 18.6.3.3.1, 18.6.3.3.3
Current Time	-	U		18.6.2.3
Errors	-	C		18.6.3.3.2, 18.6.4.4

18.5.2.10 PT-Action

18.5.2.10.1 Introduction

The PT-Action SMF service is used by a managing system to ask a managed system to perform an action on an instance of a managed object in the managed system.

The following informative table defines the mapping between System Management Function PT-Action and CMIS M-ACTION service parameters. This tutorial information has been extracted from section 9.1.7 of [OMF] and section 8.3.3 of [CMIS].

Table 15: Mapping Between System Management Function PT-Action and CMIS M-ACTION Service Parameters

SMF PT-Action Parameter	Req	Rsp	CMIS M-ACTION Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Linked Id	-	C			
Mode	M	-			
Base Object Class	M	-			
Base Object Instance	M	-			
Scope	U	-			
Filter	U	-			
Access Control	U	-			
Synchronization	U	-			
Managed Object Class	-	C			
Managed Object Instance	-	C			
Action Type	M	C(=)			
Action Information	U	-			
Current Time	-	U			
Action Reply	-	C			
Errors	-	C			

18.5.2.10.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the PT-Action SMF service defined by section 9.1.7 of [OMF]. Relevant CMIS agreements defined in subclause 18.6.3.4 are repeated here for completeness.

Table 16: Agreements On Parameter Usage Pertinent to the PT-Action SMF Service

SMF PT-Action Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		18.6.4
Linked Id	-	C		18.6.4
Mode	M	-	18.5.1.3.3	
Base Object Class	M	-		18.6.2.6
Base Object Instance	M	-		18.6.2.1
Scope	U	-		18.6.2.2.1
Filter	U	-	18.5.1.3.2	18.6.2.2.2
Access Control	U	-		18.6.2.4
Synchronization	U	-		18.6.2.2.3
Managed Object Class	-	C(e)	18.5.1.3.4	18.6.2.6
Managed Object Instance	-	C(e)	18.5.1.3.5	18.6.2.1
Action Type	M	C(=)		18.6.2.6
Action Information	U	-		
Current Time	-	U		18.6.2.3
Action Reply	-	C		
Errors	-	C		18.6.4.4

18.5.2.11 PT-Get

18.5.2.11.1 Introduction

The PT-Get SMF service 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 informative table defines the mapping between System Management Function PT-Get and CMIS M-GET service parameters. This tutorial information has been extracted from section 9.1.9 of [OMF] and section 8.3.1 of [CMIS].

Table 17: Mapping Between System Management Function PT-Get and CMIS M-GET Service Parameters

SMF PT-Get Parameter	Req	Rsp	CMIS M-GET Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Linked Id	-	C			
Base Object Class	M	-			
Base Object Instance	M	-			
Scope	U	-			
Filter	U	-			
Access Control	U	-			
Synchronization	U	-			
Attribute ID List	U	-			
Managed Object Class	-	C			
Managed Object Instance	-	C			
Current Time	-	U			
Attribute List	-	C			
Errors	-	C			

18.5.2.11.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the PT-Get SMF service defined by section 9.1.9 of [OMF]. Relevant CMIS agreements defined in subclause 18.6.3.2 are repeated here for completeness.

Table 18: Agreements On Parameter Usage Pertinent to the PT-Get SMF Service

SMF PT-Action Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		18.6.4
Linked Id	-	C		18.6.6
Base Object Class	M	-		18.6.2.6
Base Object Instance	M	-		18.6.2.1
Scope	U	-		18.6.2.2.1
Filter	U	-	18.5.1.3.2	18.6.2.2.2
Access Control	U	-		18.6.2.4
Synchronization	U	-		18.6.2.2.3
Attribute ID List	U	-		18.6.2.6
Managed Object Class	-	C(e)	18.5.1.3.4	18.6.2.6
Managed Object Instance	-	C(e)	18.5.1.3.5	18.6.2.1
Current Time	-	U		18.6.2.3
Attribute List	-	C		18.6.2.6, 18.6.3.2.1, 18.6.3.2.3
Errors	-	C		18.6.3.2.2, 18.6.4.4

18.5.2.12 PT-Event

18.5.2.12.1 Introduction

The PT-Event SMF service is used by a managed system to report an event to a managing system.

The following informative table defines the mapping between System Management Function PT-Event and CMIS M-EVENT-REPORT service parameters. This tutorial information has been extracted from section 9.1.10 of [OMF] and section 8.2.1 of [CMIS].

Table 19: Mapping Between System Management Function PT-Event and CMIS M-EVENT-REPORT Service Parameters

SMF PT-Event Parameter	Req	Rsp	CMIS M-EVENT-REPORT Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Mode	M	-			
Managed Object Class	M	U			
Managed Object Instance	M	U			
Event Type	M	C(=)			
Event Time	U	-			
Event Information	U	-			
Current Time	-	U			
Event Reply	-	C			
Errors	-	C			

18.5.2.12.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the PT-Event SMF service defined by section 9.1.10 of [OMF]. Relevant CMIS agreements defined in subclause 18.6.3.1 are repeated here for completeness.

Table 20: Agreements On Parameter Usage Pertinent to the PT-Event SMF Service

SMF PT-Action Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		18.6.4
Mode	M	-	18.5.1.3.3	
Managed Object Class	M	U>C(e)	18.5.1.3.4	18.6.2.6
Managed Object Instance	M	U>C(e)	18.5.1.3.5	18.6.2.1
Event Type	M	C(=)		18.6.2.6
Event Time	U	-		18.6.2.3
Event Information	U	-		18.6.3.1.1

SMF PT-Action Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Current Time	-	U		18.6.2.3
Event Reply	-	C		
Errors	-	C		18.6.4.4

18.5.3 State Management Function Agreements

18.5.3.1 Introduction

This subclause provides agreements pertinent to the State Management Function defined by [SMF].

The State Management Function provides for the notification of changes in the management state of existing managed object instances. The management state of a managed object represents its instantaneous condition of availability and operability from the point of view of management. The management state of a managed object is an attribute group comprising three attributes:

- operational state describes the operability of a resource;

- usage state describes whether or not a resource is actively in use and, if so, whether or not it has spare capacity; and

- administrative state describes permission to use or prohibition against using a resource, imposed through management services.

A list of the possible combinations of the operational, usage, and administrative states is given in section 8.2.4 of [SMF].

In addition to the management state group attribute, the status attributes of a managed object contain more detailed information about other aspects of the status of the corresponding resource that may affect its operability. There are a number of status attributes defined in section 8.2.5 of [SMF].

Managed objects can have other class-specific attributes describing their state of operability, availability, mode of operation, conditions of operation, or the current state of a protocol machine, or the status of a test in progress. These class-specific attributes are separate from the management state and status attributes.

The State Management Function is dependent upon the Object Management Function for reading and altering state attributes.

The State Management Function does not rely upon any specific management support objects. The State Management Function makes use of the following attributes defined in [DMI]:

operationalState;
usageState;

administrativeState;
managementState;
repairStatus;
installationStatus;
availabilityStatus;
controlStatus;
state; and
stateGroup.

The State Management Function makes use of the following notification type defined in [DMI]: stateChange.

Editor's Note: [What value does a status attributes have when the resource is "ok"? (Since all four status attributes are set-valued, the answer is probably an empty set.)]

18.5.3.2 General Agreements

These agreements address the following SMF services defined by the state management standard [SMF]:

Table 21: Scope of Agreements Relating to SMF Services Defined by the State Management Standard [SMF]

State Management SMF Service	Within Scope Of Agreements	Related Management Support Objects
State Change Reporting	Yes	Event Forwarding Discriminator

18.5.3.3 State Change Reporting

18.5.3.3.1 Introduction

The State Change Reporting SMF service enables the managed system to convey changes in the value(s) of state attributes of a managed object instance to a managing system.

The following informative table defines the mapping between System Management Function State Change Reporting and CMIS M-EVENT-REPORT service parameters. This tutorial information has been extracted from sections 9.1.1 and 11.2 of [SMF] and section 8.2.1 of [CMIS].

Table 22: Mapping Between System Management Function State Change Reporting and CMIS M-EVENT-REPORT Service Parameters

SMF State Change Report Parameter	Req	Rsp	CMIS M-EVENT-REPORT Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Mode	M	-			
Managed Object Class	M	U			
Managed Object Instance	M	U			
State Change	M	U	Event Type		C(=)
Event Time	U	-			
State Change Information					
State Change Definition					
State Attribute ID	M	-	Event Information	U	
Old State Value	U	-	Event Information		
New State Value	M	-	Event Information	U	
Additional State Change Info	U	-	Event Information		
Current Time	-	U			
Event Reply	-	C			
Errors	-	C			

18.5.3.3.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the State Change Reporting SMF service defined by section 9.3 of [SMF]. Relevant CMIS agreements defined in subclause 18.6.3.1 are repeated here for completeness.

Table 23: Agreements On Parameter Usage Pertinent to the State Change Reporting SMF Service

SMF State Change Report Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		18.6.4

SMF State Change Report Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Mode	M	-	18.5.1.3.3	
Managed Object Class	M	U>C(e)	18.5.1.3.4	18.6.2.6
Managed Object Instance	M	U>C(e)	18.5.1.3.5	18.6.2.1
State Change	M	U		18.6.2.6
Event Time	U	-		18.6.2.3
State Change Information				
State Change Definition				
State Attribute ID	M	-		18.6.3.1.1
Old State Value	U>M	-	[1]	18.6.3.1.1
New State Value	M	-		18.6.3.1.1
Additional State Change Info	U	-		18.6.3.1.1
Current Time	-	U		18.6.2.3
Event Reply	-	C		
Errors	-	C		18.6.4.4

[1] Old State Value shall be included in ALL requests.

18.5.4 Attributes For Representing Relationships Agreements

18.5.4.1 Introduction

This subclause provides agreements pertinent to the Attributes For Representing Relationships SMF defined by [ARR].

A relationship is a set of rules that describe how the operation of one part of an open system affects the operation of another part of an open system. The operation of a managed object may affect its related managed object directly or indirectly.

An reciprocal relationship is a binding between two managed objects that is represented by including, as one of a set of values of an attribute of each of the managed objects, the name of the other managed object to which it is related. A relationship attribute is a set-valued attribute of a managed object whose values are the names of other managed objects with which it has reciprocal relationships of a particular kind.

At any given time, within an open systems environment, one managed object may be a part of several different types of relationships. For each type of relationship, depending on the role of the managed object (i.e., the part played by the other managed object), the relationship can be symmetric or asymmetric. For every possible relationship role of a managed object, there exists a corresponding relationship attribute. The name of a relationship attribute of a managed object implies the relationship role of the related managed objects and the type of the reciprocal relationship. The types of relationships are defined in [ARR].

The Attributes For Representing Relationships SMF is dependent upon the Object Management Function for reading and altering relationship attributes.

The Attributes For Representing Relationships SMF does not rely upon any specific management support objects. The Attributes For Representing Relationships SMF makes use of the following attributes defined in [DMI]:

serviceProvider,
serviceUser,
peer,
primary,
secondary,
backUpObject,
backedUpObject,
member,
owner, and
relationshipGroup.

The Attributes For Representing Relationships SMF makes use of the following notification type defined in [DMI]: relationshipChange.

Editor's Note: [The name (and scope) of this SMF was changed during the ISO SC21/WG4 meeting in Seoul. Relationship modelling is now part of a separate NWI. [ARR] now defines only relationship attributes and a notification.]

18.5.4.2 General Agreements

These agreements address the following SMF services defined by the Attributes For Representing Relationships standard [ARR]:

Table 24: Scope of Agreements Relating to SMF Services Defined by the Attributes For Representing Relationships Standard [ARR]

Attributes For Representing Relationships SMF Service	Within Scope Of Agreements	Related Management Support Objects
Relationship Change Reporting	Yes	Event Forwarding Discriminator

18.5.4.3 Relationship Change Reporting

18.5.4.3.1 Introduction

The Relationship Change Reporting SMF service is used to report the change in the value of one or more relationship attributes of a managed object instance.

The following informative table defines the mapping between System Management Function Relationship Change Reporting and CMIS M-EVENT-REPORT service parameters. This tutorial information has been extracted from sections 9.1.1 and 11.2 of [ARR] and section 8.2.1 of [CMIS].

Table 25: Mapping Between System Management Function Relationship Change Reporting and CMIS M-EVENT-REPORT Service Parameters

SMF Relationship Change Report Parameter	Req	Rsp	CMIS M-EVENT-REPORT Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Mode	M	-			
Managed Object Class	M	U			
Managed Object Instance	M	U			
Relationship Change	M	U	Event Type		C(=)
Event Time	U	-			
Relationship Change Information					
Relationship Change Definition					
Relationship Attribute ID	M	-	Event Information	U	
Old Relationship Value	C	-	Event Information	U	

SMF Relationship Change Report Parameter	Req	Rsp	CMIS M-EVENT-REPORT Parameter	Req	Rsp
New Relationship Value	C	-	Event Information	U	
Additional Relationship Change Info	U	-	Event Information		
Current Time	-	U			
Event Reply	-	C			
Errors	-	C			

18.5.4.3.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the Relationship Change Reporting SMF service defined by section 9.3 of [ARR]. Relevant CMIS agreements defined in subclause 18.6.3.1 are repeated here for completeness.

Table 26: Agreements On Parameter Usage Pertinent to the Relationship Change Reporting SMF Service

SMF State Change Report Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		18.6.4
Mode	M	-	18.5.1.3.3	
Managed Object Class	M	U>C(e)	18.5.1.3.4	18.6.2.6
Managed Object Instance	M	U>C(e)	18.5.1.3.5	18.6.2.1
Relationship Change	M	U		18.6.2.6
Event Time	U	-		18.6.2.3
Relationship Change Information				
Relationship Change Definition				
Relationship Attribute ID	M	-		18.6.3.1.1
Old Relationship Value	C	-		18.6.3.1.1
New Relationship Value	C	-		18.6.3.1.1
Additional Relationship Change Info	U	-		18.6.3.1.1
Current Time	-	U		18.6.2.3

SMF State Change Report Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Event Reply	-	C		
Errors	-	C		18.6.4.4

18.5.5 Alarm Reporting Function Agreements

18.5.5.1 Introduction

This subclause provides agreements pertinent to the Alarm Reporting Function defined by [ARF].

The Alarm Reporting Function specifies a set of five generic 'alarm' notifications and their parameters and semantics. These notifications allow the reporting of alarms, errors, and related information in a standard fashion.

Control of notifications (e.g., whether a notification results in an event report) may be accomplished using the Event Report Function [ERF].

The notifications defined by the Alarm Reporting Function can report changes of state as defined by the State Management Function, or instances of the back up relationship as defined by the Attributes For Representing Relationships SMF.

The Alarm Reporting Function makes use of the Alarm Record management support object and the following attributes defined in [DMI]:

probableCause,
specificProblem,
perceivedSeverity,
backUpStatus,
backUpObjectInstance,
trendIndication,
thresholdInfo,
notificationId,
correlatedNotifications,
stateChange,
monitoredAttributes,
proposedRepairAction,
problemText, and
problemData.

The Alarm Reporting Function makes use of the following notification types defined in [DMI]:

communicationAlarm,
qualityOfServiceAlarm,

processingErrorAlarm,
equipmentAlarm, and
environmentalAlarm.

18.5.5.2 General Agreements

These agreements address the following SMF services defined by the alarm reporting standard [ARF]:

Table 27: Scope of Agreements Relating to SMF Services Defined by the Alarm Reporting Standard [ARF]

Alarm Reporting SMF Service	Within Scope Of Agreements	Related Management Support Objects
Alarm Reporting	Yes	Event Forwarding Discriminator & Alarm Record

18.5.5.3 Alarm Reporting

18.5.5.3.1 Introduction

The Alarm Reporting SMF service allows one user to notify another user of an alarm detected in a managed object instance.

The following informative table defines the mapping between System Management Function Alarm Reporting and CMIS M-EVENT-REPORT service parameters. This tutorial information has been extracted from sections 9.3 and 11.2 of [ARF] and section 8.2.1 of [CMIS].

Table 28: Mapping Between System Management Function Alarm Reporting and CMIS M-EVENT-REPORT Service Parameters

SMF Alarm Reporting Parameter	Req	Rsp	CMIS M-EVENT-REPORT Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Mode	M	-			
Managed Object Class	M	U			
Managed Object Instance	M	U			

SMF Alarm Reporting Parameter	Req	Rsp	CMIS M-EVENT-REPORT Parameter	Req	Rsp
Alarm Type	M	C(=)	Event Type		
Event Time	U	-			
Alarm Information					
Probable Cause	M	-	Event Information	U	
Specific Problems	U	-	Event Information		
Perceived Severity	M	-	Event Information	U	
Backup Object Instance	C	-	Event Information	U	
BackedUp Status	U	-	Event Information		
Trend Indication	U	-	Event Information		
Treshold Information	C	-	Event Information	U	
Notification Identifier	U	-	Event Information		
Correlated Notifications	U	-	Event Information		
State Change	C	-	Event Information		
Monitored Attributes	U	-	Event Information		
Proposed Repair Action	U	-	Event Information		
Problem Text	U	-	Event Information		
Problem Data	U	-	Event Information		
Current Time	-	U			
Event Reply	-	C			
Errors	-	C			

18.5.5.3.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the Alarm Reporting SMF service defined by section 9.3 of [ARF]. Relevant CMIS agreements defined in subclause 18.6.3.1 are repeated here for completeness.

Table 29: Agreements On Parameter Usage Pertinent to the Alarm Reporting SMF Service

SMF Alarm Reporting Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M		18.6.4
Mode	M	-	18.5.1.3.3	
Managed Object Class	M	U>C(e)	18.5.1.3.4	18.6.2.6
Managed Object Instance	M	U>C(e)	18.5.1.3.5	18.6.2.1
Alarm Type	M	C(=)		18.6.2.6
Event Time	U	-		18.6.2.3
Alarm Information				
Probable Cause	M	-		18.6.3.1.1
Specific Problems	U	-		18.6.3.1.1
Perceived Severity	M	-		18.6.3.1.1
Backup Object Instance	C	-	[1]	18.6.3.1.1
BackedUp Status	U	-		18.6.2.1
Trend Indication	U	-		18.6.3.1.1
Threshold Information	C	-		18.6.3.1.1
Notification Identifier	U	-		18.6.3.1.1
Correlated Notifications	U	-		18.6.3.1.1
State Change	C	-		18.6.3.1.1
Monitored Attributes	U	-		18.6.3.1.1
Proposed Repair Action	U	-		18.6.3.1.1
Problem Text	U	-		18.6.3.1.1
Problem Data	U	-		18.6.3.1.1
Current Time	-	U		18.6.2.3
Event Reply	-	C		
Errors	-	C		18.6.4.4

[1] To avoid ambiguity, only the Distinguished Name form of this parameter shall be used (i.e., use of Local Distinguished Name and Non-Specific forms are beyond the scope of these agreements).

Editor's Note: [We should consider additional agreements on the maximum values of Alarm Report parameters which are in alignment with the corresponding OSI/NMF Application Service parameters.]

18.5.6 Event Report Management Function Agreements

18.5.6.1 Introduction

This subclause provides agreements pertinent to the Event Report Function defined by [ERF].

The Event Report Management Function provides SMF services by which event reporting can be distributed and controlled. Event report 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 "Discriminator Construct" attribute of the "Event Forwarding Discriminator" object. Event Report Management is the ability to initiate, terminate, suspend, or resume event reporting through the manipulation of an Event Forwarding Discriminator object specified in [DMI]. In addition, Event Report Management can further alter event report distribution behavior by changing the distribution attributes in an Event Forwarding Discriminator object (e.g., Discriminator Construct).

According to the Event Reporting Model defined by [ERF], 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 SMF services defined in the subsequent subclauses. Multiple discriminators may filter the same potential event reports and hence generate multiple event reports.

The Event Report Management Function uses the State Management Function for the notification of discriminator state changes, and the Object Management Function for creating and deleting discriminators, retrieving discriminator attribute values, and notification of discriminator attribute value changes, creation, and deletion.

The Event Report Management Function makes use of the following management support objects defined in [DMI]:

eventForwardingDiscriminator, and
eventReportRecord.

The Event Report Management Function makes use of the following attributes defined in [DMI]:

discriminatorId,
discriminatorConstruct,

destinationAddress,
backupAddress,
activeAddress,
administrativeState,
operationalState,
usageState,
availabilityStatus,
allomorphicList,
packages,
daysOfWeek,
startTime,
stopTime, and
managedObjectClass.

The Event Report Management Function makes use of the following notification types defined in [DMI]:

objectCreation,
objectDeletion,
stateChange, and
attributeValueChange.

18.5.6.2 General Agreements

These agreements address the following SMF services defined by the event report standard [ERF]:

Table 30: Scope of Agreements Relating to SMF Services Defined by the Event Report Standard [ERF]

Event Report SMF Service	Within Scope Of Agreements	Related Management Support Objects
Initiation of ERF	Yes	Event Forwarding Discriminator, Event Report Record
Termination of ERF	Yes	Event Forwarding Discriminator, Event Report Record
EFD Modification, Suspension, Resumption	Yes	Event Forwarding Discriminator, Event Report Record

18.5.6.3 Initiation Of Event Report Forwarding

18.5.6.3.1 Introduction

This SMF service allows one open system to request that another open system create an Event Forwarding Discriminator, thereby requesting that new or additional event forwarding controls be imposed.

The following informative table defines the mapping between ERF Initiation of Event Report Forwarding, OMF PT-Create, and CMIS M-CREATE service parameters. This tutorial information has been extracted from sections 9.2 and 11.2 of [ERF] and section 8.3.4 of [CMIS].

Table 31: Mapping Between ERF Initiation of Event Report Forwarding, OMF PT-Create, and CMIS M-CREATE Service Parameters

SMF Initiation of ERF Parameter	Req	Rsp	OMF PT-Create & CMIS M-CREATE Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Managed Object Class	M	C			
Managed Object Instance	U	C			
Support Object Instance	U	-			
Access Control	U	-			
Reference Object Instance	U	-			
Discriminator Construct	U	C	Attribute List		
Destination Address	U	C	Attribute List		
Backup Address	U	C	Attribute List		
Active Address	U	C	Attribute List		
Administrative State	U	C	Attribute List		
Operational State	-	C	Attribute List		
Usage State	U	C	Attribute List		
Availability Status	-	C	Attribute List		
Allomorphic List	U	C	Attribute List		
Packages	U	C	Attribute List		
Days of Week	U	C	Attribute List		

SMF Initiation of ERF Parameter	Req	Rsp	OMF PT-Create & CMIS M-CREATE Parameter	Req	Rsp
Start Time	U	C	Attribute List		
Stop TIme	U	C	Attribute List		
Current Time	-	U			
Errors	-	C			

Editor's Note: [This table (and all other tables contained in the [ERF] clause) were constructed by the clause editor during alignment with Seoul output. The tables previously contained in [ERF] were deleted from the DIS editor's draft output from Seoul, but no replacement tables were provided. These tables must therefore be reviewed after final DIS text becomes available to ensure alignment. Particular attention should be paid to the list of attributes since DIS editor's draft [ERF] and [DMI] were inconsistent.]

18.5.6.3.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the Initiation of Event Report Forwarding SMF service defined by section 9.2 of [ERF]. Relevant CMIS agreements defined in subclause 18.6.3.5 are repeated here for completeness.

Table 32: Agreements On Parameter Usage Pertinent to the Initiation of Event Report Forwarding SMF Service

SMF Initiation of ERF Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		18.6.4
Managed Object Class	M	C(e)	18.5.1.3.4	18.6.2.6
Managed Object Instance	U	C(e)	18.5.1.3.5	18.6.2.1, 18.6.3.5.1
Support Object Instance	U	-		18.6.2.1
Access Control	U	-		18.6.2.4
Reference Object Instance	U	-		18.6.2.1
Discriminator Construct	U	C	[1], 18.5.1.3.2	18.6.2.6, 18.6.3.5.2

SMF Initiation of ERF Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Destination Address	U	C		18.6.2.6, 18.6.3.5.2
Backup Address	U	C		18.6.2.6, 18.6.3.5.2
Active Address	U	C		18.6.2.6, 18.6.3.5.2
Administrative State	U	C		18.6.2.6, 18.6.3.5.2
Operational State	-	C		18.6.2.6, 18.6.3.5.2
Usage State	U	C		18.6.2.6, 18.6.3.5.2
Availability Status	-	C		18.6.2.6, 18.6.3.5.2
Allomorphic List	U	C		18.6.2.6, 18.6.3.5.2
Packages	U	C		18.6.2.6, 18.6.3.5.2
Days of Week	U	C		18.6.2.6, 18.6.3.5.2
Start Time	U	C		18.6.2.6, 18.6.3.5.2
Stop Time	U	C		18.6.2.6, 18.6.3.5.2
Current Time	-	U		18.6.2.3
Errors	-	C		18.6.4.4, 18.6.3.5.2

[1] As specified in [CMIP], the value "AND {}" shall be used to represent an all-pass Discriminator Construct. If this parameter is omitted from the request, the all-pass value shall be assigned to the Discriminator Construct attribute.

18.5.6.4 Termination Of Event Report Forwarding

18.5.6.4.1 Introduction

This SMF service allows one open system to request that another open system delete one or more Event Forwarding Discriminators, thereby requesting that some event forwarding controls be terminated.

The following informative table defines the mapping between ERF Termination of Event Report Forwarding, OMF PT-Delete, and CMIS M-DELETE service parameters. This tutorial information has been extracted from sections 9.3 and 11.2 of [ERF] and section 8.3.5 of [CMIS].

Table 33: **Mapping Between ERF Termination of Event Report Forwarding, OMF PT-Delete, and CMIS M-DELETE Service Parameters**

SMF Termination of ERF Parameter	Req	Rsp	PT-Delete & CMIS M-DELETE Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Linked Id	-	C			
Base Object Class	M	-			
Base Object Instance	M	-			
Scope	U	-			
Filter	U	-			
Access Control	U	-			
Synchronization	U	-			
Managed Object Class	-	C			
Managed Object Instance	-	C			
Current Time	-	U			
Errors	-	C			

18.5.6.4.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the Termination of Event Report Forwarding SMF service defined by section 9.3 of [ERF]. Relevant CMIS agreements defined in subclause 18.6.3.6 are repeated here for completeness.

Table 34: Agreements On Parameter Usage Pertinent to the Termination of Event Report Forwarding SMF Service

SMF Termination of ERF Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		18.6.4
Linked Id	-	C		18.6.4
Base Object Class	M	-		18.6.2.6
Base Object Instance	M	-		18.6.2.1
Scope	U	-	18.5.1.3.1	18.6.2.2.1
Filter	U	-	18.5.1.3.2	18.6.2.2.2
Access Control	U	-		18.6.2.4
Synchronization	U	-		18.6.2.2.3
Managed Object Class	-	C(e)	18.5.1.3.4	18.6.2.6
Managed Object Instance	-	C(e)	18.5.1.3.5	18.6.2.1
Current Time	-	U		18.6.2.3
Errors	-	C	18.5.1.3.1	18.6.3.6.1, 18.6.4.4

18.5.6.5 EFD Modification, Suspension, and Resumption

18.5.6.5.1 Introduction

This SMF service allows one open system to request that another open system change the Administrative State attribute, or any other settable attribute, of the Event Forwarding Discriminator.

The following informative table defines the mapping between ERF Event Forwarding Discriminator Modification, Suspension, and Resumption, OMF PT-Set, and CMIS M-SET service parameters. This tutorial information has been extracted from sections 9.4 and 11.2 of [ERF] and section 8.3.2 of [CMIS].

Table 35: Mapping Between ERF Event Forwarding Discriminator Modification, Suspension, and Resumption, OMF PT-Set, and CMIS M-SET Service Parameters

SMF EFD Mod/Suspend/Resume Parameter	Req	Rsp	PT-Set & CMIS M-SET Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Linked Id	-	C			
Mode	M	-			
Base Object Class	M	-			
Base Object Instance	M	-			
Scope	U	-			
Filter	U	-			
Access Control	U	-			
Synchronization	U	-			
Managed Object Class	-	C			
Managed Object Instance	-	C			
Discriminator Construct	U	C	Mod & Attribute List	M	
Destination Address	U	C	Mod & Attribute List	M	
Backup Address	U	C	Mod & Attribute List	M	
Active Address	U	C	Mod & Attribute List	M	
Administrative State	U	C	Mod & Attribute List	M	
Usage State	U	C	Mod & Attribute List	M	
Allomorphic List	U	C	Mod & Attribute List	M	
Days of Week	U	C	Mod & Attribute List	M	
Start Time	U	C	Mod & Attribute List	M	
Stop Time	U	C	Mod & Attribute List	M	
Current Time	-	U			
Errors	-	C			

18.5.6.5.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the Event Forwarding Discriminator Modification, Suspension, and Resumption SMF service defined by section 9.4 of [ERF]. Relevant CMIS agreements defined in subclause 18.6.3.3 are repeated here for completeness.

Table 36: Agreements On Parameter Usage Pertinent to the Event Forwarding Discriminator Modification, Suspension, and Resumption SMF Service

SMF EFD Mod/Suspend/Resume Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		18.6.4
Linked Id	-	C		18.6.4
Mode	M	-	18.5.1.3.3	
Base Object Class	M	-		18.6.2.6
Base Object Instance	M	-		18.6.2.1
Scope	U	-		18.6.2.2.1
Filter	U	-	18.5.1.3.2	18.6.2.2.2
Access Control	U	-		18.6.2.4
Synchronization	U	-		18.6.2.2.3
Managed Object Class	-	C(e)	18.5.1.3.4	18.6.2.6
Managed Object Instance	-	C(e)	18.5.1.3.5	18.6.2.1
Discriminator Construct	U	C	[1], 18.5.1.3.2	18.6.2.6, 18.6.3.3.1, 18.6.3.3.3, 18.6.3.3.4
Destination Address	U	C		18.6.2.6, 18.6.3.3.1, 18.6.3.3.3, 18.6.3.3.4
Backup Address	U	C		18.6.2.6, 18.6.3.3.1, 18.6.3.3.3, 18.6.3.3.4

SMF EFD Mod/Suspend/Resume Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Active Address	U	C		18.6.2.6, 18.6.3.3.1, 18.6.3.3.3, 18.6.3.3.4
Administrative State	U	C		18.6.2.6, 18.6.3.3.1, 18.6.3.3.3, 18.6.3.3.4
Usage State	U	C		18.6.2.6, 18.6.3.3.1, 18.6.3.3.3, 18.6.3.3.4
Allomorphic List	U	C		18.6.2.6, 18.6.3.3.1, 18.6.3.3.3, 18.6.3.3.4
Days of Week	U	C		18.6.2.6, 18.6.3.3.1, 18.6.3.3.3, 18.6.3.3.4
Start Time	U	C		18.6.2.6, 18.6.3.3.1, 18.6.3.3.3, 18.6.3.3.4
Stop Time	U	C		18.6.2.6, 18.6.3.3.1, 18.6.3.3.3, 18.6.3.3.4
Current Time	-	U		18.6.2.3
Errors	-	C		18.6.3.3.2, 18.6.4.4

[1] As specified in [CMIP], the value "AND {}" shall be used to represent an all-pass Discriminator Construct.

18.6 MANAGEMENT COMMUNICATIONS

This clause identifies, in detail, use of the management communications services and protocols, based on the standards defined in [CMIS], [CMIP], [ADDRMVS/P] and [CANGETS/P].

This clause covers the agreements pertaining to the use of associations over which to carry management PDUs, agreements pertaining to the services offered to a CMIS Service User (in terms of the functions defined in clause 18.5), agreements pertaining to the protocol used to convey the management PDUs, and agreements pertaining to the services required of other layers in order to convey the management PDUs defined.

18.6.1 Association Policies

Tutorial Note: [This draft of the Association Policy subclause of the Phase 1 IAs represents the output from an interim NMSIG meeting held in Peabody MA in November 1989. The purpose of the meeting was to align the draft subclause from the July Workshop with output from the Florence meetings and with the issues from the NMSIG Issue Log. As a result of review by the December 1989 OIW NMSIG Meeting, some additional changes were made to this text.

The participants at the interim meeting summarized the issues into 8 items. These are listed here to enable reviewers to understand the premise for the subsequent text.

Issue 1: Should there be agreements about arbitration among competing requests where agents allow multiple associations to managing systems?

It was decided that this was really a matter for an agent implementation. If an agent does some form of arbitration (e.g., temporarily lock out a request to modify an object while a prior request is being honored), it must indicate this in some agreed upon way so that the managing system can distinguish between this situation and some other error, such as access denied or no such object.

This issue is not related to association types or to access control. The recommendation was placed in an appropriate subclause of the CMIS/P agreements in 18.6.3.

Issue 2: What is the retry policy, if any?

It was proposed that we make some suggestions and have them reviewed by the Workshop. See subclause 18.6.1.4.

Issue 3: What are the connect and disconnect policies, if any?

See subclause 18.6.1.4.

Issue 4: How are the roles of managing and managed system determined?

It was felt that it was necessary to determine which role a system is playing on an association and that the Application Context Name work in the Arhus output for SMO fit the bill. See subclause 18.6.1.2.

Issue 5: Handling of events vs command/control.

Issue 6: Handling of monitoring vs control.

Re Issues 5 and 6, it was felt that managed and managing systems may wish to restrict the types of functions that may be performed on a particular association. The proposal for addressing this issue is in subclause 18.6.1.2.

Issue 7: Views of a MIB on an association.

It was decided to keep the output from the July workshop which statesthat we make no agreements regarding the scope of an association as it applies to the objects made accessible over that association. The arbitration process adds a slight wrinkle though. See subclause 18.6.1.4.

Issue 8: Are we making recommendations or requirements?

This draft has both. It was never really decided if recommendations are appropriate in these agreements. If they aren't then we will have to decide whether to drop the recommendations in this draft or make them requirements.]

Associations are established using the procedures described in [ACSEP] with recommendations and extensions described in these implementation agreements.

Phase 1 IAs specify the different types of associations that may be established between managing and managed systems (see 18.6.1.2). The type of a given association is determined by the exchange of appropriate application context information between the systems using a negotiation process.

