NISTIR 4507

WORKING IMPLEMENTATION AGREEMENTS FOR OPEN SYSTEMS INTERCONNECTION PROTOCOLS

Based on the proceedings of the
NIST Workshop for Implementors of OSI
Plenary Assembly Held December 14, 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 4507

WORKING IMPLEMENTATION AGREEMENTS FOR OPEN SYSTEMS INTERCONNECTION PROTOCOLS

Based on the proceedings of the
NIST Workshop for Implementors of OSI
Plenary Assembly Held December 14, 1990
National Institute of Standards and
›chnology
Gaithersburg, MD 20899

Tim Boland, Editor

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

December 1990

Issued March 1991

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

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 1 - General Information

Output from the December 1990 NIST Workshop for Implementors of OSI

Workshop Chair: **Tim Boland**
Editor: **Brenda Gray**

Table of Contents

Foreword

This part of the Working Implementation Agreements was prepared by the Chair of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI).

Text in this part has been approved by the Plenary of the Workshop. This part replaces the previously existing chapter on this subject.

Part 1 - General Information

0 Introduction

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.

1 Scope

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

> - The companion document, "Stable Implementation Agreements for Open Systems Interconnection Protocols, Version 4 as of December 1990" records mature agreements considered advanced enough for use in product development or procurement reference.

This document supersedes the previous Working Document.

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.

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 part one). In rare exceptions, text that does not represent implementation agreements may appear temporarily; text will always be appropriately marked.

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 September 1990. Where appropriate, statements related to backward compatibility, interworking considerations, or agreement maintenance are given in this document. Architectural issues may also be considered as appropriate.

> **NOTE -** In this document, references are maintained in the individual parts as appropriate. Additional references for all of the subjects covered in this document may be found in the aligned parts of the Stable Implementation Agreements Document, Version 4, Edition 1 dated December 1990.

2 Normative References

3 Definitions

4 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.

5 Workshop Organization

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

6 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)."

7 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.

8 Structure and Operation of the Workshop

8.1 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.

8.2 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:

a) 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);

b) The SIG 2 chairperson should bring the matter before SIG 2 for action;

c) SIG 2 should respond to the concerns or needs of SIG 1 or the individual in a timely manner;

d) 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;

e) 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.

8.2.1 FTAM SIG

The charter is given as follows:

a) Scope:

1) to develop stable FTAM Agreements between vendors and users for the implementation of interoperable products;

2) 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;

3) to act as Registration Authority for OIW FTAM objects;

4) to define further FTAM functionality;

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

6) 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;

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

b) High priority work items:

1) Maintain FTAM Phase 2 and Phase 3 Agreements;

2) Maintain OIW FTAM object register;

3) Contribute to development of FTAM ISPs;

4) Specify use of general Character Set Agreements;

5) Specify requirements of FTAM to a Directory Service;

6) Specify use of Filestore Management functions;

c) Low priority work items:

1) Specify use of Security functions;

2) Specify use of Overlapped Access;

3) Specify use of ODA documents over FTAM;

4) Specify use of EDI documents over FTAM.

8.2.2 X.400 (MESSAGE HANDLING SYSTEMS) SIG

The charter is given as follows:

a) Scope of Work:

1) To develop Stable MHS Agreements among Vendors and Users for the implementation of interoperable products;

2) 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;

3) 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;

4) To Review Abstract Tests for X.400 and MOTIS and provide feedback to appropriate bodies;

b) Current Work Items:

1) MHS use of X.500 Directory;

2) Body Parts / Content Types;

3) MHS Security Issues;

4) Access Units;

5) MHS Registration Issues;

6) Maintain 1984 MHS Stable Agreements;

7) Contribute to development of MHS ISPs;

8) MHS routing;

c) Future Work Items for Next Year:

1) EDI over X.400 and MOTIS;

2) Distribution Lists over X.400 and MOTIS;

3) EDI Messaging;

4) MHS Management;

5) Character Sets and other Internationalization Considerations.

8.2.3 LOWER LAYER SIG

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

The objectives of the Lower Layer SIG are:

a) 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.;

b) Produce and maintain recommendations for implementation of these layers;

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

d) 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).

8.2.4 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:

a) to develop agreements based on IS/DIS;

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

c) to draft contributions on proposed NWIs;

d) to register security objects;

e) to provide consultancy to other SIGs;

f) to act as a well-focused group as follows:

1) to propagate security information;

2) to recommend and coordinate activities.

8.2.5 DIRECTORY SERVICES SIG

The charter is as follows:

a) To achieve the general goal of:

1) The production and promotion of·functional implementation agreements based on ISO/CCITT specifications for Directory Services in accordance with the goals and objectives of the OSI Implementors' Workshop;

b) To reach the above goal by fulfilling the objectives of:

1) providing and maintaining functional implementation agreements for Directory Services in the form of Stable and Working OIW Implementors' Agreements;

2) serving in a leadership role in the development of an International Standardized Profile for Directory Services;

3) Developing Agreements on security issues as related to Directory Services;

4) serving in a consultative role to the other SIGs in the use of Directory Services by other OSI Applications;

5) registering Directory Services objects as necessary to accomplish the other objectives of the SIG.

8.2.6 VIRTUAL TERMINAL SIG

The charter is as follows:

a) Scope:

1) 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;

b) Objectives:

1) Develop VTE-profiles to support diverse interactive applications and environments;

2) Develop Control Objects which may be referenced and used within VTE-profiles;

3) Register and maintain OIW VT objects;

4) Conduct liaison with standards organizations, other regional workshops and vendor/user groups as necessary;

5) Review and, if necessary, generate abstract test cases for VTE-profiles;

6) Harmonize OIW VTE-profiles with those from other regional workshops;

7) Adopt ISP format for OIW VTE-profiles under development;

8) Migrate existing OIW VTE-Profiles to ISP format;

9) Develop X-OSI Implementors' Agreement, if necessary;

10) Register and Maintain OIW X-OSI Objects;

11) Adopt ISP Format for OIW X-OSI Implementors' Agreements, if necessary;

12) Review and, if necessary, generate abstract test cases for X-windows;

c) High Priority:

1) Maintain stabilized OIW VTE-profiles and Control Objects;

2) Develop fully general TELNET profile in ISP format;

3) Develop Scroll Profile in ISP format;

d) Low Priority:

1) Develop abstract test cases;

2) Develop Page profile;

3) Migrate stable profiles to ISP format - Forms, TELNET, X.3, Transparent.

8.2.7 UPPER LAYERS SIG

The charter of the Upper Layers SIG is as follows:

a) Develop product level specifications for the implementation of:

1) Session service and protocol;

2) Presentation service and protocol;

3) ACSE service and protocol;

b) Remote Operations Service Element (ROSE);

c) Reliable Transfer Service Element (RTSE);

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;

The specifications developed will be done to support the requirements of all ASE SIGs;

The objectives of the Upper Layers SIG are to:

a) Study OSI Session, Presentation, ACSE, ROSE, RTSE and CCR;

b) Produce and maintain recommendations for implementations of these layers;

c) Where necessary provide input to the relevant standards bodies concerning Session, Presentation, ACSE, ROSE, RTSE, and CCR;

d) 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:

a) Align implementation agreements with other organizations such as EWOS, AOW, and ISO;

b) Develop implementor's agreements that promote the efficiency of protocol implementations;

c) Develop implementor's agreements that promote ease in the verification of interoperability;

d) Develop necessary conformance statements.

8.2.8 NETWORK MANAGEMENT SIG

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

The SIG will:

a) Agree upon NIST Implementors OSI systems management reference model;

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

c) 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;

d) 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:

a) Establish liaisons with various standards bodies;

b) Provide feedback for additional/enhanced services and protocols for OSI management.

8.2.9 OFFICE DOCUMENT ARCHITECTURE

The charter is as follows:

a) Scope:

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

b) Objectives:

1) Develop ODA document application profiles to support a diverse set of applications and environments;

2) Register and maintain ODA document application profiles;

3) Conduct liaison with standards organizations, other groups developing ODA document application profiles, vendor/user groups and testing authorities as necessary;

4) Review and, if necessary, generate abstract test cases for ODA document application profiles;

5) Harmonize OIW ODA document application profiles with those from other international groups;

6) Participate, as necessary, in the ISO ISP processing of FOD-type profiles;

c) High Priority:

1) Develop and maintain OIW ODA document application profiles;

2) Harmonize OIW ODA document application profiles with other international groups;

3) Assist in the progression of OIW ODA document application profiles through the ISO ISP process;

d) Low Priority:

1) Develop abstract test cases;

2) Integrate addenda and extensions to the base standard into OIW ODA document application profiles;

3) Develop awareness of ODA in vendor and user groups.

NOTE - The Registration SIG has effectively completed its work. The charter items below may be removed in the future.

8.2.10 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;

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

a) Define policies and procedures for the registration of objects defined by the NIST OSI Workshop;

b) Take account of currently existing OSI Workshop registration work;

c) Establish policies for the publication and promulgation of registered objects;

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

2) 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.

8.2.11 TRANSACTION PROCESSING SIG

The charter is as follows:

a) reduce TR10000-format OSI TP Profile;

b) Describe TP's use of other profile services: ACSE, CCR, Pres., Dir.;

c) Produce CCR profile covering TP requiremnts;

d) Liaise with other internal and external organizations as required;

e) Communicate with EWOS and AOW to reach goal of an aligned profile;

f) Act as registration authority for OIW TP objects, as necessary.

8.2.12 MANUFACTURING MESSAGE SPECIFICATION (MMS) SIG

The charter is as follows:

a) Scope:

1) To create an open forum for discussion and agreements pertaining to MMS and issues related to MMS;

b) Objectives:

1) To produce agreements for implementations of MMS (ISO 9506);

2) 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;

3) To produce implementation agreements on MMS Companion Standards (as recognized by ISO TC184/SC5/WG2) after those have reached ISO DIS or equivalent status;

4) Develop Conformance requirements;

5) Develop recommendations on MMS testing;

c) As Necessary:

1) Respond to defect reports as accepted;

2) Provide feedback on Addendum material;

3) 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;

4) Provide input on ISP for MMS when the ISO process for it is defined;

d) High Priority Work Items:

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

2) Produce a set of implementation agreements appropriate to that initial subset of MMS encompassing the objectives;

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

methodologies;

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

5) 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;

e) Low Priority Work Items:

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

2) Develop subsequent subsets of MMS;

3) Produce a set of implementors agreements for the subsequent subsets;

4) Provide input on Test Cases for the subsequent subsets;

5) Provide input on errors for the subsequent subsets;

6) Provide input to ISO on MMS ASE specific management entities.

8.2.13 REMOTE DATABASE ACCESS SIG

The charter is as follows:

a) Scope:

1) For all RDA Implementations based on ISO 9579:

a) Develop Implementors' agreements;

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

c) Coordinate with other organizations on matters relevant to RDA;

2) Objectives:

a) 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;

b) Provide input to ANSI and ISO on the specification of an RDA ISP;

3) High Priority Work Items:

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

1) review ULA agreements affecting RDA implementations;

2) specify limits on encodings in RDA pdus;

3) specify minimum conformance requirements for RDA implementations;

4) identify and describe recommended practices in the implementation of RDA services and protocols;

5) identify implementor defined items in ISO 9075 (SQL) affecting interoperability in an OSI environment;

6) identify implementor defined items in ISO 9579 (RDA) affecting interoperability;

7) identify RDA implementation requirements for CCR and TP;

8) harmonize ULA requirements with SQL requirements with respect to handling of variant character sets in RDA;

4) Low Priority Work Items:

a) Future RDA specializations, if any.

9 Points of Contact

OSI Workshop -Chairman	Tim Boland	NIST	(301) 975-3608
OSI Workshop - Registration	Brenda Gray	NIST	(301) 975-3664
Directory Services	You-Bong Weon-Yoon	AT&T Bell Labs	(201) 522-5073
FTAM SIG	Darryl Roberts	Unisys NCG	(805) 499-6698
Lower Layers SIG	Fred Burg	AT&T	(201) 949-0919
Manufacturing Message Secification (MMS) SIG	Herbert Falk	SISCO	(313) 774-0020
Network Management SIG - Co-Chairs	Paul Brusil George Mouradian	Mitre &T Bell Labs	(617) 271-7632 (201) 949-7671
ODA SIG	Frank Dawson	IBM	(214) 556-5052
Remote Database Access SIG	Peter Eng	IMB Canada	(416) 448-3087
Security SIG	James Galvin	Trusted Info. Systems	(301) 854-6889
Technical Liaison Committee	Einar Stefferud	NMA-Northrop	(714) 841-3711
Transaction Processing SIG	Jeff Hildebrand	Boeing Computer Services	(206) 865-4893
Upper Layers SIG	Mark Thomas	AT&T Bell Labs	(201) 522-6671
Virtual Terminal SIG	Luke Lucas	Control Data Corporation	(612) 482-2874
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

10 Profile Conformance

See Stable Implementation agreements.

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 2 - Subnetworks

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair: **Fred Burg**

Table of Contents

Foreword

This part of the Working Implementation Agreements was prepared by the Lower Layers Special Interest Group (LLSIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the Workshop. This part replaces the previously existing chapter on this subject. There are some significant technical changes to this text as previously given.

Part 2 - Subnetworks

Editor's Note: All references to Stable Agreements in this Section are to Version 4 dated December 1990.

0 Introduction

(Refer to Stable Implementation Agreements Document)

1 Scope

(Refer to Stable Implementation Agreements Document)

2 Normative References

3 Status

This material is current as of December 14, 1990.

4 Errata

Errata are reflected in replacement pages of Version 4, Stable Document, dated December 1990.

5 Local Area Networks

(Refer to Stable Implementation Agreements Document)

5.1 IEEE 802.2 Logical Link Control

(Refer to Stable Implementation Agreements Document)

5.2 IEEE 802.3 CSMA/CD Access Method

(Refer to Stable Implementation Agreements Document)

> **Editor's Note** - The following text will be added to the Stable Implementation Agreements Document at the end of 5.2.