Phase 1 IAs recommend that managed systems reserve resources for at least one association for event reporting (see 18.6.1.3).

Phase 1 IAs require the use of A-RELEASE instead of A-ABORT. Phase 1 IAs also make recommendations regarding parameters affecting the scope of managed objects and span of time for an association and synchronization among multiple associations (see 18.6.1.4).

Phase 1 IAs specify the access control information to be used in the establishment of an association (see 18.6.1.5).

18.6.1.1 ACSE Services

The A-ASSOCIATE and A-RELEASE ACSE services are used as specified in [ACSE]. The Phase 1 IAs make certain requirements as to the use of the APDU fields noted below. Usage of all other fields is left to the implementor.

AE-TITLE (Calling AP Title and Calling AE Qualifier) usage is specified in 18.6.1.2.

Application Context Name usage is specified in 18.6.1.2.

ACSE User Information consists of three parameters (specified in [CMIS]): Functional Units, Access Control and CMIS User Information. Refer to subclause 18.6.3 for agreements relating to Functional Units. Refer to subclause 18.6.1.5 for agreements relating to Access Control. The Phase 1 IAs make no agreements relating to CMIS User Information.

18.6.1.2 Association Types

The Phase 1 IAs specify that four types of association may be negotiated between managing and managed systems. These types are:

Event	M-EVENT-REPORTs may be sent by the managed system; no other CMIP PDUs are allowed
Event/Monitor	same as Event type except that, in addition, the managing system may also issue M-GET requests and receive M-GET responses over the association
Monitor/Control	managing system may issue M-GET, M-SET, M-CREATE, M-DELETE and M-ACTION requests over the association; no event reporting is allowed
Full Mgr/Agent	all functions must be supported

The negotiation process specified for the Phase 1 IAs uses the A-ASSOCIATE and A-RELEASE services as specified in [ACSEP]. Application Context Name [SMO] is used to determine the requestor's "role" in an association (managing or managed system) and to determine the type of the association. The following negotiation rules are specified by the Phase 1 IAs:

Editor's Note: [The SIG left open the question of using Application Context Names for both role and type determination. The editor investigated further to find out if there were any restrictions that would prevent such usage. Having found no restrictions, the editor updated the text to provide more detail in this direction.]

Editor's Note: [We need to assign Application Context Names. I suggest that we register appropriate object names under the NMSIG arc. I'll take a stab at the proper format (see RASIG output...clause 6 of the Working Document) and propose some names as a placeholder until we determine the actual format/names. (Wordsmithing and format advice are welcome.)

{iso(1) identified-organization(3) oiw(14) nmsig(2) manager-event-association(x)}

{iso(1) identified-organization(3) oiw(14) nmsig(2) manager-event-monitor-association(x)}

{iso(1)identified-organization(3)oiw(14)nmsig(2)manager-monitor-control-association(x)} {iso(1) identified-organization(3) oiw(14) nmsig(2) manager-full-association(x)}

{iso(1) identified-organization(3) oiw(14) nmsig(2) agent-event-association(x)}]

Tutorial Note: **[Ref: [SMO] Annex A**

The Application Context Name (ACN) indicates the role of the initiator of an association. The responder may alter the type indication to request a change in the type. Note that the proposed ACNs above follow the agreements on which system may request a particular type of association. Thus there is a single agent initiated ACN since agents (managed systems) may only initiate event reporting associations.

The ACNs in these agreements refine those defined in Annex A [SMO] and are used in the same fashion.]

Editor's Note: [We will need to add text relating to negotiation of System Management Function functional units as changes to this subclause as the relevant standards (10164-*) are updated. It is anticipated that the work in N740 will be used as the basis.]

1. A managed system may only request an Event association and, in fact, must create an Event association if it has an event to report and no suitable association already exists.

2. Managing systems may request any association type.

3. An association is created by the requesting system issuing an A-ASSOCIATE request with the requestor's AE-TITLE and the desired application context. The responding system then returns either 1) an A-ASSOCIATE response with the requestor's AE-TITLE and the application context which it wishes to accept or 2) an A-ASSOCIATE response rejecting the association.

4. Managed systems may negotiate "downward" from Full to Monitor/Control, Event/Monitor or Event by returning the new application context in the A-ASSOCIATE response to the managing system during the association creation process. In the same fashion, managed systems may negotiate from Event/Monitor to Event.

5. When a managing system receives an application context in an A-ASSOCIATE response that differs from the context sent in an A-ASSOCIATE request it may either proceed with the new context or refuse the new context by issuing an A-RELEASE request.

Editor's Note: [A-RELEASE is used when the requestor does not agree with the new context. A-ABORT is used for invalid negotiation.]

Note that a system may play both managing and managed system roles, but not on the same association.

18.6.1.3 Events

Phase 1 IAs recommend that managed systems make resources available for at least one association for the purposes of event reporting. The resources allocated to an association should be re-useable. That is, if the system must report an event to multiple managers, it may have to repeatedly utilize the resources for an association to each of the managing systems. This recommendation is made to ensure that events are not lost due to a lack of associations.

Editor's Note: [The status of 18.6.1.3 as a recommendation rather than a requirement is open for comments.]

18.6.1.4 Scope/Span of an Association

Editor's Note: [Discussions at the Florence meeting indicate the potential for an "association policy object." This object would allow for the maintenance of parameters pertaining to the behavior of an association. These parameters would include such things as number of retries and inactivity timers. This version of clause 18.6 was written so that if this proposal comes to fruition, the agreements can be migrated to the ap-object by "transferring" the parameters to the object itself.]

The Phase 1 IAs specify no process for negotiating the scope of an association as it pertains to the objects that may be managed within the context of that association.

Editor's Note: [Text in the December 1989 Workshop draft document regarding arbitration between requests from multiple managers was moved from this subclause to the CMIS/P subclause (subclause 18.6.3).]

The Phase 1 IAs specify no process for negotiating a time span of an association. The managing or managed system may terminate an association based upon an implementation specific algorithm governing association durations.

Editor's Note: [Text in the December 1989 Workshop draft document regarding potential parameters for managing time span and retries for associations was removed from this subclause. The text has been retained "off-line" at the direction of the NMSIG.

Underlying services such as ACSE may also cause the termination of an association.]

The Phase 1 IAs require that associations be terminated with A-RELEASE to avoid loss of information in an association.

Tutorial Note: **[If A-ABORT is used to terminate an association, there exists a potential for loss of information such as pending events or confirmations. A-ABORT must be used, however, when a protocol violation occurs or where an association is not yet established.]**

18.6.1.5 Other Aspects of Associations

Editor's Note: [The access control information in this subclause is based on some notes from a joint NMSIG/Security SIG meeting that took place some time ago. We should review this with the Security SIG to make sure we are still in agreement and get more information on usage and encoding. This review is tentatively planned for the March 1990 OIW.]

The Phase 1 IAs specify that the following information may be used in establishing an association. A managed system, if it requires access control information, must use this format.

Unused fields must contain nulls.

Field	Name	Purpose
1	Length	length of access control data
2	Initiating Person	
3	Process Type	
4	Process ID	
5	Authorization	password
6	Access Privileges	
7	Audit Requirements	
8	Integrity Seal	universal closed community checksum; message authentication code
9	Optional Information	0-n bytes of optional data

18.6.2 General Agreements on Users of CMIS

(Refer to the Stable Inplementation Agreements Document.)

18.6.2.1 Object Naming

(Refer to the Stable Implementation Agreements Document.)

18.6.2.2 Multiple Object Selection

(Refer to the Stable Implementation Agreements Document.)

18.6.2.2.1 Scoping

(Refer to the Stable Implementation Agreements Document.)

18.6.2.2.2 Filtering

(Refer to the Stable Implementation Agreements Document.)

18.6.2.2.3 Synchronization

(Refer to the Stable Implementation Agreements Document.)

18.6.2.2.4 Multiple Replies

(Refer to the Stable Implementation Agreements Document.)

18.6.2.3 Current/Event Time

(Refer to the Stable Implementation Agreements Document.)

18.6.2.4 Access Control

(Refer to the Stable Implementation Agreements Document.)

18.6.2.5 CMIS Functional Units

(Refer to the Stable Implementation Agreements Document.)

18.6.2.6 CMIP Parameters

(Refer to the Stable Implementation Agreements Document.)

18.6.3 Specific Agreements on Users of CMIS

(Refer to the Stable Implementation Agreements Document.)

18.6.3.1 M-Event-Report

(Refer to the Stable Implementation Agreements Document.)

18.6.3.1.1 Event Argument

(Refer to the Stable Implementation Agreements Document.)

18.6.3.1.2 Parameter Agreements

(Refer to the Stable Implementation Agreements Document.)

18.6.3.2 M-Get

(Refer to the Stable Implementation Agreements Document.)

18.6.3.2.1 Successful Response

(Refer to the Stable Implementation Agreements Document.)

18.6.3.2.2 Partially Successful Response

(Refer to the Stable Implementation Agreements Document.)

18.6.3.2.3 Multiple Replies

(Refer to the Stable Implementation Agreements Document.)

18.6.3.2.4 Parameter Agreements

(Refer to the Stable Implementation Agreements Document.)

18.6.3.3 M-Set

(Refer to the Stable Implementation Agreements Document.)

18.6.3.3.1 Successful Response

(Refer to the Stable Implementation Agreements Document.)

18.6.3.3.2 Partially Successful Response

(Refer to the Stable Implementation Agreements Document.)

18.6.3.3.3 Multiple Replies

(Refer to the Stable Implementation Agreements Document.)

18.6.3.3.4 Add/Remove Response

(Refer to the Stable Implementation Agreements Document.)

18.6.3.3.5 Parameter Agreements

(Refer to the Stable Implementation Agreements Document.)

18.6.3.4 M-Action

(Refer to the Stable Implementation Agreements Document.)

18.6.3.4.1 Multiple Objects

(Refer to the Stable Implementation Agreements Document.)

18.6.3.4.2 Parameter Agreements

(Refer to the Stable Implementation Agreements Document.)

18.6.3.5 M-Create

(Refer to the Stable Implementation Agreements Document.)

18.6.3.5.1 Managed Object Instance

(Refer to the Stable Implementation Agreements Document.)

18.6.3.5.2 Attribute Values

(Refer to the Stable Implementation Agreements Document.)

18.6.3.5.3 Parameter Agreements

(Refer to the Stable Implementation Agreements Document.)

18.6.3.6 M-Delete

(Refer to the Stable Implementation Agreements Document.)

18.6.3.6.1 Deletion of Objects Containing Objects

(Refer to the Stable Implementation Agreements Document.)

18.6.3.6.2 Parameter Agreements

(Refer to the Stable Implementation Agreements Document.)

18.6.4 Specific Agreements on CMIP

(Refer to the Stable Implementation Agreements Document.)

18.6.4.1 Invoke/Linked Identifier Size

(Refer to the Stable Implementation Agreements Document.)

18.6.4.2 Version

(Refer to the Stable Implementation Agreements Document.)

18.6.4.3 Linked Reply Values

(Refer to the Stable Implementation Agreements Document.)

18.6.4.4 Error Codes

(Refer to the Stable Implementation Agreements Document.)

18.6.5 Services Required by CMIP

CMIP requires the services provided by ACSE and ROSE. The conformance requirements for these services, and the underlying communication required to support them, are specified in subclause 5.12.1.5.

Editor's Note: [Proposed text for the ULSIG subclause 5.12.1.5. No agreements beyond the standards are made except where noted.

5.12.1.5 Network Management

ROSE Requirements:

The ROSE requirements are as specified in ISO 9596 section 5.2: Underlying Services, and section 6.2 Remote Operations.

Operations Classes

- 1, 2, and 5

Association Classes

- 3

ACSE Requirements:

all

Editor's Note: [All means what is specified in the Stable OIW agreements for ACSE in section 5.5.]

Application Contexts:

- as defined by ISO/DP 10040 ANNEX A

Editor's Note: [Pending a DIS Version of the Standard. This is beyond the standard.]

Abstract Syntaxes

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

Editor's Note: [ISSUE - Will there be a version2(2) syntax when the addendum on authentication becomes a standard?]

Associated Transfer Syntax:

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

- {joint-iso-ccitt ms(9) cmip(1) version2(2) abstractSyntax(4)}

Editor's Note: [Pending approval of the CMIP Addendum, the version2(2) is beyond the current DIS standard. As per ISO/IEC 9596 section 7.5, this abstract syntax incudes "all data types resolved by the ANY DEFINED BY X productions, in which X is of type OBJECT IDENTIFIER."]

Associated Transfer Syntax:

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

Mode-Selection

- Normal Mode

Presentation Requirements:

Presentation Functional Units:

- kernel

Presentation Contexts

- at least two presentation contexts must be supported

Mode-Selection

- Normal mode (non-X.410) shall be supported.

Session Requirements

Session Functional Units

- kernel
- full duplex

Version Number: 2

Maximum Size of User Data Parameter field: Shall be 10,240 octets. Implementations may specify in their PICS a maximum size down to 1024 octets

Editor's Note: [This is beyond the current standard.]

ASN.1 Encoding Rules

Some INTEGER types of the CMIP PCI may exceed the maximum size specified in the UNIVERSAL ASN.1 ENCODING RULES, section 5.10. See the range of values for INTEGER type Parameters in the Network Management chapter.

Editor's Note: [For example: a 32 bit unsigned integer, as specified for IEEE 802.x management statistics, can represent 2**32-1. This would require 5 octets for ASN.1 encoding. The current 4 octet restriction in the OIW ASN.1 agreements only allows integers up to 2**31-1. Specific agreements are needed in clause 18.6 regarding the length of INTEGERs.]

]

18.7 MANAGEMENT INFORMATION

This clause, which is based on ISO standards' documents [MIM] and [GDMO], deals with basic concepts and modelling techniques related to management information. It discusses (i) the information model (subclause 18.7.1), (ii) principles for naming managed objects and their attributes (subclause 18.7.2), and (iii) guidelines for defining management information (subclause 18.7.3). It is not within the scope of this clause to define specific elements of management information - such definitions can be obtained via the Management Information Library (MIL) produced by the OSI MIB Working Group (a subgroup of the NMSIG).

Tutorial Note: [Management information comprises all information in the network that is of interest to network management. A computer node in a network, a transport connection, an event log are all examples of network resources for which management information can be defined. Management information is collectively referred to as the MIB or Management Information Base.]

18.7.1 The Information Model

This subclause contains agreements related to the information model as specified in clause 5 of [MIM].

Tutorial Note: [Management information is modelled using object-oriented techniques. All "things" in the network that are to be managed, are represented in terms of managed objects. A managed object is an abstraction (or a logical view) of a "manageable" physical or logical network resource. "Manageable," in this context, means that the particular resource can be managed by using OSI Management Services and Protocols. Examples of managed objects include protocol layer entities, modems, connections, etc.

Each managed object belongs to a particular object class. An object class represents a collection of managed objects with the same, or similar properties. Each object class has a pre-defined identifier assigned to it by a standards' registration authority. A particular managed object existing in a particular network can be regarded as an instance of the object class to which it belongs. Thus, an object instance represents

an actual realisation of an object class. A managed object is identified by specifying its object class and object instance.

Managed objects contain properties which are referred to as attributes.

Managed objects participate in relationships with each other. The relationships that are of particular concern to the Management Information Model are a) the containment relationship, and b) the inheritance relationship. These relationships are used to construct management information hierarchies, as described below. Managed objects do participate in relationships other than the two mentioned above; e.g. the Service relationship, where a managed object uses the services provided by another managed object, as in the case of a Transport Layer object using the services provided by a Network Layer object. These relationships, however, are not particularly significant for the Information Model. They can be easily represented as either managed objects or attributes, contained within the managed objects participating in the relationship.

MANAGEMENT INFORMATION HIERARCHIES

The following Management Information Hierarchies are identified:

THE CONTAINMENT HIERARCHY

This hierarchy is constructed by applying the relationship "is contained in" to objects and attributes. Objects of one class may contain objects of the same or different class. Attributes are contained within objects at any level of the containment hierarchy. Attributes cannot contain objects or other attributes. All object classes must have at least one possible superior in the containment tree. The definition of a class may permit it to have more than one such superior. However, individual instances of such a class are nevertheless contained in only one instance of a possible containing class. A special object called "root" is the ultimate superior in the containment hierarchy.

The containment hierarchy is important because it is used for naming object instances. It also defines an existence dependency among its components; i.e. an object or attribute can 'exist' only if the containing object also 'exists'. If an object contains other objects, it cannot be deleted until the contained objects have been deleted. The contained objects may be deleted automatically, if this is specified in the definition of the managed object class(es) of the contained objects.

THE INHERITANCE OR OBJECT CLASS HIERARCHY

This hierarchy is constructed by applying the relationship "inherits properties of" to object classes. An object class may inherit properties of another object class, with refinement obtained by adding additional properties. The inheriting class is called the subclass in this relationship, and the parent the superclass. For example, the class "Network Entity" may be a subclass of "Layer Entity" and a superclass of "X.25 Network Entity." Each class may have zero, one or more subclasses. Subclasses

may in turn have furthur subclasses, to any degree. A special object called "top" is the ultimate superclass.

The inheritance hierarchy is useful in that it leads to a manageable and extensible technique for the definition of object classes. The inheritance hierarchy has NO relevance to object and/or instance naming.

THE REGISTRATION HIERARCHY

This hierarchy is not based on any particular relationship, and is independent of both the inheritance and containment hierarchies. It contains Object Identifiers for object classes and attributes, as assigned by the standards' registration authority.

The registration hierarchy is important because it is used for identifying object classes and attributes. It is used to ensure global uniqueness and to permit extensions without a centralized registration authority.]

18.7.1.1 Basic Concepts

The following concepts/features of the information model are supported, as specified in clause 5 of [MIM].

managed object managed object class managed object instance
attribute group attribute set-valued attribute
attribute value assertion management operation
encapsulation behaviour notification

18.7.1.2 Management Operations Supported

The following management operations are supported, as specified in clause 5.2 of [MIM].

Operations that apply to attributes :

Get attribute value
Replace attribute value
Set-to-default value
Add attribute value
Remove attribute value

Operations that apply to managed objects :

Create
Delete
Action

18.7.1.3 Filter

The concept of filter is supported as specified in clause 5.3 of [MIM]. Restrictions on its usage are specified in subclause 18.6.2.2.2 of these agreements.

18.7.1.4 Inheritance

All the inheritance related concepts (refinement, subclass, superclass, inheritance hierarchy, etc) presented in clause 5.5 of [MIM] are supported.

The following additional constraints need to be enforced for the Phase 1 IAs in order to remove potential ambiguities:

Subclasses must inherit ALL the optional attributes of their respective superclasses. Once inherited, these attributes may remain as optional attributes of the subclass or may become mandatory attributes of the subclass.

When an instance of a managed object class is created, it must support all the mandatory attributes defined for that class. The instance may support some or none of the optional attributes defined for its class. Once created, the managed object instance must support , throughout its lifetime, exactly the same set of attributes that were assigned to it at the time of creation, i.e. dynamic creation/deletion of attributes within an object instance is not allowed.

During the lifetime of a managed object instance, each of its attributes must have a value that is valid for the attribute syntax of that attribute.

The range of the attribute values for any attribute may not be redefined in the process of refinement. If it is anticipated that the range of attribute values may change, then the use of the ASN.1 enumerated type for the attribute syntax is discouraged.

Multiple inheritance is not supported for the Phase 1 IAs, since no requirements for it have been voiced within the NMSIG.

18.7.1.5 Polymorphism

Editor's Note: [Polymorphism is a very useful concept insofar as it facilitates interoperability across different versions and vendor extensions of a managed object class. However, issues and problems related to it, especially those dealing with the naming of polymorphic classes, have not been thoroughly examined or resolved in the standards. Given this, does NMSIG feel the need to incorporate polymorphism into the Phase 1 IAs ?]

Polymorphism is not supported for the Phase 1 IAs, since no requirements for it have been voiced within the NMSIG.

18.7.2 Principles of Naming

This subclause contains agreements about principles of naming as specified in clause 6 of [MIM].

18.7.2.1 Containment Hierarchy

All concepts about the containment hierarchy presented in clause 6.1 of [MIM] are supported.

18.7.2.2 Name Structure

18.7.2.2.1 Object Class Identification

A managed object class is identified by an ASN.1 object identifier, as specified in clause 6.2.1 of [MIM].

18.7.2.2.2 Object Instance Identification

The distinguished name approach is supported for the identification of managed object instances.

Editor's Note: [Many issues/questions regarding the naming of managed object instances have arisen because the related standards' text (clause 6.2.2 of [MIM]) is somewhat unclear.

The following issues related to naming managed object instances are identified :

a) Referring to the first sentence of clause 6.2.2 of [MIM], which starts with "The definition of each managed object class ...," does "an" identification attribute imply "only one" or "at least one" ? Can different name bindings for the same managed object class specify different distinguishing attributes, or is there just one distinguishing attribute per managed object class ?

b) Do name bindings get inherited ?

c) Is the distinguishing attribute of a subclass the same or different from distinguishing attribute of its superclass? If the superclass and its subclass have the same distinguishing attribute, there could be ambiguities in situations where instances of both the superclass and its subclass exist in the containment tree. If the superclass and its subclass do not have the same distinguishing attribute, polymorphism cannot be supported.

d) What is the point of reference from which managed object instances are defined - full distinguished name or partial distinguished name?]

18.7.2.2.3 Selection Of Distinguishing Attributes

The distinguishing attribute for a managed object class must be very carefully selected. It must be able to distinguish not only between instances of the object class for which it is defined, but also between instances of all other object classes that have the same superior object class. For example, consider the following figure which shows the structure of a containment tree :

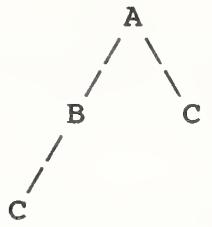


Figure 2: Example Containment Tree.

Here, A represents instances of Object Class A, B represents instances of Object Class B and C represents instances of Object Class C. As can be seen from the figure, instances of Object Class C may be contained in either instances of Object Class A, or in instances of Object Class B. When the RDN of Object Class C is defined, it is necessary to make sure that it is different from the RDN for Object Class B. If Object Class B and Object Class C were to support the same RDN, it would not be possible to unambiguously traverse down the containment tree from A.

The above example shows a simple containment tree. In the real world, however, containment trees could be much more complex, and the selection of distinguishing attributes could involve extensive checking and verification over multiple object classes.

Editor's Note: [Consider the following proposal :

> "The process of selecting the correct distinguishing attribute can be made simpler if every object class supports an additional distinguishing attribute called "My Object Class," whose value identifies the object class it is contained in. If this is done, the process of selecting and verifying the RDN of an object class would not require the consideration of object classes other than the one defining the RDN."]

The above proposal will be worked on by the NMSIG and submitted to the standards.

18.7.2.2.4 Attribute Identification

Each individual attribute of a managed object is identified by an ASN.1 object identifier, as specified in clause 6.2.4 of [SMI Part 1].

18.7.3 Guidelines for the Definition of Management Information

This subclause contains agreements about guidelines for the definition of management information, as specified in [GDMO]. These guidelines form a normative part of the standard; hence they must be strictly followed while defining management information.

18.7.3.1 Syntactical Definitions of Management Information

18.7.3.1.1 Managed Object Class Template

For Phase 1 IAs, the template supported by NMSIG for defining managed object classes is the same as the Managed Object Class template defined in clause 10.3.2 of [GDMO], with the agreement that the optional clause POLYMORPHIC SET is not to be used. The POLYMORPHIC SET clause is not supported, as per the agreements on polymorphism specified in 18.7.1.5.

Supporting productions for "propertylist" and "modifier" are adopted as specified in clause 10.3.2 of [GDMO].

Supporting definitions of the DERIVED FROM, POLYMORPHIC SET, ATTRIBUTES, GROUP ATTRIBUTES, CREATE, DELETE, ACTIONS, NOTIFICATIONS, and PACKAGE clauses of the managed object class template are adopted as defined in clause 10.3.3 of [GDMO] with the following exceptions:

> The <specific-error-label> shall not be used because the managed object class template allows for multiple specific errors to be defined within an object class, and it is not possible to unambiguously communicate over CMIP multiple specific errors pertaining to a single managed object class.
>
> For the GROUP ATTRIBUTES clause, new attributes shall not be added to the group attribute from within the managed object class template because this can lead to ambiguities. Hence, the [<attribute-label>] portion of the supporting definition for the GROUP ATTRIBUTE clause shall not be used.
>
> For the PACKAGE clause the <condition-definition> shall only reflect the capabilities of the underlying resource that the managed object class is representing.

18.7.3.1.2 Conditional Package Template

The CONDITIONAL PACKAGE template specified in clause 10.4 of [GDMO] is supported. The agreements listed in 18.7.3.1.1 for the supporting definitions of the MANAGED OBJECT CLASS template are to be applied to the CONDITIONAL PACKAGE template, too.

18.7.3.1.3 Specific Error Template

The SPECIFIC ERROR template is not supported for the reasons given in 18.7.3.1.1

18.7.3.1.4 Name Binding Template

The NAME BINDING template is supported as described in clause 10.6 of [GDMO] except that the CONSTRAINTS clause shall not be used because its usage has not been clearly specified in the standard.

18.7.3.1.5 Attribute Template

The ATTRIBUTE template described in clause 10.7 of [GDMO] is supported. The DERIVED FROM and PERMITTED VALUES clauses of the ATTRIBUTE template are not supported, in general, because their usage could lead to major ambiguities. However, usage of attributes defined in [DMI] that use the DERIVED FROM clause and are registered is allowed. The PERMITTED VALUES clause can only be used if the ATTRIBUTE SYNTAX has been previously defined; e.g., in [DMI]. The REGISTERED AS clause, which has been defined as optional, is made mandatory. The BEHAVIOUR clause is made mandatory.

18.7.3.1.6 Group Attribute Template

The GROUP ATTRIBUTE template is supported as described in clause 10.8 of [GDMO].

18.7.3.1.7 Action Template

The ACTION template is supported as described in clause 10.10 of [GDMO].

18.7.3.1.8 Notification Template

The NOTIFICATION template is supported as described in clause 10.11 of [GDMO].

18.7.3.2 Guidelines For Defining Behaviour

The following details should be provided in the definition of each managed object class:

- a textual description of the network resource it represents, including its functional role.

- a description of the relationships that instances of this managed object class participate in with instances of the same or other managed object classes.

- a description of the operations that are supported by it, with precise definition of the effects, side effects if any, constraints, response notifications, failure modes, etc.

- specification of how instances of this managed object class are created and deleted, particularly whether they can be created/deleted via the management CREATE/DELETE operations.

- a description of notifications that can be generated, the conditions that generate them (e.g., crossing of a threshold), their contents and side-effects, if any. In particular, identify all the attributes that are subject to the AttributeChange and StateChange notifications, if these notifications are supported.

- other constraints, including those involving other managed object classes.

18.7.3.3 Other Guidelines

The Systems Management functions have defined various attributes and events, as indicated in clause 18.5 of these agreements. Object Definers are encouraged to make use of these attributes and events wherever applicable.

ANNEX A -- MANAGEMENT INFORMATION LIBRARY (MIL)

MANAGEMENT INFORMATION LIBRARY

(MIL)

OSI MIB Working Group
Version 4.0

March 29, 1990

A.1 INTRODUCTION

This document is produced by the OSI MIB Working Group (a subgroup of the NMSIG). It provides definitions of management information - managed object classes, name bindings, attributes, actions and notifications. Provision of these definitions is made by: a) references to standards' documents that contain these definitions, or b) inclusion of the actual definitions in this document; in which case they will be registered in the NMSIG arc of the ISO ASN.1 Object Identifier Tree.

Management information definitions provided by the OSI MIB Working Group have been introduced to accelerate the process of defining management information. They are intended to be implementable but also serve as a basis from which other implementations may define refinements or alternatives. These definitions do not override those provided by standards' groups or other OIW SIGs.

Editor's Note: The intention is to progress these definitions to an International Management Information Library.

A.2 RULES AND PROCEDURES

The following rules and procedures apply to managed object class definitions that are to be included in the MIL :

(i) All managed object class definitions provided by the MIL must comply with the NMSIG (ISO) object templates.

(ii) A managed object class definition provided by the MIL must represent an abstraction of an identifiable logical or physical resource that can be managed via OSI management.

(iii) All managed object classes in the MIL will have registered ASN.1 object identifiers assigned either by a standards' body if it is defining the managed object class, or, if the managed object class definition is being progressed within the NMSIG, by the NMSIG in its branch of the ISO Registration Tree.

(iv) A managed object class will be selected as a candidate for inclusion into the MIL if there are at least two NMSIG members from different companies who express a requirement (strong interest) for the managed object class. If this is not a standards' defined managed object class, then there must be at least one NMSIG member who is committed to developing the definition of the managed object class.

(v) A managed object class selected for the MIL will be given a priority based on the number of members who express interest in it.

(vi) All managed object class definitions that are proposed for inclusion into the MIL will undergo a review process within the NMSIG. NMSIG member defined managed object classs will additionally undergo a ballotting process. If problems are found with a standards' defined managed object class, the appropriate standards' body will be approached. If problems are found with a member defined managed object class, it will be returned with comments.

(vii) Based on its priority, there will be a call for contributions on the definition of a managed object class at an NMSIG meeting. Contributions could be in the form of: a) identification of a standards' body that is currently working on the definition, or b) an NMSIG member definition of the managed object class.

(viii) There will be no obsolescence of any managed object class specified in the MIL.

A.3 GENERAL GUIDELINES

It is recommended that the following guidelines be used in general for all managed object definitions, unless there is a specific exception condition:

a) For the ObjectCreation Notification, send all the attributes of the created managed object instance in the CreateInfo field.

A.4 OBJECT CLASSES

A.4.1 Discriminator

This managed object class is used to define the criteria for controlling management services. Refer to [ISO Doc x] for the definition of this managed object class.

A.4.2 Event Forwarding Discriminator

This managed object class is used to define the criteria that must be satisfied by potential event reports before the event reports are forwarded to a particular destination. Refer to [ISO Doc x] for the definition of this managed object class.

A.4.3 NMSIG Agent

A.4.3.1 NMSIG Agent Definition

```
nmsig-agent    MANAGED OBJECT CLASS
   DERIVED FROM  {top}
   CHARACTERISED BY
 BEHAVIOUR DEFINITIONS   agent-behaviour
      ATTRIBUTES   nmsig-agentId  GET,

REGISTERED AS {obj-class}
```

A.4.3.2 NMSIG Agent Behaviour

agent-behaviour BEHAVIOUR

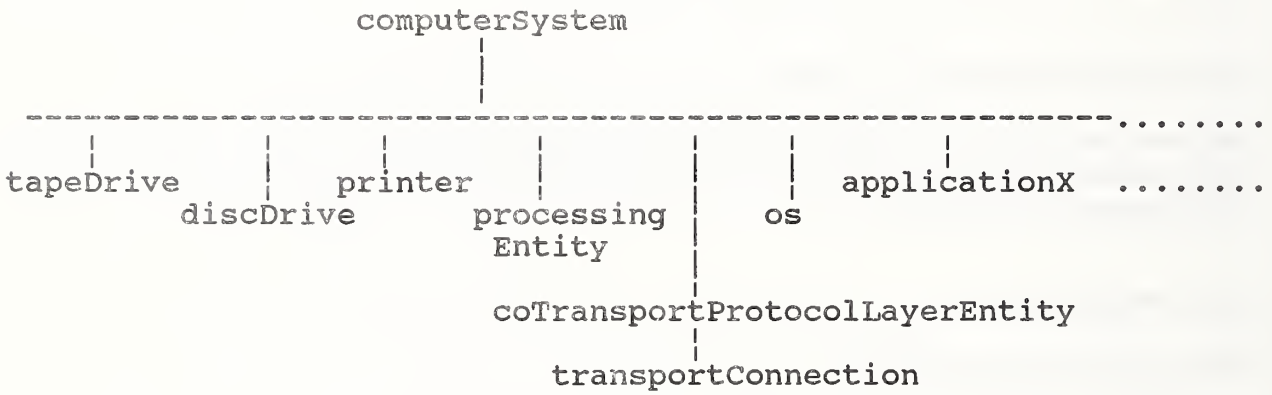
DEFINED AS

> This managed object class represents an NMSIG agent system, which is an open system that supports the NMSIG agreements to make one or more managed objects visible to other open systems that support the NMSIG agreements.
>
> An NMSIG agent system may not support more than one instances of the NMSIG Agent managed object class. If supported, this instance is assumed to be pre-existent when the NMSIG agent system comes up; i.e., management CREATE or DELETE is not supported.
>
> At this time, the NMSIG Agent managed object class only serves to name management support managed objects (e.g., EventForwardingDiscriminator).

A.4.4 NMSIG Computer System

Editor's Note: A model has been proposed for defining managed object classes related to computers, as follows:

The philosophy behind the proposed model is to define a composite or aggregrate managed object class called "computerSystem" that provides a high level view of a computer system, including its physical and logical, as well as its hardware and software components. Detailed views of these components are then modelled as object classes contained within the computerSystem object class, as shown in the CONTAINMENT TREE below. (NOTE : This is NOT an inheritance tree.)



A great benefit provided by this model is flexibility. As and when more computer components need to be specified, they can be defined as individual object classes and "plugged" into the above structure under computerSystem, without upsetting the other object classes.

The 'system' managed object class defined in [DMI] was not used because it's definition was considered to be inappropriate.