The following implementation agreements have been reached with respect to 10 BASE-T networks:

a) Auto-partition: A repeater port which connects 10 BASE-T links either through an external or internal MAU shall implement the auto-partition and reconnections algorithm as defined in IEEE 802.3, Chapter 9.

5.3 IEEE 802.4 Token Bus Access Method

(Refer to Stable Implementation Agreements Document)

5.4 IEEE 802.5 Token Ring Access Method

(Refer to Stable Implementation Agreements Document)

5.5 Fiber Distributed Data Interface (FDDI)

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

(Refer to Stable Implementation Agreements Document)

Further study is needed to confirm whether a lower default value or range for T_Req would be useful.

5.5.2 Token Ring Physical Layer (PHY,X3.148-1988)

(Refer to Stable Implementation Agreements Document)

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

(Refer to Stable Implementation Agreements Document)

6 X.25 Wide Area Networks

6.1 CCITT Recommendation X.25

(Refer to the Stable Implementation Agreements Document).

6.2 ISO 7776

(Refer to the Stable Implementation Agreements Document).

6.3 ISO 8208

(Refer to the Stable Implementation Agreements Document).

7 Integrated Services Digital Networks (ISDN)

7.1 Introduction

(Refer to the Stable Implementation Agreements Document).

7.2 Implementation Agreements

(Refer to the Stable Implementation Agreements Document).

7.2.1 Physical Layer, Basic Access at "U"

(Refer to the Stable Implementation Agreements Document).

7.2.2 Physical Layer, Basic Access at S and T

(Refer to the Stable Implementation Agreements Document).

7.2.3 Physical Layer, Primary Rate at "U"

(Refer to the Stable Implementation Agreements Document).

7.2.4 Data Link Layer, D-Channel

(Refer to the Stable Implementation Agreements Document).

7.2.5 Signaling

(Refer to the Stable Implementation Agreements Document).

7.2.6 Data Link Layer B-Channel

(Refer to the Stable Implementation Agreements Document).

7.2.7 Packet Layer

(Refer to the Stable Implementation Agreements Document).

Annex A (informative)

(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.)

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 3 - Network Layer

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair: **Fred Burg**

Table of Contents

Foreword

This part of the Working Implementation Agreements was prepared by the Lower Layers Special Interest Group (LLSIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the Workshop. This part replaces the previously existing chapter on this subject. There are some significant technical changes to this text as previously given.

Part 3 - Network Layer

Editor's Note: All references to Stable Agreements in this Section are to Version 4 dated December 1990.

0 Introduction

(Refer to Stable Implementation Agreements Document)

1 Scope

(Refer to Stable Implementation Agreements Document)

2 Normative References

3 Status

This material is current as of December 14, 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 March 1991.

4 Errata

Errata are reflected in pages of Version 4, Stable Document, dated December 1990.

5 Connectionless-Mode Network Service (CLNS)

5.1 ISO 8473

Subsets of the protocol:

(Refer to the Stable Implementation Agreements Document).

Mandatory Functions:

(Refer to the Stable Implementation Agreements Document).

Optional Functions:

(Refer to the Stable Implementations Agreements document).

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

a) NPDUs shall be scheduled based on the priority functions of ISO 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:

1) 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);

2) A minimum flow shall be provided for lower priority PDUs.[1];

b) 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:

1) "high reserved" priority will be encoded with value 14 (0000 0000 0000 1110);

2) "high" priority will be encoded with value 10 (0000 0000 0000 1010);

3) "normal" priority will be encoded with value 5 (0000 0000 0000 0101);

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

5) 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";

c) 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;

d) 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.

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.

5.3 Provision of CLNS over X.25 Subnetworks

(Refer to the Stable Agreements Document)

5.4 Provision of CLNS over ISDN

(Refer to the Stable Implementation Agreements document).

5.5 Provision of CLNS over Point-to-Point Links

(To be based on ISO 8880)

6 Connection-Mode Network Service

6.1 Mandatory Method of Providing CONS

6.1.1 General

(Refer to the Stable Implementation Agreements document).

6.1.2 X.25 WAN

(Refer to the Stable Implementation Agreements document).

6.1.3 LANs

(Refer to the Stable Implementation Agreements document).

6.1.4 ISDN

(Refer to the Stable Implementation Agreements document).

6.1.5 Priority

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

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

(Refer to the Stable Implementation Agreements Document)

6.3 Agreements on Protocols

(Refer to the Stable Implementation Agreements Document)

6.3.1 ISO 8878

(Refer to the Stable Implementation Agreements Document.)

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

(Refer to the Stable Implementation Agreements Document)

6.4 Interworking

(Refer to the Stable Implementation Agreements Document.)

7 Addressing

Refer to the Stable Implementations Agreements Document

Within routing domains intending to operate using the IS -IS Intradomain Routing Protocol defined in ISO/DIS 10589, 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.

8 Routing

8.1 ISO 9542 End System to Intermediate System Routing

(Refer to the Stable Implementation Agreements Document.)

8.2 ISO 10030 End System to Intermediate System Routing

(Refer to the Stable Implementation Agreements Document.)

The following agreements apply to the use of ISO 10030:

a) The multicast addresses corresponding to "All CONS End Systems" and "All CONS SNAREs" shall default to the following on IEEE 802.3 and IEEE 802.4 subnetworks:

All CONS End Systems	= 01-80-C2-00-00-16
All CONS SNAREs	= 01-80-C2-00-00-17

8.3 Intra-Domain Intermediate Systems to Intermediate Systems Routing

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

The following agreements apply to the use of ISO DIS 10589:

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

8.4 Inter-Domain Intermediate Systems to Intermediate Systems Routing

(Refer to Stable Implementation Agreements Document.)

9 Procedures for OSI Network Service/Protocol Identification

9.1 General

(Refer to the Stable Implementation Agreements document).

9.2 Processing of Protocol Identifiers

(Refer to the Stable Implementation Agreements document).

9.2.1 Originating NPDUs

(Refer to the Stable Implementation Agreements document).

9.2.2 Destination System Processing

(Refer to the Stable Implementation Agreements document).

9.2.3 Further Processing in Originating End System

(Refer to the Stable Implementation Agreements document).

9.3 Applicable Protocol Identifiers

(Refer to the Stable Implementation Agreements document.)

10 Migration Considerations

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

11 Use of Priority[2]

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.).

[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

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.

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 3.5.1; for Transport Layer see 4.5.1.2.6, and for Upper Layers the clause will be included at a later date).

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:

a) low priority

b) normal priority

c) high priority

d) 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).

12 Conformance

(Agreements to be added at a later date)

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 4 - Transport Layer

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair: **Fred Burg**

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 4 - Transport Layer

[illegible]

[illegible]

Table of Contents

Foreword

This part of the Working Implementation Agreements was prepared by the Lower Layers Special Interest Group (LLSIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the Workshop. This part replaces the previously existing chapter on this subject. There are some significant technical changes to this text as previously given.

Part 4 - TRANSPORT LAYER

Editor's Note: All references to Stable Agreements in this Section are to Version 4 dated December 1990.

0 Introduction

(Refer to Stable Implementation Agreements Document)

1 Scope

(Refer to the Stable Implementation Agreements document).

2 Normative References

3 Status

This material is current as of December 1990.

4 Errata

Errata are reflected in pages of Version 4, Stable Document, dated December 1990.

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.

5 Provision of Connection Mode Transport Services

(Refer to the Stable Implementation Agreements document).

5.1 Transport Class 4

5.1.1 Transport Class 4 Overview

(Refer to the Stable Implementation Agreements document).

5.1.2 Protocol Agreements

5.1.2.1 General Rules

(Refer to the Stable Implementation Agreements Document.)

5.1.2.2 Transport Class 4 Service Access Points or Selectors

(Refer to the Stable Implementation Agreements Document.)

5.1.2.3 Retransmission Timer

(Refer to Stable Implementation Agreements Document)

5.1.2.4 Keep-Alive Function

(Refer to the Stable Implementation Agreements Document.)

5.1.2.5 Congestion Avoidance Policies

(Refer to the Stable Implementation Agreements Document).

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 part 3 clause 11 shall always be used in an instance of communications. In other words an explicit priority parameter shall be sent.

Additional requirements of systems implementing priority are defined below:

a) 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];

b) 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;

[1] Refer to part 3 clause 11 for an overview on the use of priority.

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

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

1) "high reserved" priority will be encoded with value "zero" (0000 0000 0000 0000),

2) "high" priority will be encoded with value 5 (0000 0000 0000 0101),

3) "normal" priority will be encoded with value 10 (0000 0000 0000 1010),

4) "low" priority will be encoded with value 14 (0000 0000 0000 1110)

d) 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";

e) The exchange of priority parameters by Transport entities is performed as described below[4]:

1) 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.

2) 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. 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.

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

4) 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

[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.

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.

5.2 Transport Class 0

(Refer to Stable Implementation Agreements Document)

5.2.1 Transport Class 0 Overview

(Refer to Stable Implementation Agreements Document)

5.2.2 Protocol Agreements

5.2.2.1 General Rules

(Refer to Stable Implementation Agreements Document)

5.2.2.2 Transport Class 0 Service Access Points

(Refer to Stable Implementation Agreements Document)

5.2.3 Rules for Negotiation

(Refer to Stable Implementation Agreements Document.)

5.3 Transport Class 2

(Refer to Stable Implementation Agreements Document.)

5.3.1 Transport Class 2 Overview

(Refer to Stable Implementation Agreements Document.)

5.3.2 Protocol Agreements

(Refer to Stable Implementation Agreements Document)

6 Provision of Connectionless Transport Service

(Refer to Stable Implementation Agreements Document.)

7 Transport Protocol Identification

(Refer to the Stable Implementation Agreements document.)

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 5 - Upper Layers

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair: **Mark Thomas**
SIG Editor: **Scott Beale**

Table of Contents

Foreword

This part of the Working Implementation Agreements was prepared by the Upper Layers Special Interest Group (ULIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the above-mentioned Workshop. This part replaces the previously existing chapter on this subject.

Annexes A, B, and C are for information purposes only.

Part 5 - Upper Layers

Editor's Note - All references to Stable Agreements in this section are to Version 4 dated December 1990.

0 Introduction

(Refer to Stable Agreements Document)

1 Scope

(Refer to Stable Agreements Document)

2 Normative References

Information Processing Systems - Open Systems Interconnection - Service Definition for the Association Control Service Element - Addendum 1: Peer-Entity Authentication During Association Establishment, ISO 8649/DAD1 (ISO/IEC JTC1/SC21 N3771)

Information Processing Systems - Open Systems Interconnection - Protocol Specification for the Association Control Service Element - Addendum 1: Peer-Entity Authentication During Association Establishment, ISO 8650/DAD1 (ISO/IEC JTC1/SC21 N3772)

3 Status

This version of the upper layer agreements is under development.

4 Errata

4.1 ISO Defect Solutions

ISO 8649 defect solutions:
002

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

(Refer to Stable Agreements Document)

5 Association Control Service Element

5.1 Introduction

(Refer to Stable Agreements Document)

5.2 Services

(Refer to Stable Agreements Document)

5.3 Protocol Agreements

5.3.1 Application Context

(Refer to Stable Agreements Document)

5.3.2 AE Title

(Refer to Stable Agreements Document)

5.3.3 Peer Entity Authentication

If supported, peer-entity authentication during association establishment shall be implemented as specified in Addendum 1 to ISO 8650 (ISO 8650/DAD1).

5.4 ASN.1 Encoding Rules

(Refer to Stable Agreements Document)

5.5 Connectionless

(Refer to Stable Agreements Document)

6 ROSE

(Refer to Stable Agreements Document)

7 RTSE

(Refer to Stable Agreements Document)

8 Presentation

8.1 Introduction

(Refer to Stable Agreements Document)

8.2 Service

(Refer to Stable Agreements Document)

8.3 Protocol Agreements

8.3.1 Transfer Syntaxes

(Refer to Stable Agreements Document)

8.3.2 Presentation Context Identifier

(Refer to Stable Agreements Document)

8.3.3 Default Context

(Refer to Stable Agreements Document)

8.3.4 P-Selectors

(Refer to Stable Agreements Document)

8.3.5 Provider Abort Parameters

(Refer to Stable Agreements Document)

8.3.6 Provider Aborts and Session Version

(Refer to Stable Agreements Document)

8.3.7 CPC-Type

(Refer to Stable Agreements Document)

8.3.8 Presentation-context-definition-result-list

(Refer to Stable Agreements Document)

8.3.9 RS-PPDU

(Refer to Stable Agreements Document)

8.4 Presentation ASN.1 Encoding Rules

8.4.1 Invalid Encoding

(Refer to Stable Agreements Document)

8.5 General

8.5.1 Presentation Data Value (PDV)

(Refer to Stable Agreements Document)

8.6 Connection Oriented

(Refer to Stable Agreements Document)

8.7 Connectionless

(Refer to Stable Agreements Document)

9 Session

9.1 Introduction

(Refer to Stable Agreements Document)

9.2 Services

(Refer to Stable Agreements Document)

9.3 Protocol Agreements

9.3.1 Concatenation

(Refer to Stable Agreements Document)

9.3.2 Segmenting

(Refer to Stable Agreements Document)

9.3.3 Reuse of Transport Connection

(Refer to Stable Agreements Document)

9.3.4 Use of Transport Expedited Data

(Refer to Stable Agreements Document)

9.3.5 Use of Session Version Number

(Refer to Stable Agreements Document)

9.3.6 Receipt of Invalid SPDUs

(Refer to Stable Agreements Document)

9.3.7 Invalid SPM Intersections

(Refer to Stable Agreements Document)

9.3.8 S-Selectors

(Refer to Stable Agreements Document)

9.4 Connectionless

(Refer to Stable Agreements Document)

10 Universal ASN.1 Encoding Rules

10.1 Tags

(Refer to Stable Agreements Document)

10.2 Definite Length

(Refer to Stable Agreements Document)

10.3 External

(Refer to Stable Agreements Document)

10.4 Integer

(Refer to Stable Agreements Document)

10.5 String Types

(Refer to Stable Agreements Document)

10.6 Bit String

(Refer to Stable Agreements Document)

11 Character Sets

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

12 Conformance

(Refer to Stable Agreements Document)

13 Specific ASE Requirements

13.1 FTAM Phase 2

(Refer to Stable Agreements Document)

13.2 MHS

(Refer to Stable Agreements Document)

13.3 DS Phase 1

(Refer to Stable Agreements Document)

13.4 Virtual Terminal

(Refer to Stable Agreements Document)

13.5 MMS

13.5.1 ACSE Requirements

ACSE Functional Units: Kernel

Application Context: "ISO MMS" {ISO(1) Standard(0) 9506 Part(2) mms-application-context-version 1(3)} - implies use of ACSE and MMS ASE

13.5.2 Presentation Requirements

Presentation Functional Units: Kernel

At least 2 Presentation Contexts must be supported.

Abstract Syntaxes:

a) "mms-abstract-syntax-major-version 1" {ISO(1) Standard(0) 9506 Part(2) mms-abstract-syntax-major-version 1 (1)}

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

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

13.5.3 Session Requirements

Session Functional Units:

a) Kernel

b) Duplex

Version Number: 2

Maximum size of User Data parameter field: 10,240

13.6 Transaction Processing

13.6.1 ACSE Requirements

ACSE Functional Units: Kernel

The application context is user-defined.

13.6.2 Presentation Requirements

Presentation Functional Units: Kernel

Presentation Contexts:

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

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

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

Associated Transfer Syntax:

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

b) "ISO 10026-TP" { joint-iso-ccitt(2) transaction-processing(?) abstract-syntax(2) tp-apdus(1) }

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

d) At least one user-defined abstract syntax.

13.6.3 Session Requirements

Session Functional Units:

a) kernel

b) duplex

c) 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

13.7 Network Management

13.7.1 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

13.7.2 ACSE Requirements

ACSE Functional Units: kernel

Application Contexts: as defined by [SMO]

AE-Title: The association responder shall support both forms of the AE-Title. The association requestor may use either form of the AE-Title.

13.7.3 Presentation Requirements

Presentation Functional Units: kernel

Presentation Contexts: At least 2 must be supported.

Abstract Syntaxes:

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

b) "CMIP-PCI" {joint-iso-ccitt(2) ms(9) cmip(1) cmip-pci(1) abstractSyntax(4)}

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

13.7.4 Session Requirements

Session Functional Units:

a) kernel

b) duplex

Version Number: 2

Maximum size of User Data parameter field: 10,240

Annex A (informative)

Recommended Practices

(Refer to Stable Agreements Document)

Annex B (informative)

Object Identifier Register

B.1 Register Index

(Refer to Stable Agreements Document)

B. 2 Object Identifier Descriptions

(Refer to Stable Agreements Document)

Annex C (informative)

Backward Compatibility

	Version & Section	
Issue	Changed	Backward Compatibility
Restrictions on minimum number of octets implementations shall be able to receive.	V1E2 5.5.3.2	Interworking problems may occur, since implementations could send more than 128 octets. [An implementation that conforms to versions previous to V1E2 as an initiator and V3E1 as a responder will be able to interoperate.]
Agreements on AE Title, AP Title, and AE Qualifier changed.	V1E3 section 5.5.3.3 & V1E4 section 5.5.3.3	Interworking problems may occur between implementations that expect different forms of AP itle and AE Qualifier to be used. [Implementations that accept any form of these parameters will interwork with initiators that conform to earlier versions.]
Restrictions on encoding of "Presentation Context Identifier."	V2E1 section 5.8.3.3	Interworking problems may occur since implementations could encode negative numbers. [An implementation that conforms to versions previous to V2E1 as a responder and V3E1 as an initiator will be able to interoperate.]
Mode selector as first element in set	V1E4 section 5.6.3.4	This will cause interworking problems for those implementations that don't encode "mode selector" as the first element in the set. [An implementation that conforms to versions previous to V1E4 as an initiator and V3E1 as a responder will be able to interoperate.]

Version & Section		
Issue	Changed	Backward Compatibility
Restrictions on encoding of "protocol version" and "presentatation requirements."	V2E1 section 5.8.4.2	This will cause interworking problems for those implementations expecting "protocol version" and "presentation requirements" to be encoded in the primitive form. [An implementation that conforms to versions previous to V2E1 as an initiator and V3E1 as a responder will be able to interoperate.]
Restrictions on encoding of "presentation selector."	V2E1 section 5.8.4.3	This will cause interworking problems for those implementations expecting "presentation selector" to be encoded in the primitive form. [An implementation that conforms to versions previous to V2E1 as an initiator and V3E1 as a responder will be able to interoperate with either version.]
Use of default values for Minor syncpoint changed.	V2E3 section 5.11.1.1.1	No backwards compatibility
Addition and deletions of abstract syntaxes.	V2E1 section 5.11.1.3.1	No backwards compatibility
Value for session functional unit "resynchronize" changed.	V2E4 section 5.11.1.4.1	No backwards compatibility
Restrictions on inclusion of "Transfer-syntax-name" in CP PPDU and CPC type.	V3E1 section 5.8.6	Interworking problems will occur for those implementations that expect "Transfer-syntax-name" parameter to be present in the PDV-List even though one transfer syntax was negotiated. [An implementation conforming to V3E1 as an initiator and versions previous to V3E1 as a responder will be able to interoperate.]

Version & Section		
Issue	Changed	Backward Compatibility
Encloding restrictions on ASN.1 INTEGER type describing PCI.	V3E1 section 5.10.4	Interworking problems will occur since implementations conforming to previous versions could encode PCI integer lengths greater than 4. [Responders that accept integers describing PCI that are encoded in greater than 4 octets and Initiators that conform to V3E1 will be able to interoperate.]
Encoding restrictions on BIT STRING, OCTET STRING, and CHARACTER STRING.	V3E1 section 5.10.5	Implementations that conform to previous versions can expect these strings to have nested constructed encodings and therefore interworking problems will occur. [Responders that accept nested constructed encodings and Initiators that conform to V3E1 will be able to interoperate.]
No extra trailing bits allowed in BIT STRING.	V3E1 section 5.10.6	Interworking problems will occur when implementations that conform to previous versions send extra trailing bits. [Responders accepting extra trailing bits and Initiators that conform to V3E1 will be able to interoperate.]
Restriction on usage of "token item field" and "user data."	V3E1 section 5.9.3.1	Interworking problems will occur since implementations that conform to V1E1 do not expect the "token item field" to be encoded when a category 0 SPDU is concatenated to a category 2 SPDU.
Restrictions on CPC-type values when multiple transfer syntaxes are proposed.	V2E2 section 5.8.3.9	Interworking problems may occur between initiators that send CPC-type values and receivers that do not examine them.

Version & Section		
Issue	Changed	Backward Compatibility
References to ISO 8649 and ISO 8650 changed.	V1E3 section "References."	Interworking problems will occur for those implementations that conform to ISO DIS 8649 and 8650. V1E3 references IS versions of 8649 and 8650.
References to ISO 8326, ISO 8327, ISO 8822, and ISO 8823 changed.	V1E4 section References.	Interworking problems will occur for those implementations that conform to 8326/DAD2, 8327/DAD2, DIS 8822, and DIS 8823. V1E4 referenced 8326/AD2, 8327/AD2, IS 8822, and IS 8823.
AE Title changed according to Amendment 1 to ISO 8650.	V3E1 section 5.5.3.2	Interworking problems will occur between initiators that use AE-title- form 1 and responders that accept only AE-Title-form 2.
Restrictions on usage of "direct references" in ABRT APDU.	V3E1 section 5.5.4	Interworking problems will occur for those implementations that expect the "direct reference" parameter to be included in the ABRT APDU. [An implementation that conforms to V3E1 as an initiator and versions previous to V3E1 as a responder will be able to interoperate.]

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 6 - Registration Authority Procedures for the OSI Implementors Workshop (OIW)

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair: **Einar Stefferud**
SIG Editor: **Charlie Combs**

Table of Contents

Foreword

This part of the Working Implementation Agreements was prepared by the Registration Special Interest Group (RSIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the above-mentioned Workshop. This part replaces the previously existing chapter on this subject.

Annex A is for information purposes only.

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

Editor's Note - Refer to the Stable Implementations Agreement, Version 4, Edition 1, December 1990.

Annex A (informative)

Guidelines for Registering Changes to Technical Objects

Editor's Note - Part 6 of the OIW Agreements document describes the registration process for registering technical information objects that are defined in OIW implementation agreements. However, the process does not describe a criteria for determining when a change in an object definition is of sufficient magnitude to require registration of the changed object with a new OID. Criteria are needed when changes are proposed to technical object definitions that have already been registered. The following draft tutorial text is informative and is presented for consideration by the OIW to clarify the rules. This text is under review and is to be revised in order to condense it and make it more comprehensible.

A.1 Introduction

The registration procedures for technical information objects in OIW Implementation Agreements assumes that each object is uniquely different in particular ways from all other registered technical information objects, and requires that there is exactly one definition for each registered object identifier (OID). Therefore, when an object definition is to be changed, it must receive a new OID if the change is "sufficiently significant", in order to signal to all concerned parties that something significant has been changed.

There is no rule to prevent two different OIDs from being assigned to identical copies of a single object definition, but it is hard to imagine any value to be derived from doing such a thing, and it could easily generate confusion among implementors. Therefore, assignment of two OIDs to identical object definitions is strongly discouraged.

This text attempts to clarify what is meant by the phrase "sufficiently significant". Unfortunately, there are gray areas and it is not possible to set forth precise rules for decisions. Decisions require assessment of subjective factors such as the impact of a change on interoperation among existing and future implementations, and possible confusion among implementors.

For example, we should note that sometimes a change may be desired to specifically reduce confusion that stems from different interpretations of a given definition. In this case, the change might require some implementations to be modified to conform to a chosen interpretation, but who is to say that the definition was changed, versus saying that the original intent was finally made clear? It is a matter of judgement by the responsible OIW SIG to decide whether a new OID should be assigned in this case, or not.

In other cases, there may be no implementations at the time when a change is being considered, so the impact may have less significance, if confusion among the implementors can be avoided. One way to avoid confusion is to assign a new identifier to the revised definition, and retire use of the old OID and its definition.

When reviewing a potential change to a registered object, it is necessary to determine how the proposed change might be applied to the registered definition without modifying the operational implications for implementations of the registered technical object. Many changes will be in the gray area between an obvious "editorial change" (meaning no new registration) and "new object" (meaning that the change implies that this is sufficiently significant to create a new technical object that must be registered with a new OID). When a change is sufficiently significant enough to require a new OID, then the old object should remain unchanged with its old OID.

These guidelines are presented to assist the OIW SIGs to achieve a consistent approach for assessing and judging proposed changes to any registered technical object.

A.2 Assessing Changes To Registered Technical Objects

Technical objects registered in OIW Implementation Agreement texts include functional definitions describing the states of a given object and the actions that can change the states of that object. The definitions and actions are usually presented as descriptive text, while the states may be defined by data structures and a set of values such as constants or ranges, having particular syntaxes.

Both the text descriptions and the state data structures represent a collection of facts about the registered technical object. If any of the facts are to change, then it must be determined whether a change (when applied to current and future implementations) will require creation of a new technical object to be registered with a new OID.

The description of a technical object represents a unique set of facts that define the object. But as independent statements, each fact expressed in the definition has a relative importance to the whole set of facts. It should be recognized that there are instances when modifying or deleting one or more of the facts may (or may not) be of sufficient significance to require registration of the modified technical object with a new OID.

An Example:

Change#1. A given registered object includes a range of values for a particular attribute called TIMEOUT. It includes the following two facts:

a) a definition of the TIMEOUT attribute;

b) the range of values for the TIMEOUT attribute.

If the range of the TIMEOUT attribute is changed from 10..100 to be 10..1000, it is possible that the change is not significant enough to warrant a new registration, if the parameter is only applied locally. (We will assume that this is the case for this example.)

Change#2. Suppose the same attribute is to be deleted. Then some assessment is needed, regarding the impact of the change to the global operational environment in which the technical object is to function, to determine if a new registration is required with a new OID.

The relative significance of the two changes to the operational requirements are clear in these two cases. Changing the values of the range of the TIMEOUT parameter is a relatively minor change which affects only local operation. Depending on other operational considerations, and the relation of the TIMEOUT to other facts about the technical object it could be changed without a new registration. But the elimination of the TIMEOUT attribute altogether would be much more significant, and more than likely require a new registration, since current implementations would be expecting the existence of such an attribute in any operating environment, and future implementations would not include it.

It is important to avoid setting the change assessment criteria so low as to require assignment of new OIDs for insignificant changes. A too low threshold can cause proliferation of slightly different technical objects and produce considerable confusion.

It is probably correct to err on the conservative side if there is significant doubt or question about the potential impact of a proposed change. However, a more reasoned choice is factual analysis of the change, applying the criteria as a guide. This can best be done by the authors of the proposed change, working with the SIG that is responsible for the impacted implementation agreement text.

A.3 Change Assessment Criteria

The significant components of a technical definition, to which the criteria are to be applied include (1) the text description of the technical object, (2) the definitions of the state values, and (3) the definitions of the data structures. The criteria is not intended to be regulatory in nature, but to provide some direction in reviewing each of the three components when evaluating change proposals for registered technical objects.

A.3.1 The Technical Object Description

Does the definition as changed describe a uniquely different set of functions or state conditions, or change the relations between functions of the registered object?