A.4.4.1 NMSIG Computer System Definition

```
nmsig-computerSystem   MANAGED OBJECT CLASS
  DERIVED FROM  {top}
  CHARACTERISED BY
    BEHAVIOUR DEFINITIONS  computerSystem-behaviour
    ATTRIBUTES   nmsig-systemId  GET,
                 AdministrativeState  GET-REPLACE
                 HealthState  GET,
                 OperationalState   GET,
                 nmsig-systemTime   GET,                          nmsig-peripheralNames
GET,
                 nmsig-userFriendlyLabel  GET-REPLACE
    NOTIFICATIONS     ObjectCreationUnConfirmed,
                      ObjectDeletionUnConfirmed,
                      AttributeChangeUnConfirmed,
                      StateChangeUnConfirmed,
                      ProcessingErrorAlarmUnConfirmed,
                      EnvironmentalAlarmUnConfirmed,
                      EquipmentAlarmUnConfirmed
```

REGISTERED AS {obj-class}

A.4.4.2 NMSIG Computer System Behaviour

computerSystem-behaviour BEHAVIOUR

DEFINED AS

The nmsig-computerSystem managed object class is a composite or aggregate object class that provides a high level view of a general purpose business computer system, including its physical, logical, hardware and software components.

The Computer System managed object class supports all the values of the administrative state. It supports only the 'enabled' and 'disabled' values of the operational state.

The 'enabled' value of the operational state indicates that the underlying computer system resources are together capable of providing minimal computing services. These enabled resources may or may not be modelled as managed objects, and may or may not include the entire set of resources which together are viewed as the computer system.

The 'disabled' value of the operational state indicates that the underlying computer system resources are incapable of providing minimal services at the current time.

The peripheralNames attribute specifies the names of auxiliary devices that are used by the underlying computer system resource.

The CreateInfo field of the ObjectCreation notification shall contain all the attributes of the created computer sytem instance.

The DeleteInfo field of the ObjectDeletion notification shall be NULL.

Attributes that are subject to the AttributeChange notification are: nmsig-peripheralNames, nmsig-userFriendlyLabel, HealthState.

Attributes that are subject to the StateChange notification are: AdministrativeState and OperationalState.

A.4.5 NMSIG Connection Oriented Tranport Protocol Layer Entity

A.4.5.1 NMSIG CO Transport Protocol Layer Entity Definition

```
nmsig-coTransportProtocolLayerEntity    MANAGED OBJECT CLASS

  DERIVED FROM          {top}
  CHARACTERIZED BY
     BEHAVIOUR DEFINITIONS   coTransportProtocolLayerEntity-behaviour
     ATTRIBUTES          nmsig-coTransportProtocolLayerEntityId  GET,
                    AdministrativeState  GET-REPLACE,
                        OperationalState  GET,
                        HealthState GET,
                        nmsig-localTransportAddresses  GET,
                        nmsig-maxConnections  GET,
                        nmsig-openConnections  GET,
                        OutgoingConnectionsRequestCounter  GET,
                        IncomingConnectionsRequestCounter  GET,
                        OutgoingConnectionRejectErrorCounter  GET,
                        IncomingConnectionRejectErrorCounter  GET,
                        OutgoingDisconnectErrorCounter  GET,
                        IncomingDisconnectErrorCounter  GET,
                        nmsig-incomingNormalDisconnectCounter  GET,
                        nmsig-outgoingNormalDisconnectCounter  GET,
                        OctetsSentCounter   GET,
                        OctetsReceivedCounter  GET,
                        IncomingTemporalErrorCounter  GET,
                        OutgoingTemporalErrorCounter  GET,
                        nmsig-checksumTPDUsDiscardedCounter  GET,
                        nmsig-transportEntityType GET,
                        nmsig-productInfo GET,
                        nmsig-entityUpTime GET
            NOTIFICATIONS       ObjectCreationUnConfirmed,
                        ObjectDeletionUnConfirmed,
                        AttributeChangeUnConfirmed,
                        StateChangeUnConfirmed,
                        ProcessingErrorAlarmUnConfirmed,
                        nmsig-counterWrapUnConfirmed

REGISTERED AS       {obj-class}
```

A.4.5.2 NMSIG CO Transport Protocol Layer Entity Behaviour

coTransportProtocolLayerEntity-behaviour BEHAVIOUR

DEFINED AS

The managed object class nmsig-coTransportProtocolLayerEntity represents an instantiation of any connection-oriented transport layer protocol e.g. the ISO Transport Protocol layer or the Internet Transmission Control Protocol (TCP). The transport protocol layer is layer four of the OSI Reference model. It provides for the transparent transference of data between two peer entities. It relieves its users from any concerns about the detailed way in which supporting communication media are utilized to achieve this transfer. The connection oriented transport protocol layer entity makes use of a transport connection for the purpose of transferring data.

This managed object class represents a "generic" view of a connection oriented transport protocol layer entity. It does not concern itself with the details of specific transport protocols like ISO TP or TCP. Transport entities that are tied to a specific protocol can be defined as its subclasses; in fact their definitions are being progressed within various standards' bodies. The purpose of defining this managed object class, however, is to provide a common base that will facilitate the high level management of similar but slightly differing resources.

The connection oriented transport protocol layer entity supports all values of the administrative and operational states.

The 'enabled' value of the operational state indicates that the underlying transport protocol layer entity resource is capable of supporting transport connections but currently has no open transport connections.

The 'disabled' value of the operational state indicates that the underlying transport protocol layer entity resource is not capable of supporting any transport connections.

The 'active' value of the operational state indicates that the underlying transport protocol layer entity resource is currently supporting at least one transport connections and is capable of supporting additional transport connections.

The 'busy' value of the operational state indicates that the underlying transport protocol layer entity resource is supporting the maximum number of transport connections that it is capable of supporting.

The CreateInfo field of the ObjectCreation notification shall contain all the attributes of the created connection-oriented transport protocol layer entity instance.

The DeleteInfo field of the ObjectDeletion notification shall contain all the attributes of the deleted connection-oriented transport protocol layer entity instance.

Attributes that are subject to the AttributeChange notification are: nmsig-localTransportAddresses, nmsig-maxConnections, nmsig-productInfo, HealthState.
Attributes that are subject to the StateChange notification are: AdministrativeState and OperationalState.

The counterWrap notification is emitted when any of the counter attributes wrap.

A.4.6 NMSIG Connectionless Network Protocol Layer Entity

A.4.6.1 NMSIG Connectionless Network Protocol Layer Entity Definition

```
nmsig-clNetworkProtocolLayerEntity    MANAGED OBJECT CLASS

   DERIVED FROM            {top}
   CHARACTERIZED BY
        BEHAVIOUR DEFINITIONS  clNetworkProtocolLayerEntity-behaviour
     ATTRIBUTES            nmsig-clNetworkProtocolLayerEntityId  GET,
                    AdministrativeState  GET-REPLACE,
                    OperationalState  GET,
                    HealthState  GET,
                    nmsig-localNetworkAddresses  GET,
                    nmsig-nPDUTimeToLive  GET-REPLACE,
                    PDUsSentCounter  GET,
                    PDUsReceivedCounter  GET,
                    nmsig-PDUsForwardedCounter  GET,
                    nmsig-PDUsReasmbldOKCounter  GET,
                    nmsig-PDUsReasmblFailCounter  GET,
                    nmsig-PDUsDiscardedCounter   GET,
                    nmsig-networkEntityType GET,
                    nmsig-productInfo GET,
                    nmsig-entityUpTime GET

     NOTIFICATIONS       ObjectCreationUnConfirmed,
                    ObjectDeletionUnConfirmed,
                    AttributeChangeUnConfirmed,
                    ProcessingAlarmUnConfirmed,
                    StateChangeUnConfirmed,
                    nmsig-counterWrapUnConfirmed

     PACKAGE        nmsig-clNetworkProtocolLayerEntityRedirection
               PRESENT IF connectionless network protocol layer
                    entity supports redirection of recd PDUs


REGISTERED AS      {obj-class}
```

A.4.6.2 NMSIG Connectionless Network Protocol Layer Entity Behaviour

clNetworkProtocolLayerEntity-behaviour BEHAVIOUR

DEFINED AS

The managed object class nmsig-clNetworkProtocolEntity represents an instantiation of a connectionless network protocol layer. The network layer is layer three of the OSI Reference Model. It provides network services for the transparent transfer of data between peer transport entities. It relieves the transport protocol layer from the need to know anything about the underlying network technologies used to achieve data transfer. The connectionless network protocol layer does not make use of a network connection for the purposes of transferring data. No dynamic peer to peer agreement is involved in the process of data transfer.

An instance of this managed object class supports only one type of protocol and one address domain.

This managed object class represents a "generic" view of a connectionless network protocol layer entity. It does not concern itself with the details of specific network protocols. Network entities that are tied to a specific network protocol can be defined as its subclasses; in fact their definitions are being progressed within various standards' bodies. The purpose of defining this managed object class, however, is to provide a common base that will facilitate the high level management of similar but slightly differing resources.

The NMSIG connectionless network protocol layer entity managed object class supports all the values of the administrative state attribute. It supports only the 'disabled' and 'enabled' values of the operational state attribute.

The 'enabled' value of the operational state indicates that the underlying connectionless network protocol layer entity resource is capable of providing connectionless network layer services.

The 'disabled' value of the operational state indicates that the underlying connectionless network protocol layer entity resource is incapable of supporting any network services at the current time.

The CreateInfo field of the ObjectCreation notification shall contain all the attributes of the created connectionless network protocol layer entity instance.

The DeleteInfo field of the ObjectDeletion notification shall contain all the attributes of the deleted connectionless network protocol layer entity instance.

Attributes that are subject to the AttributeChange notification are: nmsig-localNetworkAddresses, nmsig-nPDUTimeToLive, nmsig-productInfo, and HealthState

Attributes that are subject to the StateChange notification are: AdministrativeState and OperationalState.

The counterWrap notification is emitted when any of the counter attributes wrap.

A.4.6.3 NMSIG CL Network Protocol Layer Entity Redirection Package

```
nmsig-clNetworkProtocolLayerEntityRedirection  CONDITIONAL PACKAGE
        BEHAVIOUR DEFINITIONS  clNetworkProtocolLayerEntityRedirection-
                                    behaviour
        ATTRIBUTES  nmsig-PDUsRedirected  GET

REGISTERED AS  {package}

clNetworkProtocolLayerEntityRedirection-behaviour  BEHAVIOUR

    DEFINED AS
```

This package reflects the redirection capability of the underlying connectionless network protocol layer entity resource.

A.4.7 NMSIG Equipment

A.4.7.1 NMSIG Equipment Definition

```
nmsig-equipment   MANAGED OBJECT CLASS
  DERIVED FROM  {top}
  CHARACTERIZED BY
    BEHAVIOUR DEFINITIONS  equipment-behaviour
    ATTRIBUTES    nmsig-equipmentId  GET,
                  OperationalState  GET,
                  HealthState  GET,
                  AdministrativeState  GET-REPLACE,
                  nmsig-locationName  GET-REPLACE,
                  nmsig-contactNames  ADD-REMOVE,
                  nmsig-equipmentPurpose      GET-REPLACE,
                  nmsig-productInfo   GET,
                  nmsig-vendorName    GET-REPLACE,
                  nmsig-userFriendlyLabel  GET-REPLACE

  NOTIFICATIONS    EnvironmentalAlarmUnConfirmed,
                  EquipmentAlarmUnConfirmed,
                  ObjectCreationUnConfirmed,
                  ObjectDeletionUnConfirmed,
                  AttributeChangeUnConfirmed,
                  StateChangeUnconfirmed

REGISTERED AS  {obj-class}
```

A.4.7.2 NMSIG Equipment Behaviour

equipment-behaviour BEHAVIOUR

DEFINED AS

The NMSIG equipment managed object class represents physical entities. Instances of this managed object class are located in specific geographic locations and support some type of functions. For example, a PBX, which may be regarded as an instance of this managed object class, performs switching functions. Multiplexers, amplifiers, and repeaters which can also be regarded as instances of this managed object class perform transmission functions. Equipment may be nested in equipment, thereby creating a containment relationship. For example, a line card is contained in an equipment shelf which is nested in a relay rack which is part of a switch.

Instances of this managed object class may be endpoints of a circuit or facility.

The NMSIG Contact Names attribute specifies who (persons or organizations) are to be contacted about the equipment.

The NMSIG Location Name attribute identifies where the equipment is located.

The NMSIG Vendor Name attribute identifies the organization from whom the equipment was obtained (i.e., purchased, leased, etc.).

The NMSIG equipment managed object class supports all permissible values of the administrative and operational states.

The CreateInfo field of the ObjectCreation notification shall contain all the attributes of the created equipment instance.

The DeleteInfo field of the ObjectDeletion notification shall contain all the attributes of the deleted equipment instance.

Attributes that are subject to the AttributeChange notification are: nmsig-locationName, nmsig-contactNames, nmsig-equipmentPurpose, nmsig-productInfo, nmsig-vendorName, nmsig-userFriendlyLabel, HealthState.

Attributes that are subject to the StateChange notification are: AdministrativeState and OperationalState.

A.4.8 NMSIG IEEE 802.3

A.4.8.1 NMSIG IEEE 802.3 Definition

```
nmsig-IEEE-802.3    MANAGED OBJECT CLASS
   DERIVED FROM  {top}
      CHARACTERIZED BY
        BEHAVIOUR DEFINITIONS  iEEE-802.3-behaviour
      ATTRIBUTES   nmsig-IEEE-802.3Id   GET,
                 OperationalState   GET,
                 AdministrativeState   GET-REPLACE,
                 nmsig-macAddress   GET-REPLACE,
                 nmsig-IEEE-802.3State   GET-REPLACE,
                 nmsig-multicastAddressList   GET-REPLACE,
                 HealthState  GET

   OPERATIONS      DELETE
                 ACTIONS   nmsig-executeSelfTest

   NOTIFICATIONS    ObjectCreationUnConfirmed,
                 ObjectDeletionUnConfirmed,
                 AttributeChangeUnConfirmed,
                 StateChangeUnconfirmed

REGISTERED AS  {obj-class}
```

A.4.8.2 NMSIG IEEE 802.3 Behaviour

```
iEEE-802.3-behaviour  BEHAVIOUR

  DEFINED AS
```

The managed object class nmsig-IEEE-802.3 represents an instantiation of an IEEE 802.3 CSMA/CD MAC. It may contain either an nmsig-IEEE-802.3-XMT managed object, an nmsig-802.3-RCV managed object, or both of these subordinate objects, as shown in the following figure.

```
+-----------------------------------------+
|  NMSIG IEEE 802.3                       |
|                                         |
| +---------------+     +---------------+ |
| |  NMSIG IEEE   |     |  NMSIG IEEE   | |
| |  802.3 XMT    |     |  802.3 RCV    | |
| +---------------+     +---------------+ |
|                                         |
|                                         |
+-----------------------------------------+
```

The NMSIG IEEE 802.3 managed object class supports only the 'enabled' and 'disabled' values of the operational state attribute. The 'enabled' value indicates that the underlying IEEE 802.3 resource is available for use, and the 'disabled' value indicates that the underlying IEEE 802.3 resource is not available for use.

The NMSIG IEEE 802.3 managed object class supports the DELETE operation; this operation serves to reinitialize the CSMA/CD MAC.

The NMSIG IEEE 802.3 managed object class supports an nmsig-executeSelfTest ACTION; this action causes a self test to be performed on the referenced managed object instance.

The CreateInfo field of the ObjectCreation notification shall contain all the attributes of the created IEEE 802.3 instance.

The DeleteInfo field of the ObjectDeletion notification shall contain all the attributes of the deleted IEEE 802.3 instance.

Attributes that are subject to the AttributeChange notification are: nmsig-macAddress, nmsig-multicastAddressList, HealthState.

Attributes that are subject to the StateChange notification are: AdministrativeState and OperationalState.

A.4.9 NMSIG IEEE 802.3 RCV

A.4.9.1 NMSIG IEEE RCV Definition

```
nmsig-IEEE-802.3-RCV    MANAGED OBJECT CLASS
    DERIVED FROM  {top}
        CHARACTERIZED BY
        BEHAVIOUR DEFINITIONS  IEEE-802.3-RCV-behaviour
        ATTRIBUTES   nmsig-IEEE-802.3-RCVId   GET,
                OperationalState   GET,
                AdministrativeState   GET-REPLACE,
                HealthState   GET,
                nmsig-multicastRcvState   GET-REPLACE,
                PDUsReceivedCounter   GET,
                nmsig-PDUsFCSErrorCounter    GET,
                nmsig-PDUsAlignmentErrorCounter    GET,
                nmsig-PDUsInRangeLengthErrorCounter   GET,
                nmsig-PDUsOutRangeLengthErrorCounter   GET,
                nmsig-PDUsTooLongErrorCounter   GET
                OctetsReceivedCounter   GET,
                nmsig-multicastPDUsRcvCounter   GET,
                nmsig-broadcastPDUsRcvCounter   GET,
                nmsig-internalMACRcvErrorCounter   GET,
                nmsig-sourceAddrLastFCSErrorPDU   GET,
                nmsig-sourceAddrLastAlignmentErrorPDU   GET,
                nmsig-sourceAddrLastInRangeLengthErrorPDU   GET,
                nmsig-sourceAddrLastOutRangeLengthErrorPDU    GET,
nmsig-sourceAddrLastTooLongErrorPDU   GET,
                nmsig-FCSErrorThreshold   GET-REPLACE,
                nmsig-alignmentErrorThreshold   GET-REPLACE,
                nmsig-inRangeThreshold   GET-REPLACE,
                nmsig-outRangeThreshold   GET-REPLACE,
                nmsig-frameTooLongThreshold   GET-REPLACE,
                nmsig-internalMACRcvErrorThreshold   GET-REPLACE,
                nmsig-enablePromiscuousState  GET-REPLACE

  NOTIFICATIONS   ObjectCreationUnConfirmed,
                ObjectDeletionUnConfirmed,
                AttributeChangeUnConfirmed,
                StateChangeUnConfirmed,
                ProcessingAlarmUnConfirmed,
                nmsig-counterWrapUnConfirmed,
                CommunicationAlarmUnConfirmed

REGISTERED AS  {obj-class}
```

A.4.9.2 NMSIG IEEE 802.3 RCV Behaviour

iEEE-802.3-RCV-behaviour BEHAVIOUR

DEFINED AS

The managed object class nmsig-IEEE-802.3-RCV represents an instantiation of an IEEE 802.3 CSMA/CD MAC receiver. This object may be contained within an nmsig-IEEE-802.3 managed object.

The NMSIG IEEE 802.3 RCV managed object class supports only the 'enabled' and 'disabled' values of the operational state attribute. The 'enabled' value indicates that the underlying IEEE 802.3 RCV resource is available for use, and the 'disabled' value indicates that the underlying IEEE 802.3 RCV resource is not available for use.

The definitive description of the counter attributes, their operation and precedence is specified in the [IEEE Doc X].

The NMSIG IEEE 802.3 RCV managed object class supports several threshold attributes; all are associated with the generation of a Communication Alarm notification.

The CreateInfo field of the ObjectCreation notification shall contain all the attributes of the created IEEE 802.3 RCV instance.

The DeleteInfo field of the ObjectDeletion notification shall contain all the attributes of the deleted IEEE 802.3 RCV instance.

Attributes that are subject to the AttributeChange notification are: nmsig-multicastRcvState, nmsig-promiscuousRcvState, nmsig-FCSErrorThreshold, nmsig-alignmentErrorThreshold, nmsig-inRangeThreshold, nmsig-outRangeThreshold, HealthState, nmsig-frameTooLongThreshold and nmsig-internalMACRcvErrorThreshold.

Attributes that are subject to the StateChange notification are: AdministrativeState and OperationalState.

The counterWrap notification is emitted when any of the counter attributes wrap.

A.4.10 NMSIG IEEE 802.3 XMT

A.4.10.1 NMSIG IEEE 802.3 XMT Definition

```
nmsig-IEEE-802.3-XMT    MANAGED OBJECT CLASS
   DERIVED FROM  {top}
      CHARACTERIZED BY
      BEHAVIOUR DEFINITIONS  iEEE-802.3-XMT-behaviour
      ATTRIBUTES   nmsig-IEEE-802.3-XMTId   GET,
                 OperationalState   GET,
                 AdministrativeState   GET-REPLACE,
                 HealthState  GET,
                 nmsig-XmtState  GET-REPLACE,
                 PDUsSentCounter   GET,
                 nmsig-singleCollisionPDUsCounter   GET,
                 nmsig-multipleCollisionPDUsCounter   GET,
                 nmsig-lateCollisionsCounter   GET,
                 nmsig-PDUsAbortedExcessiveCollisionsCounter   GET,
nmsig-carrierSenseErrorsCounter   GET,
                 nmsig-collisionPDUsCounter   GET,
                 OctetsSentCounter   GET,
                nmsig-multicastPDUsXmtCounter   GET,
                nmsig-broadcastPDUsXmtCounter   GET,
                nmsig-PDUsLostInternalMACXmtErrorCounter   GET,
                nmsig-PDUsExcessiveDeferralCounter   GET,
                nmsig-collisionPDUsThreshold   GET-REPLACE,
                nmsig-lateCollisionsThreshold   GET-REPLACE,
                nmsig-PDUsAbortedExcessColThreshold   GET-REPLACE,
nmsig-carrierSenseErrorsThreshold   GET-REPLACE,
                nmsig-internalMACXmtErrorThreshold   GET-REPLACE,
nmsig-excessiveDeferralThreshold   GET-REPLACE

   NOTIFICATIONS   ObjectCreationUnConfirmed,
                ObjectDeletionUnConfirmed,
                AttributeChangeUnConfirmed,
                CommunicationAlarmUnConfirmed,
                StateChangeUnConfirmed,
                ProcessingAlarmUnConfirmed,
                nmsig-counterWrapUnConfirmed
REGISTERED AS  {obj-class}
```

A.4.10.2 NMSIG IEEE 802.3 XMT Behaviour

iEEE-802.3-XMT-behaviour BEHAVIOUR

DEFINED AS

The managed object class nmsig-IEEE-802.3-XMT represents an instantiation of an IEEE 802.3 CSMA/CD MAC transmitter. This object may be contained within an nmsig-IEEE-802.3 managed object.

The NMSIG IEEE 802.3 XMT managed object class supports only the 'enabled' and 'disabled' values of the operational state attribute. The 'enabled' value indicates that the underlying IEEE 802.3 XMT resource is available for use, and the 'disabled' value indicates that the underlying IEEE 802.3 XMT resource is not available for use.

The NMSIG IEEE 802.3 XMT managed object class supports both the 'locked' and 'unlocked' values of the administrative state attribute. Unlocking the administrative state serves to enable transmit on the underlying IEEE 802.3 XMT resource.

The definitive description of the counter attributes, their operation and precedence is specified in the [IEEE Doc X].

The NMSIG IEEE 802.3 XMT managed object class supports several threshold attributes; all are associated with the generation of a CommunicationAlarm notification.

The CreateInfo field of the ObjectCreation notification shall contain all the attributes of the created IEEE 802.3 XMT instance, including those inherited from the nmsig-equipment managed object class.

The DeleteInfo field of the ObjectDeletion notification shall contain all the attributes of the deleted IEEE 802.3 XMT instance, including those inherited from the nmsig-equipment managed object class.

Attributes that are subject to the AttributeChange notification are: nmsig-collisionPDUsThreshold, nmsig-lateCollisionsThreshold, nmsig-PDUsAbortedExcessColThreshold, nmsig-carrierSenseErrorThreshold, nmsig-internalMACXmtErrorThreshold, nmsig-excessiveDeferralThreshold, HealthState.

Attributes that are subject to the StateChange notification are: AdministrativeState and OperationalState.

The counterWrap notification is emitted when any of the counter attributes wrap.

A.4.11 NMSIG LAN MAC Bridge

A.4.11.1 NMSIG LAN MAC Bridge Definition

```
nmsig-LAN-MAC-Bridge    MANAGED OBJECT CLASS
    DERIVED FROM  {nmsig-equipment}
      CHARACACTERIZED BY
        BEHAVIOUR DEFINITIONS  lAN-MAC-Bridge-behaviour
        ATTRIBUTES   nmsig-packetLossRate      GET,
                  nmsig-packetLossRateThreshold  GET-REPLACE

    NOTIFICATIONS    CommunicationAlarm

REGISTERED AS  {obj-class}
```

A.4.11.2 NMSIG LAN MAC Bridge Behaviour

lAN-MAC-Bridge-behaviour BEHAVIOUR

DEFINED AS

A LAN MAC bridge is a device which interconnects two or more MAC domains. A MAC domain is an instance of a MAC algorithm (e.g., a Collision Domain or a Token Domain).

The LAN MAC bridge contains two or more MAC ports each associated with a MAC Domain and operating at layer two of the OSI Model. The function of the LAN MAC bridge is to forward frames from any one MAC Domain to one or more of the other MAC domains. This managed object class represents the LAN MAC bridge device. The definition of this managed object class is based upon the IEEE 802.1 D specification.

The NMSIG LAN MAC bridge managed object class supports only the 'enabled' and 'disabled' values of the operational state attribute. The 'enabled' value indicates that the underlying LAN MAC bridge resource is available for use, and the 'disabled' value indicates that the underlying LAN MAC bridge resource is not available for use.

The CreateInfo field of the ObjectCreation notification shall contain all the attributes of the created LAN MAC Bridge instance.

The DeleteInfo field of the ObjectDeletion notification shall contain all the attributes of the deleted LAN MAC Bridge instance.

Attributes, additioal to those that have been inherited from Equipment, that are subject to the AttributeChange notification are: nmsig-packetLossRateThreshold.

A.4.12 NMSIG MAC Port

A.4.12.1 NMSIG MAC Port Definition

```
nmsig-MAC-Port   MANAGED OBJECT CLASS
   DERIVED FROM  {top}
   CHARACTERIZED BY
      BEHAVIOUR DEFINITIONS  mAC-Port-behaviour
      ATTRIBUTES   nmsig-MAC-PortId   GET,
                nmsig-MAC-PortInNonUCastPktsCounter  GET,
                nmsig-MAC-PortOutNonUCastPktsCounter  GET,
                nmsig-MAC-PortInUCastPktsCounter  GET,
                nmsig-MAC-PortOutUCastPktsCounter  GET,
                nmsig-MAC-PortOutDelayDiscPktsCounter GET,
                nmsig-MAC-PortOutQLen   GET,
                nmsig-MAC-PortInOctetRate   GET,
                nmsig-MAC-PortInOctetRateThreshold  GET-REPLACE,
                AdministrativeState  GET-REPLACE,
                OperationalState  GET,
                HealthState  GET,
                nmsig-broadcastForwardingState  GET-REPLACE,
                nmsig-multicastForwardingState  GET-REPLACE

   NOTIFICATIONS    ObjectCreationUnConfirmed,
                ObjectDeletionUnConfirmed,
                AttributeChangeUnConfirmed,
                StateChangeUnConfirmed,
                nmsig-counterWrapUnConfirmed,
                CommunicationAlarmUnConfirmed

REGISTERED AS  {obj-class}
```

A.4.12.2 NMSIG MAC Port Behaviour

mAC-Port-behaviour BEHAVIOUR

DEFINED AS

This managed object class represents a MAC Port. A MAC Port is contained in a LAN MAC Bridge. It provides the physical connection to a MAC Domain.

The NMSIG MAC Port managed object class supports only the 'enabled' and 'disabled' values of the operational state attribute. The 'enabled' value indicates that the underlying MAC Port resource is available for use, and the 'disabled' value indicates that the underlying MAC port resource is not available for use.

The CreateInfo field of the ObjectCreation notification shall contain all the attributes of the created MAC Port instance.

The DeleteInfo field of the ObjectDeletion notification shall contain all the attributes of the deleted MAC Port instance.

Attributes that are subject to the AttributeChange notification are: HealthState, nmsig-MAC-PortInOctetsRateThreshold, nmsig-broadcastForwardingState and nmsig-multicastForwardingState.

Attributes that are subject to the StateChange notification are: AdministrativeState and OperationalState.

The counterWrap notification is emitted when any of the counter attributes wrap.

A.4.13 NMSIG Network

A.4.13.1 NMSIG Network Definition

```
nmsig-network   MANAGED OBJECT CLASS

   DERIVED FROM  {top}
     CHARACTERIZED BY
     BEHAVIOUR DEFINITIONS  network-behaviour
     ATTRIBUTES   nmsig-networkId   GET,
                nmsig-networkPurpose  GET,
                nmsig-userFriendlyLabel  GET-REPLACE

     NOTIFICATIONS  ObjectCreationUnConfirmed,
                  ObjectDeletionUnConfirmed,
                  AttributeChangeUnConfirmed

REGISTERED AS  {obj-class}
```

A.4.13.2 NMSIG Network Behaviour

network-behaviour BEHAVIOUR

DEFINED AS

The NMSIG Network managed object class represents a collection of connecting and interconnected resources (logical and physical) capable of exchanging information. A network may be contained in another network, thereby creating a superior/subordinate relationship.

The CreateInfo field of the ObjectCreation notification shall contain all the attributes of the created network instnace.

The DeleteInfo field of the ObjectDeletion notification shall contain all the attributes of the deleted network instance.

Attributes that are subject to the AttributeChange notification are: nmsig-networkPurpose, nmsig-userFriendlyLabel

A.4.14 NMSIG Processing Entity

A.4.14.1 NMSIG Processing Entity Definition

```
nmsig-processingEntity   MANAGED OBJECT CLASS

    DERIVED FROM   {nmsig-equipment}
      CHARACTERIZED BY
        BEHAVIOUR DEFINITIONS  processingEntity-behaviour
        ATTRIBUTES       nmsig-cPU-Type  GET,
                    nmsig-memorySize  GET,
                    nmsig-osInfo   GET,
                    nmsig-entityUpTime  GET

        OPERATIONS      DELETE

        NOTIFICATIONS   ProcessingAlarmUnConfirmed

REGISTERED AS  {obj-class}
```

A.4.14.2 NMSIG Processing Entity Behaviour

processingEntity-behaviour BEHAVIOUR

DEFINED AS

The NMSIG processing entity managed object class represents the physical portion of the computer system that performs the processing function. A processing entity may be composed of such components as arithmetic logic units (ALUs) registers for processing memory, limited storage often in the form of Random Access Memory (RAM), and various other types of memory used in the processing function. It does not include components such as disk drives, data bases, etc.

Some processing entities may have input/output channels, particularly when hardware is shared between elements of the processing entity. In other cases, the input/output may be viewed as components of a superior object, e.g. a computer system, or even shared among several computer systems.

The NMSIG processing entity managed object class supports all the values of the administrative state. It supports only the enabled and disabled values of the operational state. An instance of the NMSIG Processing Entity managed object class must be created before any of its subordinates are created.

The CreateInfo field of the ObjectCreation notification shall contain all the attributes of the created processing entity instance.

The DeleteInfo field of the ObjectDeletion notification shall contain all the attributes of the deleted processing entity instance.

Attributes, additional to those inherited from Equipment, that are subject to the AttributeChange notification are: nmsig-cPU-Type, nmsig-memorySize, nmsig-osInfo

A.4.15 NMSIG Root

A.4.15.1 NMSIG Root Definition

```
nmsig-root       MANAGED OBJECT CLASS

   DERIVED FROM   top
   CHARACTERIZED BY
      BEHAVIOUR DEFINITIONS  root-behaviour

REGISTERED AS {obj-class}
```

A.4.15.2 NMSIG Root Behaviour

```
root-behaviour   BEHAVIOUR

    DEFINED AS
```

This managed object class is used to represent the most superior object instance in the containment tree. The purpose of this managed object class is to serve as the common point from which all instances of managed object classes are named.

A single instance of this managed object class is always present in every system, with a distinguished name that is a null sequence (i.e. a SEQUENCE OF with a length of zero).

A.4.16 NMSIG Transport Connection

A.4.16.1 NMSIG Transport Connection Definition

```
nmsig-transportConnection          MANAGED OBJECT CLASS
   DERIVED FROM             {top}
   CHARACTERIZED BY
      BEHAVIOUR DEFINITIONS  transportConnection-behaviour
      ATTRIBUTES          nmsig-transportConnectionId  GET,
                      nmsig-localTransportConnectionEndpoint GET,
                     nmsig-remoteTransportConnectionEndpoint GET,
                       nmsig-transportConnectionReference  GET,
                       nmsig-localNetworkAddress  GET,
                       nmsig-remoteNetworkaddress  GET,
                       nmsig-inactivityTimeout  GET,
                       nmsig-maxPDuSize  GET,
                       PDUsSentCounter  GET,
                       PDUsReceivedCounter  GET,
                       OctetsSentCounter  GET,
                       OctetsReceivedCounter  GET,
                          Peer GET

      OPERATIONS          DELETE   deletes contained objects

      NOTIFICATIONS        ObjectCreationUnConfirmed,
                       ObjectDeletionUnConfirmed,
                       RelationshipChangeUnConfirmed,
                       nmsig-counterWrapUnConfirmed

      PACKAGE       nmsig-transportConnectionRetransmission                PRESENT
IF transport protocol supports retransmission

REGISTERED AS  {obj-class}
```

A.4.16.2 NMSIG Transport Connection Behaviour

transportConnection-behaviour BEHAVIOUR

DEFINED AS

The managed object class nmsig-transportConnection represents an active transport connection (e.g., an OSI transport connection or a TCP connection). A transport connection is established and used by two peer connection oriented transport protocol layer entities for the purpose of transferring data. A connection oriented transport protocol layer entity may support multiple transport connections.

This managed object class represents a "generic" view of a transport connection. It does not concern itself with the details of specific transport protocols like ISO TP or TCP. Transport connections that are tied to a specific protocol can be defined as its subclasses; in fact their definitions are being progressed within various standards' bodies. The purpose of defining this managed object class, however, is to provide a common base that will facilitate the high level management of similar but slightly differing resources.

The expected real effect of the DELETE operation when applied to an instance of the NMSIG transport connection managed object class is that the underlying transport connection resource is aborted.

The CreateInfo field of the ObjectCreation notification shall contain all the attributes of the created transport connection instance.

The DeleteInfo field of the ObjectDeletion notification shall contain all the attributes of the created transport connection instance. In addition it shall also contain a 'cause' parameter defined as follows:

```
cause  ::=  SEQUENCE {
                 INTEGER  (unknown (0),
                          user    (1),
                          provider (2)),
                 INTEGER  (unknown (0),
                          local   (1),
                          remote  (2)),
                 INTEGER  (unknown (0),
                          excessiveIdle (1),
                          excessiveRtx  (2))
                   }
```

The counterWrap notification is emitted when any of the counter attributes wrap.

The RelationshipChange notification is emitted whenever the peer attribute changes in value.

A.4.16.3 NMSIG Transport Connection Retransmission Package

```
nmsig-transportConnectionRetransmission  CONDITIONAL PACKAGE
      BEHAVIOUR DEFINITIONS  transportConnectionRetransmission-              behaviour
      ATTRIBUTES  nmsig-maxRetransmissions  GET,
               nmsig-retransmissionTimerInitialValue GET,
               PDUsRetransmittedErrorCounter  GET,
               PDUsRetransmittedRate  GET,
               PDUsRetransmittedRateThreshold  GET-REPLACE,
               nmsig-octetsRetransmitted  GET

      NOTIFICATIONS   AttributeChange
                  CommunicationAlarmUnConfirmed

REGISTERED AS  {package}
transportConnectionRetransmission-behaviour  BEHAVIOUR
```

DEFINED AS

This package reflects the retransmitting capability of the underlying transport protocol resource.

Attributes that are subject to the AttributeChange notification are: PDUsRetransmittedRateThreshold.

A.4.17 NMSIG Transport Connection Profile

A.4.17.1 NMSIG Transport Connection Profile Definition

```
nmsig-transportConnectionProfile   MANAGED OBJECT CLASS
  DERIVED FROM   {top}
    CHARACTERIZED BY
     BEHAVIOUR DEFINITIONS  trasnportConnectionProfile-behaviour
       ATTRIBUTES   nmsig-transportConnectionProfileId  GET,
                  nmsig-inactivityTimeout  GET-REPLACE,
                nmsig-maxTPDuSize  GET-REPLACE

     OPERATIONS   CREATE,
                DELETE

     NOTIFICATIONS  ObjectCreation
                 ObjectDeletion
                 AttributeChange

REGISTERED AS  {obj-class}
```

A.4.17.2 NMSIG Transport Connection Profile Behaviour

transportConnectionProfile-behaviour BEHAVIOUR

DEFINED AS

This managed object class represents the collection of characteristic attributes which supply default and initially advertised attribute values to be used by instances of the NMSIG Transport Connection managed object class when they are created. There can be only one instance of the NMSIG Transport Connection Profile managed object class for each instance of the NMSIG CO Transport Protocol Layer Entity managed object class.

The CreateInfo field of the ObjectCreation notification shall contain all the attributes of the created transport connection profile instance.

The DeleteInfo field of the ObjectDeletion notification shall contain all the attributes of the deleted transport connection profile instance.

Attributes that are subject to the AttributeChange notification are: nmsig-inactivityTimeout, nmsig-maxTPDuSize.

A.4.18 NMSIG Transport Connection Retransmisslon Profile

A.4.18.1 NMSIG Transport Connection Retransmisslon Profile Definition

```
nmsig-transportConnectionRetransmissionProfile   MANAGED OBJECT CLASS
  DERIVED FROM   nmsig-transportConnectionProfile
    CHARACTERIZED BY
      BEHAVIOUR DEFINITIONS  transportConnectionProfile-behaviour
        ATTRIBUTES  nmsig-maxRetransmissions  GET-REPLACE,
                 nmsig-retransmissionTimerInitialValue GET-REPLACE

REGISTERED AS  {obj-class}
```

A.4.18.2 NMSIG Transport Connection Retransmission Profile Behaviour

transportConnectionRetransmissionProfile-behaviour BEHAVIOUR

DEFINED AS

This managed object class represents the collection of characteristic attributes which supply default and initially advertised attribute values to be used by instances of the NMSIG Transport Connection managed object class that support retransmission, when they are created. There can be only one instance of the NMSIG Transport Connection Retransmission Profile managed object class for each instance of the NMSIG CO Transport Protocol Layer Entity managed object class.

Attributes, additional to those inherited from the transport connection profile managed object class, that are subject to the AttributeChange notification are : nmsig-maxRetransmissions, nmsig-retransmissionTimerInitialValue

A.4.19 Top

This managed object class represents the root of the inheritance tree.

Refer to [ISO Doc x] for the definition of this managed object class.

A.5 NAME BINDINGS

This clause provides definitions of NAME BINDINGS for the managed object classes defined by the OSI MIB Working Group. NAME BINDINGs for managed object classes defined by other groups can be found in the document referenced under the managed object class definition in section 3.

A.5.1 Event Forwarding Discriminator Name Bindings

```
EventForwardingDiscriminator-nb-1   NAME BINDING

EventForwardingDiscriminator   IS NAMED BY   nmsig-agent
        WITH ATTRIBUTE   DiscriminatorId

REGISTERED AS   {nmsig-nb}
```

A.5.2 NMSIG Agent Name Bindings

```
nmsig-agent-nb-1   NAME BINDING

nmsig-agent IS NAMED BY nmsig-root
       WITH ATTRIBUTE  nmsig-agentId

REGISTERED AS    {nmsig-nb}
```

A.5.3 NMSIG Computer System Name Bindings

```
nmsig-computerSystem-nb-1  NAME BINDING

nmsig-computerSystem IS NAMED BY nmsig-network
   WITH ATTRIBUTE  nmsig-systemId

REGISTERED AS    {nmsig-nb}


nmsig-computerSystem-nb-2  NAME BINDING

nmsig-computerSystem IS NAMED BY nmsig-computerSystem
   WITH ATTRIBUTE  nmsig-systemId

REGISTERED AS    {nmsig-nb}


nmsig-computerSystem-nb-3  NAME BINDING

nmsig-computerSystem IS NAMED BY nmsig-root
```

WITH ATTRIBUTE nmsig-systemId

REGISTERED AS {nmsig-nb}

A.5.4 NMSIG CO Transport Protocol Layer Entity Name Bindings

nmsig-coTransportProtocolLayerEntity-nb-1 NAME BINDING

nmsig-coTransportProtocolLayerEntity IS NAMED BY nmsig-computerSystem
WITH ATTRIBUTE nmsig-coTransportEntityId

REGISTERED AS {nmsig-nb}

nmsig-coTransportProtocolLayerEntity-nb-2 NAME BINDING

nmsig-coTransportProtocolLayerEntity IS NAMED BY nmsig-equipment
WITH ATTRIBUTE nmsig-coTransportEntityId

REGISTERED AS {nmsig-nb}

A.5.5 NMSIG CL Network Protocol Layer Entity Name Bindings

nmsig-clNetworkProtocolLayerEntity-nb-1 NAME BINDING

nmsig-clNetworkProtocolLayerEntity IS NAMED BY nmsig-computerSystem
WITH ATTRIBUTE nmsig-clNetworkProtocolEntityId

REGISTERED AS {nmsig-nb}

nmsig-clNetworkProtocolLayerEntity-nb-2 NAME BINDING

nmsig-clNetworkProtocolLayerEntity IS NAMED BY nmsig-equipment
WITH ATTRIBUTE nmsig-clNetworkProtocolEntityId

REGISTERED AS {nmsig-nb}

A.5.6 NMSIG Equipment Name Bindings

nmsig-equipment-nb-1 NAME BINDING

nmsig-equipment IS NAMED BY nmsig-equipment
WITH ATTRIBUTE nmsig-equipmentId

REGISTERED AS {nmsig-nb}

```
nmsig-equipment-nb-2   NAME BINDING

nmsig-equipment IS NAMED BY  nmsig-network
   WITH ATTRIBUTE  nmsig-equipmentId
REGISTERED AS  {nmsig-nb}


nmsig-equipment-nb-3   NAME BINDING

nmsig-equipment IS NAMED BY  nmsig-root
   WITH ATTRIBUTE  nmsig-equipmentId

REGISTERED AS  {nmsig-nb}
```

A.5.7 NMSIG IEEE 802.3 Name Bindings

```
nmsig-IEEE-802.3-nb-1  NAME BINDING

nmsig-IEEE-802.3 IS NAMED BY nmsig-network
  WITH ATTRIBUTE  nmsig-IEEE-802.3Id

REGISTERED AS {nmsig-nb}


nmsig-IEEE-802.3-nb-2  NAME BINDING

nmsig-IEEE-802.3 IS NAMED BY nmsig-computerSystem
  WITH ATTRIBUTE  nmsig-IEEE-802.3Id

REGISTERED AS {nmsig-nb}
```

A.5.8 NMSIG IEEE 802.3 RCV Name Bindings

```
nmsig-IEEE-802.3-RCV-nb-1  NAME BINDING

nmsig-IEEE-802.3-RCV IS NAMED BY nmsig-IEEE-802.3
  WITH ATTRIBUTE  nmsig-IEEE-802.3-RCVId

REGISTERED AS {nmsig-nb}
```

A.5.9 NMSIG IEEE 802.3 XMT Name Bindings

```
nmsig-IEEE-802.3-XMT-nb-1  NAME BINDING

nmsig-IEEE-802.3-XMT IS NAMED BY nmsig-IEEE-802.3
  WITH ATTRIBUTE  nmsig-IEEE-802.3-XMTId
```

REGISTERED AS {nmsig-nb}

A.5.10 NMSIG LAN MAC Bridge Name Bindings

```
nmsig-LAN-MAC-Bridge-nb-1  NAME BINDING
nmsig-LAN-MAC-Bridge IS NAMED BY nmsig-network
  WITH ATTRIBUTE  nmsig-equipmentId

REGISTERED AS {nmsig-nb}
```

A.5.11 NMSIG MAC Port Name Bindings

```
nmsig-MAC-Port-nb-1  NAME BINDING

nmsig-MAC-Port  IS NAMED BY  nmsig-LAN-MAC-Bridge
  WITH ATTRIBUTE  nmsig-MAC-PortId

REGISTERED AS  {nmsig-nb}
```

A.5.12 NMSIG Network Name Bindings

```
nmsig-network-nb-1    NAME BINDING

nmsig-network  IS NAMED BY  nmsig-network
   WITH ATTRIBUTE  nmsig-networkId

REGISTERED AS  {nmsig-nb}


nmsig-network-nb-2    NAME BINDING

nmsig-network  IS NAMED BY  nmsig-root
   WITH ATTRIBUTE  nmsig-networkId

REGISTERED AS  {nmsig-nb}
```

A.5.13 NMSIG Processing Entity Name Bindings

```
nmsig-processingEntity-nb-1 NAME BINDING

nmsig-processingEntity IS NAMED BY nmsig-computerSystem
  WITH ATTRIBUTE  nmsig-equipmentId

REGISTERED AS {nmsig-nb}
```

A.5.14 NMSIG Transport Connection Name Bindings

```
nmsig-transportConnection-nb-1   NAME BINDING

nmsig-transportConnection
   IS NAMED BY  nmsig-coTransportProtocolLayerEntity
   WITH ATTRIBUTE   nmsig-transportConnectionId

REGISTERED AS  {nmsig-nb}
```

A.5.15 NMSIG Transport Connection Profile Name Bindings

```
nmsig-transportConnectionProfile-nb-1   NAME BINDING

nmsig-transportConnectionProfile
   IS NAMED BY nmsig-coTransportProtocolLayerEntity
   WITH ATTRIBUTE   nmsig-transportConnectionProfileId

REGISTERED AS  {nmsig-nb}
```

A.5.16 NMSIG Transport Connection Retransmission Profile Name Bindings

```
nmsig-transportConnectionRetransmissionProfile-nb-1   NAME BINDING

nmsig-transportConnectionRetransmissionProfile
   IS NAMED BY nmsig-coTransportProtocolLayerEntity
   WITH ATTRIBUTE   nmsig-transportConnectionProfileId

REGISTERED AS  {nmsig-nb}
```

A.6 ATTRIBUTES

This clause provides definitions of attributes contained in the managed object classes defined by the OSI MIB Working Group. Attribute definitions for managed object classes defined by other groups can be found in the document referenced under the managed object class definition in section 3.

A.6.1 Administrative State

Refer to [ISO Doc x] for the definition of this attribute.

A.6.2 Begin Time

Refer to [ISO Doc x] for the definition of this attribute.

A.6.3 Destination Address

Refer to [ISO Doc x] for the definition of this attribute.

A.6.4 Discriminator Construct

Refer to [ISO Doc x] for the definition of this attribute.

A.6.5 Discriminator Id

Refer to [ISO Doc x] for the definition of this attribute.

A.6.6 End Time

Refer to [ISO Doc x] for the definition of this attribute.

A.6.7 Health State

Refer to [ISO Doc x] for the definition of this attribute.

A.6.8 Incoming Connection Reject Error Counter

Refer to [ISO Doc X] for the definition of this attribute.

A.6.9 Incoming Connection Requests Counter

Refer to [ISO Doc X] for the definition of this attribute.

A.6.10 Incoming Disconnect Error Counter

Refer to [ISO Doc X] for the definition of this attribute.

A.6.11 Incoming Temporal Error Counter

Refer to [ISO Doc X] for the definition of this attribute.

A.6.12 NMSIG Alignment Error Threshold

```
nmsig-alignmentErrorThreshold  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   GaugeThreshold
        MATCHES FOR  Equality
        BEHAVIOUR  alignmentErrorThreshold-behaviour

REGISTERED AS     {nmsig-attr}

GaugeThreshold ::= {as defined in ISO Doc X}

alignmentErrorThreshold-behaviour  BEHAVIOUR

    DEFINED AS
```

This attribute specifies a threshold which is applied against the alignment error rate. The alignment error rate is defined as the number of PDUs received with alignment errors divided by the total number of PDUs received. A communication alarm notification is emitted when the alignment error rate exceeds the threshold value.

A.6.13 NMSIG Agent Id

```
nmsig-agentId  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX PrintableString
        BEHAVIOUR  agentId-behaviour
REGISTERED AS     {nmsig-attr}

agentId-behaviour  BEHAVIOUR

      DEFINED AS
```

This is the distinguishing attribute for the managed object class NMSIG Agent.

A.6.14 NMSIG Broadcast Forwarding State

```
nmsig-broadcastForwardingState  ATTRIBUTE
```

```
        WITH ATTRIBUTE SYNTAX  State
        MATCHES FOR  Equality
        BEHAVIOUR  broadcastForwardingState-behaviour

REGISTERED AS   {nmsig-attr}

State ::=  ENUMERATED {off (0),
                       on (1)}


broadcastForwardingState-behaviour   BEHAVIOUR

    DEFINED AS

        This attribute specifies whether broadcast PDUs are being forwarded.
```

A.6.15 NMSIG Broadcast PDUs Rcv Counter

```
nmsig-broadcastPDUsRcvCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
        BEHAVIOUR  broadcastPDUsRcvCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

broadcastPDUsRcvCounter-behaviour  BEHAVIOUR

    DEFINED AS

        This attribute specifies the number of broadcast PDUs received ok by the underlying
        NMSIG IEEE 802.3 RCV resource.
```

A.6.16 NMSIG Broadcast PDUs Xmt Counter

```
nmsig-broadcastPDUsXmtOkCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
        BEHAVIOUR  broadcastPDUsXmtOkCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

broadcastPDUsXmtOkCounter-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of broadcast PDUs which were transmitted ok by the underlying NMSIG IEEE 802.3 XMT resource.

A.6.17 NMSIG Carrier Sense Errors Counter

```
nmsig-carrierSenseErrorsCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
        BEHAVIOUR  carrierSenseErrorsCounter-behaviour

REGISTERED AS      {nmsig-attr}

Count ::= {as defined in ISO Doc X}

carrierSenseErrorsCounter-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of carrier sense Errors which were detected by the underlying NMSIG IEEE 802.3 XMT resource.

A.6.18 NMSIG Carrier Sense Errors Threshold

```
nmsig-carrierSenseErrorsThreshold  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   GaugeThreshold
        MATCHES FOR  Equality
        BEHAVIOUR  carrierSenseErrorsThreshold-behaviour

REGISTERED AS      {nmsig-attr}

GaugeThreshold ::=  {as defined in ISO Doc X}

carrierSenseErrorsThreshold-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies a threshold which is applied against the carrier sense error rate. The carrier sense error rate is defined as the carrier sense errors detected per second. A communication alarm notification is emitted when the carrier sense error rate exceeds the threshold value.

A.6.19 NMSIG Checksum TPDUs Discarded Counter

```
nmsig-checksumTPDUsDiscardedCounter   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Count
        MATCHES FOR   Equality, Ordering
        BEHAVIOUR  checksumTPDUsDiscardedCounter-behaviour
```

```
REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}
checksumTPDUsDiscardedCounter-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of TPDUs discarded due to a bad checksum.

A.6.20 NMSIG Collision PDUs Counter

```
nmsig-collisionPDUsCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  Count
        MATCHES FOR   Equality, Ordering
            BEHAVIOUR  collisionPDUsCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

collisionPDUsCounter-behaviour  BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of collision PDUs which were detected by the underlying NMSIG IEEE 802.3 XMT resource.

A.6.21 NMSIG Collision PDUs Threshold

```
nmsig-collisionPDUsThreshold  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   GaugeThreshold
        MATCHES FOR  Equality
            BEHAVIOUR  collisionPDUsThreshold-behaviour

REGISTERED AS     {nmsig-attr}

GaugeThreshold ::=  {as defined in ISO Doc X}

collisionPDUsThreshold-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies a threshold which is applied against the collision PDU rate. The collision PDU rate is defined as the collision PDUs detected per second. A communication alarm notification is emitted when the collision PDU rate exceeds the threshold value.

A.6.22 NMSIG CO Transport Protocol Layer Entity Id

```
nmsig-coTransportEntityId  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX PrintableString
        MATCHES FOR  Equality
            BEHAVIOUR  coTransportEntityId-behaviour

REGISTERED AS    {nmsig-attr}

coTransportEntityID-behaviour  BEHAVIOUR

    DEFINED AS
```

This is the distinguishing attribute for the managed object class connection oriented transport protocol layer entity.

A.6.23 NMSIG Connectionless Network Protocol Layer Entity Id

```
nmsig-clNetworkProtocolLayerEntityId  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX PrintableString
        MATCHES FOR  Equality
            BEHAVIOUR  clNetworkProtocolLayerEntityId-behaviour

REGISTERED AS     {nmsig-attr}

clNetworkProtocolLayerEntityId-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute is the distinguishing attribute for the managed object class clNetworkProtocolLayerEntity.

A.6.24 NMSIG Contact Names

```
nmsig-contactNames   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   AnyName
        MATCHES FOR  Set Comparison, Set Intersection
            BEHAVIOUR  contactNames-behaviour

REGISTERED AS    {nmsig-attr}

AnyName ::=  SET OF (CHOICE {dn   DistinguishedName,
                             ps   PrintableString})

contactNames-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies name(s) of one or more contacts.

A.6.25 NMSIG CPU Type

```
nmsig-cPU-Type   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  PrintableString
      MATCHES FOR  Equality
          BEHAVIOUR  cPU-Type-behaviour

REGISTERED AS   {nmsig-attr}

cPU-Type-behaviour  BEHAVIOUR

    DEFINED AS

          This attribute specifies the type of the Central Processor Unit in a processing entity.
```

A.6.26 NMSIG Enable Promiscuous State

```
nmsig-enablePromiscuousState  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  State
        MATCHES FOR  Equality
            BEHAVIOUR  enablePromiscuousState-behaviour

REGISTERED AS   {nmsig-attr}

State ::=  ENUMERATED {off (0),
                on (1)}

enablePromiscuousState-behaviour  BEHAVIOUR

    DEFINED AS

          This attribute specifies whether the IEEE 802.3 RCV is operating in promiscuous
          mode.
```

A.6.27 NMSIG Entity Up Time

```
nmsig-entityUpTime    ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  INTEGER
        MATCHES FOR   Equality, Ordering
            BEHAVIOUR  entityUpTime-behaviour

REGISTERED AS     {nmsig-attr}

entityUpTime-behaviour  BEHAVIOUR
```

DEFINED AS

This attribute specifies the time interval (in seconds) that has elapsed since the time that the value of the entity's operational state changed from 'disabled' to some other value, or since the time that the entity was created into a non disabled state.

A.6.28 NMSIG Equipment Id

```
nmsig-equipmentId   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  PrintableString
        MATCHES FOR  Equality
             BEHAVIOUR  equipmentId-behaviour

REGISTERED AS     {nmsig-attr}

equipmentId-behaviour  BEHAVIOUR
```

DEFINED AS

This is the distinguishing attribute of the NMSIG equipment managed object class.

A.6.29 NMSIG Equipment Purpose

```
nmsig-equipmentPurpose  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX PrintableString
        MATCHES FOR  Equality
             BEHAVIOUR  equipmentPurpose-behaviour

REGISTERED AS     {nmsig-attr}

equipmentPurpose-behaviour  BEHAVIOUR
```

DEFINED AS

This attribute specifies what the equipment is used for (e.g., switching, processing, etc.).

A.6.30 NMSIG Excessive Deferral Threshold

```
nmsig-excessiveDeferralThreshold  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   GaugeThreshold
        MATCHES FOR  Equality
             BEHAVIOUR  excessiveDeferralThreshold-behaviour

REGISTERED AS     {nmsig-attr}

GaugeThreshold ::=  {as defined in ISO Doc X}
```

```
excessiveDeferralThreshold-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies a threshold which is applied against the excessive deferral rate. The excessive deferral rate is defined as the number of excessive deferral PDUs transmitted per second. A communication alarm notification is emitted when the excessive deferral rate exceeds the threshold value.

A.6.31 NMSIG FCS Error Threshold

```
nmsig-FCSErrorThreshold  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  GaugeThreshold
        MATCHES FOR  Equality
             BEHAVIOUR  fCSErrorThreshold-behaviour

REGISTERED AS     {nmsig-attr}

GaugeThreshold ::=  {as defined by ISO Doc X}

fCSErrorThreshold-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies a threshold which is applied against the FCS error rate. The FCS error rate is defined as the number of PDUs received which had FCS errors divided by the total number of PDUs received. A communication alarm notification is emitted when the FCS error rate exceeds the threshold value.

A.6.32 NMSIG IEEE 802.3 Id

```
nmsig-IEEE-802.3Id   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   PrintableString
        MATCHES FOR  Equality
             BEHAVIOUR  iEEE-802.3Id-behaviour

REGISTERED AS     {nmsig-attr}

iEEE-802.3Id-behaviour  BEHAVIOUR

    DEFINED AS
```

This attribute is the distinguishing attribute of the NMSIG IEEE 802.3 managed object class.

A.6.33 NMSIG IEEE 802.3 RCV Id

```
nmsig-IEEE-802.3-RCVId   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   PrintableString
        MATCHES FOR  Equality
             BEHAVIOUR  iEEE-802.3-RCVId-behaviour

REGISTERED AS     {nmsig-attr}
iEEE-802.3-RCVId-behaviour  BEHAVIOUR
```

DEFINED AS

This attribute is the distinguishing attribute of the NMSIG IEEE 802.3 RCV managed object class.

A.6.34 NMSIG IEEE 802.3 State

```
nmsig-IEEE-802.3State  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX EnableState
        MATCHES FOR  Equality
             BEHAVIOUR  iEEE-802.3State-behaviour

REGISTERED AS   {nmsig-attr}

EnableState ::=  ENUMERATED {disable (0),
                          enable (1)}

iEEE-802.3State-behaviour  BEHAVIOUR
```

DEFINED AS

This attribute specifies whether the IEEE 802.3 object is enabled or not. The 'enabled' and 'disabled' values of this attribute correspond to the 'enabled' and 'disabled' values of the OperationalState attribute. (This attribute was introduced as a GET-REPLACE attribute which can be used by management to enable or disable the underlying IEEE 802.3 resource.)

A.6.35 NMSIG IEEE 802.3 XMT Id

```
nmsig-IEEE-802.3-XMTId   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   PrintableString
        MATCHES FOR  Equality
             BEHAVIOUR  iEEE-802.3-XMTId-behaviour

REGISTERED AS     {nmsig-attr}

iEEE-802.3-XMTId-behaviour  BEHAVIOUR
```

DEFINED AS

This attribute is the distinguishing attribute of the NMSIG IEEE 802.3 XMT managed object class.

A.6.36 NMSIG In-Range Threshold

```
nmsig-inRangeThreshold  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   GaugeThreshold
        MATCHES FOR  Equality
             BEHAVIOUR  inRangeTheshold-behaviour

REGISTERED AS     {nmsig-attr}

GaugeThreshold ::=  {as defined by ISO Doc X}

inRangeTheshold-behaviour   BEHAVIOUR
```

DEFINED AS

This attribute specifies a threshold which is applied against the in-range length error rate. The in-range length error rate is defined as the number of PDUs received that had in-range length errors divided by the total number of PDUs received. A communication alarm notification with the specified severity is emitted when the in-range length error rate exceeds the threshold value.

A.6.37 NMSIG Inactivity Timeout

```
nmsig-inactivityTimeout          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX INTEGER
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  inactivityTimeout-behaviour

REGISTERED AS      {nmsig-attr}

inactivityTimeout-behaviour  BEHAVIOUR
```

DEFINED AS

This attribute specifies the maximum amount of time (in 1/100ths of a second) that the transport connection can remain up when there is no activity (i.e. data flow) on it. A value of 0 for this attribute indicates that an inactivity timeout is not supported on the transport connection.

A.6.38 NMSIG Incoming Normal Disconnect Counter

```
nmsig-incomingNormalDisconnectCounter        ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  incomingNormalDisconnectCounter-behaviour

REGISTERED AS      {nmsig-attr}

Count ::= {as defined in ISO Doc X}

incomingNormalDisconnectCounter-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of incoming transport connections that were disconnected due to normal reasons.

A.6.39 NMSIG Internal MAC Rcv Error Threshold

```
nmsig-internalMACRcvErrorThreshold  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX    GaugeThreshold
        MATCHES FOR  Equality
             BEHAVIOUR  internalMACRcvErrorThreshold

REGISTERED AS      {nmsig-attr}

GaugeThreshold ::=  {as defined in ISO Doc X}

internalMACRcvErrorThreshold   BEHAVIOUR

    DEFINED AS
```

This attribute specifies a threshold which is applied against the internal MAC receive error rate. This rate is defined as the number of internal MAC receive errors detected per second. A communication alarm notification is emitted when the internal MAC receive error rate exceeds the threshold value.

A.6.40 NMSIG Internal MAC Rcv Error Counter

```
nmsig-internalMACRcvErrorCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX    Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  internalMACRcvErrorCounter

REGISTERED AS      {nmsig-attr}

Count ::= {as defined in ISO Doc X}
```

internalMACRcvErrorCounter BEHAVIOUR

DEFINED AS

This attribute specifies the number of internal MAC receive errors detected by the underlying NMSIG IEEE 802.3 RCV resource.

A.6.41 NMSIG Internal MAC Xmt Error Threshold

```
nmsig-internalMACXmtErrorThreshold  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   GaugeThreshold
        MATCHES FOR  Equality
             BEHAVIOUR  internalMACXmtErrorThreshold-behaviour

REGISTERED AS     {nmsig-attr}

GaugeThreshold ::=  {as defined in ISO Doc X}

internalMACXmtErrorThreshold-behaviour   BEHAVIOUR
```

DEFINED AS

This attribute specifies a threshold which is applied against the internal MAC transmit error rate. This rate is defined as the number of internal MAC transmit errors detected per second. A communication alarm notification is emitted when the internal MAC transmit error rate exceeds the threshold value.

A.6.42 NMSIG Late Collision Counter

```
nmsig-lateCollisionsCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  lateCollisionsCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

lateCollisionsCounter-behaviour   BEHAVIOUR
```

DEFINED AS

This attribute specifies the number of late collisions detected by the underlying NMSIG IEEE 802.3 XMT resource.

A.6.43 NMSIG Late Collisions Threshold

```
nmsig-lateCollisionThreshold  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   GaugeThreshold
        MATCHES FOR  Equality
             BEHAVIOUR  lateCollisionThreshold-behaviour

REGISTERED AS      {nmsig-attr}

GaugeThreshold ::=  {as defined in ISO Doc X}

lateCollisionThreshold-behaviour   BEHAVIOUR
```

DEFINED AS

This attribute specifies a threshold which is applied against the late collision rate. The late collision rate is defined as the number of late collision PDUs transmitted divided by the total number of PDUs transmitted. A communication alarm notification is emitted when the late collision rate exceeds the threshold value.

A.6.44 NMSIG Local Network Address

```
nmsig-localNetworkAddress       ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  OCTET STRING
        MATCHES FOR   Equality
             BEHAVIOUR  localNetworkAddress-behaviour

REGISTERED AS       {nmsig-attr}

localNetworkAddress-behaviour   BEHAVIOUR
```

DEFINED AS

This attribute identifies the local network address of the transport connection (e.g., the local IP address for TCP or the local NSAP for OSI TP).

A.6.45 NMSIG Local Network Addresses

```
nmsig-localNetworkAddresses       ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  LocalNetworkAddresses
        MATCHES FOR  Set Comparison, Set Intersection
             BEHAVIOUR  localNetworkAddresses-behaviour

REGISTERED AS       {nmsig-attr}

LocalNetworkAddresses  ::=  SET OF OCTET STRING

localNetworkAddresses-behaviour   BEHAVIOUR
```

DEFINED AS

This attribute specifies a set of local network addresses supported by a network protocol layer entity.

A.6.46 NMSIG Local Transport Addresses

```
nmsig-localTransportAddresses    ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  TransportAddresses
        MATCHES FOR  Set Comparison, Set Intersection
             BEHAVIOUR  localTransportAddresses-behaviour

REGISTERED AS  {nmsig-attr}

TransportAddresses ::=  SET OF SEQUENCE {
transportConnectionEndpoint OCTET STRING,
                    networkAddress  OCTET STRING}

localTransportAddresses-behaviour   BEHAVIOUR
```

DEFINED AS

This attribute specifies the set of transport addresses that a connection oriented transport protocol layer entity provides to its users. A transport address consists of a transport connection endpoint and a network address.

A.6.47 NMSIG Local Transport Connection Endpoint

```
nmsig-localTransportConnectionEndpoint      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  OCTET STRING
        MATCHES FOR  Equality
             BEHAVIOUR  localTransportConnectionEndpoint-behaviour

REGISTERED AS      {nmsig-attr}

localTransportConnectionEndpoint-behaviour   BEHAVIOUR
```

DEFINED AS

This attribute identifies the local transport connection endpoint (e.g., it represents the source port for TCP or the local t-selector for OSI TP).

A.6.48 NMSIG Location Name

```
nmsig-locationName   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   AnyName
        MATCHES FOR  Equality
            BEHAVIOUR  locationName-behaviour

REGISTERED AS     {nmsig-attr}

AnyName ::=  CHOICE {dn   DistinguishedName,
                     ps   PrintableString}

locationName-behaviour   BEHAVIOUR

   DEFINED AS
```

This attribute specifies the name of a location (e.g., Hilo Hawaii USA).

A.6.49 NMSIG MAC Address

```
nmsig-macAddress   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   OctetString
        MATCHES FOR  Equality
            BEHAVIOUR  macAddress-behaviour

REGISTERED AS     {nmsig-attr}

macAddress-behaviour   BEHAVIOUR

   DEFINED AS
```

This attribute specifies a MAC address.

A.6.50 NMSIG MAC Port Id

```
nmsig-MAC-PortId   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   PrintableString
        MATCHES FOR  Equality
            BEHAVIOUR  mAC-PortID-behaviour

REGISTERED AS     {nmsig-attr}

mAC-PortID-behaviour   BEHAVIOUR

   DEFINED AS
```

This attribute is the distinguishing attribute of the NMSIG MAC Port managed object class.

A.6.51 NMSIG MAC Port In Non-Unicast Packets Counter

```
nmsig-MAC-PortInNonUCastPktsCounter   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  mAC-PortInNonUCastPktsCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

mAC-PortInNonUCastPktsCounter-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of non-unicast (i.e., subnet broadcast or subnet multicast) packets that were received at the MAC port.

A.6.52 NMSIG MAC Port In Octet Rate

```
nmsig-MAC-PortInOctetRate  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  Gauge
        MATCHES FOR  Equality, Ordering
             BEHAVIOUR  mAC-PortInOctetRate

REGISTERED AS     {nmsig-attr}

Gauge ::=  {as defined in ISO doc X}

mAC-PortInOctetRate   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the rate of octets arriving at the MAC port per second.

A.6.53 NMSIG MAC Port In Octet Rate Threshold

```
nmsig-MAC-PortInOctetRateThreshold   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  GaugeThreshold
        MATCHES FOR  Equality
             BEHAVIOUR  mAC-PortInOctetRateThreshold

REGISTERED AS     {nmsig-attr}

GaugeThreshold ::=  {as defined in ISO Doc X}

mAC-PortInOctetRateThreshold   BEHAVIOUR

    DEFINED AS
```

This attribute specifies a threshold which is applied against the in octet rate. A communication alarm notification is emitted when the in octet rate exceeds the threshold value.

A.6.54 NMSIG MAC Port In Unicast Packets Counter

```
nmsig-MAC-PortInUCastPktsCounter   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  mAC-PortInUCastPktsCounter

REGISTERED AS     {nmsig-attr}
Count ::= {as defined in ISO Doc X}

mAC-PortInUCastPktsCounter   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of unicast packets received on the MAC port.

A.6.55 NMSIG MAC Port Out Delay Discarded Packets Counter

```
nmsig-MAC-PortOutDelayDiscPktsCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  mAC-PortOutDelayDiscPktsCounter

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

mAC-PortOutDelayDiscPktsCounter   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of packets that were discarded at the MAC port because the maximum packet hold time was exceeded.

A.6.56 NMSIG MAC Port Out Non-Unicast Packets

```
nmsig-MAC-PortOutNonUCastPktsCounter   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  mAC-PortOutNonUCastPktsCounter

REGISTERED AS     {nmsig-attr}
```

Count ::= {as defined in ISO Doc X}

mAC-PortOutNonUCastPktsCounter BEHAVIOUR

DEFINED AS

This attribute specifies the number of non-unicast packets that were sent out of the MAC port.

A.6.57 NMSIG MAC Port Out Queue Length

```
nmsig-MAC-PortOutQLen  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  INTEGER
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  mAC-PortOutQLen
```

REGISTERED AS {nmsig-attr}

mAC-PortOutQLen BEHAVIOUR

DEFINED AS

This attribute specifies the number of packets that are currently queued for output on the MAC port.

A.6.58 NMSIG MAC Port Out Unicast Packets Counter

```
nmsig-MAC-PortOutUCastPktsCounter   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  mAC-PortOutUCastPktsCounter
```

REGISTERED AS {nmsig-attr}

Count ::= {as defined in ISO Doc X}

mAC-PortOutUCastPktsCounter BEHAVIOUR

DEFINED AS

This attribute specifies the number of unicast packets that were sent out of this MAC port.

A.6.59 NMSIG Max Connections

```
nmsig-maxConnections          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX INTEGER
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  maxConnections-behaviour

REGISTERED AS      {nmsig-attr}

maxConnections-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the maximum number of simultaneously open transport connections allowed by the transport protocol layer entity.