Example: A registered object is defined as a set of functions: Fn(a,b,c,d). Each term of the function "Fn" defines a substantive fact that is a function or a state condition of the technical object.

If the proposed change adds another function "Fn+1" or adds another state (a,b,c,d,e), or modifies the relationship between functions or state conditions (a,c,b,d) of the object, or deletes a defined function or state (a,b,_,d) then the proposed technical object definition should be seriously considered for registration as a new technical object with a new OID, provided that the fact (or facts) proposed for change will have a significant impact upon the implementation and operation of the technical object.

The most difficult assessment is for changes made to the functional description of the technical object. Editorial changes can be made to correct grammar or to improve clarity, without changing the definition. For changes that require additions or deletions of text to the definition, an assessment must be made to determine whether the changes will be optional or will apply only to a local entity, or will be extensive enough to require all implementations to represent completely the new functional requirements of the technical object.

Deciding what changes mean with respect to the functional definition requires a subjective judgement and will require that each SIG establish some guidelines for its particular object types. Consistency of registration policy is highly desired.

One approach is to rule that any change, other than a spelling or grammar change requires a new technical object registration. As stated earlier, the consequence of such a rule would be proliferation of many registered technical objects with very similar definitions. This can create considerable confusion for implementors. The opposite extreme is to treat any change to the functional description as an editorial change, with only changes to the other criteria (state values and data structures) triggering a new registration and new OID.

Between the two is a more acceptable view that provides for assessment of the characteristics of the proposed change and a decision, albeit subjective, as to whether a given change is an editorial change NOT affecting implementation; or it is a change in functionality that MAY affect implementation. Note the emphasis - if it does NOT affect implementation, then it is an editorial change, but if it MAY affect implementation, then the change should be seriously considered for registration as a new technical information object with a new OID.

A.3.2 Evaluating the State Values

Within a registered technical object description there may be a number of constants, ranges of values, and syntaxes specifically defined for the object. They are all subject to change. The evaluation criteria applied to requests for changes to state values has to consider the kind of operation that the technical object is performing.

Example: The range of accepted values is changed from 1-128 to 0-127, or the default value of a parameter is changed from 32 to 128.

Understanding the implications of what is changing helps to measure the impact of the proposed changes. The shift of the range from 1-128, to 0-127 could be trivial, depending on the scope of its use and would not alone necessarily warrant a new registration. However to change a default value from 32 to 128 (if the attribute applies to the availability or limit of some external system or network resource) would clearly be cause for much concern over how the change impacts implementations of the technical object.

A.3.3 Evaluating The Data Structure

Each technical information object may have one or more data structures defined within the description of the object. Changes can be made to the data structures in a umber of ways. Data field sizes can change, and the number of data fields can change. As with the state values, they should be considered in a very broad sense.

Example: A defined data field is changed from 3 octets to 4 and another field is reduced from 2 octets to 1.

Changing the data structure is probably the clearest case of a requirement to change an implementation. One must be aware that all syntactical changes in a technical definitions need not be mandatory, they may be optional. But if the changes are mandatory, they will most likely affect every implementation, and in such a way that they will not interoperate properly with old implementations. Such cases warrant that the change be registered as a new technical object with a new OID.

A.3.4 A Final Observation

It is most important to be consistent in making subjective judgements concerning changes to registered technical objects rather than trying to be 'correct'. If there is real doubt about the nature of the change, then the change should most likely result in registration of a new object with a new OID.

A.4 The Change Process

Responsibility for evaluating the change requests is assigned to each SIG. The SIG makes its determinations by voting on changes to each registered object as it is defined in the SIG text in the OIW Implementation Agreements Documents. Any SIG approved changes must also be voted in the OIW Plenary using the rules of the SIG and the Plenary. By this means each SIG will have agreed whether a change is "editorial" or "technical" before the change reaches a plenary vote.

An object definition in a Working Agreement text is not registered until it has been voted into the Stable Agreements Document, so it is relatively easy to change an "as yet unregistered" object in the Working Agreements Document.

With respect to these guidelines, it should be emphasized that it is most important to be consistent in making subjective judgements concerning changes to registered technical objects rather than being 'correct'. There may be more than one interpretation or 'correct' decision regarding any change, but if the guidelines are consistent, then the implementations are more likely to be consistent too.

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 7 - 1984 Message Handling Systems

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair: **Barbara Nelson (Retix)**
SIG Editor: **Charles Combs (MCI)**

Table of Contents

Foreword

This part of the Working Implementation Agreements was prepared by the Message Handling Systems Special Interest Group (X.400 SIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the Workshop. This part replaces the previously existing chapter on this subject. There is no significant technical change from this text as previously given.

Part 7 1984 Message Handling Systems

See part 7 of the *Stable Implementation Agreements* document.

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 8 - 1988 Message Handling Systems

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair: **Barbara Nelson (Retix)**
SIG Editor: **Charles Combs (MCI)**

Table of Contents

Foreword

This part of the Working Implementation Agreements was prepared by the Message Handling Systems Special Interest Group (X.400 SIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the Workshop. This part replaces the previously existing chapter on this subject. There is no significant technical change from this text as previously given.

Part 8 1988 Message Handling Systems

See part 8 of the *Stable Implementation Agreements* document.

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 9 - FTAM Phase 2

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair **Darryl Roberts**
SIG Editor **Larry Friedman**

Table of Contents

Foreword

This part of the Working Implementation Agreements was prepared by the File Transfer, Access and Management Special Interest Group (FTAM SIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the above-mentioned Workshop. This part replaces the previously existing chapter on this subject.

Part 9 - ISO File Transfer, Access and Management Phase 2

NOTE - See Stable Document for text on this subject.

Annex A (normative)

FTAM Document Types

(See Stable Document.)

Annex B (normative)

Constraint Sets

(See Stable Document.)

Annex C (normative)

Abstract Syntaxes

(See Stable Document.)

Annex D (informative)

FTAM-1 Document Type Tutorial

NOTE - This text does not represent an implementation agreement.

D.1 Introduction

This annex is informative. It does not specify any additional requirements.

The purpose of this tutorial is to clarify the mechanism to convey lines of text in a FTAM-1 document type.

ISO 8571-2 defines a number of document types for files. One of these document types is FTAM-1. ISO defines the FTAM-1 document type for usage with files that contain unstructured text. A file that has a document type of FTAM-1 consists of one FADU that consists of zero or more character strings. In order to reduce ambiguities it is useful to assume that one character string correspond to one Data Element.

FTAM-1 document type parameters are defined in ISO 8571-2 clause B.1. These parameters are used to define:

- the allowed character sets that may be contained in the strings (universal-class-number);
- the maximum allowed length of a string (maximum-string-length);
- the significance of the end of string (string-significance)

D.2 Document type Parameters

D.2.1 Universal-Class-Number

The universal-class-number parameter determines the character sets that are allowed to be used in a FTAM-1 file. The values of the universal-class-number parameter are ASN.1 types whose definition can be found in ISO 8824. The important thing for this discussion is that some string classes allow only graphic characters to be used while other string classes allow both graphic and control characters to be used. (Control characters include "format effector" characters such as carriage return <CR> and line feed <LF>).

D.2.2 Maximum-String-Length

The maximum-string-length parameter determines the maximum number of characters allowed in a string of the FTAM-1 file, it does not determine the maximum number of octets allowed in the string.

GeneralStrings illustrate how the number of octets in a string can differ from the number of characters in an string. GeneralStrings can contain escape sequences that are used for purposes such as invoking different character sets. An escape sequence is considered to be a bit string, not a character string. Therefore, the combined length of any escape sequences contained in a GeneralString contributes to the number of octets in the GeneralString but does not contribute to the number of characters in the GeneralString.

The length value of the ASN.1 encoding of a character string always reflects the number of octets in the character string. This value will always be greater than or equal to the number of octets in the string. The character string must be processed to determine the actual number of characters in the string.

NIST/OIW FTAM Phase 2 agreements state that a conformant FTAM implementation must support a maximum-string-length parameter of at least 134 for a FTAM-1 file (see part 9 clause 10). There is no minimum requirement for maximum-string-length in the FTAM phase 3 agreements. The minimum requirement implies that a minimally conformant NIST/OIW FTAM responding implementation will not accept a FTAM-1 file whose actual maximum-string-length parameter has a value greater than 134. The relaxation rules for FTAM-1 files allow a FTAM-1 file to be opened for read using a maximum-string-length parameter that is greater than or equal to the value of the maximum-string-length file attribute actually associated with the file, a smaller value is not allowed (see ISO 8571-2 B.1 clause 11.1.1.2). This implies that a minimally conformant NIST/OIW FTAM initiating implementation can not read a FTAM-1 file whose actual string length parameter has a value greater than 134.

To increase interoperability, a sending FTAM system should be able to divide a file with string-significance of not-significant into strings of no more than 134 characters. A receiving FTAM system should be able to use the strings to form the file which was sent. If a file has a maximum-string-length associated with it that is greater than 134, relaxation to the minimally conformant value is not possible and interworking might not be possible.

D.2.3 String-Significance

The string -significance parameter determines the significance of the character strings (semantics of string boundaries). Fixed string-significance means that each string contains exactly the number of characters defined by the maximum-string-length parameter. Variable string-significance means that the length of each string is less than or equal to the maximum-string-length parameter. When string-significance is fixed or variable, then maximum-string-length is present with a value > 0, the boundaries of the character strings are preserved and contribute to the document's semantic. A value of not-significant means that the length of each string is less than or equal to the maximum-string-length parameter and that the boundaries of the character strings are not necessarily preserved when the file is stored and do not contribute to the document's semantics. In this case, string-significance is not maintained, thus the sender entity explicitly declares that string boundaries have no meaning.

Note the NIST/OIW FTAM Phase 2 agreements require the support of only the not-significant value for string-significance. Fixed and variable string-significance are outside the scope of the Phase 2 agreements, but are required in the Phase 3 agreements.

It is in the area of not-significant strings where most interoperability problems have occurred.

> **NOTE -** the difference between variable significance and not-significant significance. If a file has a significance of fixed or variable, it is the responsibility of any storer of the file to "remember" where the boundaries of each character string are located within the file. The storer of a file with a significance of not-significant has no such responsibility.

D.3 New Line Function

When a sequence of characters are being displayed on a character imaging device, the term "new line function" is used to mean the repositioning of the current character display position one row down and back to column one. A new line function may be implemented in a variety of ways. A UNIX system implements the new line function with a <LF> character (sometimes called <NL>). A MS-DOS system implements the new line function with a <CR><LF> character sequence. A typical word processor will implement a new line function as a "wrap around" function that depends upon a defined page width. A record oriented file system may interpret an end of record condition as implying a new line function.

ISO suggests (see FTAM agreements part 9 clause 10.1.2 and ISO 646 clause 4.1.2.2) that a new line function be accomplished with a <CR><LF> combination. If there is a prior arrangement,e.g. a bilateral agreement, between a sender and a receiver, and only in this case, may a vertical format effector, i.e. a <LF> be used to accomplish a new line function. The NIST/OIW FTAM agreements contain no such prior arrangement (see NIST/OIW Part 9 clause 10.1.2).

It is strongly suggested that files being sent to a remote FTAM represent the local new line function as a <CR><LF> pair and files received from a remote FTAM have <CR><LF> pairs converted to the local new line function.

It is important to realize that a new line function represents a display positioning functions and it does not represent anything more that. A new line function is not intended to act as wither a string terminator or a string separator.

All Format Effector characters except <CR><LF> are outside the scope of the NIST Agreements.

D.4 Character Strings Versus Lines

A line of characters is generally considered to be a sequence of graphic characters followed by a new line function (or possible by an end of line condition).

A character string is simply that, a string of characters from one or more character sets. Characters within a string come from allowed character sets. It is the "universal-class-number" parameter defined in ISO 8571-2 B.1 that determines which character sets may be used to compose a string. For example, a GraphicString consists of characters from any graphic character set but may not contain characters from a control character set (it can not contain format effectors); a GeneralString consists of characters from any graphic character and characters from any control character set (it can contain format effectors).

Text files will be transferred using the Document type FTAM-1. The supported character sets are:

- IA5String (line boundaries via format effectors, preferably <CR><LF>)
- GeneralString (i.e. ISO 646 International Reference Version and ISO 8859-1. Line boundaries via format effectors. preferably <CR><LF>)
- VisibleString (IA5 String without control characters, line boundaries via Data Element boundaries)
- GraphicString (i.e. ISO 646 International Reference Version without control characters and ISO 8859-1, line boundaries via Data Element boundaries)

NOTE - A string is really a language (programming or otherwise) concept. File systems generally have no concept of a string, although a file system, especially a record oriented file system may have some concept of a line.

The standard gives no relation between character string and a line of characters. A character string may contain a portion of a line of characters or it may contain multiple lines of characters. A character string can contain zero, one, or many <CR><LF> pairs. A character string may or may not end with a <CR><LF> pair. In fact, an entire file of character strings may not contain a single <CR><LF> pair, even when those characters are allowed to be used in the character strings.

The following figure is an example of IA5String and GeneralString when string-significance is not-significant could be broken up into lines of text.

<table>
<tr><td colspan="2">String-1</td><td colspan="2">String-2</td><td>String-3</td><td>String-4</td><td>String-5</td></tr>
<tr><td>Line-1 <CR><LF></td><td colspan="2">Line-2 <CR><LF></td><td>Line-3 <CR><LF></td><td colspan="2">Line-4 <CR><LF></td><td>Line-5 <CR><LF></td></tr>
</table>

The following figure is an example of VisibleString and GraphicString when string-significance is fixed or variable could be broken up into lines of text.

String-1	String-2	String-3	String-4	String-5
Line-1	Line-2	Line-3	Line-4	Line-5

Note that the minimum requirement of 134 for maximum-string-length (see Maximum-String-length above) does not limit the length of a line; however, the additional rule that is needed for increasing document semantics interoperability is: "A line of characters corresponds to a character string".