A.6.60 NMSIG Max Retransmissions

```
nmsig-maxRetransmissions  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  INTEGER
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  maxRetransmissions-behaviour

REGISTERED AS      {nmsig-attr}

maxRetransmissions-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the maximum number of times a TPDU is to be retransmitted before the transport connection is aborted.

A.6.61 NMSIG Max TPDU Size

```
nmsig-maxTPDUSize             ATTRIBUTE
        WITH ATTRIBUTE SYNTAX INTEGER
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  maxTPDUSize-behaviour

REGISTERED AS       {nmsig-attr}

maxTPDUSize-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the maximum TPDU size (in terms of octets) that can be handled by the local transport protocol layer entity.

A.6.62 NMSIG Memory Size

```
nmsig-memorySize   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  INTEGER
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  memorySize-behaviour

REGISTERED AS   {nmsig-attr}

memorySize-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the amount of random access memory (in kilobytes) that is owned by a processing entity. (1 Kilobyte = 1024 bytes.)

A.6.63 NMSIG Multicast Address List

```
nmsig-multicastAddressList   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   AddressList
        MATCHES FOR  Set Comparison, Set Intersection
             BEHAVIOUR  multicastAddressList-behaviour

REGISTERED AS     {nmsig-attr}

AddressList ::=  SET OF OCTET STRING

multicastAddressList-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies a multicast address list.

A.6.64 NMSIG Multicast Forwarding State

```
nmsig-multicastForwardingState  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  State
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  multicastForwardingState-behaviour

REGISTERED AS   {nmsig-attr}

State ::=  ENUMERATED {off (0),
                   on (1)}

multicastForwardingState-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies whether multicast PDUs are being forwarded.

A.6.65 NMSIG Multicast PDUs Rcv Counter

```
nmsig-multicastPDUsRcvCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  multicastPDUsRcvCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

multicastPDUsRcvCounter-behaviour  BEHAVIOUR

    DEFINED AS

          This attribute specifies the number of multicast PDUs received ok by the underlying
          NMSIG IEEE 802.3 RCV resource.
```

A.6.66 NMSIG Multicast PDUs Xmt Counter

```
nmsig-multicastPDUsXmtCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  multicastPDUsXmtCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

multicastPDUsXmtCounter-behaviour   BEHAVIOUR

    DEFINED AS

          This attribute specifies the number of multicast PDUs which were transmitted ok by
          the underlying NMSIG IEEE 802.3 XMT resource.
```

A.6.67 NMSIG Multicast Receive State

```
nmsig-multicastReceiveState  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  State
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  multicastReceiveState-behaviour

REGISTERED AS   {nmsig-attr}
```

```
State ::=  ENUMERATED {off (0),
                      on (1)}

multicastReceiveState-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the multicast receive state of the underlying NMSIG IEEE 802.3 RCV resource.

A.6.68 NMSIG Multiple Collision PDU Counter

```
nmsig-multipleCollisionPDUCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  multipleCollisionPDUCounter

REGISTERED AS      {nmsig-attr}
Count ::= {as defined in ISO Doc X}

multipleCollisionPDUCounter   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of multiple collision PDUs detected by the underlying NMSIG IEEE 802.3 XMT resource.

A.6.69 NMSIG Network Entity Type

```
nmsig-networkEntityType ATTRIBUTE
      WITH ATTRIBUTE SYNTAX NetworkEntityType
      MATCHES FOR  Equality
           BEHAVIOUR  networkEntityType-behaviour

REGISTERED AS      {nmsig-attr}

NetworkEntityType ::=  INTEGER {other(0),
                       oSI CLNP (1),
                       internet IP (2)} (0..256)

networkEntityType-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the type of the network protocol layer entity.

A.6.70 NMSIG Network Id

```
nmsig-networkId   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  PrintableString
        MATCHES FOR  Equality
            BEHAVIOUR  networkId-behaviour

REGISTERED AS     {nmsig-attr}

networkId-behaviour   BEHAVIOUR

    DEFINED AS
```

This is the distinguishing attribute of the NMSIG network managed object class.

A.6.71 NMSIG Network Purpose

```
nmsig-networkPurpose   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  PrintableString
        MATCHES FOR  Equality
            BEHAVIOUR  networkPurpose-behaviour

REGISTERED AS     {nmsig-attr}

networkPurpose-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies what the network is used for (e.g., manufacturing control, airline reservation, etc.)

A.6.72 NMSIG NPDU Time To Live

```
nmsig-nPDUTimeToLive        ATTRIBUTE
        WITH ATTRIBUTE SYNTAX INTEGER
        MATCHES FOR   Equality, Ordering
            BEHAVIOUR  nPDUTimeToLive-behaviour

REGISTERED AS     {nmsig-attr}

nPDUTimeToLive-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the maximum amount of time (in units of 10 ms) that an NPDU can exist in the network. This attribute is used to limit the lifetime of NPDUs during unstable network situations.

A.6.73 NMSIG Octets Retransmitted Error Counter

```
nmsig-octetsRetransmittedErrorCounter   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  octetsRetransmitterErrorCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

octetsRetransmitterErrorCounter-behaviour   BEHAVIOUR

    DEFINED AS

          This attribute specifies the total number of octets that were retransmitted.
```

A.6.74 NMSIG OS Info

```
nmsig-osInfo   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  OsInfo
        MATCHES FOR Set Comparison, Set Intersection
             BEHAVIOUR  osInfo-behaviour

REGISTERED AS   {nmsig-attr}

OsInfo  ::=  SET  OF (CHOICE {osName  [0] DistingishedName,
                        osSpec  [1] ProductInfo})

ProductInfo ::=  SEQUENCE {manufacturer  PrintableString,
                      productLabel  PrintableString,
                      release       PrintableString,
                      serialNumber  PrintableString}

osInfo-behaviour   BEHAVIOUR

    DEFINED AS

          This attribute specifies the names and releases of operating systems supported by
          the processing entity
```

A.6.75 NMSIG Open Connections

```
nmsig-openConnections          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX INTEGER
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  openConnections-behaviour
```

```
REGISTERED AS     {nmsig-attr}

openConnections-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of currently established transport connections.

A.6.76 NMSIG Out-Range Threshold

```
nmsig-outRangeThreshold  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   GaugeThreshold
        MATCHES FOR  Equality
             BEHAVIOUR  outRangeThreshold-behaviour

REGISTERED AS     {nmsig-attr}

GaugeThreshold ::=  {as defined in ISO Doc X}

outRangeThreshold-behaviour   BEHAVIOUR
    DEFINED AS
```

This attribute specifies a threshold which is applied against the out-range length error rate. This rate is defined as the number of PDUs received with out-range length errors divided by the total number of PDUs received. A communication alarm notification is emitted when the out-range length error rate exceeds the threshold value.

A.6.77 NMSIG Outgoing Normal Disconnect Counter

```
nmsig-outgoingNormalDisconnectCounter        ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  outgoingNormalDisconnectCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

outgoingNormalDisconnectCounter-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of outgoing transport connections that were disconnected due to normal reasons.

A.6.78 NMSIG Packet Loss Rate

```
nmsig-packetLossRate   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  Gauge
        MATCHES FOR  Equality, Ordering
             BEHAVIOUR  packetLossRate-behaviour

REGISTERED AS     {nmsig-attr}

Gauge ::=  {as defined in ISO Doc X}

packetLossRate-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the rate of packets dropped per second.

A.6.79 NMSIG Packet Loss Rate Threshold

```
nmsig-packetLossRateThreshold   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  GaugeThreshold
        MATCHES FOR  Equality
             BEHAVIOUR  packetLossRateThreshold

REGISTERED AS     {nmsig-attr}

GaugeThreshold ::=  {as defined in ISO Doc X}

packetLossRateThreshold    BEHAVIOUR

    DEFINED AS
```

This attribute specifies a threshold which is applied against the packet loss rate. A communication alarm notification is emitted when the packet loss rate exceeds the threshold value.

A.6.80 NMSIG PDU Too Long Threshold

```
nmsig-PDUTooLongThreshold  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   GaugeThreshold
        MATCHES FOR  Equality
             BEHAVIOUR  pDUTooLongThreshold-behaviour

REGISTERED AS     {nmsig-attr}

GaugeThreshold ::=  {as defined by ISO Doc X}
```

```
pDUTooLongThreshold-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies a threshold which is applied against the "PDU too long" error rate. The PDU too long error rate is defined as the number of PDUs received that were too long divided by the total number of PDUs received. A communication alarm notification is emitted when the "PDU too long" error rate exceeds the threshold value.

A.6.81 NMSIG PDUs Aborted Excessive Collisions Counter

```
nmsig-PDUsAbortedExcessiveCollisionsCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  pDUsAbortedExcessiveCollisionsCounter-behaviour

REGISTERED AS    {nmsig-attr}

Count ::= {as defined in ISO Doc X}

pDUsAbortedExcessiveCollisionsCounter-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of PDUs which were aborted by the underlying NMSIG IEEE 802.3 XMT resource due to excessive collisions.

A.6.82 NMSIG PDUs Aborted Excessive Collisions Threshold

```
nmsig-PDUsAbortedExcessColThreshold  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   GaugeThreshold
        MATCHES FOR  Equality
             BEHAVIOUR  pDUsAbortedExcessColThreshold-behaviour

REGISTERED AS     {nmsig-attr}

GaugeThreshold ::=  {as defined in ISO Doc X}

pDUsAbortedExcessColThreshold-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies a threshold which is applied against the PDUs aborted due to excessive collision rate. This rate is defined as the number of PDUs aborted due to excessive collision divided by the total number of PDUs transmitted. A communication alarm notification is emitted when the PDUs aborted due to excessive collision rate exceeds the threshold value.

A.6.83 NMSIG PDUs Alignment Error Counter

```
nmsig-PDUsAlignmentErrorCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  pDUsAlignmentErrorCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

pDUsAlignmentErrorCounter-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of PDUs with an alignment error detected by the underlying NMSIG IEEE 802.3 RCV resource.

A.6.84 NMSIG PDUs Excessive Deferral Counter

```
nmsig-PDUsExcessiveDeferralCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  pDUsExcessiveDeferralCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

pDUsExcessiveDeferralCounter-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of PDUs for which the underlying NMSIG IEEE 802.3 XMT resource detected excessive deferral.

A.6.85 NMSIG PDUs Discarded Counter

```
nmsig-PDUsDiscardedCounter          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  pDUsDiscardedCounter-behaviour

REGISTERED AS     {nmsig-attr}
```

```
Count ::= {as defined in ISO Doc X}

pDUsDiscardedCounter-behaviour   BEHAVIOUR

    DEFINED AS

          This attribute specifies the number of PDUs that were discarded by a network
          protocol layer entity.
```

A.6.86 NMSIG PDUs FCS Error Counter

```
nmsig-PDUsFCSErrorCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  pDUsFCSErrorCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

pDUsFCSErrorCounter-behaviour   BEHAVIOUR

    DEFINED AS

          This attribute specifies the number of PDUs with an FCS error detected by the
          underlying NMSIG IEEE 802.3 RCV resource.
```

A.6.87 NMSIG PDUs Forwarded Counter

```
    nmsig-PDUsForwardedCounter         ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  pDUsForwardedCounter-behaviour

    REGISTERED AS     {nmsig-attr}

    Count ::= {as defined in ISO Doc X}

      pDUsForwardedCounter-behaviour   BEHAVIOUR
    DEFINED AS
          This attribute specifies the number of PDUs forwarded by a network protocol layer
          entity.
```

A.6.88 NMSIG PDUs In-Range Length Error Counter

```
nmsig-PDUsInRangeLengthErrorCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  pDUsInRangeLengthErrorCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

pDUsInRangeLengthErrorCounter-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of PDUs with an in-range length error detected by the underlying NMSIG IEEE 802.3 RCV resource.

A.6.89 NMSIG PDUs Lost Internal MAC Xmt Error Counter

```
nmsig-PDUsLostInternalMACXmtErrorCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  pDUsLostInternalMACXmtErrorCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

pDUsLostInternalMACXmtErrorCounter-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of PDUs which were lost by the underlying NMSIG IEEE 802.3 XMT resource due to an internal MAC transmit error.

A.6.90 NMSIG PDUs Out-Range Error Counter

```
nmsig-PDUsOutRangeLengthErrorCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  pDUsOutRangeLengthErrorCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

pDUsOutRangeLengthErrorCounter-behaviour   BEHAVIOUR
```

DEFINED AS

This attribute specifies the number of PDUs with an out-range length error detected by the underlying NMSIG IEEE 802.3 RCV resource.

A.6.91 NMSIG PDUs Reassemble Fail Counter

```
nmsig-PDUsReasmblFailCounter          ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  pDUsReasmblFailCounter-behaviour

REGISTERED AS      {nmsig-attr}

Count ::= {as defined in ISO Doc X}

pDUsReasmblFailCounter-behaviour   BEHAVIOUR
```

DEFINED AS

This attribute specifies the number of PDUs that could not be reassembled successfully by a network protocol layer entity.

A.6.92 NMSIG PDUs Reassembled OK Counter

```
nmsig-PDUsReasmbldOKCounter           ATTRIBUTE
        WITH ATTRIBUTE SYNTAX Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  pDUsReasmbldOKCounter-behaviour

REGISTERED AS      {nmsig-attr}

Count ::= {as defined in ISO Doc X}

pDUsReasmbldOKCounter-behaviour   BEHAVIOUR
```

DEFINED AS

This attribute specifies the number of PDUs that were reassembled successfully by a network protocol layer entity.

A.6.93 NMSIG PDUs Too Long Error Counter

```
nmsig-PDUsTooLongErrorCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
```

```
            BEHAVIOUR  pDUsTooLongErrorCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

pDUsTooLongErrorCounter-behaviour   BEHAVIOUR

    DEFINED AS

          This attribute specifies the number of PDUs with a "PDU too long" error detected by
          the underlying NMSIG IEEE 802.3 RCV resource.
```

A.6.94 NMSIG Peripheral Names

```
nmsig-peripheralNames  ATTRIBUTE
       WITH ATTRIBUTE SYNTAX  PeripheralNames
       MATCHES FOR Set Comparison, Set Intersection
           BEHAVIOUR  peripheralNames-behaviour

REGISTERED AS     {nmsig-attr}

PeripheralNames ::=  SET OF AnyName

AnyName ::=  CHOICE {dn   DistinguishedName,
                     ps   PrintableString}

peripheralNames-behaviour   BEHAVIOUR

    DEFINED AS
          This attribute specifies the names of auxiliary devices.
```

A.6.95 NMSIG Product Info

```
nmsig-productInfo     ATTRIBUTE
       WITH ATTRIBUTE SYNTAX   ProductInfo
       MATCHES FOR  Equality
            BEHAVIOUR  productInfo-behaviour

REGISTERED AS     {nmsig-attr}

ProductInfo ::=  SEQUENCE {manufacturer  PrintableString,
                     productLabel  PrintableString,
                     release       PrintableString,
                     serialNumber  PrintableString}

productInfo-behaviour   BEHAVIOUR
```

```
    DEFINED AS
```

This attribute specifies product information of the underlying resource.

A.6.96 NMSIG Promiscuous Receive State

```
nmsig-promiscuousReceiveState  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  State
        MATCHES FOR   Equality, Ordering
            BEHAVIOUR  promiscuousReceiveState-behaviour

REGISTERED AS   {nmsig-attr}

State ::=  ENUMERATED {off (0),
                on (1)}

promiscuousReceiveState-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the promiscuous receive state of the underlying NMSIG IEEE 802.3 RCV resource.

A.6.97 NMSIG Remote Network Address

```
nmsig-remoteNetworkAddress       ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  OCTET STRING
        MATCHES FOR  Equality
            BEHAVIOUR  remoteNetworkAddress-behaviour

REGISTERED AS       {nmsig-attr}

remoteNetworkAddress-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute identifies the remote network address of the transport connection (e.g., it represents the remote IP address for TCP or the remote NSAP for OSI TP).

A.6.98 NMSIG Remote Transport Connection Endpoint

```
nmsig-remoteTransportConnectionEndpoint      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  OCTET STRING
        MATCHES FOR  Equality
            BEHAVIOUR  remoteTransportConnectionEndpoint-behaviour
```

```
REGISTERED AS      {nmsig-attr}

remoteTransportConnectionEndpoint-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute identifies the remote transport connection endpoint (It represents the destination port for TCP or the remote t-selector for OSI TP).

A.6.99 NMSIG Retransmission Timer Initial Value

```
nmsig-retransmissionTimerInitialValue  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  INTEGER
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  retransmissionTimerInitialValue-behaviour

REGISTERED AS     {nmsig-attr}

retransmissionTimerInitialValue-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the initial value (in 1/100ths of a second) of the retransmission timer used by a transport connection.

A.6.100 NMSIG Single Collision PDUs Counter

```
nmsig-singleCollisionPDUsCounter  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   Count
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  singleCollisionPDUsCounter-behaviour

REGISTERED AS     {nmsig-attr}

Count ::= {as defined in ISO Doc X}

singleCollisionPDUsCounter-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the number of single collision PDUs detected by the underlying NMSIG IEEE 802.3 XMT resource.

A.6.101 NMSIG Source Address Last Alignment Error PDU

```
nmsig-sourceAddrLastAlignmentErrorPDU  ATTRIBUTE
```

```
        WITH ATTRIBUTE SYNTAX   OCTET STRING
        MATCHES FOR  Equality
            BEHAVIOUR  sourceAddrLastAlignmentErrorPDU-behaviour

REGISTERED AS     {nmsig-attr}

sourceAddrLastAlignmentErrorPDU-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the source address of the last alignment error PDU detected by the underlying NMSIG IEEE 802.3 RCV resource.

A.6.102 NMSIG Source Address Last FCS Error PDU

```
nmsig-sourceAddrLastFCSErrorPDU  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   OCTET STRING
        MATCHES FOR  Equality
            BEHAVIOUR  sourceAddrLastFCSErrorPDU-behaviour

REGISTERED AS     {nmsig-attr}

sourceAddrLastFCSErrorPDU-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the source address of the last FCS error PDU detected by the underlying NMSIG IEEE 802.3 RCV resource.

A.6.103 NMSIG Source Address Last In-Range Length Error PDU

```
nmsig-sourceAddrLastInRangeLengthErrorPDU  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   OCTET STRING
        MATCHES FOR  Equality
            BEHAVIOUR  sourceAddrLastInRangeLengthErrorPDU-behaviour

REGISTERED AS     {nmsig-attr}

sourceAddrLastInRangeLengthErrorPDU-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the source address of the last in-range length error PDU detected by the underlying NMSIG IEEE 802.3 RCV resource.

A.6.104 NMSIG Source Address Last Out-Range Length Error PDU

```
nmsig-sourceAddrLastOutRangeLengthErrorPDU  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   OCTET STRING
        MATCHES FOR  Equality
             BEHAVIOUR  sourceAddrLastOutRangeLengthErrorPDU-behaviour

REGISTERED AS     {nmsig-attr}

sourceAddrLastOutRangeLengthErrorPDU-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the source address of the last out-range length error PDU detected by the underlying NMSIG IEEE 802.3 RCV resource.

A.6.105 NMSIG Source Address Last Too Long Error PDU

```
nmsig-sourceAddrLastTooLongErrorPDU  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   OCTET STRING
        MATCHES FOR  Equality
             BEHAVIOUR  sourceAddrLastTooLongErrorPDU

REGISTERED AS     {nmsig-attr}

sourceAddrLastOutRangeLengthErrorPDU-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the source address of the last "PDU too long" error PDU detected by the underlying NMSIG IEEE 802.3 RCV resource.

A.6.106 NMSIG System Id

```
nmsig-systemId  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX PrintableString
        MATCHES FOR  Equality
             BEHAVIOUR  systemId-behaviour

REGISTERED AS     {nmsig-attr}

systemId-behaviour   BEHAVIOUR

    DEFINED AS
```

This is the distinguishing attribute of the NMSIG computer system managed object class.

A.6.107 NMSIG System Time

```
nmsig-systemTime  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX GeneralizedTime
        MATCHES FOR   Equality, Ordering
             BEHAVIOUR  systemTime-behaviour

REGISTERED AS      {nmsig-attr}

systemTime-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the current time clocked at the computer system.

A.6.108 NMSIG Transport Connection Id

```
nmsig-transportConnectionId  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX PrintableString
        MATCHES FOR  Equality
             BEHAVIOUR  transportConnectionId-behaviour

REGISTERED AS     {nmsig-attr}

transportConnectionId-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute is the distinguishing attribute for the managed object class transportConnection.

A.6.109 NMSIG Transport Connection Profile Id

```
nmsig-transportConnectionProfileId  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX PrintableString
        MATCHES FOR  Equality
             BEHAVIOUR  transportConnectionProfileId-behaviour

REGISTERED AS     {nmsig-attr}

transportConnectionProfileId-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute is the distinguishing attribute for the managed object class nmsig-transportConnectionProfile.

A.6.110 NMSIG Transport Connection Reference

```
nmsig-transportConnectionReference      ATTRIBUTE
        WITH ATTRIBUTE SYNTAX  OCTET STRING
        MATCHES FOR  Equality
             BEHAVIOUR  transportConnectionReference-behaviour

REGISTERED AS      {nmsig-attr}

transportConnectionReference-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute identifies the local transport connection reference that is established by the two transport connection endpoints (e.g., the local socket number for TCP or the local connection reference for OSI).

A.6.111 NMSIG Transport Entity Type

```
nmsig-transportEntityType ATTRIBUTE
      WITH ATTRIBUTE SYNTAX TransportEntityType
      MATCHES FOR  Equality
           BEHAVIOUR  transportEntityType-behaviour

REGISTERED AS      {nmsig-attr}

TransportEntityType ::=  INTEGER {other(0),
                         oSI TP (1),
                           tCP (2),
                           sNA (3)} (0..256)

transportEntityType-behaviour   BEHAVIOUR

    DEFINED AS
```

This attribute specifies the type of the transport protocol layer entity.

A.6.112 NMSIG User Friendly Label

```
nmsig-userFriendlyLabel  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX PrintableString
        MATCHES FOR  Equality
             BEHAVIOUR  userFriendlyLabel-behaviour

REGISTERED AS      {nmsig-attr}

userFriendlyLabel-behaviour   BEHAVIOUR
```

```
    DEFINED AS

        This attribute specifies a user friendly name.
```

A.6.113 NMSIG Vendor Name

```
nmsig-vendorName   ATTRIBUTE
        WITH ATTRIBUTE SYNTAX   AnyName
        MATCHES FOR  Equality
            BEHAVIOUR  vendorName-behaviour

REGISTERED AS     {nmsig-attr}

AnyName ::=  CHOICE {dn   DistinguishedName,
                     ps   PrintableString}

vendorName-behaviour   BEHAVIOUR

    DEFINED AS

        This attribute specifies the name of a vendor.
```

A.6.114 NMSIG Xmt State

```
nmsig-XmtState  ATTRIBUTE
        WITH ATTRIBUTE SYNTAX EnableState
        MATCHES FOR  Equality, Ordering
            BEHAVIOUR  xmtState-behaviour

REGISTERED AS   {nmsig-attr}

EnableState ::=  ENUMERATED {disable (0),
                             enable (1)}

xmtState-behaviour   BEHAVIOUR
    DEFINED AS
```

This attribute specifies whether the transmitting capability of the unserlying IEEE 802.3 resource is enabled or not. The 'enabled' and 'disabled' values of this attribute correspond to the 'enabled' and 'disabled' values of the OperationalState attribute of the IEEE 802.3 XMT managed object class. (This attribute was introduced as a GET-REPLACE attribute which can be used by management to enable or disable the transmitting capability of the underlying IEEE 802.3 resource.)

A.6.115 Object Class

Refer to [ISO Doc x] for the definition of this attribute.

A.6.116 Octets Received Counter

Refer to [ISO Doc X] for the definition of this attribute.

A.6.117 Octets Sent Counter

Refer to [ISO Doc X] for the definition of this attribute.

A.6.118 Operational State

Refer to [ISO Doc x] for the definition of this attribute.

A.6.119 Outgoing Connection Reject Error Counter

Refer to [ISO Doc X] for the definition of this attribute.

A.6.120 Outgoing Connections Request Counter

Refer to [ISO Doc X] for the definition of this attribute.

A.6.121 Outgoing Disconnect Error Counter

Refer to [ISO Doc X] for the definition of this attribute.

A.6.122 Outgoing Temporal Error Counter

Refer to [ISO Doc X] for the definition of this attribute.

A.6.123 PDUs Received Counter

Refer to [ISO Doc X] for the definition of this attribute.

A.6.124 PDUs Sent Counter

Refer to [ISO Doc X] for the definition of this attribute.

A.6.125 PDUs Retransmitted Error Counter

Refer to [ISO Doc X] for the definition of this attribute.

A.7 ACTIONS

This clause provides definitions of actions supported by managed object classes defined by the OSI MIB Working Group. Action definitions for managed object classes defined by other groups can be found in the document referenced under the managed object class definition in section 3.

A.7.1 NMSIG Execute Self Test

```
nmsig-executeSelfTest  ACTION
          ACTION BEHAVIOUR  selfTestBehaviour
          WITH RESULT SYNTAX  SelfTestResult

REGISTERED AS   {nmsig-action}

selfTestBehaviour  BEHAVIOUR

          DEFINED AS
```

This action requests a self test sequence be executed on the referenced managed object instance. This action is always confirmed. The confirmation contains the operational state of the managed object under test following test completion, and optionally indicates the success or failure of the self test.

```
          SelfTestResult ::= SEQUENCE
                      {
                        operationalState   OperationalState,
                        testResult         BOOLEAN OPTIONAL
                       }
```

A.8 NOTIFICATIONS

This clause provides definitions of notifications emitted by managed object classes defined by the OSI MIB Working Group. Notification definitions for managed object classes defined by other groups can be found in the document referenced under the managed object class definition in section 3.

A.8.1 Attribute Change Unconfirmed

Refer to [ISO Doc x] for the definition of this notification.

A.8.2 Communication Alarm Unconfirmed

Refer to [ISO Doc x] for the definition of this notification.

A.8.3 Equipment Alarm Unconfirmed

Refer to [ISO Doc x] for the definition of this notification.

A.8.4 Environmental Alarm Unconfirmed

Refer to [ISO Doc x] for the definition of this notification.

A.8.5 NMSIG Counter Wrap Unconfirmed

```
nmsig-counterWrapUnconfirmed   NOTIFICATION
     BEHAVIOUR  counterWrap-behaviour
     WITH DATA SYNTAX  WrapInfo

REGISTERED AS  {notification}

counterWrap-behaviour   BEHAVIOUR

      DEFINED AS

           This notification indicates that a counter has wrapped.

WrapInfo  ::=  Attribute { -- attribute ID and value of counter
                         attribute that wrapped   }
```

A.8.6 Object Creation Unconfirmed

Refer to [ISO Doc x] for the definition of this notification.

A.8.7 Object Deletion Unconfirmed

Refer to [ISO Doc x] for the definition of this notification.

A.8.8 Processing Error Alarm Unconfirmed

Refer to [ISO Doc x] for the definition of this notification.

A.8.9 Relationship Change Unconfirmed

Refer to [ISO Doc x] for the definition of this notification.

A.8.10 State Change Unconfirmed

Refer to [ISO Doc x] for the definition of this notification.

A.9 REFERENCES

This clause lists the names of documents that were referenced in the earlier clauses.

Table of Contents

List of Figures

19 REMOTE DATABASE ACCESS

19.1 INTRODUCTION

Remote Database Access (RDA) specifies the communications service and protocol for accessing the capabilities of a database server from a client application. Figure 19.1 depicts RDA's placement within the application layer and its relationship to supporting OSI protocols:

```
+------------------------------------------+
|                                          |
|      Remote Database Access              |
|                                          |
+---------------+--------------------------+
|               |                          |
|  ASCE         |          TP              |
|               +-----+          +---------+
|                     |          |  CCR    |
+---------------------+----------+---------+
|                                          |
|               Presentation               |
|                                          |
+------------------------------------------+
|                                          |
|                 Session                  |
|                                          |
+------------------------------------------+
```

Figure Figure 19.1. Placement of RDA within the Application Layer..

This is an implementation agreement for RDA developed by the Implementors Workshop sponsored by the U.S. National Institute of Standards and Technology. This document addresses both the RDA generic model, service, and protocol, as well as the SQL Specialization, ISO 9579 parts 1 and 2, respectively. It is the intent of the workshop to expand this agreement to include other parts of 9579 as they are developed.

19.2 SCOPE

This implementation agreement addresses remote database interaction between a database server and a client application. The database server is an open system that provides database storage facilities and supplies database processing services to clients at other open systems.

The RDA communications service provides the protocol for RDA client interaction with an RDA server. The RDA client initiates an RDA dialogue and requests RDA operations to be performed by the RDA server on behalf of a client applications. The RDA server, located within the database server, provides database services to RDA clients.

More specifically, this document describes implementation agreements in the following areas:

1. the RDA generic model, service, and protocol,

2. the RDA SQL Specialization, and

3. SQL language restrictions.

19.3 REFERENCES

The following documents contain provisions which, through reference in this text, constitute provisions of this International Standardized Profile. At the time of publication, the additions indicated were valid. All documents are subject to revision, and parties to agreements based on this International Standardized Profile are warned against automatically applying any more recent additions of the documents listed below, since the nature of references made by ISPs to such documents is that they may be specific to a particular addition. Members of IEC and ISO maintain registers of currently valid International Standards and ISPs, and CCITT maintains published additions of its current recommendations.

ISO 9579-1 Information Processing Systems - Open Systems Interconnection - Remote Database Access - Part 1: Generic Model, Service, and Protocol

ISO 9579-2 Information Processing Systems - Open Systems Interconnection - Remote Database Access - Part 2: SQL Specialization

ISO/IEC/TR10000-1:1990(E) Information Technology - Framework and Taxonomy of International Standardized Profiles - Part 1: Framework

Note: Work on ISO 9579 is ongoing.

19.4 DEFINITIONS

19.5 ABBREVIATIONS

19.6 RDA DIALOGUE STATE MODEL AGREEMENTS

19.7 GENERIC RDA AGREEMENTS

19.7.1 Functional Units

19.7.2 Optional Negotiable Facilities

19.7.2.1 Open/Close Within Transaction

19.7.3 Services

19.7.3.1 R-BeginDialogue

19.7.3.1.1 Optional Parameters

19.7.3.1.2 Parameter Restrictions

19.7.3.2 R-EndDialogue

19.7.3.3 R-Open

19.7.3.4 R-Close

19.7.3.5 R-Execute

19.7.3.6 R-Define

19.7.3.7 R-Invoke

19.7.3.8 R-Drop

19.7.3.9 R-BeginTransaction

19.7.3.10 R-Commit

19.7.3.11 R-Rollback

19.7.3.12 R-Status

19.7.3.13 Cancel

19.8 SQL SPECIALIZATION AGREEMENTS

19.8.1 Functional Units

19.8.2 Optional Facilites

19.8.3 Services

19.8.4 SQL Language Agreements

19.8.4.1 Language/Protocal Mapping

19.8.4.2 Implementor Defined Items

19.8.4.3 SQL Functional Restrictions

19.8.4.4 SQL State and Error Messages

19.8.5 Conformance Requirements

19.8.6 Recommended Practices

Table of Contents

20 Manufacturing Message Specification (MMS)

20.1 INTRODUCTION

This section defines Implementors Agreements based on ISO Manufacturing Message Specification (MMS), as defined in ISO 9506. This International Standard has two parts. Part 1 of the IS defines the Virtual Manufacturing Device (VMD) as well as defining the services, and Part 2 defines the Protocol. Future parts may define companion standards.

MMS, as described in the IS, is based on the following ISO documents: ACSE Service and Protocol (ISO 8649, ISO 8650), Presentation Service and Protocol (ISO 8822, ISO 8823), ASN.1 Abstract Syntax Notation and Basic Encoding Rules (ISO 8824, ISO 8825), and Session Service and Protocol (ISO 8326, ISO 8327). These services and protocols are defined architecturally in the OSI Reference Model (ISO 7498). These Agreements provide detailed guidance for the implementor, and eliminate ambiguities in interpretations.

The agreements can be used over any T-Profile (see ISO DTR 10000) specifying the OSI connection-mode transport service. In addition, these MMS agreements can be used over the Transport profiles used in support of MAP (Manufacturing Automation Protocol) or TOP (Technical and Office Protocols).

20.1.1 References

Application Layer - MMS

ISO/IEC 9506-1: 1990 Industrial automation systems - Manufacturing Message Specification Part 1: Service definition

ISO/IEC 9506-2: 1990 Industrial automation systems - Manufacturing Message Specification Part 2: Protocol specification

20.2 SCOPE AND FIELD OF APPLICATION

There will be a phased grouping of implementation agreements. These agreements will be based on selected subsets of MMS services as defined in ISO 9506-1. Agreements will be defined in phases which will be added as needed.

20.2.1 Phase I Agreements

These agreements will be implementation agreements pertaining to the services as specified as table 20.1.