D.5 Implementation Problems

The lack of an equivalence between a line of characters and a character string can cause implementation problems. It is common for a record oriented file system to store a line of characters as a record. How does such a system decide how large a record to allocate for a line of characters? A line of characters may be contained in a part of one string, one or more strings, or it may actually consist of an entire file. How does such a system identify the end of a line (record)? It must scan the string for a <CR><LF> pair (or end of transmission) and probably remove the <CR><LF> before storing a record. What happens if the line is bigger than the size of the record allocated? The system would likely break the string and store it in the available record size. Breaking the string adds a new line function which was not present in the original file.

Another problem can occur when a system whose new line function is implemented by a <CR><LF> pair send a file to a system whose new line function is implemented by <LF>. For example, a MS-DOS system could send a file that contains <CR><LF> pairs and also contains single <LF> characters to a UNIX system. The UNIX system would likely translate both <CR><LF> and <LF> to UNIX new line functions, i.e. a <LF> . If the UNIX system then sends the file back to the MS-DOS system the original single <LF> characters will be sent as <CR><LF> pairs.

D.6 Imaging a File

There is no relation between a character string and a line of characters (see ISO 8571-2 B.1 clause 7) except when character strings that come from character sets that do not contain format effector characters (for example, VisibleStrings and GraphicStrings) are transferred to a device such as a printer. In this case the end of a string implies the invocation of the device's new line function. This means that, in this case, a string is equivalent to a line.

In order for the rendition of such a file made of character strings belonging to a set that does not contain format effector characters (For example, VisibleString, and GraphicString) transferred first to disk and then to a character imaging device to be equivalent to the rendition of the same file transferred directly to a character imaging device, the storer of the file must " remember" where the end of string occurs as the file is transferred to disk. This would be easy for a record oriented file system to accomplish since one string could occupy one record, but would present difficulties for a byte stream oriented file system. One solution for a byte stream oriented file system would be to write a "newline" after each string when the file is written to disk. The file could then be correctly sent to a printer. This solution would require the newlines be stripped from the disk file it is later sent to a remote system using FTAM.

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 10 - ISO FTAM Phase 3

Output from the September 1990 NIST Workshop for Implementors of OSI

SIG Chair **Darryl Roberts**
SIG Editor **Larry Friedman**

Table of Contents

Foreword

This part of the Working Implementation Agreements was prepared by the File Transfer, Access and Management Special Interest Group (FTAM SIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the above-mentioned Workshop. This part replaces the previously existing chapter on this subject. There is no significant technical change from this text as previously given.

Part 10 - ISO File Transfer, Access and Management Phase 3

Editor's Note: For current Stable FTAM Phase 3 Agreements, consult the aligned part in the Stable Implementation Agreements, Version 4, dated December 1990.

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 11 - Directory Services Protocols

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair: **You Bong Weon–Yoon**
SIG Editor: **Michael Ransom**

Table of Contents

Foreward

This part of the Working Implementation Agreements was prepared by the Directory Services Special Interest Group (DSSIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the above mentioned Workshop. This part replaces the previously existing chapter on Directory Services Protocol. There is no significant technical change from this text as previously given.

PART 11 - DIRECTORY SERVICES PROTOCOLS

0 Introduction

Refer to clause 0 of Stable Agreements Version 4 as of December 14, 1990.

1 Scope

Refer to clause 1 of Stable Agreements Version 4 as of December 14, 1990.

2 Normative references

Refer to clause 2 of Stable Agreements Version 4 as of December 14, 1990.

3 Status

Refer to clause 3 of Stable Agreements Version 4 as of December 14, 1990.

4 Use of the Directory

This clause will contain introductory text.

4.1 MHS

(TBD)

4.2 FTAM

(TBD)

5 Directory ASEs and Application Contexts

Refer to clause 5 of Stable Agreements Version 4 as of December 14, 1990.

6 Schema

Refer to clause 6 of Stable Agreements Version 4 as of December 14, 1990.

6.1 Support of Structures and Naming Rules

Refer to 6.1 of Stable Agreements Version 4 as of December 14, 1990.

6.2 Support of Object Classes and Subclasses

Refer to 6.2 of Stable Agreements Version 4 as of December 14, 1990.

6.3 Support of Attribute Types

Refer to 6.3 of Stable Agreements Version 4 as of December 14, 1990.

6.4 Support of Attribute Syntaxes

Refer to 6.4 of Stable Agreements Version 4 as of December 14, 1990.

6.5 Naming Contexts

Refer to 6.5 of Stable Agreements Version 4 as of December 14, 1990.

6.6 Common Profiles

Refer to 6.6 of Stable Agreements Version 4 as of December 14, 1990.

6.6.1 OIW Directory Common Application Directory Profile

Refer to 6.6.1 of Stable Agreements Version 4 as of December 14, 1990.

6.6.1.1 Standard Application Specific Attributes and Attribute Sets

Refer to 6.6.1.1 of Stable Agreements Version 4 as of December 14, 1990.

6.6.1.2 Standard Application Specific Object Classes

Refer to 6.6.1.2 of Stable Agreements Version 4 as of December 14, 1990.

6.6.2 OIW Directory Strong Authentication Directory Profile

Refer to 6.6.2 of Stable Agreements Version 4 as of December 14, 1990.

6.6.2.1 Other Profiles Supported

Refer to 6.6.2.1 of Stable Agreements Version 4 as of December 14, 1990.

6.6.2.2 Standard Application Specific Object Classes

Refer to 6.6.2.2 of Stable Agreements Version 4 as of December 14, 1990.

6.7 Restrictions on Object Class Definitions

Refer to 6.7 of Stable Agreements Version 4 as of December 14, 1990.

7 Pragmatic Constraints

Refer to clause 7 of Stable Agreements Version 4 as of December 14, 1990.

7.1 General Constraints

Refer to 7.1 of Stable Agreements Version 4 as of December 14, 1990.

7.1.1 Character Sets

Refer to 7.1.1 of Stable Agreements Version 4 as of December 14, 1990.

7.1.2 APDU Size Considerations

Refer to 7.1.2 of Stable Agreements Version 4 as of December 14, 1990.

7.1.3 Service Control (SC) Considerations

Refer to 7.1.3 of Stable Agreements Version 4 as of December 14, 1990.

7.1.4 Priority Service Control

Refer to 7.1.4 of Stable Agreements Version 4 as of December 14, 1990.

7.2 Constraints on Operations

Refer to 7.2 of Stable Agreements Version 4 as of December 14, 1990.

7.2.1 Filters

Refer to 7.2.1 of Stable Agreements Version 4 as of December 14, 1990.

7.2.2 Errors

Refer to 7.2.2 of Stable Agreements Version 4 as of December 14, 1990.

7.2.3 Error Reporting – Detection of Search Loop

Refer to 7.2.3 of Stable Agreements Version 4 as of December 14, 1990.

7.3 Constraints Relevant to Specific Attribute Types

Refer to 7.3 of Stable Agreements Version 4 as of December 14, 1990.

8 Conformance

Refer to clause 8 of Stable Agreements Version 4 as of December 14, 1990.

8.1 DUA Conformance

Refer to 8.1 of Stable Agreements Version 4 as of December 14, 1990.

8.2 DSA Conformance

Refer to 8.2 of Stable Agreements Version 4 as of December 14, 1990.

8.3 DSA Conformance Classes

Refer to 8.3 of Stable Agreements Version 4 as of December 14, 1990.

8.4 Authentication Conformance

Refer to 8.4 of Stable Agreements Version 4 as of December 14, 1990.

8.5 Directory Service Conformance

Refer to 8.5 of Stable Agreements Version 4 as of December 14, 1990.

8.6 The Directory Access Profile

Refer to 8.6 of Stable Agreements Version 4 as of December 14, 1990.

8.7 The Directory System Profile

Refer to 8.7 of Stable Agreements Version 4 as of December 14, 1990.

8.8 Digital Signature Protocol Conformance Profile

Refer to 8.8 of Stable Agreements Version 4 as of December 14, 1990.

8.9 Strong Authentication Protocol Conformance Profile

Refer to 8.9 of Stable Agreements Version 4 as of December 14, 1990.

9 Distributed Operations

Refer to clause 9 of Stable Agreements Version 4 as of December 14, 1990.

9.1 Referrals and Chaining

Refer to 9.1 of Stable Agreements Version 4 as of December 14, 1990.

9.2 Trace Information

Refer to 9.2 of Stable Agreements Version 4 as of December 14, 1990.

10 Underlying Services

Refer to clause 10 of Stable Agreements Version 4 as of December 14, 1990.

10.1 ROSE

Refer to 10.1 of Stable Agreements Version 4 as of December 14, 1990.

10.2 Session

Refer to 10.2 of Stable Agreements Version 4 as of December 14, 1990.

10.3 ACSE

Refer to 10.3 of Stable Agreements Version 4 as of December 14, 1990.

11 Access Control

Refer to clause 11 of Stable Agreements Version 4 as of December 14, 1990.

12 Test Considerations

Refer to 12 of Stable Agreements Version 4 as of December 14, 1990.

12.1 Major Elements of Architecture

Refer to 12.1 of Stable Agreements Version 4 as of December 14, 1990.

12.2 Search Operations

Refer to 12.2 of Stable Agreements Version 4 as of December 14, 1990.

13 Errors

Refer to clause 13 of Stable Agreements Version 4 as of December 14, 1990.

13.1 Permanent vs. Temporary Service Errors

Refer to 13.1 of Stable Agreements Version 4 as of December 14, 1990.

13.2 Guidelines for Error Handling

Refer to 13.2 of Stable Agreements Version 4 as of December 14, 1990.

13.2.1 Introduction

Refer to 13.2.1 of Stable Agreements Version 4 as of December 14, 1990.

13.2.2 Symptoms

Refer to 13.2.2 of Stable Agreements Version 4 as of December 14, 1990.

13.2.3 Situations

Refer to 13.2.3 of Stable Agreements Version 4 as of December 14, 1990.

13.2.4 Error Actions

Refer to 13.2.4 of Stable Agreements Version 4 as of December 14, 1990.

13.2.5 Reporting

Refer to 13.2.5 of Stable Agreements Version 4 as of December 14, 1990.

14 Specific Authentication Schemes

Refer to clause 14 of Stable Agreements Version 4 as of December 14, 1990.

14.1 Specific Strong Authentication Schemes

Refer to 14.1 of Stable Agreements Version 4 as of December 14, 1990.

14.1.1 ElGamal

Refer to 14.1.1 of Stable Agreements Version 4 as of December 14, 1990.

14.1.1.1 Background

Refer to 14.1.1.1 of Stable Agreements Version 4 as of December 14, 1990.

14.1.1.2 Digital Signature

Refer to 14.1.1.2 of Stable Agreements Version 4 as of December 14, 1990.

14.1.1.3 Verification

Refer to 14.1.1.3 of Stable Agreements Version 4 as of December 14, 1990.

14.1.1.4 Known Constraints on Parameters

Refer to 14.1.1.4 of Stable Agreements Version 4 as of December 14, 1990.

14.1.1.5 Note on subjectPublicKey

Refer to 14.1.1.5 of Stable Agreements Version 4 as of December 14, 1990.

14.1.2 One–Way Hash Functions

Refer to 14.1.2 of Stable Agreements Version 4 as of December 14, 1990.

14.1.2.1 SQUARE–MOD–N Algorithm

Refer to 14.1.2.1 of Stable Agreements Version 4 as of December 14, 1990.

14.1.2.2 MD2 Algorithm

Refer to 14.1.2.2 of Stable Agreements Version 4 as of December 14, 1990.

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.

14.1.2.4 Use of One–Way Hash Functions in Forming Signatures

Refer to 14.1.2.4 of Stable Agreements Version 4 as of December 14, 1990.

14.1.3 ASN.1 for Strong Authentication Algorithms

Refer to 14.1.3 of Stable Agreements Version 4 as of December 14, 1990.

14.1.4 Note on the ENCRYPTED MACRO

Refer to 14.1.4 of Stable Agreements Version 4 as of December 14, 1990.

14.2 Protected Simple Authentication

Refer to 14.2 of Stable Agreements Version 4 as of December 14, 1990.

14.3 Simple Authentication

Refer to 14.3 of Stable Agreements Version 4 as of December 14, 1990.

Annex A
(normative)
Maintenance of Attribute Syntaxes

Refer to Annex A of Stable Agreements Version 4 as of December 14, 1990.

A.1 Introduction

Refer to A.1 of Stable Agreements Version 4 as of December 14, 1990.

A.2 General Rules

Refer to A.2 of Stable Agreements Version 4 as of December 14, 1990.

A.3 Checking Algorithms

Refer to A.3 of Stable Agreements Version 4 as of December 14, 1990.

A.3.1 distinguishedNameSyntax

Refer to A.3.1 of Stable Agreements Version 4 as of December 14, 1990.

A.3.2 integerSyntax

Refer to A.3.2 of Stable Agreements Version 4 as of December 14, 1990.

A.3.3 telephoneNumberSyntax

Refer to A.3.3 of Stable Agreements Version 4 as of December 14, 1990.

A.3.4 countryName

Refer to A.3.4 of Stable Agreements Version 4 as of December 14, 1990.

A.3.5 preferredDeliveryMethod

Refer to A.3.5 of Stable Agreements Version 4 as of December 14, 1990.

A.3.6 presentationAddress

Refer to A.3.6 of Stable Agreements Version 4 as of December 14, 1990.

A.4 Matching Algorithms

Refer to A.4 of Stable Agreements Version 4 as of December 14, 1990.

A.4.1 UTCTimeSyntax

Refer to A.4.1 of Stable Agreements Version 4 as of December 14, 1990.

A.4.2 distinguishedNameSyntax

Refer to A.4.2 of Stable Agreements Version 4 as of December 14, 1990.

A.4.3 caseIgnoreListSyntax

Refer to A.4.3 of Stable Agreements Version 4 as of December 14, 1990.