Table 20.1. Phase I Services

```
Initiate
Conclude
Reject
Abort

Status
GetNameList
Identify
UnsolicitedStatus
GetCapabilityList

InitiateDownloadSequence
DownloadSegment
TerminateDownloadSequence
InitiateUploadSequence
UploadSegment
TerminateUploadSequence
DeleteDomain
GetDomainAttributes

Read
Write
InformationReport
GetVariableAccessAttributes

Input
Output

CreateProgramInvocation
DeleteProgramInvocation
Start
Stop
Resume
Reset
Kill
GetProgramInvocationAttributes
```

20.3 STATUS

20.3.1 Status of Phase 1 Agreements

Phase 1 is in progress.

Editor's Note: Portions of these agreements may be a candidate for stability in September 1990, and should be examined closely.

20.4 ERRATA

None at time of publication.

20.5 SPECIFIC SERVICE AGREEMENTS

20.5.1 Initiate

20.5.1.1 Max Serv Outstanding

An MMS Implementation which intends to conform only with the Client Conformance Requirements for Requester CBBs shall:

1. propose 1 or greater for the value of the Proposed Max Serv Outstanding Called parameter in the Initiate service when initiating the application association (calling).

2. offer 1 or greater for the value of the Negotiated Max Serv Outstanding Calling parameter in the Initiate service when receiving the application association initiation (called).

An MMS Implementation which intends to conform to one or more Server Conformance Requirements for Responder CBBs shall:

1. propose 1 or greater for the value of the Proposed Max Serv Outstanding Calling parameter in the Initiate service when initiating the application association (calling).

2. offer 1 or greater for the value of the Negotiated Max Serv Outstanding Called parameter in the Initiate service when receiving the application association initiation (called).

20.5.1.2 Version Number

The value of the proposed Version Number in the Initiate request and the negotiated Version Number in the Initiate response service primitivesshall be one unless interoperability with DIS based implementations, as described in section 20.6 is desired.

20.5.1.3 Minimum Supported PDU Size

MMS implementations must be able to parse and process 64 octets of MMS pdu as they would be encoded in ASN.1 Basic Encoding Rules. However, it is recommended that 512 be supported.

20.5.1.4 Max Supported PDU Size

The max_mms_pdu_size is defined as the maximum number of octets in an MMS pdu encoded using the negotiated transfer syntax. This size shall apply to all MMS PDU's with the exception of the initiate-Request PDU, initiate-Response PDU, and initiate-Error PDU. The max_mms_pdu_size shall be negotiated during connection initiation using the Local Detail Calling and Local Detail Called parameters of the MMS initiate service.

The semantics of these parameters follows:

Local Detail Calling

> The local detail calling parameter in the initiate request primitive shall specify the max_mms_pdu_size guaranteed to be supported by the calling MMS-user. If the local detailcalling parameter is absent from the request primitive, then the calling MMS-user guarantees support for an unlimited max_mms_pdu_size.
>
> If present in the request or indication primitives, the local_detail_calling parameter shall not be less than 64.

Local Detail Called

> The local detail called parameter in the initiate response shall specify the negotiated max_mms_pdu_size for the application association.

If the local detail calling parameter was omitted in the indication primitive, then the local_detail_called parameter:

1. may be omitted from the response, indicating that the calling MMS-user and the called MMS-user are prepared to support an unbounded max_mms_pdu_size, or,

2. may be specified in the response, indicating a requirement to support the specified value for max_mms_pdu_size.

If the local detail calling parameter was included in the request, then this parameter shall appear in the response and its value shall be less than or equal to the value of the local detail calling parameter of the request.

If present in the response, the local detail called parameter shall not be less than 64.

The negotiated max_mms_pdu_size shall be applied as follows:

> Any received MMSpdu which is less than or equal to the negotiated max_mms_pdu_size shall be properly parsed and processed.
>
> When rejecting an MMS-pdu because it exceeds the negotiated max_mms_pdu_size, an MMS implementation shall use a pdu type of pdu_error and a reject code of invalid_pdu in the resulting reject PDU.

An MMS implementation shall not send an MMSpdu whose size exceeds the negotiated max_mms_pdu_size.

If an MMS implementation is unable to send a service response because the response would exceed the max_mms_pdu_size, then it shall return a Service response (-) with an error class of SERVICE and an error code of OTHER.

20.5.1.5 Negotiation of MMS Abstract Syntaxes

On initiate response, the MMS responder shall not accept more than one presentation context derived from an MMS abstract syntax (in this context, only the core MMS abstract syntax and the Companion Standard defined abstract syntaxes, are considered MMS abstract syntaxes).

20.5.2 Scattered Access

It is strongly recommended that for services which use variable access, a Variable List Name or List of Variable be used instead of Scattered Access.

No implementations shall be required to propose or accept the VSCA Parameter CBB.

20.5.3 Floating Point

It is stongly recommended for services which use floating point types or values, that the MMS choice of floating-point in the Data and Type specification productions be used instead of the real choice.

No implementations shall be required to propose or accept the REAL parameter CBB.

20.5.4 Start Stop,Resume, and Reset Services

A ProgramInvocationState of non-existent shall be returned in a Result(-) when a request to Start, Stop, Resume, or Reset a non-existent Program Invocation is received.

20.5.5 FileName

Restrictions for the use of the type FileName in the MMS Abstract Syntax are specified in section 9.9.1.

20.5.6 Domain Management Agreements

20.5.6.1 List of Capabilities

Only one capability will be described in each Visible String of the SEQUENCE OF.

20.5.6.2 Initiate Download Sequence Service

The List of Capability parameter will follow the limitations of 20.5.6.1.

The syntax and semantics of the capabilities are defined by the Server in the PICS. Any deviation from the defined syntax and semantics is grounds for the Server to return a service error with Error Class = RESOURCE and Error Code = CAPABILITY-UNKNOWN.

20.5.6.3 Download Segment Service

If a negative response to a Download Segment request is received, an MMS server will not send any more Download Segment requests, but will next send either a Terminate Download Sequence request or an Abort request. A client who receives another Download Segment indication should issue either a service error, specifying an Error Class = SERVICE and an Error Code = PRIMITIVES-OUT-OF-SEQUENCE, or an Abort request.

20.5.6.4 Terminate Download Sequence Service

If a Server has not received a response to a Download Segment request with a value of the More Follows parameter = FALSE, it will not issue a Terminate Download Sequence service request unless that request specifies a Discard parameter value of TRUE. If a Client receives a Terminate Download Sequence request in which the Discard parameter is either absent or has a value FALSE, and it has not previously issued a parameter value of More Follows = FALSE in response to a Download Segment request, it shall behave exactly as if it had received a Terminate Download Sequence service request with the Discard parameter = TRUE.

20.5.6.5 Initiate Upload Sequence Service

The List of Capability parameter will follow the limitations of 20.5.6.1.

20.5.6.6 Upload Segment Service

If a Client receives a negative confirm to an Upload Segment request, it will not send any more Upload Segment requests. The next service primitive sent will be either a Terminate Upload Sequence request or an Abort request. A Server who receives another Upload Segment indication should issue either a service error, specifying an Error Class = SERVICE and an Error Code = PRIMITIVES-OUT-OF-SEQUENCE, or an Abort request.

20.5.6.7 Get Domain Attributes Service

The List of Capability parameter will follow the limitations of 20.5.6.1.

20.5.7 Get Capability List Service

The List of Capability parameter will follow the limitations of 20.5.6.1.

20.6 INTEROPERABILITY AGREEMENTS

There is an installed base of realDIS 9506 based implementations. Providing support for application connectivity to both DIS and IS is desired as a migration strategy. These implementation agreements will allow IS based implementations to interoperate with DIS based implementations as described in Appendix A. To achieve this interoperability, the IS implementation shall support all of the agreements in this section.

It was found that the Abstract Syntax name object identifiers of both DIS and IS were identical. There, the use of Version 0 allows differentiation between an IS and a DIS based implementation. The value of zero is a valid value for these parameters in the DIS and not in the IS.

Note: The value of zero is a valid value for these parameters in the DIS and not in the IS.

Tutorial Note:

There are three types of implementations when considering MMS interoperabilty.

IMP-1: An implementation based on DIS 9506 as described in Appendix A.

IMP-2: An implementation based on IS 9506 with no interoperability agreements applied.

IMP-3: An implementation based on IS 9506 which includes the interoperabilty agreements described below.

IMP-1, IMP-2, and IMP-3 can interoperate with each other in all combinations with the exception of the IMP-1 and IMP-2 combination. The remainder of this section describes additional agreements which change an IMP-2 implementation into an IMP-3 implementation.

20.6.1 Calling MMS-user Interoperability Agreements

A calling MMS-user shall be capable of receiving and supporting a negotiatedVersionNumber parameter in the Initiate Service Confirm of zero.

A calling MMS-user which has received a negotiatedVersionNumber parameter in the Initiate Service Confirm of zero shall support the modifications described in section 20.6.3.

A calling MMS-user shall ignore the Application Context Name parameter in the A-Associate Confirm.

A calling MMS-user which has received a negotiatedVersionNumber of zero shall be capable of receiving and supporting an InitiateResponse which has been encoded according to the modifications described in Appendix A, specifically the capability of receiving and supporting a negotiatedParameterCBB containing exactly 7 bits.

20.6.2 Called MMS-user Interoperability Agreements.

A called MMS-user shall be capable of receiving and supporting a proposedVersionNumber parameter in the Initiate Service Indication of zero.

A called MMS-user which has received a proposedVersionNumber parameter in the Initiate Service Indication of 0 shall support the modifications in section 20.6.3.

A called MMS-user shall ignore the Application Context Name parameter in the A-Associate Indication.

A called MMS-user shall be capable of receiving and supporting an InitiateRequest which has been encoded according to the modifications described in Appendix A, specifically the capability of receiving and supporting a proposedParameterCBB containing exactly 7 bits.

20.6.3 General Interoperability Agreements

20.6.3.1 VMD Logical Status

If the current VMD State is SUPPORT-SERVICES-ALLOWED and the association minor version number is zero, then the vmdLogicalStatus parameter shall have a value of state-changes-allowed in a status response or a unsolicitedStatus request.

20.6.3.2

Further agreements are required to complete this section.

20.7 ANNEX A: DIS 9506 MODIFICATIONS REQUIRED FOR INTEROPERABILITY

This annex is an integral part of chapter 20. It documents the modifications to DIS 9506 required to describe implementations for which the IS agreements provide interoperability. This appendix as applied to DIS 9506 is referred to as Version 0.

20.7.1 References

[1] MMS/1 Manufacturing Message Specification - ISO DIS 9506 - Service Definition, December 1987

[2] MMS/2 Manufacturing Message Specification -ISO DIS 9506 - Protocol Specification, December 1987

[3] NBS OSI Implementors Workshop Agreements - December 1987

20.7.2 General

20.7.2.1 Implementation Base

Version 0 is based upon Reference [3] in 20.7.1 as it applies to MMS.

20.7.2.2 Rules of Extensibility

The following sentence is appended to the last paragraph in section 8.2.1.1.5.2 Proposed Parameter CBB and the last paragraph in section 8.2.1.2.5.2 Negotiated Parameter CBB of DIS 9506-1.

"Any additional bits shall be ignored."

20.7.3 Modifications to the Protocol definitions

20.7.3.1 Page 39, Section 7.5.2 of DIS 9506-2

CHANGE

```
reportEventEnrollmentStatus [60] IMPLICIT ReportEventEnrollmentStatus-Request,
```

TO

```
reportEventEnrollmentStatus [60] ReportEventEnrollmentStatus-Request,
```

20.7.3.2 Page 49, Section 7.6.4, DIS 9506-2

CHANGE

```
ApplicationReference ::= SEQUENCE {
 ap-title        ISO-8650-ACSE-1.AP-title OPTIONAL,
 ap-invocation-idISO-86 50-ACSE-1.AP-invocation-id OPTIONAL,
 ae-qualifierISO-8650-ACSE-1.AE-qualifier OPTIONAL,
    ae-invocation-idISO-8650-ACSE-1.AE-invocation-id OPTIONAL
        }
```

TO

```
    ApplicationReference ::= SEQUENCE {
     ap-title       [0] OBJECT IDENTIFIER OPTIONAL,
     ap-invocation-id [1] INTEGER OPTIONAL,
     ae-qualifier[2] INTEGER OPTIONAL,
     ae-invocation-id[3] INTEGER OPTIONAL
     }
```

20.7.3.3 Page 95, Section 12.2.1 of DIS 9506-2

CHANGE

```
    structure [2] IMPLICIT SEQUENCE OF SEQUENCE {
```

TO

```
    structure [2] IMPLICIT SEQUENCE {
```

20.7.3.4 Page 96, Section 12.3.1 of DIS 9506-2

CHANGE

```
    named [4] IMPLICIT SEQUENCE {
```

TO

```
    named [5] IMPLICIT SEQUENCE {
```

20.7.3.5 Page 98, Section 12.4.2 of DIS 9506-2

CHANGE

```
    generalized-time [10] IMPLICIT GeneralizedTime,
```

TO

```
    generalized-time [11] IMPLICIT GeneralizedTime,
```

20.7.3.6 Page 138, Section 15.14 of DIS 9506-2

CHANGE

```
        additionalDetail      [9] IMPLICIT EE-Additional-Detail OPTIONAL
```

TO

```
        additionalDetail      [9] EE-Additional-Detail OPTIONAL
```

20.7.3.7 Page 166, Section 17.10 of DIS 9506-2

CHANGE the transfer syntax object identifier value from

```
        { iso asn1(1) basic-encoding(1) }
```

TO

```
        { joint-iso-ccitt asn1(1) basic-encoding(1) }
```

20.7.4 Behavioral Requirements

20.7.4.1 Filenames

File Names are specified in accordance with the NBS Implementors'agreements for FTAM Reference [3] in 20.7.2.

20.7.4.2 Identify Service

In the Identify service, the vendor, model and revision fields may be of any length, but only the first 64, 16, and 16 octets respectively are treated as significant.

20.7.4.3 Initiate Service

An MMS Client will:

1. propose 1 or greater for the value of the Proposed Max Serv Outstanding Called parameter in the Initiate service when initiating the application association (calling).

2. offer 1 or greater for the value of the Negotiated Max Serv Outstanding Calling parameter in the Initiate service when receiving the application association initiation (called).

An MMS Server will:

1. propose 1 or greater for the value of the Proposed Max Serv Outstanding Calling parameter in the Initiate service when initiating the application association (calling).

2. offer 1 or greater for the value of the Negotiated Max Serv Outstanding Called parameter in the Initiate service when receiving the application association initiation (called).

20.7.4.3.1 Minimum Segment Size

MMS implementations are able to parse and process 512 octets of MMSpdu as they are encoded in ASN.1 basic encoding rules.

20.7.4.3.2 Maximum Segment Size

The Max Segment Size is defined as the maximum number of octets in an MMS pdu encoded using the negotiated transfer syntax. This size will apply to all MMS pdus with the exception of the initiate-Request PDU, initiate-Response PDU, and the initiate-Error PDU. The max segment size will be negotiated during connection initiation using the Proposed Max Segment Size and Negotiated Max Segment Size parameters of the MMS initiate service.

The Max Segment Size will be applied as follows:

> Any received MMSpdu which is less than or equal to the Max Segment Size will be properly parsed and processed.
>
> An MMS implementation will not send an MMSpdu whose size exceeds the Max Segment Size.

20.7.4.4 Abstract Syntax Name

The ASN.1 object identifier value for the abstract syntax name will be the same as specified on page 166, section 17.10 DIS 9506-2.

20.7.4.5 Application Context Name

The ASN.1 object identifier value for the application context name will be the same as specified on page 166, section 17.11 DIS 9506-2.

An MMS-user ignores the Application Context Name in the A-Associate indication and the A-Associate confirm.

20.7.4.6 Minor Version Number

The Minor Version Number is zero.

20.7.5 Parameter CBB Subset

The following subset of MMS Parameter CBBs were considered during preparation of this appendix.

```
STR1,
NEST,
VADR,
VNAM
```

20.7.6 Service Subset

The following subset of MMS services were considered during preparation of this appendix.

```
Initiate,
Conclude,
Cancel,
Status,
GetNameList,
Identify,
UnsolicitedStatus,
GetCapabilityList,
InitiateDownloadSequence,
DownloadSegment,
TerminateDownloadSequence,
InitiateUploadSequence,
UploadSegment,
TerminateUploadSequence,
RequestDomainDownload,
RequestDomainUpload,
LoadDomainContent,
StoreDomainContent,
DeleteDomain,
GetDomainAttributes,
Read,
Write,
InformationReport,
GetVariableAccessAttributes,
Input,
Output,
TakeControl,
RelinquishControl,
ReportSemaphoreStatus,
ReportPoolSemaphoreStatus,
ReportSemaphoreEntryStatus,
CreateProgramInvocation,
DeleteProgramInvocation,
Start,
Stop,
Resume,
Reset,
Kill,
GetProgramInvocationAttributes,
ObtainFile,
GetEventConditionAttributes,
ReportEventConditionStatus,
GetAlarmSummary,
ReadJournal,
WriteJournal,
InitializeJournal,
CreateJournal,
DeleteJournal,
ReportJournalStatus
```

Editor's Note: The text of the entire working document has been marked as "pending stable", with the exceptions of section 20.3, 20.4, and 20.6.3.2.

21. Character Set Usage in OSI Applications

This International Standardized Profile is defined within the context of Functional Standardization, in accordance with the principles specified by ISO TR 10000, "Taxonomy Framework and Directory of Profiles." The context of Functional Standardization is one part of the overall field of Information Technology (IT) standardization activities, covering base standards, profiles, and registration mechanisms. A profile defines a combination of base standards that collectively perform a specific well-defined IT function. Profiles standardize the use of options and other variations in the base standards, and provide a basis for the development of uniform, internationally recognized system tests.

This International Standardized Profile was developed in close cooperation between the three International OSI Workshops: the NIST OSI Implementors Workshop (NIST OIW), the European Workshop for Open Systems (EWOS), and the AsiaOceania Workshop (AOW). The text is harmonized between these three Workshops and was ratified by the Workshops' plenary assemblies.

This International Standardized Profile contains an informative Annex A - Character Set Technology.

21.1. Scope

This International Standardized Profile describes Information Processing Character Set agreements covering character set usage in referencing Application Service Elements and OSI Applications. These agreements are based upon ISO Character Set International Standards and CCITT Character Set Recommendations. The informative Annex A summarizes the character set practices within referencing Application Service Elements and OSI Applications including all relevant encoding information drawn from the appropriate ISO Registers, ISO Standards, and CCITT Recommendations.

21.1.1. Recording Additional Character Sets

This International Standardized Profile does not prevent Application Service Elements from adding new graphic character sets or control function sets. When new character sets are to be added, however, they shall be recorded in this International Standardized Profile.

21.1.2. General Applicability of Character Sets

For the purpose of this International Standardized Profile when new character sets are to be added, efforts shall be made to obtain agreement on their uses among Application Service Elements so that they are generally applicable.

21.1.3. Minimum Number of Character Sets

The number of character sets supported will be kept to the minimum possible so as to maximize interoperability.

21.2. References

The following International Standards and CCITT Recommendations are referenced in this International Standardized Profile:

International Information Exchange for Videotex, CCITT Recommendation T.100, 1985.

International Alphabet No. 5, CCITT Recommendation T.50, 1985.

Coded Character Sets for Telematic Services, CCITT Recommendation T.51, 1985.

Character Repertoire and Coded Character Sets for the International Teletex Service, CCITT Recommendation T.61, 1985.

Information processing — 8-bit single-byte coded graphic character sets — Part 7: Latin/Greek alphabet, DIS 8859-7, 1987.

Information processing — ISO 7-bit and 8-bit coded character sets — Code extension techniques, IS 2022, 1986.

Data processing — Procedure for registration of escape sequences, IS 2375, 1985.

Information processing — ISO 8-bit code for information interchange — Structure and rules for implementation, IS 4873, 1986.

Information Processing — ISO 7-bit and 8-bit coded character sets — Additional control functions for character-imaging devices, IS 6429, 1983.

Information Processing — ISO 7-bit coded character set for information interchange, IS 646, 1983.

Information processing — Coded character sets for text communication — Part 1: General introduction, IS 6937/1, 1983.

Information Processing — Coded character sets for text communication — Part 2: Latin alphabetic and non-alphabetic graphic characters, IS 6937/2, 1983.

Text Communication — Registration of graphic character subrepertoires, IS 7350, 1984.

Information Processing Systems — Open Systems Interconnection — Specification of Abstract Syntax Notation One (ASN.1), IS 8824, 1987.

Information Processing Systems — Open Systems Interconnection — Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1), IS 8825, 1987.

Information processing — 8-bit single-byte coded graphic character sets — Part 1: Latin alphabet No. 1, IS 8859-1, 1987.

International Register of Coded Character Sets to be Used With Escape Sequences, International Register of Coded Character Sets, 1989.

21.3. Definitions

21.3.1. character data:

Character data is defined to be graphic characters and control functions as defined by ISO 2022 and the appropriate International Standards.

21.3.2. composite graphic symbol:

A composite graphic symbol is defined for the purposes of this International Standardized Profile as a non-spacing diacritical in combination with an alphabetic as in ISO 6937.

21.4. Abbreviations

21.4.1. ASN.1:

ASN.1 is an abbreviation for Abstract Symbolic Notation One.

21.4.2. IRV

IRV is an abbreviation for International Reference Version.

21.5. Position within the Taxonomy

<<The formal position of this International Standardized Profile within the taxonomy is currently unknown.>>

It may be referenced from the ISP for any application service element or OSI application.

21.6. Conformance

Implementations claiming conformance to this ISP must designate one or more of the Character Set Profiles defined herein.

Imaging of Graphic Characters is not required by this ISP. Imaging conformance may be defined in the specific Upper Layers Requirements section of the referencing ISP. If no imaging requirements are specified, then there are no conformance requirements.

21.6.1. Processed Character Data

Processed character data is character data which must be processed by the Application Service Element or OSI Application, for example, store and forward character data.

Senders of character data must not produce invalid character codes or invalid designating or invoking escape sequences.

21.6.1.1. Non-supported Character Sets

If an implementation receives a designating escape sequence for a character set that it is not able to interpret, then it shall regard that sequence as "invalid data." If possible, it will signal this error in a way that is appropriate to the protocol definition. For applications for which there is no protocol, then no error need be returned. It will not be required to interpret any following characters that are within that data item.

21.6.1.2. Reserved Character Codes

If an implementation receives a coded character that is specified in the standard to be "reserved for future standardization," it shall not be considered an error. An imaging device shall indicate receipt of such a reserved character to the user in am implementation defined way, e.g. by making available a character that need not be distinguishable from one of the other characters specified in the standard.

If receivers reject or discard invalid character codes, an appropriate error code must be returned.

21.6.1.3. Validation of Character Codes

Character codes within the scope of a standard for which there is no definition in the code table are defined to be invalid character codes. An invalid escape sequence is any designating or invoking escape sequence which is not defined in these agreements.

Implementations must conform to the following statement.

- Originators of data shall not produce invalid character codes or invalid designating or invoking escape sequences.

21.6.2. Unprocessed Character Data

Unprocessed character data is character data which is not processed by the Application Service Element or OSI Application, for example, character matching.

21.6.2.1. Validation of Character Codes

Character codes within the scope of a standard for which there is no definition are defined to be invalid character codes. An invalid escape sequence is any designating or invoking escape sequence which is not defined in these agreements.

Implementations must conform to the following statements.

- Receivers need not validate character codes or designating or invoking escape sequences.
- Senders who do not originate data need not validate character codes.

21.7. General Agreements

The agreements recorded in this section cover all character set usage except where explicitly noted to the contrary. Additional agreements specific to individual character sets are recorded in the individual character set profiles.

21.7.1. Encoding

The following agreements cover various aspects of character encoding.

21.7.1.1. Overprint, Composite Characters

A composite graphic symbol is considered as one character for purposes of comparison and character string length computation.

With the exception of composite graphic symbols, 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. So for example, the sequence "a BACKSPACE ¨" must be processed as three characters rather than as the single character ä.

21.7.1.2. Code Extension Facilities for GeneralString and GraphicString

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 from ISO 2022 have been applied: ESC 2/0 4/3, ESC 2/0 4/9, and ESC 2/0 5/10. These sequences indicate:

- 8-bit environment;
- the G0, and G1 graphic sets shall be used;
- the designating escape sequences also invoke the G0 and G1 sets into columns 02 to 07 and 10 to 15 respectively;
- no locking shift functions shall be used;
- the graphic character sets may comprise 94 and/or 96 characters,
- a G2 set shall be used; and,
- characters from G2 may be accessed by use of the single-shift 2 control function.

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

21.7.1.3. Initial Conditions for TeletexString

For TeletexString (T61String) the initial condition is described in CCITT T.61 Annex A, Clause A.2.

21.7.2. Comparisons

This section contains agreements concerning comparison of characters during processing.

21.7.2.1. Matching Characters

A character submitted for matching with another character does not have to be drawn from the same coded character set. However, the match is restricted to characters taken from any pair of coded character sets for which equality or inequality is defined. The identifications of such pairs of coded character sets are shown in the following list. The result of comparing characters from a pair of different coded character sets not in this list is undefined.

(ISO 646,	ISO 6937-2)
(ISO 646,	ISO 8859-1)
(ISO 6937-2,	ISO 8859-1)

Character matching is defined for characters, not their coded representations. The character must take into account any code extension techniques. For example, the character named "SMALL LETTER a WITH DIAERESIS" of ISO 8859 must match the character named "small a with diaeresis or umlaut mark" of ISO 6937 even though the former character is encoded in a single octet and the latter in two octets.

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 names of the composite graphic symbols are specified in the standard itself. However in the present edition there are some systematic differences between the naming conventions used in the standard and those used in the ISO Character Set Register as shown below:

ISO 6937 name:	capital A with acute accent
ISO Register Name:	CAPITAL LETTER A WITH ACUTE ACCENT

In this case, two characters are equal if, and only if, their names differ only by the inclusion of the word LETTER in the ISO Register Name. For those characters whose names do not follow this convention, the following list defines the match:

ISO 6937 Name	ISO Register Name

<<Editor's Note: to be filled in>

If a character set registration does not provide character names then matching will be defined by exact matching on an octet by octet basis.

<<Editor's Note: The problem of matching Oriental language character sets is for further study.>>

In comparing strings all control functions except code designation and invocation extension facilities shall be ignored. SPACE is treated as a graphic character in such comparisons.

In comparing strings when a character code is encountered for which no other match is defined, matching will be defined by exact matching on an octet by octet basis.

21.7.2.2. CaseIgnore Comparisons

In character comparisons in which case is ignored, the matching rules of clause 21.7.2.1 are relaxed in that the characters are equal if their names as defined in clause 21.7.2.1 differ only by one name having SMALL where the other name has CAPITAL.

21.7.2.3. Ordering and Comparing Characters

An agreement on comparison, other than equality or inequality, between characters requires a definition of a collating sequence. This document contains no such agreements.

The collating sequence of letters, accented letters and other graphic symbols is not currently defined in any International Standard or Recommendation.

Preferred collating sequences might vary between countries.

21.7.2.4. Comparing Encoded ASN.1 Character 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 result independent of each character string's encoding, for example, the comparison is independent of the Basic Encoding Rules for ASN.1:

- as constructed or primitive form, or,
- as definite or indefinite length form.

21.8. Character Set Profiles

A Character Set Profile summarizes implementation agreements specific to a particular character set. Character Set Profiles are identified in the following manner:

CSn-m

where:
CS means Character Set
n = 1 designates a profile for a graphic character set
n = 2 designates a profile for a control function set
m is a number uniquely identifying the Character Set Profile.

The values of n and m are defined in this agreement. Names of Character Set Profiles are also defined in this International Standardized Profile.

This section covers agreements about Character Set Standards and Recommendations including:

- subrepertoires supported,
- standardized options selected,
- 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,
- the designation of component character sets within the ISO 2022 Code Extension Model where there is a choice to be made.

The General Agreements of the preceding section apply to each of these Character Set Profiles.

21.8.1. CS1-1 ISO 646 Graphic Character Set

21.8.1.1. Base Standard

International Standard 646 - 1983, *Information Processing — ISO 7-bit coded character set for information interchange.*

<<Editor's Note: These agreements will be based on the new DIS 646.>>

21.8.1.2. Subrepertoire or Version

International Reference Version

21.8.1.3. Standard Options Selected

Composite graphic symbols are covered by General Agreements.

21.8.1.4. Character Set Components and Designated Position

IRV of ISO 646 number 2 in G0

<<Editor's Note: This will change to number 6.>>

Space is in 2/0

21.8.1.5. Other Agreements

None.

21.8.2. CS1-2 JIS X0208

<<Editor's Note: to be defined.>>

21.8.3. CS1-3 CCITT Recommendation T.61 Graphic Character Sets Basic Teletex Profiles

21.8.3.1. Base Standard

CCITT Recommendation T.61 - 1985, *Character Repertoire and Coded Character Sets for the International Teletex Service.*

<<Editor's Note: These references will be updated as soon as the 1989 versions are published.>>

21.8.3.2. Subrepertoire or Version

None

21.8.3.3. Standard Options Selected

None

21.8.3.4. Character Set Components and Designated Position

Teletex Primary Graphic Set 102 in G0

Teletex Supplementary Graphic Set 103 in G2

SPACE in 2/0

21.8.3.5. Other Agreements

Support for CCITT Recommendation T.61 as an ASN.1 GeneralString is outside of this International Standardized Profile.

Support of the graphic set components of T.61 as an ASN.1 GraphicString is outside the scope of this International Standardized Profile.

Use of CCITT Recommendation T.61 except where mandated by standards is outside the scope of this International Standardized Profile. Exceptions to this rule for specific Application Service Element protocol elements must be documented by the referencing Application Service Elements or OSI Applications.

21.8.4. CS1-4 ISO 8859-1 Latin Alphabet No. 1

21.8.4.1. Base Standard

International Standard 8859-1 - 1987, *Information processing — 8-bit single-byte coded graphic character sets — Part 1: Latin alphabet No. 1.*

21.8.4.2. Subrepertoire or Version

Not applicable.

21.8.4.3. Standard Options Selected

Not applicable.

21.8.4.4. Character Set Components and Designated Position

ASCII Graphic Character Set number 6 in G0

Right hand part of Latin Alphabet No. 1 number 100 in G1

21.8.4.5. Other Agreements

None.

21.8.5. CS1-5 ISO 6937-2 Coded Character Sets for Text Communication

21.8.5.1. Base Standard

International Standard 6937/2 - 1983, *Information Processing — Coded character sets for text communication — Part 2: Latin alphabetic and non-alphabetic graphic characters.*

<<Editor's Note: Includes Addendum 1 as soon as it is published.>>

21.8.5.2. Subrepertoire or Version

Full number 0

Minimum number 1

Teletex number 3

Western European Data Processing number 9

21.8.5.3. Standard Options Selected

Not applicable

21.8.5.4. Character Set Components and Designated Position

IRV of ISO 646 number 2 in G0

<<Editor's Note: This will change to number 6.>>

Supplementary set of Latin Text Processing number 142 in G2

21.8.5.5. Other Agreements

For subrepertoires 2 and 5, the supplementary set may be omitted at the discretion of the sending application.

21.8.6. CS1-6 ISO 8859/7 Greek Supplementary Set

<<Editor's Note: to be defined.>>

21.8.7. CS1-7 CCITT Recommendation T.61 Graphic Character Sets Basic Teletex Profiles (1984)

21.8.7.1. Base Standard

CCITT Recommendation T.61 - 1981, *Character Repertoire and Coded Character Sets for the International Teletex Service.*

21.8.7.2. Subrepertoire or Version

None

21.8.7.3. Standard Options Selected

None

21.8.7.4. Character Set Components and Designated Position

Teletex Primary Graphic Set 102 in G0

Teletex Supplementary Graphic Set 103 in G2

SPACE in 2/0

21.8.7.5. Other Agreements

Support for CCITT Recommendation T.61 as an ASN.1 GeneralString is outside of this International Standardized Profile.

Support of the graphic set components of T.61 as an ASN.1 GraphicString is outside the scope of this International Standardized Profile.

Use of CCITT Recommendation T.61 except where mandated by standards is outside the scope of this International Standardized Profile. Exceptions to this rule for specific Application Service Element protocol elements must be documented in the referencing Application Service Elements or OSI Applications.

This profile is intended for use with the X.400-1984 implementation agreements only.

21.8.8. CS 1-8 CCITT Recommendation T.61 Graphic Character Sets

<<Editor's Note: to be defined.>>

21.8.9. Korean National Character Set

<<Editor's Note: to be defined.>>

21.8.10. CS2-1 ISO 646 C0 Control Functions

21.8.10.1. Base Standard

International Standard 646 - 1983, *Information Processing — ISO 7-bit coded character set for information interchange.*

21.8.10.2. Subrepertoire or Version

None.

21.8.10.3. Standard Options Selected

None.

21.8.10.4. Character Set Components and Designated Position

ISO 646 C0 Set number 1 in C0

DELETE in 7/15

21.8.10.5. Other Agreements

When a single format effector for vertical (or horizontal) movement is optionally permitted to effect a combined vertical and horizontal movement, implementations shall not use this single format effector for effecting the combined vertical and horizontal movement.

21.8.11. CS2-2 ISO 6429 Additional Control Functions

21.8.11.1. Base Standard

International Standard 6429 - 1983, *Information Processing — ISO 7-bit and 8-bit coded character sets — Additional control functions for character-imaging devices.*

21.8.11.2. Subrepertoire or Version

None.

21.8.11.3. Standard Options Selected

None.

21.8.11.4. Character Set Components and Designated Position

C1 Control Set of ISO 6429-1983 number 77 in C1

21.8.11.5. Other Agreements

None.

21.8.12. CS2-3 CCITT Recommendation T.61 Control Sets

21.8.12.1. Base Standard

CCITT Recommendation T.61 - 1985, *Character Repertoire and Coded Character Sets for the International Teletex Service.*

<<Editor's Note: These references will be updated as soon as the 1989 versions are published.>>

21.8.12.2. Subrepertoire or Version

None.

21.8.12.3. Standard Options Selected

Teletex optional repertoire of control functions is not selected.