Annex B
(informative)
Glossary

Refer to Annex B of Stable Agreements Version 4 as of December 14, 1990.

Annex C
(informative)
Requirements for Distributed Operations

Refer to Annex C of Stable Agreements Version 4 as of December 14, 1990.

C.1 General Requirements

Refer to C.1 of Stable Agreements Version 4 as of December 14, 1990.

C.2 Protocol Support

Refer to C.2 of Stable Agreements Version 4 as of December 14, 1990.

C.2.1 Usage of ChainingArguments

Refer to C.2.1 of Stable Agreements Version 4 as of December 14, 1990.

C.2.2 Usage of Chaining Results

Refer to C.2.2 of Stable Agreements Version 4 as of December 14, 1990.

Annex D
(informative)
Guidelines for Applications Using the Directory

Refer to Annex D of Stable Agreements Version 4 as of December 14, 1990.

D.1 Tutorial

Refer to D.1 of Stable Agreements Version 4 as of December 14, 1990.

D.1.1 Overview

Refer to D.1.1 of Stable Agreements Version 4 as of December 14, 1990.

D.1.2 Use of the Directory Schema

Refer to D.1.2 of Stable Agreements Version 4 as of December 14, 1990.

D.1.2.1 Use of Existing Object Classes

Refer to D.1.2.1 of Stable Agreements Version 4 as of December 14, 1990.

D.1.2.2 Kinds of Object Classes

Refer to D.1.2.2 of Stable Agreements Version 4 as of December 14, 1990.

D.1.2.3 Use of Unregistered Object Classes

Refer to D.1.2.3 of Stable Agreements Version 4 as of December 14, 1990.

D.1.2.4 Side Effects of Creating Unregistered Object Classes

Refer to D.1.2.4 of Stable Agreements Version 4 as of December 14, 1990.

D.2 Creation of New Object Classes

Refer to D.2 of Stable Agreements Version 4 as of December 14, 1990.

D.2.1 Creation of New Subclasses

Refer to D.2.1 of Stable Agreements Version 4 as of December 14, 1990.

D.2.2 Creation of New Attributes

Refer to D.2.2 of Stable Agreements Version 4 as of December 14, 1990.

D.3 DIT Structure Rules

Refer to D.3 of Stable Agreements Version 4 as of December 14, 1990.

Annex E
(informative)
Template for an Application Specific Profile for Use of the Directory

Refer to Annex E of Stable Agreements Version 4 as of December 14, 1990.

Annex F
(informative)
Bibliography

Refer to Annex F of Stable Agreements Version 4 as of December 14, 1990.

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 12 - Security

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair **Dr. James Galvin**
SIG Editors **Maj Doug Naegele, USAF**

Table of Contents

Foreword

This part of the Working Implementation Agreements was prepared by the Security Special Interest Group (SECSIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the above-mentioned Workshop. This part replaces the previously existing chapter on this subject. There is no significant technical change from this text as previously given.

Part 12 - Security

Editor's Note - This part points to Stable Security Agreements which are contained in the aligned part of the Stable Implementation Agreements, Version 4 dated December 1990.

Working Agreements for Open Systems Interconnection Protocols: Part 13 - Security

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair **Dr James Galvin**
SIG Editor **Maj Doug Naegele, USAF**

Table of Contents

List of Tables

Foreword

This part of the Working Implementation Agreements was prepared by the Security Special Interest Group (SECSIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the above-mentioned Workshop. This part replaces the previously existing chapter on this subject. There is significant technical change from this text as previously given.

Part 13 - Security

0 Introduction

1 Scope

2 Normative References

ISO/IEC 9594-8 (CCITT X.509 Recommendation)*Information Technology - Open Systems Interconnection - The Directory - Part 8: Authentication Framework.*

ISO 8649: 1988/DAD 1 *Service Definition for the Association Control Service Element, Addendum 1: Peer-Entity Authentication During Association Establishment.*

ISO 8650: 1988/DAD 1 *Protocol Specification for the Association Control Service Element, Addendum 1: Peer-Entity Authentication During Association Establishment.*

ISO/IEC 9594-3 (CCITT X.511 Recommendation) *Information Technology - Open Systems Interconnection - The Directory - Part 3: Abstract Service Definition.*

ISO 10021-4 (CCITT X.411 Recommendation) *Information Processing Systems - Text Communication - MOTIS - Message Transfer System : Abstract Service Definition and Procedures.*

ISO 7498-2 *Information Processing Systems - Open Systems Interconnection Reference Model - Part 2: Security Architecture, February 1989.*

3 Definitions

Editor's Note: This clause will contain all unique terms used in this part, to be determined.
Refer to ISO 7498/2 for definitions of security relevant terms. This base standard contains detailed descriptions of accepted security terms. Refer to ISO TR-10000 for general ISO definitions used in this part. The following security terms are not defined in ISO 7498/2:

o Authentication

o Mechanism

Editor's Note: The above two terms will be defined as a work item.

4 Symbols and Abbreviations

5 Architectures

Editor's Note: Clause 5 below is evolving and is not considered stable enough to consider for insertion into the OIW Stable Implementation Agreements in the near future.

5.1 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.

5.2 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

5.3 Security Profiles

5.3.1 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.

5.3.2 Generic Security Profile

5.3.2.1 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.

5.3.2.2 Mapping/Discussion of Mechanisms to Provide Security Services

5.3.2.3 Description of Generic Security Classes

o Profile 0 Null

o 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 5.3.2 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.

5.4 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 (5.3.2).

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).

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

5.i Specification of Security Profile
5.i.1 Discussion of Threats and Mapping to Security Services
5.i.2 Mechanisms to Provide Security Services
5.i.3 Security Classes

6 Key Management

7 Lower Layers Security

8 Upper Layers Security

9 Message Handling System (MHS) Security

All current MHS security relevant text appears in part 8.

10 Directory Services Security

11 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 straight forward to implement, and to avoid any unnecessary computational and managerial overhead. The approach also takes into consideration the need for different levels of security services within different network management domains, and the near term requirement for interoperability of network management entities over disparate network types.

11.1 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.

11.2 Security Services

11.2.1 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 authenticated 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 here and the decision of which to employ is left for further study.

11.2.2 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.

11.3 Security Mechanisms

11.3.1 Peer Entity Authentication

In order to simplify the management aspects associated with various phases of authentication procedures, the authentication scheme proposed is the same as that used for secure messages on the Internet [Ref. B26], which is compatible with the Directory Authentication Framework standard [Ref. ISO/IEC 9594-8]. 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.

One suitable location to convey authentication information is the association control service element (ACSE) authentication field, described in the addenda to the ACSE service definition and protocol specification covering peer-entity authentication during association establishment [Ref. ISO 8699, ISO 8650].

11.3.1.1 ACSE Authentication Field Usage

ACSE authentication extensions [Ref. ISO 8699, ISO 8650] support two-way authentication through the definition of a new functional unit. When this functional unit is employed, additional parameters are provided by the A-ASSOCIATE service to indicate this requirement and convey authentication information between entities. The ASN.1 definition for this information is given below.

```
from ISO 8650: 1988/DAD1

Authentication ::= SEQUENCE {
        mechanism-name [0] IMPLICIT OBJECT IDENTIFIER OPTIONAL,
        authentication-value [1] CHOICE {
                charstring [0] IMPLICIT GraphicString,
                bitstring [1] IMPLICIT BITSTRING,
                external [2] IMPLICIT EXTERNAL,
                other [3] ANY DEFINED BY mechanism-name }  }
```

It is proposed that support of ACSE authentication functional unit be mandatory for network management. This may require that a mechanism-name be defined and registered for network management, and that the corresponding authentication-value (i.e., CHOICE other [3]) defined by that mechanism be conveyed. Since it is intended that the ASN.1 definitions of the authentication field arguments, the procedures for handling those arguments, and their mapping onto ACSE be consistent with other application layer protocols, this defined mechanism conceivably may be utilized by other application protocols.

11.3.1.2 Authentication Value Definition

The authentication scheme used for privacy enhancement of Internet electronic mail [Ref. B26] relies on a key management architecture based on the use of public key certificates. Certificates are issued by an issuing authority (IA) and contain the public key of a principal, its identity, and other related information such as the serial number and validity period of the certificate, and the identity of the issuer. The certification authority (CA) acting on behalf of the IA applies a digital signature to a certificate providing non-forgable assurance of the identity of a principal and binding it to the given public key. The according ASN.1 definitions are given below.

```
from ISO 9594-8, Annex G:

Certificate ::= SIGNED SEQUENCE {
        version  [0] Version DEFAULT v1988,
        serialNumber  CertificateSerialNumber,
        signature  AlgorithmIdentifier,
        issuer  Name,
        validity ,
        subject  Name,
        subjectPublicKeyInfo  }

SIGNED MACRO ::=
        BEGIN
        TYPE NOTATION ::= type(ToBeSigned)
        VALUE NOTATION ::= value (VALUE
                SEQUENCE{
```

```
                              ToBeSigned,
                              AlgorithmIdentifier,
                              ENCRYPTED OCTETSTRING } )
```

Credentials are used to establish the identity of a user and provide the means for meaningful utilization of public key certificates. In addition to conveying the certificate of a principal, they can protect against replay attacks and, in situations where a hierarchy of certification authorities exists, they can convey the chain of CA certificates that are needed by the recipient to verify the senders certificate. One definition of credentials taken from the abstract service definition of the directory bind operation [Ref. ISO 9594-3] is given below.

from ISO 9495-3, Annex A, and ISO 9495-8:

```
Credentials ::= CHOICE {
            simple  [0] SimpleCredentials,
            strong  [1] StrongCredentials,
            externalProcedure [2] EXTERNAL }

SimpleCredentials ::= SEQUENCE {
            name [0] DistinguishedName,
            validity [1] SET { time1 [0] UTCTime OPTIONAL,
                               time2   [1] UTCTime OPTIONAL,
                               random1 [2] BITSTRING
                               random2 [3] BITSTRING} OPTIONAL,
            password [2] OCTETSTRING OPTIONAL }

StrongCredentials ::= SET {
            certification-path [0] CertificationPath OPTIONAL,
            bind-token  [1] Token }

Token ::= SIGNED SEQUENCE {
            algorithm [0] AlgorithmIdentifier,
            name [1] DistinguishedName,
            time [2] UTCTime,
            random [3] BITSTRING }

CertificationPath ::= SEQUENCE {
      userCertificate Certificate,
      theCACertificates SEQUENCE OF CertificatePair OPTIONAL}

CertificatePair ::= SEQUENCE {
      forward [0] Certificate OPTIONAL,
      reverse [1] Certificate OPTIONAL }
```

It is proposed that this definition be utilized by network management as the ACSE authentication value definition.

11.3.2 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 and connectionless integrity, they are to be considered jointly.

11.3.3 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.

12 Security Algorithms

12.1 Integrity

12.1.1 MD4 Hash Algorithm

12.1.2 Other Algorithms

12.2 Authentication

12.2.1 MD4 with RSA signature Algorithm

12.2.2 Other Algorithms

12.3 ASN.1 Definitions

This section defines object identifiers assigned to algorithms. The following definitions take the form of the ASN.1 module, "oIWAlgorithmObjectIdentifiers":

```
OIWAlgorithmObjectIdentifiers {iso(1) identified-organization(3)
                                      oiw(14) secsig(3)
                                      OIWALGORITHMOBJECTIDENTIFIERS(1)}
DEFINITIONS ::=
BEGIN

EXPORTS
        md4, md4WithRSA

IMPORTS
        -- to be determined

-- categories of object identifiers

algorithm OBJECT IDENTIFIER ::={iso(1) identified-organization(3)
                                      oiw(14) secsig(3) algorithm(2)}

encriptionAlgorithm OBJECT IDENTIFIER ::= {algorithm 1}

hashAlgorithm OBJECT IDENTIFIER       ::= {algorithm 2}

signatureAlgorithm OBJECT IDENTIFIER  ::= {algorithm 3}

--algorithms

md4 ALGORITHM
        PARAMETER NULL
        ::= {hashAlgorithm 1}

md4WithRsa ALGORITHM
        PARAMETER NULL
        ::= {signatureAlgorithm 1}

END -- of Algorithm Object Identifier Definitions
```

Annex A (normative)

ISPICS Requirements List

Annex B (normative)

Errata

Table B.1 WIA PART 13 CHANGES

NO. OF ERRATA	TYPE	REFERENCED DOCUMENT	CLAUSE	NOTES
	TECHNICAL	WIA PART - 13	12	ADDED NEW
	EDITORIAL	WIA PART - 13	11	EDITED
	TECHNICAL	WIA PART - 13	2	ADDED REF
	EDITORIAL	WIA PART - 13	ALL	NOTES ADDED, EDITING

Annex C (normative)

Security Labels

Annex D (informative)

Bibliography

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

D.2 ISO/IEC DP 9796 Data Cryptographic Techniques

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

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

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

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

D.7 Access Control Information Specification (ACIS)

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

D.25 RFC-1113, Privacy Enhancement for Internet Electronic Mail: Part I - Message Encipherment and Authentication Procedures.

D.26 RFC-1114, Privacy Enhancement for Internet Electronic Mail: Part II - Certificate-Based Key Management.

D.27 RFC-1115, Privacy Enhancement for Internet Electronic Mail: Part III - Algorithms, Modes, and Identifiers (August 1989).

D.28 Network Layer ISO/IEC JTC1 SC6

D.29 Transport Layer ISO/IEC JTC1 SC6 6285

D.30 Lower Layer ISO/IEC JTC1 SC6 6227

Annex E (informative)

STATUS

Table E.1 ISO STATUS

DOCUMENT	WD	CD	DIS	IS
ISO/IEC JTC1 SC21/WG1 N5044	X	X	X	6/91
NETWORK LAYER ISO/IEC JTC1 SC6	X	7/91		
TRANSPORT LAYER ISO/IEC JTC1 SC6	X	X	7/91	
LOWER LAYER ISO/IEC JTC1 SC6 6227	X			

NOTE - This table was not included in any motion presented to the Plenary in December 1990.

Annex F (informative)

Security-SIG Management Plan

DOCUMENT	NEXT MILESTONE	DATE
ISO/IEC JTC1 SC21 N3614		
ISO/IEC DP 9796		
SDN-601/NIST IR 90-4262	COMPLETED	
SDN-701/NIST IR 90-4250	COMPLETED	
SDN-702/NIST IR 90-4250	COMPLETED	
ISO/IEC JTC1 SC21/WG1 N5002		
SDN-902/NIST IR 90-4262	COMPLETED	
SDN-903/NIST IR 90-4262	COMPLETED	
ISO/IEC JTC1 SC21/WG1 N4110		
SDN-301/NIST IR 90-4250	COMPLETED	
SDN-401/NIST IR 90-4250	COMPLETED	
SDN-906/NIST IR 90-4262	COMPLETED	
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		
DOCUMENT	NEXT MILESTONE	DATE
ISO/IEC JTC1 SC21/WG7 N4022		

ISO/IEC JTC1 SC21/WG1 N5048		
ISO/IEC JTC1 SC21/WG1 N5049		
ISO/IEC JTC1 SC21/WG1 N5044	IS	6/91
NETWORK LAYER ISO/IEC JTC1 SC6	CD	7/91
TRANSPORT LAYER ISO/IEC JTC1 SC6 6285	DIS	7/91
LOWER LAYER ISO/IEC JTC1 SC6 6227	WD	N/A

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 14 - Virtual Terminal

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair **Luke Lucas**
SIG Editors **Luke Lucas**

Table of Contents

Foreword

This part of the Working Implementation Agreements was prepared by the Virtual Terminal Special Interest Group (VTSIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the above-mentioned Workshop. This part replaces the previously existing chapter on this subject. There is no significant technical change from this text as previously given.

Three normative annexes are given.

Part 14 ISO Virtual Terminal Protocol

Editor's Note - References to Stable Agreements in this part refer to Version 4 dated December 1990.

0 Introduction

See Stable Agreements.

1 Scope

1.1 Phase Ia Agreements

See Stable Agreements.

1.2 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.

1.3 Phase II Agreements

See Stable Agreements regarding X.3 profile.

The Page profiles are intended for applications which require page-oriented operation.

2 Normative References

3 Status

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

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.

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.

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.

4 Errata

5 Conformance

See Stable Agreements.

6 Protocol

See Stable Agreements.

7 OIW Registered Control Objects

7.1 Sequenced Application (SA)

See Stable Agreements.

7.2 Unsequenced Application (UA)

See Stable Agreements.

7.3 Sequenced Terminal (ST)

See Stable Agreements.

7.4 Unsequenced Terminal (UT)

See Stable Agreements.

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.

7.5.1 Entry Number

To be supplied by the Registration Authority.

7.5.2 Name of Sponsoring Body

NIST/OSI Workshop for Implementors of OSI, VTSIG.

7.5.3 Date

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

7.5.4 Identifier

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

7.5.5 Descriptor Value

"OIW VT CO for Termination Conditions"

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)* }
```

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

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.

7.5.9 Usage

Defined in profile.

8 OIW Defined VTE-Profiles

8.1 Telnet Profile

See Stable Agreements.

8.2 Transparent Profile

See Stable Agreements.

8.3 Forms Profile

See Stable Agreements.

8.4 X3 Profile

See Stable Agreements.

8.5 Scroll Profile

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

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".

8.5.2 Association Requirements

8.5.2.1 Functional Units

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

8.5.2.2 Mode

This profile operates in A-mode.

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".
```

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.

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.

8.5.6 Profile Notes

8.5.6.1 Definitive Notes

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.

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).

8.5.6.2 Informative Notes

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.

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.

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

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.

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

8.5.7 Specific Conformance Requirements

None.

Annex A (normative)

Specific ASE Requirements

See Stable Agreements.

Annex B (normative)

Clarifications

See Stable Agreements.

Annex C (normative)