21.8.12.4. Character Set Components and Designated Position

Teletex Primary Set of Control Functions number 106 in C0

Teletex Supplementary Set of Control Functions number 107 in C1

21.8.12.5. Other Agreements

None.

21.8.13. CS2-4 CCITT Recommendation T.61 Control Sets (1984)

21.8.13.1. Base Standard

CCITT Recommendation T.61 - 1981, *Character Repertoire and Coded Character Sets for the International Teletex Service.*

21.8.13.2. Subrepertoire or Version

None.

21.8.13.3. Standard Options Selected

Teletex optional repertoire of control functions is not selected.

21.8.13.4. Character Set Components and Designated Position

Teletex Primary Set of Control Functions number 106 in C0

Teletex Supplementary Set of Control Functions number 107 in C1

21.8.13.5. Other Agreements

This profile is intended for use with the X.400-1984 implementation agreements only.

Annex A

Character Set Technology

(This Annex does not form part of these agreements.)

A.1. Introduction

This Annex presents information from Information Processing Character Set Standards which is relevant to the implementation of OSI Services. The intent is to collect into one place the most relevant information for implementors from character set standards specified in OSI and OSI related standards.

A.2. Scope

Material in this Annex is drawn from ISO and CCITT Character Set standards and Recommendations. Topics covered include Character Set Extension Techniques and Character Set Encodings. ASN.1 Basic Encoding Rules are reviewed also. Rationale for the implementation agreements in the ISP is provided where appropriate.

A.3. Field of Application

This annex covers character set information for ASN.1 Basic Encoding Rules as used by OSI services. It also includes information pertaining to OSI Interchange Formats such as Office Document Architecture.

A.4. Character Set Standards

The following character set standards have some relevance to this material.

International Information Exchange for Videotex, CCITT Recommendation T.100, 1985.

International Alphabet No. 5, CCITT Recommendation T.50, 1985.

Coded Character Sets for Telematic Services, CCITT Recommendation T.51, 1985.

Character Repertoire and Coded Character Sets for the International Teletex Service, CCITT Recommendation T.61, 1985.

Information processing — 8-bit single-byte coded graphic character sets — Part 7: Latin/Greek alphabet, DIS 8859-7, 1987.

Information processing — ISO 7-bit and 8-bit coded character sets — Code extension techniques, IS 2022, 1986.

Data processing — Procedure for registration of escape sequences, IS 2375, 1985.

Information processing — ISO 8-bit code for information interchange — Structure and rules for implementation, IS 4873, 1986.

Information Processing — ISO 7-bit and 8-bit coded character sets — Additional control functions for character-imaging devices, IS 6429, 1983.

Information Processing — ISO 7-bit coded character set for information interchange, IS 646, 1983.

Information processing — Coded character sets for text communication — Part 1: General introduction, IS 6937/1, 1983.

Information Processing — Coded character sets for text communication — Part 2: Latin alphabetic and non-alphabetic graphic characters, IS 6937/2, 1983.

Text Communication — Registration of graphic character subrepertoires, IS 7350, 1984.

Information Processing Systems — Open Systems Interconnection — Specification of Abstract Syntax Notation One (ASN.1), IS 8824, 1987.

Information Processing Systems — Open Systems Interconnection — Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1), IS 8825, 1987.

Information processing — 8-bit single-byte coded graphic character sets — Part 1: Latin alphabet No. 1, IS 8859-1, 1987.

International Register of Coded Character Sets to be Used With Escape Sequences, International Register of Coded Character Sets, 1989.

A.5. Introduction to Character Set Standards

A brief introduction to reading a character set standard is presented here for the uninitiated. Most of the character set standards described in this Annex use the term "bit combinations" to refer to the ordered string of bits which compose a character. Most implementations of these standards allocate an 8-bit byte to a character and consequently tend to intermix the notions of bytes and characters. In the OSI environment, 8-bit bit combinations are normally referred to as "octets."

A character set standard generally presents its character encodings in a table composed of 16 rows and 8 or 16 columns depending on whether a 7-bit or an 8-bit character set is being defined. A given character code is generally referenced by naming its column and then its row. Thus in ISO 646 the capital letter A is referred to as 4/1. Some standards precede single digits with a zero so that in ISO 8859/1 the capital letter A is referred to as 04/01. This positional notation is especially important in the consideration of the code extension techniques. Code extension techniques describe characters in terms of their position only, without regard for any possible previously assigned interpretations.

A.6. Definitions

The following definitions drawn from relevant character set standards are provided to assist in understanding the material in this annex. These definitions were drawn from International Standards which were current at the time of drafting this document. Any conflict between these definitions and those of the relevant International Standards shall be resolved by using the definition in the International Standard.

bit combination: An ordered set of bits that represents a character or is used as a part of the representation of a character.

byte: A bit string that is operated upon as a unit and the size of which is independent of redundancy or framing techniques.

character: A member of a set of elements used for the organization, control or representation of data.

code extension: The techniques for the encoding of characters that are not included in the character set of a given code.

control character: A control function the coded representation of which consists of a single bit combination.

control function: An action that affects the recording, processing, transmission or interpretation of data and that has a coded representation consisting of one or more bit combinations.

graphic character: A character, other than a control function, that has a visual representation normally handwritten, printed or displayed.

A.7. ISO 2022 Information Processing — ISO 7-bit and 8-bit coded character sets — Code extension techniques

This International Standard was originally written to establish extension techniques for the 7-bit codes of ISO 646. It has been revised twice so that it now also provides the basic framework for an 8-bit code family which is compatible with the 7-bit codes. The four interrelated clauses cover

- the extension of the 7-bit code remaining in a 7-bit environment;
- the structure of a family of 8-bit codes;
- the extension of an 8-bit code remaining in an 8-bit environment;
- the relationship between the 7-bit code and an 8-bit code.

The middle two clauses are of special relevance to this document although portions of the others should be read and understood in order to set the context for the relevant material.

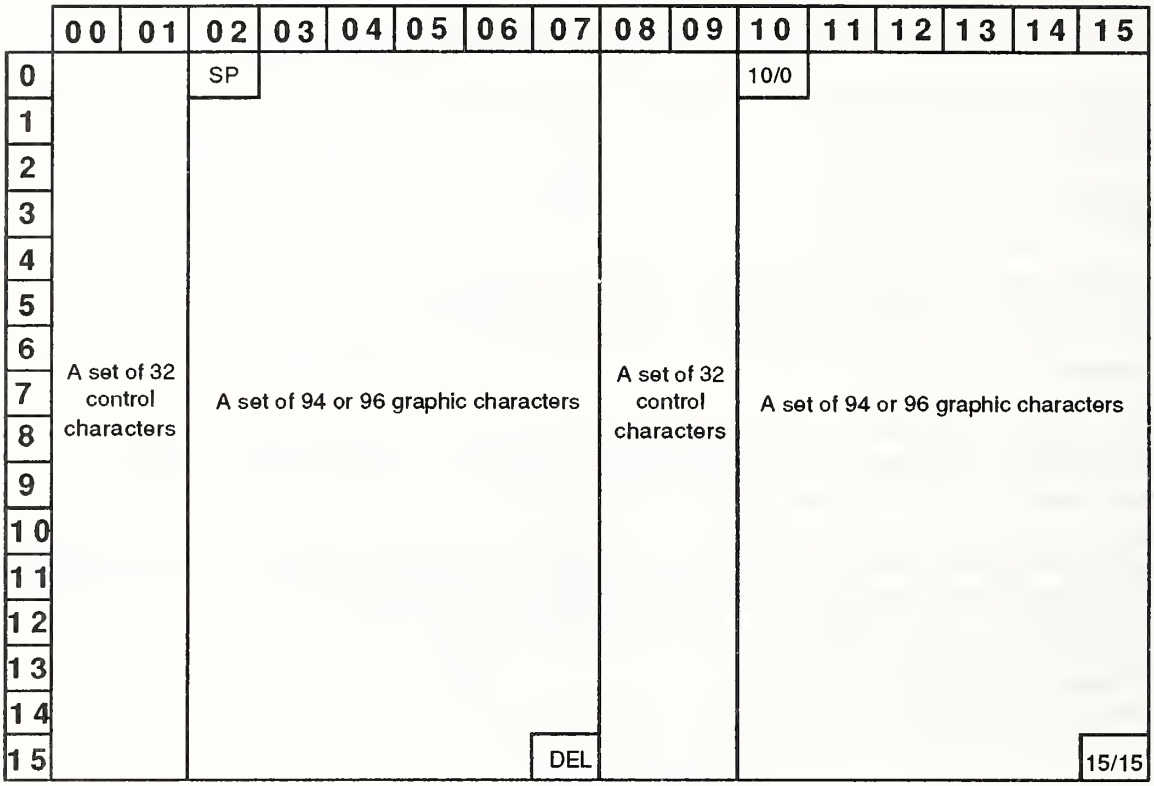
Some underlying assumptions from the standard are recorded here in order to understand the context of these agreements. Clause 2 notes that code extension techniques are designed to be used for data to be processed serially in a forward direction.

A.7.1. Structure of a Family of 8-bit codes

Clause 7 of the standard describes a family of 8-bit codes obtained from the 7-bit set. The family of 8-bit codes is obtained by the addition of one bit to each of the bit combinations of the 7-bit code producing a set of 256 8-bit combinations. The characters of the 7-bit code are assigned to the 128 bit combinations for which the eighth bit is set to ZERO. The 128 additional bit combinations for which the eighth bit is set to ONE are available for assignment. The 8-bit code table of clause 7.1 is a 16 by 16 array of columns numbered 00 to 15 and rows numbered 0 to 15. Columns 08 and 09 are provided for control characters and columns 10 to 15 for graphic characters.

The following figure shows the basic code structure for 8-bit character codes. This structure is followed by the standards described in this annex.

8-bit Code Structure


00
01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
SP
10/0
A set of 32 control characters
A set of 94 or 96 graphic characters
A set of 32 control characters
A set of 94 or 96 graphic characters
DEL
15/15

The family concept is described in clause 7.2 as

a) a set of 32 additional control characters can be selected for columns 08 and 09;

b) a set of 94 or 96 additional graphic characters can be selected for columns 10 to 15. If a set of 94 graphic characters is invoked in columns 10 to 15, positions 10/0 and 15/15 shall not be used.

Three control functions were provided by ISO 646 for purposes of code extension. ISO 2022 uses these three and adds 7 more for use in the 8-bit environment. For reference purposes the corresponding characters from the 7-bit environment are shown also. The following table shows these control functions.

7-bit Name	Abbreviation	8-bit Name	Abbreviation
ESCAPE	ESC	ESCAPE	ESC
SHIFT-OUT	SO	LOCKING-SHIFT ZERO	LS0
SHIFT-IN	SI	LOCKING-SHIFT ONE	LS1
LOCKING-SHIFT TWO	LS2	LOCKING-SHIFT TWO	LS2
LOCKING-SHIFT THREE	LS3	LOCKING-SHIFT THREE	LS3
SINGLE-SHIFT TWO	SS2	SINGLE-SHIFT TWO	SS2
SINGLE-SHIFT THREE	SS3	SINGLE-SHIFT THREE	SS3
		LOCKING-SHIFT ONE RIGHT	LS1R
		LOCKING-SHIFT TWO RIGHT	LS2R
		LOCKING-SHIFT THREE RIGHT	LS3R

A.7.2. Elements of Code Extension in an 8-bit Environment

The elements of code extension in an 8-bit environment are shown in the following table taken from Clause 8.1 of the standard:

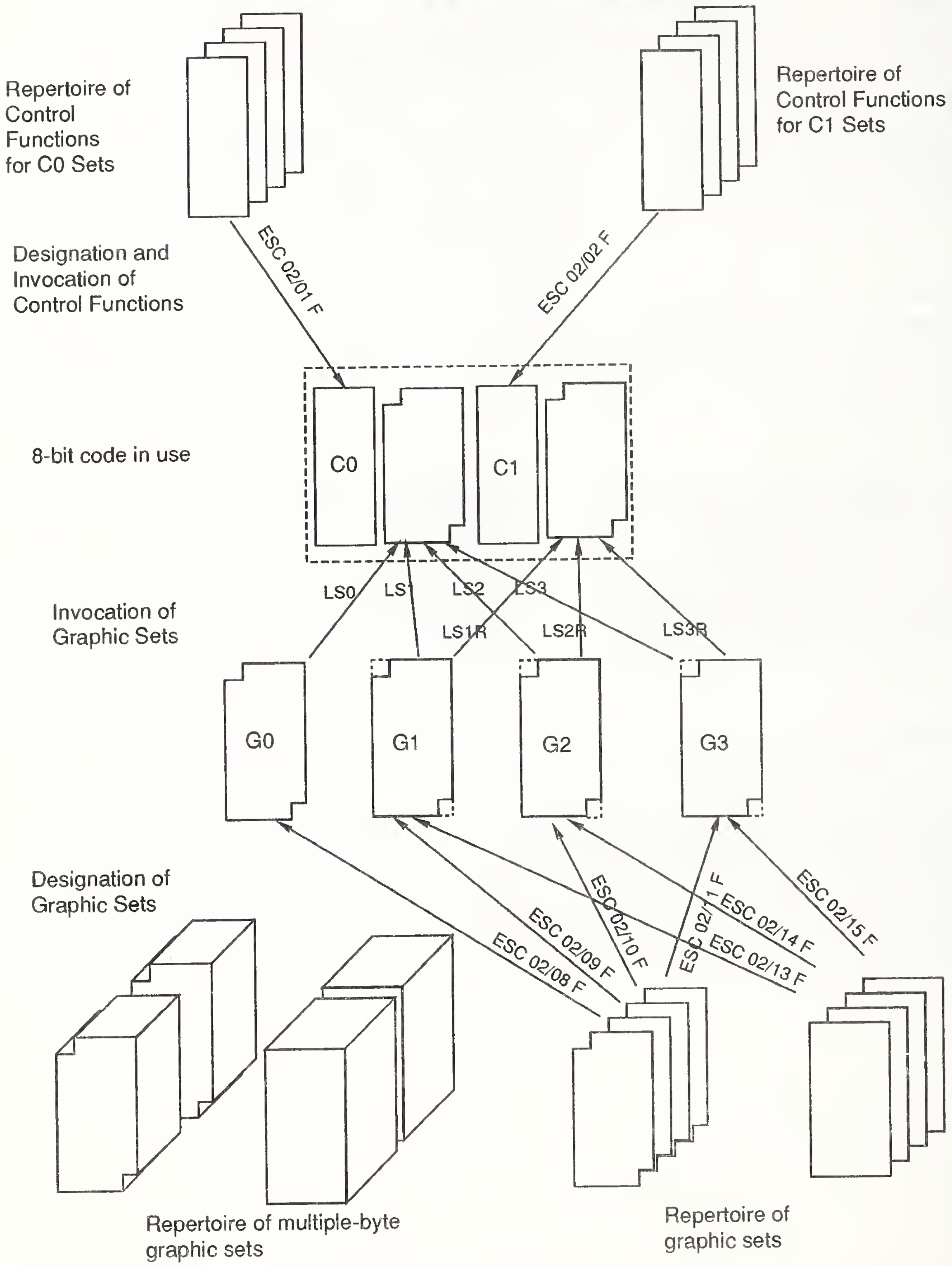
Set	Description	Columns occupied
C0	32 control characters	00 to 01
C1	32 control characters	08 to 09
G0	94 graphic characters	02 to 07
G1	94 or 96 graphic characters	02 to 07 or 10 to 15
G2	94 or 96 graphic characters	02 to 07 or 10 to 15
G3	94 or 96 graphic characters	02 to 07 or 10 to 15

A.7.3. Multiple Character Sets

<<Describe multi-level designation and invocation here.>>

The standard defines a graphic character set extension strategy in which a designating escape sequence is used to select up to four graphic character sets from the International Character Set Register. An invocation sequence is then used to select up to two graphic sets from the designated sets for concise access to the characters. The following figure shows the technique for the 8-bit environment.

Code Extension in an 8-bit Environment


Repertoire of Control Functions for C0 Sets
Repertoire of Control Functions for C1 Sets
Designation and Invocation of Control Functions
ESC 02/01 F
ESC 02/02 F
8-bit code in use
C0
C1
Invocation of Graphic Sets
LS0
LS1
LS2
LS3
LS1R
LS2R
LS3R
G0
G1
G2
G3
Designation of Graphic Sets
ESC 02/08 F
ESC 02/09 F
ESC 02/10 F
ESC 02/11 F
ESC 02/13 F
ESC 02/14 F
ESC 02/15 F
Repertoire of multiple-byte graphic sets
Repertoire of graphic sets

The standard defines two terms for use in describing code extension practices: to designate and to invoke. They are defined as follows:

to designate: To identify a set of characters that are to be represented, in some cases immediately and in others on the occurrence of a further control function, in a prescribed manner.

to invoke: To cause a designated set of characters to be represented by the prescribed bit combinations whenever those bit combinations occur, until an appropriate code extension function occurs.

Designation of a character set is usually achieved by employing an escape sequence defined by the standard along with values assigned by a registration authority. In many cases, designation of a character set also implies invocation. In other cases a character set must be explicitly invoked usually by using a shift function.

The following table defines the use of the locking shift functions in an 8-bit environment for extension of the graphic set.

Function	Abbreviation	Set Invoked	Columns affected
LOCKING-SHIFT ZERO	LS0	G0	02 to 07
LOCKING-SHIFT ONE	LS1	G1	02 to 07
LOCKING-SHIFT ONE RIGHT	LS1R	G1	10 to 15
LOCKING-SHIFT TWO	LS2	G2	02 to 07
LOCKING-SHIFT TWO RIGHT	LS2R	G2	10 to 15
LOCKING-SHIFT THREE	LS3	G3	02 to 07
LOCKING-SHIFT THREE RIGHT	LS3R	G3	10 to 15

The meanings of control characters in columns 00, 01, 08 and 09 shall not be affected by the occurrence of these locking shift functions.

Clause 6.4 states that at the beginning of any information interchange, except where interchanging parties have agreed otherwise, all designations shall be defined by the use of appropriate escape sequences, and the shift status shall be defined by the use of the appropriate locking shift functions.

A.7.4. Announcement of Extension Facilities

A code extension facility consists of the elements of code extension employed as well as the means by which these elements are designated and invoked. Thus the control function sets, the graphic character sets, and the character shifting codes must be specified. Specification of control function sets and graphic character sets also specifies the designation and invocation sequences required to use their codes.

Clause 9 of ISO 2022 describes how the various extension facilities are to be made known. If an announcement is to be embedded in the interchanged information, the form is described. The announcement may be omitted by agreement between the interchanging parties. Some restrictions are imposed on the defined announcer sequences. For example the sequence ESC 02/00 04/03 specifies that 1) the G0 and G1 sets shall be used in an 8-bit environment only, 2) the designating escape sequences also invoke the G0 and G1 sets into columns 02 to 07 and 10 to 15, respectively, and 3) no locking shift functions shall be used.

A.7.5. Composite Graphic Characters

Clause 6.1.8 of the standard addresses methods for the representation of additional graphic characters by the combination of two or more graphic characters in the same position. Two methods are provided for:

a) graphic characters having implicit forward motion (spacing characters) used in conjunction with BACKSPACE or CARRIAGE RETURN;

b) graphic characters having no implicit forward motion (non-spacing characters) used in combination with spacing graphic characters.

Method b allows for the specification of characters with diacritical marks. The technique is known colloquially as the "dead key" approach. A non-spacing accent grave character is immediately followed by the character it modifies.

A.7.6. International Register of Coded Character Sets to be used with Escape Sequences

ISO 2375 specifies procedures to be used to assign meanings to the final bit combinations of escape sequences defined in ISO 2022. The International Register of Coded Character Sets to be used with escape sequences is the document which records these assignments. The current International Registration Authority for ISO 2375 is the European Computer Manufacturers Association (ECMA).

A.8. Character Sets

Several character set standards are described here. The standards chosen for description are each used by one or more known OSI applications. The usage of these standards is summarized in tabular form.

A.8.1. ISO 646 *7-bit coded character set for information processing interchange* and CCITT Recommendation T.50 *International Alphabet No. 5*

This International Standard specifies a set of 128 characters with their coded representation. The 128 bit combinations of the 7-bit code represent control characters and graphic characters. The allocation of characters to bit combinations is based on the following principles:

- the bit combinations 0/0 to 1/15 represent 32 control characters;
- the bit combination 2/0 represents the character SPACE, which is interpreted as both a control character and a graphic character;
- the bit combinations 2/1 to 7/14 represent up to 94 graphic characters;
- the bit combination 7/15 represents the control character DELETE.

The 7-bit code table consists of 128 positions arranged in 8 columns and 16 rows. The columns are numbered from 0 to 7, and the rows are numbered 0 to 15.

Most of these characters are mandatory and unchangeable, but provision is made for some flexibility to accommodate national and other requirements. The standard provides guidance on how to exercise the options offered in order to define specific national versions and application-oriented versions. It further specifies an International Reference Version in which all options have been exercised.

<<Editor's Note: A revision of ISO 646 which has achieved DP status revises this table.>>

X3.4-1977 ASCII

	0	1	2	3	4	5	6	7
0	NUL	DLE	SP	0	@	P	`	p
1	SOH	DC1	!	1	A	Q	a	q
2	STX	DC2	"	2	B	R	b	r
3	ETX	DC3	#	3	C	S	c	s
4	EOT	DC4	$	4	D	T	d	t
5	ENQ	NAK	%	5	E	U	e	u
6	ACK	SYN	&	6	F	V	f	v
7	BEL	ETB	'	7	G	W	g	w
8	BS	CAN	(	8	H	X	h	x
9	HT	EM	)	9	I	Y	i	y
10	LF	SUB	*	:	J	Z	j	z
11	VT	ESC	+	;	K	[	k	{
12	FF	FS	,	<	L	\	|	¦
13	CR	GS	–	=	M	]	m	}
14	SO	RS	.	>	N	^	n	~
15	SI	US	/	?	O	_	o	DEL

ISO 646-1983 IRV

	0	1	2	3	4	5	6	7
0	NUL	TC7	SP	0	@	P	`	p
1	TC1	DC1	!	1	A	Q	a	q
2	TC2	DC2	"	2	B	R	b	r
3	TC3	DC3	#	3	C	S	c	s
4	TC4	DC4	¤	4	D	T	d	t
5	TC5	TC8	%	5	E	U	e	u
6	TC6	TC9	&	6	F	V	f	v
7	BEL	TC10	'	7	G	W	g	w
8	FE0	CAN	(	8	H	X	h	x
9	FE1	EM	)	9	I	Y	i	y
10	FE2	SUB	*	:	J	Z	j	z
11	FE3	ESC	+	;	K	[	k	{
12	FE4	IS4	,	<	L	\	|	|
13	FE5	IS3	–	=	M	]	m	}
14	SO	IS2	.	>	N	^	n	‾
15	SI	IS1	/	?	O	_	o	DEL

ISO 646 International Reference Version

A.8.2. ISO 8859 *Information Processing — 8-bit single-byte coded character sets*

This International Standard is a multiple part standard. Each part specifies a set of up to 191 graphic characters and the coded representation of each of these characters by means of a single 8-bit byte. The use of control functions for the coded representation of composite characters is prohibited. Each set is intended for a group of languages. Part 1 of ISO 8859 specifies a set of 191 graphic characters identified as Latin alphabet No. 1. This set of graphic characters is suitable for use in a version of an 8-bit code according to ISO 2022.

The standard specifically notes that it is not intended for use with CCITT defined Telematic services. If information coded according to ISO 8859 is to be transferred to such services, it will have to conform at the coding interface to their requirements.

ISO 8859/1-1987 Latin Alphabet No. 1

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0			SP	0	@	P	`	p			NBSP	°	À	Ð	à	ð
1			!	1	A	Q	a	q			¡	±	Á	Ñ	á	ñ
2			"	2	B	R	b	r			¢	²	Â	Ò	â	ò
3			#	3	C	S	c	s			£	³	Ã	Ó	ã	ó
4			$	4	D	T	d	t			¤	´	Ä	Ô	ä	ô
5			%	5	E	U	e	u			¥	µ	Å	Õ	å	õ
6			&	6	F	V	f	v			¦	¶	Æ	Ö	æ	ö
7			'	7	G	W	g	w			§	·	Ç	×	ç	÷
8			(	8	H	X	h	x			¨	¸	È	Ø	è	ø
9			)	9	I	Y	i	y			©	¹	É	Ù	é	ù
10			*	:	J	Z	j	z			ª	º	Ê	Ú	ê	ú
11			+	;	K	[	k	{			«	»	Ë	Û	ë	û
12			,	<	L	\	l	|			¬	¼	Ì	Ü	ì	ü
13			−	=	M	]	m	}			SHY	½	Í	Ý	í	ý
14			.	>	N	^	n	‾			®	¾	Î	Þ	î	þ
15			/	?	O	_	o	DEL			¯	¿	Ï	ß	ï	ÿ

ISO 8859/1 - 1987 Latin Alphabet No. 1

A.8.3. ISO 6937 *Information Processing — Coded Character Sets for Text Communication*

This International Standard specifies repertoires of graphic characters and control functions, and their coded representation for use in text communication. This International Standard consists, at present, of two parts, as follows:

- ISO 6937/1, General Introduction.
- ISO 6937/2, Latin Alphabetic and non-alphabetic graphic characters.

The specifications are based on the 7-bit coded character set specified in ISO 646, the 7-bit and 8-bit code extension techniques of ISO 2022, and the definitions of additional control functions given in ISO 6429.

ISO 6937 was developed in parallel with CCITT Recommendations which in the standard are referred to as S.61 and S.100. These CCITT Recommendations were moved to a new section in 1984 and were renumbered T.61 and T.100. This 1984 designation is being carried forward in the 1988 CCITT Recommendations.

A.8.3.1. ISO 6937/1 *Information Processing — Coded Character Sets for Text Communication — Part 1: General Introduction*

Annex A of this International Standard describes a method of identification of graphic characters and control functions which is used in other parts of the standard to define the characters of the standard.

A.8.3.2. ISO 6937/2 *Information Processing — Coded Character Sets for Text Communication — Part 2: Latin Alphabetic and Non-alphabetic Graphic Characters*

This part of the standard

a) defines a repertoire of Latin alphabetic and non-alphabetic characters for the communication of text in European languages;

b) specifies coded representations for the graphic characters;

c) specifies rules for the definition and use of graphic character subrepertoires.

A graphic subrepertoire is a subset of the defined character repertoire. Because the number of characters defined by this standard is so large, this subsetting facility allows for the use of well defined subsets of the characters available. Rules for the definition of subrepertoires are defined in clause 5. The procedure for registration of subrepertoires is given in ISO 7350. Three standard subrepertoires are defined in Annex A of the standard.

Graphic characters which represent accented letters and umlauts are specified using a two byte sequence composed of the diacritical character immediately followed by the character modified. The allowable combinations are carefully defined in the standard and only these combinations are permitted.

ISO 6937/2-1983 Addendum 1
Full Repertoire

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0			SP	0	@	P	`	p			NBSP	°		―	Ω	ĸ
1			!	1	A	Q	a	q			¡	±	`	¹	Æ	æ
2			"	2	B	R	b	r			¢	²	´	®	Đ	đ
3			#	3	C	S	c	s			£	³	^	©	ª	ð
4			¤	4	D	T	d	t			$	×	˜	™	Ħ	ħ
5			%	5	E	U	e	u			¥	µ	¯	♪		ı
6			&	6	F	V	f	v				¶	˘	¬	Ĳ	ĳ
7			'	7	G	W	g	w			§	·	˙	¦	Ŀ	ŀ
8			(	8	H	X	h	x				÷	¨		Ł	ł
9			)	9	I	Y	i	y			‘	’			Ø	ø
10			*	:	J	Z	j	z			“	”	˚		Œ	œ
11			+	;	K	[	k	{			«	»	¸		º	ß
12			,	<	L	\	l	|			←	¼	_	⅛	Þ	þ
13			-	=	M	]	m	}			↑	½	˝	⅜	Ŧ	ŧ
14			.	>	N	^	n	‾			→	¾	˛	⅝	Ŋ	ŋ
15			/	?	O	_	o	DEL			↓	¿	ˇ	⅞	ŉ	SHY

ISO 6937-2 Latin Alphabetic and non-Alphabetic Characters

A.8.4. CCITT Recommendation T.51 *Coded Character Sets for Telematic Services*

This Recommendation specifies a primary set and a supplementary set of graphic characters which are to be the respective supersets of various primary and supplementary character sets to be used in various telematic services. The Recommendation also describes those code extension mechanisms which are relevant to existing telematic services.

A.8.5. CCITT Recommendation T.61 *Character Repertoire and Coded Character Sets for the International Teletex Service*

This Recommendation contains detailed definitions of the repertoires of graphic characters and control functions to be used in the basic International Teletex service, and their coded representations for communication.

A.9. ASN.1 Character String Types

Character String Types are sequences of zero, one or more characters from some specified character set. ISO 8824 defines 8 such types: NumericString, PrintableString, TeletexString (T61String), VideotexString, VisibleString (ISO646String), IA5String, GraphicString, GeneralString.

A.9.1. Universal Class Numbers and Registration Numbers

The type of each character string is identified by a Universal Class number. Universal Class numbers are assigned by ISO 8824. No other standard or private user may define these numbers. The character sets associated with each type are identified by the ISO Character Set Registration Numbers as shown in the following table:

Name of Character String Type	Universal Class Number	ISO Character Set Registration Numbers
NumericString	18	Not Registered
PrintableString	19	Not Registered
TeletexString (T61String)	20	87, 102, 103, 106, 107 + SPACE + DELETE
VideotexString	21	1, 72, 73, 102, 108, 128, 129 + SPACE + DELETE
VisibleString (ISO646String)	26	2 + SPACE
IA5String	22	1, 2 + SPACE + DELETE
GraphicString	25	All G sets + SPACE
GeneralString	27	All G sets and all C sets + SPACE + DELETE

NumericString and PrintableString do not have Registration Numbers assigned to them since their character sets are defined in table 4 and 5 respectively of ISO 8824.

A.9.2. Initial States

Some character string types allow multiple character sets through code extension techniques. For these types, at the beginning of each string there are initial default character sets to be designated in G0 and/or C0 and/or C1 and for each character set there is an assumed escape sequence. The following table drawn from ISO 8825 describes these initial states.

Name of Character String Type	Initial G0 (Reg. No.)	Initial C0 (Reg. No.)	Initial C1 (Reg. No.)	Initial ESC Seq and Lock Shift Function	Code Extension
NumericString	2	None	None	ESC 2/8 4/0 LS0	No
PrintableString	2	None	None	ESC 2/8 4/0 LS0	No
TeletexString (T61String)	102	106	107	ESC 2/8 4/0 LS0 ESC 2/1 4/5 ESC 2/2 4/8	Yes
VideotexString	102	1	73	ESC 2/8 7/5 LS0 ESC 2/1 4/0 ESC 2/2 4/1	Yes
VisibleString (ISO646String)	2	None	None	ESC 2/8 4/0 LS0	No
IA5String	2	1	None	ESC 2/8 4/0 LS0 ESC 2/1 4/0	No
GraphicString	2	None	None	ESC 2/8 4/0 LS0	Yes
GeneralString	2	1	None	ESC 2/1 4/0 LS0 ESC 2/1 4/0	Yes

For example, VideotexString initial G0 set is Primary Teletex Graphic Set (ISO Registration Number 102), initial C0 set is ISO 646 C0 set (ISO Registration Number 1), initial C1 set is Attribute Control Set for Videotex (ISO Registration Number 73), initial escape sequence and locking shift function is ESC 2/8 7/5 LS0, and ESC 2/2 4/1, and code extensions are permitted.

A.10. Use of ASN.1 OctetString as a Character String

<<Editor's Note: Add a description of ODA treatment of character sets.>>

A.11. Escape Sequences for Character Set Designation

This information is extracted from the ISO Register. In some cases, the defaults supplied by ASN.1 make the use of these escape sequences unnecessary. In some cases, this information is carried by application protocol elements.

Graphic Set Designation

Set No.	G0	G1	G2	Name
2	ESC 2/8 4/0			ISO 646 IRV
6	ESC 2/8 4/2			ISO 646 USA
87	ESC 2/4 2/8 4/2	ESC 2/4 2/9 4/2		JIS X0208
100		ESC 2/13 4/1	ESC 2/14 4/1	ISO 8859/1 Right Hand Part
102	ESC 2/8 7/5			CCITT T.61 Primary
103			ESC 2/10 7/6	CCITT T.61 Supp
126		ESC 2/13 4/6		ISO 8859/7 Greek
142			ESC 2/14 4/10	ISO 6937/2 Ad1 Supp

Control Set Designation

Set No.	C0	C1	Name
1	ESC 2/1 4/0		ISO 646 C0
106	ESC 2/1 4/5		CCITT T.61 Primary
107		ESC 2/2 4/8	CCITT T.61 Suppl.

<<Editor's Note: Add 6429 designation.>>

<<Editor's Note: Add DIS 10538 amd DIS 10367?>>

Table of Contents

22 ODA FOR RASTER DOCUMENTS

This is the definition of an Open Document Architecture (ODA) document application profile (DAP) named NIST ODA Raster DAP. This Document Application Profile is suitable for interchanging a document in formatted form. The document contains only raster images. This Document Application Profile has been prepared by the ODA Special Interest Group of the NIST OSI Implementors Workshop (OIW). The Document Application Profile is defined in accordance with IS 8613-1 and CCITT T.411 and follows the standardized proforma and notation defined the proposed Draft Addendum to IS 8613-1 Annex F (to be published). The Document Application Profile is based on ODA as defined in IS 8613 and the Draft Addendum to IS 8613, Part 7.

22.1 SCOPE AND FIELD OF APPLICATION

This document application profile specifies an interchange format suitable for transfer of structured documents between equipment designed for raster processing. Such documents contain only bi-tone raster graphics content, such as engineering drawings and illustrations, although there is no restriction on the minimum size of the image.

This document defines a document application profile that allows large format raster documents to be interchanged in a formatted form in accordance with ISO 8613.

It is assumed that, when negotiation is performed by the service using this document application profile, all non-basic features are subject to negotiation.

This document application profile is independent of the processes carried out in an end system to create, edit, or reproduce raster documents. It is also independent of the means to transfer the document which, for example, may be by means of communication links or storage media.

The features of a document which can be interchanged using this document application profile fall into the following categories:

- o Page format features - these concern how the layout of each page of a document will appear when reproduced;
- o Raster graphics layout and imaging features - these concern how the document content will appear within pages of the reproduced document; and
- o Raster graphics coding - these concern the raster graphics representations and control functions that make up the document raster graphics content.

22.2 REFERENCES

The following references are required in order to implement this document application profile:

ISO 8613-1 - Information processing: Text and Office Systems; Open Document Architecture (ODA) and Interchange Format Part 1: Introduction and General Principles (1989)

ISO 8613-2 - Information processing: Text and Office Systems; Open Document Architecture (ODA) and Interchange Format Part 2: Document Structures (1989)

ISO 8613-4 - Information processing: Text and Office Systems; Open Document Architecture (ODA) and Interchange Format Part 4: Document Profile (1989)

ISO 8613-5 - Information processing: Text and Office Systems; Open Document Architecture (ODA) and Interchange Format Part 5: Open Document Interchange Format (1989)

ISO 8613-7 - Information processing: Text and Office Systems; Open Document Architecture (ODA) and Interchange Format Part 7: Raster Graphics Content Architectures (1989)

ISO 8613-7 - Draft Addendum: Tiled Raster Graphics Addendum to ISO 8613, Part 7 (January 1990)

ISO 8613-1 - Draft Addendum: Document Application Profile Proforma and Notation (to be published)

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

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

CCITT T.6 - Facsimile Coding Schemes and Coding Control Functions for Group 4 Facsimile Apparatus, (1988)

CCITT T.411 Open Document Architecture (ODA) and Interchange Format - Introduction and general principles (1989)

CCITT T.412 Open Document Architecture (ODA) and Interchange Format - Document structures (1989)

CCITT T.414 Open Document Architecture (ODA) and Interchange Format - Document profile (1989)

CCITT T.415 Open Document Architecture (ODA) and Interchange Format - Open document interchange format (1989)

CCITT T.417 Open Document Architecture (ODA) and Interchange Format - Raster graphics content architecture (1989)

CCITT T.503 Document Application Profile for the Interchange of Group 4 Facsimile Documents

22.3 DEFINITIONS AND ABBREVIATIONS

The definitions given in ISO 8613-1 are applicable to this document.

22.4 RELATIONSHIP TO OTHER DAPS

Functionally, this DAP is approximately the same as the CCITT Recommendation T.503, A Document Application Profile for the Interchange of Group 4 Facsimile Documents.

22.5 CONFORMANCE

In order to conform to this document application profile, a data stream representing a document must meet the requirements specified in subclause 22.5.1.

Subclause 22.5.2 specifies the requirements for implementations that originate and/or receive data streams conforming to this document application profile.

22.5.1 Data stream conformance

The following requirements apply to the encoding of data streams that conform to these agreements.

o The data stream shall be encoded in accordance with the ASN.1 encoding rules defined in ISO 8825;

o The data stream shall be structured in accordance with the interchange format defined in subclause 22.8 of this document application profile;

o The document shall be structured in accordance with only the formatted document architecture class specified in subclause 22.7 of this document application profile. In addition, the document shall contain all mandatory constituents specified for that class and may optionally contain constituents permitted for that class as specified in subclause 22.7;

o Each constituent shall contain all those attributes specified as required for that constituent in this profile. Other attributes may be specified provided that they are permitted for that constituent;

o The attributes shall have values within the range of permissible values specified in this profile;

o The encoded document shall be structured in accordance with the abstract document architecture defined in ISO 8613-2;

o The encoded document shall be structured in accordance with the characteristics defined in subclause 22.6 of this document application profile and shall contain only those features defined in subclause 22.6.

22.5.2 Implementation conformance

This subclause states the requirements for implementations claiming conformance to this document application profile.

An implementation claiming to originate and/or receive data streams conforming to this document application profile must complete a conformance test to be defined by a test plan to be developed by DoD and NIST.

A conforming receiving implementation must be capable of receiving any data stream conforming to this document application profile. "Receiving" means not rejecting a data stream conforming to this document application profile and usually, but not always, involves recognizing and further processing the data stream elements. The explicit meaning of "receiving" is determined by a conformance test plan to be developed by DoD and NIST.

22.6 CHARACTERISTICS SUPPORTED BY THIS DAP

This clause describes the characteristics of documents which can be represented by data steams conforming to this profile. This clause also describes how these characteristics are represented in terms of divisional components of the data streams.

22.6.1 Overview

This document application profile describes the features of ISO 8613 that are needed to support the interchange of documents containing only raster graphics content. It specifies interchange formats for the transfer of structured documents with simple layout structures.

This document application profile describes documents which can be interchanged in the formatted form, which facilitates the reproduction of a document as intended by the originator.

Only one category of content is allowed within the document, namely, a raster graphics content in the formatted processable form, which facilitates the reproduction of the document content as intended by the originator or facilitates the revision of the document content.

This subclause describes the layout features that can be represented in documents conforming to this document application profile. The features are described in terms that are typical of the user-perceived capabilities and semantics found in current document image processors.

For the purpose of interchange, a document is represented as a collection of **constituents**, each of which is a set of attributes. The constituents that make up a formatted document are defined below in this subclause and are illustrated in the following figure.

Document Profile
Specific Layout Structure
Presentation Style (optional)
Content Portion Description

Constituents defined as **required** must occur in any document that conforms to this profile. Constituents listed as **optional** may or may not be present in the document, depending on the requirements of the particular document.

The required constituents include:

o a document profile,

o layout object descriptions representing a specific layout structure, and

o content portion description.

The only optional constituent is the presentation style.

22.6.2 Logical Characteristics

Not applicable.

22.6.3 Layout Characteristics

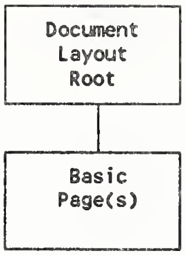
This subclause describes the features of the layout objects that can be represented in documents conforming to this document application profile.

22.6.3.1 Overview of the Layout Characteristics

The document structure allows the document content to be laid out and presented in one or more pages. Each page consist of only a single raster graphics content representing an engineering drawing, illustration, or other raster scanned image.

A specific layout structure of the document conforming to this application profile consists of a two level hierarchy of a document layout root and a set of basic pages. The basic page contains the content information.

The following is a document layout structure derived from this document application profile:



22.6.3.2 DocumentLayoutRoot

A DocumentLayoutRoot is the top level in a document layout structure. A DocumentLayoutRoot may consist of a sequence of one or more BasicPage constituent constraints.

22.6.3.3 BasicPage

A BasicPage is a basic layout object that corresponds to the area used for presenting the raster image content of the document.

22.6.3.3.1 Page Dimensions

A wide variety of page dimensions are supported including large format raster documents. The dimensions of the pages may be specified as any value, in BMU measurement units, including the larger sizes produced from roll paper. These sizes apply to both portrait and landscape orientations and provide the capability to interchange foldout size images. The dimensions may be specified in both portrait and landscape orientations, as well.

Dimensions equivalent to or less than the actual (nominal) page sizes of ANSI E in both portrait and landscape orientations are basic values. Larger dimensions (F-K) including those produced from roll paper are non-basic and their use must be indicated in the document profile. Although ISO A0-A4 sizes are not generally used, the A0-A3 sizes do fall within the range of the ANSI E sizes and therefore could be considered basic values, see table 2.

The default dimensions are the CARA of North American Letter (A). Any default page dimensions may be specified in the document profile subject to the maximum dimensions defined above by using the Page-dimensions attribute. The Page-position attribute may be used to specify the position of the pel array image on the page. Although actual page dimensions may be used allowing for the raster content to completely fill a page leaving no borders, it is advised that the assure reproduction area (ARA) listed in table 1 be used wherever feasible. See ISO 8613-2, subclause 7.3, General rules for positioning pages on presentation surfaces.

22.6.3.3.2 Nominal Page Sizes

The nominal page sizes that may be specified are listed in Table 1. These may be specified in portrait or landscape orientations. All values of nominal page size up to ANSI E size are basic. All sizes larger than ANSI E size and roll paper are non-basic and their use in a document must be indicated in the document profile using the Medium-type attribute, see table 2.

Any of the nominal page sizes defined in Table 1, subject to the restriction specified above, may be specified as the default value in the document profile.

Table 1 also includes the recommended assured reproduction area (ARA). Information loss may occur when a document is reproduced if the dimension of the BasicPage exceed the ARA for the specified nominal page size.

Table 1 Dimensions for Various Page Sizes

Page Type	Size	Size (BMU)	ARA (BMU)
- Metric	(mm)		
ISO-A4	210X297	9920 x 14030	9240 x 13200
ISO-A3	297X420	14030 x 19840	13200 x 18480
ISO-A2	420X594	19840 x 28060	18898 x 27118
ISO-A1	594X840	28060 x 39680	26173 x 37843
ISO-A0	840X1188	39680 x 56120	37843 x 54283
- ANSI, North American(NA)	(inches)		
NA-A	8.5X11	10200 x 13200	9240 x 12400
NA-L	8.5X14	10200 x 16800	9240 x 15480
NA-B	11X17	13200 x 20400	12744 x 19656
NA-C	17X22	20400 x 26400	19500 x 25800
NA-D	22X34	26400 x 40800	25800 x 39600
NA-E	34X44	40800 x 52800	39600 x 52200
NA-F	28X40	33600 x 48000	31400 x 47400
NA-G	11X90	13200 x 108000	12400 x 106800
NA-H	28X143	33600 x 171600	31400 x 170400
NA-J	34X176	40800 x 211200	39600 x 210000
NA-K	40X143	48000 x 171600	47400 x 170400
- Foldouts			
Small	11X14	13200 x 16800	12744 x 15480
NA-B	11X17	(same as NA-B above)	

These page sizes are for the portrait orientation.

Table 2 Layout Attributes

Attributes	Basic Values	Default Values	Non-Basic Values
Dimensions*	NA A-F, L ISO A4-A1 Small Foldout	CARA NA-A	NA G-K ISO A0 11" roll
Medium-type* (Nominal page size)	NA A-F, L ISO A4-A1 Small Foldout	NA-A	NA G-K ISO A0 11" roll

* see Table 1

22.6.4 Document Layout Characteristics

A document layout structure contains only a basic page with content information.

Each raster graphics content must be allocated to only a single basic page.

A page containing tiled raster graphics content may consist of as many tiles as is necessary to represent the image in digital form.

22.6.5 Content Layout and Imaging Characteristics

A document may contain only raster graphics content portions as specified in ISO 8613-7.

22.6.5.1 Raster Graphics Content Architecture

Only the formatted processable raster graphics content architecture class is supported by the profile. The content architecture class associated with a basic page is specified using the document architecture class attribute Content-architecture-class. The default value that must be specified in the document profile is formatted processable raster content architectures.

When using raster graphics content, only one content portion may be associated with a basic page.

22.6.5.2 Raster Graphics Encoding Methods

The CCITT T.6 (untiled), Tiled, and Bitmap encoding methods are supported by this profile as basic. CCITT T.4 one dimensional and CCITT T.4 two dimensional are not supported. Only the CCITT Recommendation T.6 Group 4 compression algorithm shall be used except where it is more efficient to retain an image or tile image in bitmap format or to specify a tile as being either all background or all foreground.

The 'uncompressed' mode of encoding lines in CCITT T.6 encoding method is not supported by this profile. In other words, uncompressed data can not occur within a T.6 encoded data stream. The default mode of 'compressed' is to be used for the attribute Compression indicating that the code extension technique in T.6 encoding is **not** used. Therefore, the Compression attribute will not appear in the description of the raster content.

In a content portion, it is required that both the Number-of-lines and Number-of-pels-per-line parameter of the Coding-attributes attribute be specified. The value of these parameters shall be a positive number. Otherwise, no constraints are placed on these parameters by this profile. This profile places no constraints on the size of the pel arrays that may be used as long as the size does not exceed the page dimension size.

The type of coding method used is specified by the attribute Type-of-coding. The use of this attribute is mandatory in the Document-architecture-defaults of the document profile to define the default value of either T.6 encoding (untiled) or Tiled encoding. The use of this attribute in the description of the content portions is non-mandatory. If this attribute is not specified for a particular content portion, then the default value specified in the Document-architecture-defaults of the document profile is used.

If the Tiled encoding method is used, the default value of 512 for the Number-of-pels-per-tile-line and Number-of-lines-per-tile must be used. No other values are supported, therefore these two attributes do not need to be specified. If the Tile-types attribute is not present, then all tiles will be T.6 encoded. If it is present, then there must be a value specified for each tile in which case only null background, null foreground, T.6 encoded, or bitmap encoded values are supported. T.4 one dimensional and T.4 two dimensional encodings are not supported. There are no restrictions on the use of the Tiling-offset other than that specified in IS 8613-7 Addendum.

See table 3 for a tabulated list of the attributes and their basic, default, and non-basic values.

22.6.5.3 Raster Presentation

Raster presentation is controlled by the presentation attributes specified in ISO 8613-7. This document application profile provides for additional constraints on these presentation attributes as specified below.

The basic Pel-path supported by this profile are 0 and 90 degrees. A Pel-path of 180 and 270 degrees are non-basic.

The basic Line-progression supported by this profile is 270 degrees. A Line-progression of 90 degrees is non-basic.

The basic Pel-spacing supported by this profile are the ratios equal to 6 and 4 BMU between adjacent pels. This corresponds to equivalent resolutions of 200 and 300 pels per 25.4mm (1 in.), respectively when the BMU is interpreted as 1/1200 inch. A value for Pel-spacing other than these ratios are non-basic, i.e., 5, 3, 2, and 1 BMU. This corresponds to equivalent resolutions of 240, 400, 600, and 1200 pels per 25.4mm (1 in.).

There are no restrictions on the use of the Clipping attribute. The Spacing-ratio and Image-dimensions attributes are not supported.

See table 4 for a tabulated list of the attributes and their basic, default, and non-basic values.

Table 3 Content Coding Attributes

Attributes	Basic Values	Default Values	Non-Basic Values
Number-of-pels-per-line	any positive integer	None	None
Number-of-lines	any positive integer	None	None
Tiling-offset*	(any non-neg integer < 512, any non-neg integer < 512)	(0,0)	None
Tile-types*	T.6 encoded bitmap encoded null background null foreground	T.6 encoded	None
Type-of-coding	T.6 encoding (untiled) bitmap (untiled) tiled	T.6 encoding	None

* Only used if Type-of-coding is "tiled"

Table 4 Presentation Attributes

Attributes	Basic Values	Default Values	Non-Basic Values
Pel-path	0, 90 deg	0 deg	180, 270 deg
Line-progression	270 deg	270 deg	90 deg
Pel-spacing	6, 4 SMU (200, 300)	4 SMU (300)	16,12,8,5 3,2,1 SMU
Clipping	Two Coord. Pairs (any non-negative integer, any non-negative integer)	(0,0), (N-1, L-1)	None

22.6.6 Miscellaneous Features

Specification of the attribute Application-comments is optional. When used in conjunction with the Type-of-coding of 'Tiled', it contains a sequence of positive integers, one for each tile in the content portion. The integer is an index representing the octet offset to the beginning of the respective tile starting from the location of the first tile, the first tile will be at offset zero (0). The integers will be sequenced in the same order as the tiles. The tiles will be sequenced primarily in the Pel-path and secondarily in the Line-progression direction as defined by the presentation attributes.

22.6.7 Document Management Features

Every document interchanged in accordance with this document application profile must include a document profile containing information which relates to the document as a whole. The document profile used in this document application profile must identify the contents as raster graphics data.

The features specified by the document profile are listed below. A definition of the information contained in these features is given in the corresponding attribute definitions in ISO 8613-4.

Presence of document constituents:

- o specific layout structure;
- o presentation styles (optional).

Document characteristics:

- o document application profile;
- o document application profile defaults;
- o document architecture class;
- o content architecture class;
- o interchange format class;
- o ODA version date;
- o raster graphics content defaults.

Non-basic document characteristics:

- o page dimensions;
- o medium type;
- o raster graphics presentation features.

The attributes applicable to the document profile are defined in Table 5. The folowing notation is used in the class column of this table:

- m mandatory attribute
- nm non-mandatory attribute
- d defaultable attribute

Capital letters (M, NM, and D) are used for groups of attributes.

Table 5 Document Profile Attributes

Attribute	Class	Permissible Values
Specific-layout-structure	m	present
Presentation-styles	nm	present
Document-characteristics	M	
Document-architecture-class	m	formatted
Document-application-profile	m	{1 3 14 11 0 1 1}
Content-architecture-classes	m	{2 8 2 7 2}
Interchange-format-class	m	B
ODA-version	m	ISO 8613, 1989-07-04
Document-architecture-defaults	M	
Content-architecture-class	m	formatted processable
Type-of-coding	nm	T.6 Encoding (default) Tiled Encoding
Page-dimensions	nm	See list in table 1, (Default value is NA-A, 9240 x 13200 BMU)
Medium-types	nm	See list in table 1, (Default value is NA-A, 9240 x 13200 BMU)
Page-position	nm	any coordinate pair within page
Raster-gr-content-defaults	NM	
Pel-path	nm	90, 180, 270 degrees (0 is normal default)
Line-progression	nm	90 degrees (270 is normal default)
Clipping	nm	any coordinate pair within page

Pel-spacing	nm	6 BMU (200 pels/in.) 5 BMU (240 pels/in.) 3 BMU (400 pels/in.) 2 BMU (600 pels/in.) 1 BMU (1200 pels/in.) (Normal default is 4 BMU (300 pels/in.))
Non-basic-doc-characteristics	NM	
Page-dimensions	nm	See table 1, NA-F through NA-K, roll paper
Medium-types	nm	See table 1, NA-F through NA-K, roll paper
Raster-gr-presentation-features	NM	
Pel-path	nm	180, 270 degrees
Line-progression	nm	90 degrees
Pel-spacing	nm	5 BMU (240 pels/in.) 3 BMU (400 pels/in.) 2 BMU (600 pels/in.) 1 BMU (1200 pels/in.)
Document-management-attributes	M	
Document Reference	m	Any string of characters

22.7 SPECIFICATION OF CONSTITUENT CONSTRAINTS

22.7.1 Document Profile Constraints

22.7.1.1 Macro Definitions

```
-- Basic page dimensions. --
DEFINE(BasicPageDimension,"
    { #horizontal       { <=40800 },       #vertical          { <=52800},
-- Any size equal to or smaller than the actual page size of ISO A1 and ANSI E portrait. --
    | #horizontal       { <=52800 },       #vertical          { <=40800 } }
```

```
-- Any size equal to or smaller than the actual page size of ISO A1 and ANSI E landscape. --
")

-- Non-basic page dimensions. --
DEFINE(NonBasicPageDimensions,"
    { #horizontal       {40801..48000},   #vertical           {52801..211200}
-- Any size larger than the range of basic values in ANSI E portrait and equal to or smaller than the full
size of ANSI K portrait. --
    | #horizontal       {52801..211200},  #vertical           {40801..48000}}
-- Any size larger than the range of basic values in ANSI E landscape and equal to or smaller than the
full size of ANSI K landscape. --
")

DEFINE(NominalPageSizes,"

-- ISO Page Sizes --

    #horizontal         {9920},           #vertical           {14030}
-- ISO A4 Portrait (210mm x 297mm) --
    | #horizontal       {14030},          #vertical           {9920}
-- ISO A4 Landscape (297mm x 210mm) --
    | #horizontal       {14030},          #vertical           {19843}
-- ISO A3 Portrait (297mm x 420mm) --
    | #horizontal       {19843},          #vertical           {14030}
-- ISO A3 Landscape (420mm x 297mm) --
    | #horizontal       {19843},          #vertical           {28063}
-- ISO A2 Portrait (420mm x 594mm) --
    | #horizontal       {28063},          #vertical           {19843}
-- ISO A2 Landscape (594mm x 420mm) --
    | #horizontal       {28063},          #vertical           {39732}
-- ISO A1 Portrait (594mm x 841mm) --
    | #horizontal       {39732},          #vertical           {28063}
-- ISO A1 Landscape (841mm x 594mm) --
    | #horizontal       {39732},          #vertical           {56173}
-- ISO A0 Portrait (841mm x 1189mm) --
    | #horizontal       {56173},          #vertical           {39732}
-- ISO A0 Landscape (1189mm x 841mm) --

-- ANSI Page Sizes --

    | #horizontal       {10200},          #vertical           {13200}
-- ANSI A Portrait (8.5in x 11in) --
    | #horizontal       {13200},          #vertical           {10200}
-- ANSI A Landscape (11in x 8.5in) --
    | #horizontal       {10200},          #vertical           {16800}
-- ANSI Legal Portrait (8.5in x 14in) --
    | #horizontal       {16800},          #vertical           {10200}
-- ANSI Legal Landscape (14in x 8.5in) --
    | #horizontal       {13200},          #vertical           {20400}
```

```
-- ANSI B Portrait (11in x 17in) --
   | #horizontal      {20400},        #vertical        {13200}
-- ANSI B Landscape (17in x 11in) --
   | #horizontal      {20400},        #vertical        {26400}
-- ANSI C Portrait (17in x 22in) --
   | #horizontal      {26400},        #vertical        {20400}
-- ANSI C Landscape (22in x 17in) --
   | #horizontal      {26400},        #vertical        {40800}
-- ANSI D Portrait (22in x 34in) --
   | #horizontal      {40800},        #vertical        {26400}
-- ANSI D Landscape (34in x 22in) --
   | #horizontal      {40800},        #vertical        {52800}
-- ANSI E Portrait (34in x 44in) --
   | #horizontal      {52800},        #vertical        {40800}
-- ANSI E Landscape (44in x 34in) --
   | #horizontal      {33600},        #vertical        {48000}
-- ANSI F Portrait (28in x 40in) --
   | #horizontal      {48000},        #vertical        {33600}
-- ANSI F Landscape (40in x 28in) --
   | #horizontal      {13200},        #vertical        {108000}
-- ANSI G Portrait (11in x 90in) --
   | #horizontal      {108000},       #vertical        {13200}
-- ANSI G Landscape (90in x 11in) --
   | #horizontal      {33600},        #vertical        {171600}
-- ANSI H Portrait (28in x 143in) --
   | #horizontal      {171600},       #vertical        {33600}
-- ANSI H Landscape (143in x 28in) --
   | #horizontal      {40800},        #vertical        {211200}
-- ANSI J Portrait (34in x 176in) --
   | #horizontal      {211200},       #vertical        {40800}
-- ANSI J Landscape (176in x 34in) --
   | #horizontal      {48000},        #vertical        {171600}
-- ANSI K Portrait (40in x 143in) --
   | #horizontal      {171600},       #vertical        {48000}
-- ANSI K Landscape (143in x 40in) --

-- Foldouts --

  | #horizontal       {13200},        #vertical        {16800}
-- Foldout Portrait (11in x 14in) --
  | #horizontal       {16800},        #vertical        {13200}
-- Foldout Landscape (14in x 11in) --
   | #horizontal      {13200},        #vertical        {>= 16801}
-- Any portrait size larger than the typical foldout size (11in x 14in) including 11 inch roll paper --
   | #horizontal      {>= 16801},     #vertical        {13200}
-- Any landscape size larger than the typical foldout size (14in x 11in) including 11 inch roll paper --
")

DEFINE(FDA,"          formatted (0)")
DEFINE(DAC,"
Document-profile{#Document-characteristics
 {#Document-architecture-class}}  ")

DEFINE(FPR,"            {2 8 2 7 2}")  -- Raster formatted processable --
```

22.7.1.2 Constituent Constraints

22.7.1.2.1 DocumentProfile

```
{

-- Presence of document constituents --

$FDA:  REQ    Specific-layout-structure          {'present'};
       PERM   Presentation-styles                {'present'};

-- Document characteristics --

REQ    Document-application-profile    {-- To be supplied --};
                      -- Proposed Object ID is { 1 3 14 11 0 1 1 } --

REQ    Doc-appl-profile-defaults                 {

-- Document architecture defaults --

       REQ    #content-architecture-class       {$FPR},
       PERM   #dimensions                       {$BasicPageDimensions
                                                 $NonBasicPageDimensions},
       PERM   #medium-type                      {
              REQ  #nominal-page-size           {$NominalPageSizes},
              REQ  #side-of-sheet               {ANY_VALUE} },
       PERM   #type-of-coding                   {'T6 encoding'
                                                 | 'tiled encoding'},
       PERM   #page-position                    {ANY_VALUE},
       PERM   raster-gr-contents-defaults       {
              PERM  #pel-path                   {ANY_VALUE},
              PERM  #line-progression           {ANY_VALUE},
              PERM  #pel-spacing                {ANY_RATIO = 6/1 4/1},
              DIS   #compression                {'uncompressed'},
              PERM  #clipping                   {ANY_VALUE},

REQ    Document-architecture-class              {$FDA};
REQ    Content-architecture-classes             {$FPR};
REQ    Interchange-format-class                 {if-b};
REQ    ODA-version
```

```
        {#standard-or-recommendation  {<character-string-constraint>
        ::= "ISO 8613"},
        #publication-date         {<character-string-constraint>
        ::= "1989-07-04"} };
```

-- Non-basic document characteristics --

```
PERM   #Page-dimensions                        {$NonBasicPageDimensions};
PERM   #Medium-types                           {
       REQ      #nominal-page-size             {$NominalPageSize},
       REQ      #side-of-sheet                 {ANY_VALUE},
PERM  #Ra-gr-presentation-features             {
       PERM    #pel-path                       {'180-degrees'
                                               '270-degrees'},
       PERM    #line-progression               {'90-degrees'},
       PERM    #pel-spacing                    {ANY_RATIO <> 6/1 4/1},
       DIS       #compression                  {'uncompressed'}};
```

-- Document management attributes --

```
REQ  Document-reference                        {ANY_VALUE};
```

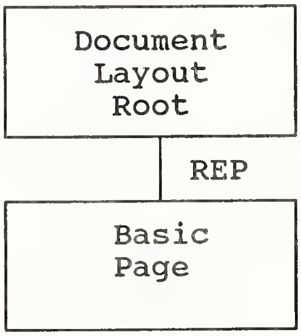
22.7.2 Logical Constituent Constraints

No logical constituents applicable in this subclause.

22.7.3 Layout Constituent Constraints

22.7.3.1 Diagrams of Relationships of Layout Constituents

The notation used for the structure diagrams is that specified in Annex A of ISO 8613-2.



22.7.3.2 Macro Definitions

Note - Construction expressions for repeated basic pages within a root?

22.7.3.3 Factor Constraints

```
FACTOR:        ANY-LAYOUT           {

SPECIFIC:
REQ  Object-type                         {VIRTUAL};
REQ  Object-identifier                   {ANY_VALUE};
PERM Subordinates                        {VIRTUAL};
PERM User-visible-name                   {ANY_VALUE};
PERM User-readable-comment               {ANY_VALUE};
}

FACTOR:        ANY-PAGE    :ANY-LAYOUT  {

SPECIFIC:
REQ    Object-type                       {'basic-page'};
PERM   Dimensions                        {$BasicPageDimensions
                                         $NonBasicPageDimensions};
PERM Page-position                       {ANY_VALUE};
}
```

22.7.3.4 Constituent Constraints

22.7.3.4.1 LayoutDocumentRoot

```
LayoutDocumentRoot          : ANY-LAYOUT          {

SPECIFIC:
REQ    Object-type                       {'document_layout_root'};
REQ    Subordinates                      {SUB_ID_OF(BasicPage)};
}
```

22.7.3.4.2 BasicPage

```
BasicPage                   : ANY-PAGE    {

SPECIFIC:
REQ    Object-type                       {'basic_page'};
PERM   Medium-type                       {#nominal-page-size
                                         {NON_BASIC}, #side-of-sheet
```

```
                                                              {ANY_VALUE}};
PERM   Application-comments                                   {SEQ_INTEGERS};
                                                      -- See subclause 22.8.2 --
PERM   Content-portions                                       {ANY_VALUE};
PERM   Dimensions                                             {#horizontal{
                                                              #fixed{ANY_VALUE}},
                                                              #vertical{#fixed{ANY_VALUE}}
                                                              };
PERM   Position                                               {#fixed{ANY_VALUE}};
PERM   Presentation-style                                     {STYLE_ID_OF(PStyle3};
PERM   Presentation-attributes                                {
       PERM   #raster-attributes                              {
                PERM   Pel-path                               {ANY_VALUE},
                PERM   Line-progression                       {ANY_VALUE},
                PERM   Pel-spacing                            {ANY_VALUE},
                PERM   Clipping                               {ANY_VALUE} } }; }
```

22.7.4 Layout Style Constraints

No layout style constraints applicable in this subclause.

22.7.5 Presentation Style Constraints

22.7.5.1 Macro Definitions

```
DEFINE(R-Pres-Attr,"
PERM   Pel-path                                               {ANY_VALUE};
PERM   Line-progression                                       {ANY_VALUE};
PERM   Pel-spacing                                            {ANY_VALUE};
PERM   Clipping                                               {ANY_VALUE};
 ")
```

22.7.5.2 Factor Constraints

```
FACTOR:          ANY-PRESENTATION-STYLE {
REQ    Presentation-style-identifier                          {ANY_VALUE};
PERM   ser-readable-comments                                  {ANY_VALUE};
PERM   User-visible-name                                      {ANY_VALUE};
}
```

22.7.5.3 Constituent Constraints

22.7.5.3.1 PStyle3

```
PStyle3         :ANY-PRESENTATION-STYLE  {

REQ    Content-architecture-class         {$FPR};
PERM   Presentation-attributes            {$R-Pres-Attr};
}
```

22.7.6 Content Portion Constraints

22.7.6.1 Raster Graphics Content Portion

```
DEFINE(T6,                                "{2 8 3 7 0}")
       DEFINE(Bitmap,                     "{2 8 3 7 3}")
DEFINE(Tiled,                             "{2 8 3 7 5}")

PERM   Content-identifier-layout          {CONTENT_ID_OF(raster-content-portion)};
PERM   Type-of-coding                     {$T6 $Bitmap $Tiled};
PERM   Coding-attributes                  {
PERM   #Number-of-lines                   {ANY_VALUE},
PERM   #Number-of-pels-per-line           {ANY_VALUE},
PERM   #Number-of-pels-per-tile-line      {512},
PERM   #Number-of-lines-per-tile          {512},
PERM   #Tiling-offset                     {ANY_VALUE},
PERM   #Tile-types                        {'null background'
                                          'null foreground'
                                          'T.6 encoded'
                                          'bitmap encoded'}
                                          };
PERM   Content-information                {RASTER};
```

22.7.7 Additional Usage Constraints

No other usage constraints are currently defined.

22.8 INTERCHANGE FORMAT

Interchange format class "B" is to be used in this application profile, as defined in ISO 8613-5.

The encoding is in accordance with the Basic Encoding Rules for Abstract Syntax Notation One (ASN.1), as defined in ISO 8825.

22.8.1 ASN.1 Generation Constraints

The following are additional constraints imposed on the ASN.1 generation beyond those defined in ISO 8824 and ISO 8825.

22.8.2 Encoding of Application Comments

ISO 8613-5 define the encoding of the attribute Application Comments as an octet string. This document application profile requires that the encoding within that octet string be in accordance with the ASN.1 syntax specified
in the following module definition.

```
NISTDAPSpecification
DEFINITION                              ::=      BEGIN
EXPORTS Object-Appl-Comm-Encoding;

Object-Appl-Comm-Encoding  ::=  IMPLICIT SEQUENCE OF
                              INTEGER
END
```

22.8.3 Encoding of Raster Content Information

The encoding of raster content information in the bitmap encoding scheme is that specified in subclause 9.3 of the raster graphics content architecture part of ISO 8613-7, that is, the first pel in the order of bits is allocated to the most significant bit of an octet. The encoding of the code words in the Group 4 facsimile encoding scheme is such that the first or only bit of the first code word shall be placed in the least significant bit of the first octet. Subsequent bits of the first and following code words are placed in the direction of more significant bits in the first and following octets.

Note: DoD is currently encoding of the code words in the most to least significant sequence. A letter has been sent through ANSI/X3V1 to ISO/IEC JTC1/SC18/WG5 requesting a re-evaluation of the bit order. DoD's decision on the matter is pending a reply to the letter.

Table of Contents

23 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 Agreements Document, Version 3 dated June 1990.

☆ U.S. GOVERNMENT PRINTING OFFICE 1990 — 261-913 20635

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

NIST-114A
(REV. 3-89)

U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY

BIBLIOGRAPHIC DATA SHEET

1. PUBLICATION OR REPORT NUMBER
NISTIR 4382

2. PERFORMING ORGANIZATION REPORT NUMBER

3. PUBLICATION DATE
AUGUST 1990

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 NIST, SEE INSTRUCTIONS)
U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
GAITHERSBURG, MD 20899

7. CONTRACT/GRANT NUMBER

8. TYPE OF REPORT AND PERIOD COVERED
FINAL

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (STREET, CITY, STATE, ZIP)
Same as item 6.

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 Working Agreements on Implementation details of 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 (6 TO 12 ENTRIES; ALPHABETICAL ORDER; CAPITALIZE ONLY PROPER NAMES; AND SEPARATE KEY WORDS BY SEMICOLONS)

Local area networks; network protocols; NIST/OSI Workshop; open systems interconnection

13. AVAILABILITY

☒ UNLIMITED

□ FOR OFFICIAL DISTRIBUTION. DO NOT RELEASE TO NATIONAL TECHNICAL INFORMATION SERVICE (NTIS).

□ ORDER FROM SUPERINTENDENT OF DOCUMENTS, U.S. GOVERNMENT PRINTING OFFICE, WASHINGTON, DC 20402.

☒ ORDER FROM NATIONAL TECHNICAL INFORMATION SERVICE (NTIS), SPRINGFIELD, VA 22161.

14. NUMBER OF PRINTED PAGES
435

15. PRICE
A19

ELECTRONIC FORM