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) }
```

Working Agreements for Open Systems Interconnection Protocols:
Part 15 - Transaction Processing

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair	**Jeff Hildebrand**
SIG Editor	**Jeff Hildebrand**

Table of Contents

Foreword

This part of the Working Agreements was prepared by the Transaction Processing Special Interest Group (TPSIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the above-mentioned Workshop. This part replaces the previously existing chapter on this subject. There is no significant technical change from this text as previously given. References are made to other sections of both the Working and Stable agreements.

Annex A is normative and contains all PDU specifications.

Annex B is normative and contains all notes used in Annex A.

Part 15 - Transaction Processing

0 Introduction

The NIST/OIW Transaction Processing (TP) SIG is developing implementation agreements for the TPmodel, 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:

a) Atomicity: a property of a set of related operations such that the operations are either all performed, or none of them are performed.

b) 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.

c) Isolation: a property of a set of related operations uch that the partial results of the operations are not accessible, except by operations of the set.

d) 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.

1 Scope

This profile will address the following areas:

a) Specification of functional units:

1) Kernel

2) Polarized Control

3) Shared Control

4) Handshake

5) Commit

6) Unchained Transactions

b) Combining functional units into profiles:

1) ATP-1 Polarized application transaction

2) ATP-2 Polarized provider supported chained transaction

3) ATP-3 Polarized provider supported unchained transaction

4) ATP-4 Shared application transaction

5) ATP-5 Shared provider supported chained transaction

6) ATP-6 Shared provider supported unchained transaction

c) Agreements covering TP services and generation of TP protocol.

d) Agreements covering the use of the following OSI services by TP:

1) ACSE for association management

2) CCR for support of provider supported ACID properties

3) Presentation services

4) Directory services

e) Agreements with regard to implementation issues not specified in ISO 10026.

f) Statement of requirements to meet conformance to the agreements.

g) Additionally, the following interoperability issues will be addressed:

1) TP usage by other OSI standards

2) Application context

3) Security

2 Normative References

The following documents contain provisions which, through reference in this text, constitute provisions of this profile. At the time of publication, the editions indicated were valid. All documents are subject to revision, and parties to agreements based on this profile are warned against automatically applying any more recent editions of the documents listed below, since the nature of references made by profiles to such documents, is that they may be specific to a particular edition. ISO and NIST OIW maintain registers of currently valid standards and agreements used in this profile.

ISO

ISO 10026-1: Information Processing Systems - Open Systems Interconnection - Distributed Transaction Processing - Part 1: Model

ISO 10026-2: Information Processing Systems - Open Systems Interconnection - Distributed Transaction Processing - Part 1: Service

ISO 10026-3: Information Processing Systems - Open Systems Interconnection - Distributed Transaction Processing - Part 1: Protocol

ISO 9804:1990 Information Processing Systems - Open Systems Interconnection - Service Definition of Common Application Service Elements - Concurrency, Commitment and Recovery

ISO 9805:1990 Information Processing Systems - Open Systems Interconnection - Specification of Protocols for Common Application Service Elements - Concurrency, Commitment and Recovery

ISO 8649:1988 Information Processing Systems - Open Systems Interconnection - Service Definition for the Association Control Service Element

ISO 8650:1988 Information Processing Systems - Open Systems Interconnection - Protocol specification for the Association Control Service Element

ISO 8822:1989 Information Processing Systems - Open Systems Interconnection Connection Oriented Presentation Service Definition

ISO 9594: Information Technology - Open Systems Interconnection - The Directory

3 Definitions

4 Abbreviations

5 Taxonomy

5.1 Functional Units

This subclause specifies the PDUs which comprise each functional unit.

a) Kernel

   1) TP-BEGIN-DIALOGUE-RI

2) TP-BEGIN-DIALOGUE-RC

3) TP-END-DIALOGUE-RI

4) TP-REJECT-RI

5) TP-U-ERROR-RI

6) TP-U-ERROR-RC

7) TP-P-ERROR-RI

8) TP-ABORT-RI

9) TP-BID-RI

10) TP-BID-RC

11) TP-ASSOCIATION-ESTABLISHMENT-RI

12) TP-ASSOCIATION-ESTABLISHMENT-RC

b) Polarized Control

1) TP-GRANT-CONTROL-RI

2) TP-REQUEST-CONTROL-RI

c) Shared Control

1) No PDUs specified

d) Handshake

1) TP-HANDSHAKE-RI

2) TP-HANDSHAKE-RC

3) TP-HANDSHAKE-AND-GRANT-CONTROL-RI

4) TP-HANDSHAKE-AND-GRANT-CONTROL-RC

e) Commit

1) TP-PREPARE-RI

2) TP-DEFER-RI (Grant-Control/End-Dialogue)

f) Unchained Transactions

1) TP-DEFER-RI (Next-Transaction)

2) TP-UNCHAIN-RI

3) TP-BEGIN-TRANSACTION-RI

5.2 Profiles

This subclause specifies which functional units combine to form each profile. Refer to Annex A for the specification of how a specific profile uses a PDU and its parameters.

5.2.1 ATP-1 Polarized application transaction

KERNEL + POLARIZED CONTROL + HANDSHAKE

5.2.2 ATP-2 Polarized provider supported chained transaction

KERNEL + POLARIZED CONTROL + COMMIT + HANDSHAKE

5.2.3 ATP-3 Polarized provider supported unchained transaction

KERNEL + POLARIZED CONTROL + COMMIT + HANDSAKE + UNCHAINED

5.2.4 ATP-4 Shared application transaction

KERNEL + HANDSHAKE (TBD)

5.2.5 ATP-5 Shared provider supported chained transaction

KERNEL + COMMIT + HANDSHAKE (TBD)

5.2.6 ATP-6 Shared provider supported unchained transaction1

KERNEL + COMMIT + HANDSHAKE (TBD) + UNCHAINED

5.3 TP Use of OSI Services

5.3.1 ACSE - Association Management

5.3.2 CCR - Provider Supported ACID Properties

5.3.3 Presentation Services

5.3.4 Directory Services

5.4 Interoperability Issues

5.4.1 Application Context

5.4.2 Security

5.4.3 Recovery

5.4.4 Recommended Practices

5.5 Conformance Statement

An implementation conformant to a profile must be able to implement the functional units of that profile; it may additionally implement other functional units without being nonconformant.

An implementation conforming to a given profile may accept a dialogue outside this profile, if it does this it:

a) Does not violate conformance to the original profile.

b) Must now conform to the accepted profile.

Annex A (normative)

OSI Transaction Processing Protocol PDUs

This annex is normative, and details all the protocol PDUs used in this profile and by OSI Transaction Processing to deliver an interoperable transaction processing environment. The format of the table is for a PDU to be specified once for all profiles and any differences noted in the Type/Length/Value Allowed column. The intent of this approach is to provide the user with a single PDU specification which is: complete, compact, and easily compared between profiles.

Usage of this annex requires the user to:

a) Identify the necessary PDUs for a profile by consulting the section on profile specification.

b) Have an understanding of how the PDU tables are constructed:

1) The Item number uniquely identifies each parameter within the annex.

2) The parameter column provides the name of each PDU parameter.

3) The status columns indicate requirements for the field:

a) M = Mandatory

b) C = Conditional

c) O = Optional

d) NA = Not applicable

e) X = Excluded

f) TBD = To be determined

g) U = ACSE service-user option

h) NU = Not used

4) The REFERENCE column points to the page within the referenced document where this parameter is defined.

5) The PROFILE ID column, if present, defines how this paramter is used by a specific profile. The column will have an identifier in the form of the profile number, eg., 1 = ATP-1.

6) The T/L/V ALLOWED column specifies the range of type, length, or values this parameter can assume or contain. This column can have multiple definitions based on which profile is being described. When multiple definitions are possible this column will be defined in conjunction with the Profile ID column.

7) The NOTES column points to note contained in Annex B. These notes can be editorial or tutorial in nature.

A.1 Transaction Processing Protocol PDUs

A.1.1 TP-Begin-Dialogue-RI

TP-BEGIN-DIALOGUE-RI, Sending - to begin a dialogue

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
101	Initiating-TPSU-Title	O(1)			O	0..2**31-1	1
102	Recipient-TPSU-Title	M(1)			M	0..2**31-1	1
103	Selected-Functional-Units	M(1)			M		1
104	Commit	O		1,4	X		1
				2,3,5,6	M		1
105	Polarized-Control	O		1,2,3	M		1
				4,5,6	X		1
106	Handshake	O			TBD		1
107	Unchained-Transactions	O		3,6	M		1
				1,2,4,5	X		1
108	Initial-Coordination-Level	C		3,6	M		1
				1,2,4,5	N/A		1
109	Invocation-data	O(1)			O		1,2
110	Dialogue/Channel-Identifier	M(1)			M	0..2**31-1	1

TP-BEGIN-DIALOGUE-RI, Sending - to begin a TP channel

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
201	Dialogue/Channel-Identifier	M		2,3,5,6	M	0..2**31-1	1
				1,4	N/A		1
202	Channel-utilization	C(1)		2,3,5,6	M		1
				1,4	N/A		1

TP-BEGIN-DIALOGUE-RI, Receiving - to begin a dialogue

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
301	Initiating-TPSU-Title	O(1)			O	0..2**31-1	1
302	Recipient-TPSU-Title	M(1)			●	0..2**31-1	1
303	Selected-Functional-Units	M(1)			M		1
304	Commit	O		1,4	X		1
				2,3,5,6	M		1
305	Polarized-Control	O		1,2,3	M		1
				4,5,6	X		1
306	Handshake	O			TBD		1
307	Unchained-Transactions	O		3,6	M		1
				1,2,4,5	X		1
308	Initial-Coordination-Level	C		3,6	M		1
				1,2,4,5	N/A		1
309	Invocation-data	O(1)			O		1,2
310	Dialogue/Channel-Identifier	M(1)			M	0..2**31-1	1

TP-BEGIN-DIALOGUE-RI, Receiving - to begin a TP channel

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
401	Dialogue/Channel-Identifier	M		2,3,5,6	M	0..2**31-1	1
				1,4	N/A		1

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
402	Channel-Utilization	M(1)		2,3,5,6	M		1
				1,4	N/A		1

A.1.2 TP-BEGIN-DIALOGUE-RC

TP-BEGIN-DIALOGUE-RC, Sending

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
501	Dialogue/Channel-Identifier	M			M	0..2**31-1	

TP-BEGIN-DIALOGUE-RC, Receiving

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
601	Dialogue/Channel-Identifier	M			M	0..2**31-1	

A.1.3 TP-REJECT-RI

TP-REJECT-RI, Sending

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
701	Type	M			M		
702	Diagnostic	C		1,4	M	1-5, 42	3,5
				2,3,5,6	M	1-6, 42	3,5
703	User-data	O			O		2,4
704	Dialogue/Channel-Identifier	M			M	0..2**31-1	

TP-REJECT-RI, Receiving

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
801	Type	●			●		
802	Diagnostic	C		1,4	M	1-5, 42	3,5
				2,3,5,6	M	1-6, 42	3,5
803	User-data	O			O		2,4
804	Dialogue/Channel-Identifier	M			M	0..2**31-1	

A.1.4 TP-BID-RI

No parameters

A.1.5 TP-BID-RC

TP-BID-RC, Sending

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
901	Result	M			M		

TP-BID-RC, Receiving

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/Ł/V ALLOWED	NOTES
1001	Result	M			M		

A.1.6 TP-END-DIALOGUE-RI

No parameters

A.1.7 TP-U-ERROR-RI

No parameters

A.1.8 TP-U-ERROR-RC

No parameters

A.1.9 TP-P-ERROR-RI

TP-P-ERROR-RI, Sending

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1101	Diagnostic	M		1,4	N/A		6
				2,3,5,6	M		

TP-P-ERROR-RI, Receiving

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1201	Diagnostic	M		1,4	N/A		6
				2,3,5,6	M		

A.1.10 TP-ABORT-RI

TP-ABORT-RI, Sending

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1301	Type	M			M		
1302	Diagnostics	C		1,2,4,5	M	1, 2	7,9
				3,6	M	1, 2, 3	7,9
1303	User-data	C			O		2,8

TP-ABORT-RI, Receiving

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1401	Type	M			M		
1402	Diagnostics	C		1,2,4,5	M	1, 2	7,9
				3,6	M	1, 2, 3	7,9
1403	User-data	C			O		2,8

A.1.11 TP-REQUEST-CONTROL-RI

No parameters

A.1.12 TP-GRANT-CONTROL-RI

No parameters

A.1.13 TP-HANDSHAKE-RI

TP-HANDSHAKE-RI, Sending

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1501	Type	M			M		
1502	Confirmation	C			M		

TP-HANDSHAKE-RI, *Receiving*

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1601	Type	M			M		
1602	Confirmation	C			M		10

A.1.14 TP-HANDSHAKE-RC

No parameters

A.1.15 TP-HANDSHAKE-AND-GRANT-CONTROL-RI

TP-HANDSHAKE-AND-GRANT-CONTROL-RI, Sending

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1701	Confirmation	M		1,2,3	M		
				4,5,6	X		

TP-HANDSHAKE-AND-GRANT-CONTROL-RI, Receiving

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1801	Confirmation	M		1,2,3	M		
				4,5,6	X		

A.1.16 TP-HANDSHAKE-AND-GRANT-CONTROL-RC

No parameters

A.1.17 TP-DEFER-RI

TP-DEFER-RI, Sending

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
1901	End-dialogue	O			M		
1902	Grant-control	O		1,2,3	M		
				4,5,6	X		
1903	Next-Transaction	O		3,6	M		
				1,2,4,5	X		

TP-DEFER-RI, Receiving

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
2001	End-dialogue	O			M		
2002	Grant-control	O		1,2,3	M		
				4,5,6	X		
2003	Next-Transaction	O		3,6	M		
				1,2,4,5	X		

A.1.18 TP-PREPARE-RI

TP-PREPARE-RI, Sending

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
2101	Data-permitted	O			TBD		

TP-PREPARE-RI, Receiving

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
2201	Data-permitted	O			TBD		

A.1.19 TP-UNCHAIN-RI

No parameters

A.1.20 TP-BEGIN-TRANSACTION-RI

TP-BEGIN-TRANSACTION-RI, Sending

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
2301	Chain	M		3,6	M		
				1,2,4,5	X		

TP-BEGIN-TRANSACTION-RI, Receiving

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
2401	Chain	M		3,6	M		
				1,2,4,5	X		

A.1.21 TP-ASSOCIATION-ESTABLISHMENT-RI

TP-ASSOCIATION-ESTABLISHMENT-RI, Sending

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
2501	Protocol Version	M			M		
2502	Contention winner assignment	M			M		
2503	Bid-Mandatory	M			M		

TP-ASSOCIATION-ESTABLISHMENT-RI, Receiving

	BASE STANDARD ISO 10026-3			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
2601	Protocol Version	M			M		
2602	Contention winner assignment	M			M		
2703	Bid-Mandatory	M			M		

A.2 ACSE PROTOCOL PDUs

This subclause shows TP's use of ACSE services and parameters. The reader should consult the upper layer agreements for a detailed discussion of this service. This ISP only specifies PDU parameters necessary for the Transaction Processing ISP.

A.2.1 AARQ

	BASE STANDARD ISO 8650			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T/L/V ALLOWED	NOTES
2801	Protocol Version	O			M		
2802	Application Context Name	M			M		
2803	Calling AP Title	U					
2804	Calling AE Qualifier	U					
2805	Calling AP Invocation Identifier	M					
2806	Calling AE Invocation Identifier	M					
2807	Called AP Title	U					
2808	Called AE Qualifier	U					
2809	Called AP Invocation Identifier	U					
2810	Called AE Invocation Identifier	U					
2811	Implementation Information	O					
2812	User Information	U					

A.2.2 AARE

	BASE STANDARD ISO 8650			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
2901	Protocol Version	O			M		
2902	Application Context Name	M			M		
2903	Responding AP Title	U					
2904	Responding AE Qualifier	U					
2905	Responding AP Invocation Identifier	U					
2906	Responding AE Invocation Identifier	U					
2907	Result	M			M		
2908	Result Source - Diagnostic	M			M		
2909	User Information	U			M		
2910	Implementation Information	U			M		

A.2.3 RLRQ

	BASE STANDARD ISO 8650			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
3001	Reason	U			NU		
3002	User information	U			NU		

A.2.4 RLRE

	BASE STANDARD ISO 8650			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
3101	Reason	U			NU		
3102	User information	U			NU		

A.2.5 ABRT

	BASE STANDARD ISO 8650			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
3201	Abort Source	M					
3202	User Information	U			NU		

A.3 PRESENTATION SERVICE PARAMETERS

This subclause shows TP's use of Presentation services and parameters. The reader should consult the Upper Layer agreements for a detailed discussion of these services.

A.3.1 P-TOKEN-PLEASE

P-TOKEN-PLEASE, Sending

	BASE STANDARD ISO 8822			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
3301	Tokens						1
3302	User-data	NU			NU		

P-TOKEN-PLEASE, Receiving

	BASE STANDARD ISO 8822			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
3401	Tokens						1
3402	User-data	NU			NU		

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

A.3.2 P-TOKEN-GIVE

P-TOKEN-GIVE, Sending

	BASE STANDARD ISO 8822			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
3501	Tokens	M			M		

P-TOKEN-GIVE, Receiving

	BASE STANDARD ISO 8822			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
3601	Tokens	M			M		

A.3.3 P-DATA

P-DATA, Sending

	BASE STANDARD ISO 8822			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
3701	User-data	M			M		

P-DATA, Receiving

	BASE STANDARD ISO 8822			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
3801	User-data	M			M		

A.4 CCR SERVICE PARAMETERS

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

A.4.1 C-BEGIN

C-BEGIN, Sending

	BASE STANDARD ISO 9805			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
3901	Atomic Action Id.- Master's Name	M			M		
3902	Atomic Action Id.- Suffix	M			M		1
3903	Branch Id.-Superior's Name	M			M		
3904	Branch Id.-Suffix	M			M		1
3905	User Data	C			M		

C-BEGIN, Receiving

	BASE STANDARD ISO 9805			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
4001	Atomic Action Id.- Master's Name	M			M		
4002	Atomic Action Id.- Suffix	M			M		1
4003	Branch Id.-Superior's Name	M			M		
4004	Branch Id.-Suffix	M			M		1
4005	User Data	C			M		

A.4.2 C-PREPARE

C-PREPARE, Sending

	BASE STANDARD ISO 9805	PROFILE

ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
4101	User-data	C			M		

C-PREPARE, Receiving

	BASE STANDARD ISO 9805			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
4201	User-data	C			M		

A.4.3 C-READY

C-READY, Sending

	BASE STANDARD ISO 9805			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
4301	User-data	NU			NU		

C-READY, Receiving

	BASE STANDARD ISO 9805			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
4401	User-data	NU			NU		

A.4.4 C-COMMIT

C-COMMIT, Sending

	BASE STANDARD ISO 9805			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
4501	User-data	C			M		

C-COMMIT, Receiving

	BASE STANDARD ISO 9805			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
4601	User-data	C			M		

A.4.5 C-ROLLBACK

C-ROLLBACK, Sending

	BASE STANDARD ISO 9805			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
4701	User-data	C			M		

C-ROLLBACK, Receiving

	BASE STANDARD ISO 9805				PROFILE		
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
4801	User-data	C			M		

A.4.6 C-RECOVER

C-RECOVER, Sending

	BASE STANDARD ISO 9805			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
4901	Recovery State	M			M		
4902	Atomic Action Identifier	M			M		
4903	Branch Identifier	M			M		
4904	User-data	C			M		

C-RECOVER, *Receiving*

	BASE STANDARD ISO 9805			PROFILE			
ITEM#	PARAMETER	STATUS	REFERENCE	PROFILE ID	STATUS	T\L\V ALLOWED	NOTES
5001	Recovery State	M			M		
5002	Atomic Action Identifier	M			M		
5003	Branch Identifier	M			M		
5004	User-data	C			M		

Annex B (normative)

NOTES

1 Status reflects the base standard value when PDUs are expressed as seperate PDUs and not as a combined PDU, which is how the base standard expresses them.

2 May need to determine limits on the amount and type of data passed in this manner.

3 User/Provider division of values is unclear in the Standard's ASN.1.

4 Parameter is present on user rejects.

5 Parameter is present on provider rejects.

6 Presently defined values only applicable to profiles 2,3,5 and 6.

7 May want to specify meanings for reason codes, Permanent and Transient failure.

8 Parameter is present on user abort.

9 Parameter is present on provider abort.

10 Parameter is present only on Handshake when the Shared Control functional unit is active.

11 Only if CCR is used, else the parameter is a user option.

12 Parameter becomes mandatory if the association is being established for recovert purposes (TP Channels).

13 Must decide which CCR ASN.1 Choice to use.

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 16 - Office Document Architecture Level 3 DAP

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair **Frank Dawson**
SIG Editor **Frank Dawson**

Table of Contents

Foreword

This part of the Working Implementation Agreements was prepared by the Office Document Architecture Group (ODASIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop Charter.

Text in this part has been approved by the Plenary of the Workshop. This part replaces the previously existing chapter on this subject.

Part 16 - Office Document Architecture Level 3 DAP

NOTE - Text for the proposed draft International Standardized Profile F0D36 will be placed here.

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 17 - Office Document Architecture Level 2 DAP

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair **Frank Dawson**
SIG Editor **Frank Dawson**

Table of Contents

Foreword

This part of the Working Implementation Agreements was prepared by the Office Document Architecture Group (ODASIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop Charter.

Text in this part has been approved by the Plenary of the Workshop. This part replaces the previously existing chapter on this subject. There are some significant technical changes to this text as previously given.

Part 17 - Office Document Architecture Level 2 DAP

NOTE - Text reflecting the proposed draft International Standardized Profile F0D26 will be inserted here; previous text was sent in (camera-ready form and as such could not be updated).

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 18 - Network Management

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Co-Chair	**Paul Brusil**
SIG Co-Chair	**George Mouradian**
SIG Editor	**Robert Aronoff**

Table of Contents

List of Tables

List of Figures

Foreword

This part of the Working Implementation Agreements was prepared by the Network Management Special Interest Group (NMSIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the above-mentioned Workshop. This part replaces the previously existing chapter on this subject.

18 Network Management

0 Introduction

(Refer to the Stable Implementation Agreements Document.)

1 Scope

(Refer to the Stable Implementation Agreements Document.)

2 Normative References

The following documents are referenced in the statements of the agreements relating to OSI network management. The notation "*" indicates that tentative object identifiers contained in these DIS-level documents are superseded by the NMSIG Phase 1 object identifiers contained in ANNEX B.2 of these agreements.

[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.

[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.

[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.

[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 I989.

[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.

[DIR] ISO 9594 - Information Processing Systems - Open Systems Interconnection - The Directory, 1988.

[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.

[ERMF]* ISO/IEC DIS 10164-5, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 5: Event Report Management 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.

[ISPFRM] ISO/IEC TR 10000-1, Information Technology - Framework and Taxonomy of International Standardized Profiles - Part 1: Framework, ISO/IEC JTC1/SGFS N184, 9 February 1990.

[ISPSRVC] ISO/IEC TR 8509, Information Processing Systems - Open Systems Interconnection - Service Conventions, TC97/SC16/1646.

[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.

[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.

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

[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.

[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.

[SD35] EWOS/EG/NM/90/xx, Information Technology - Profiles AOMnn OSI Management - Management Communications Protocols - Part x: AOM12 - Full CMIP for Managing & Managed Systems, 7 September 1990.

[SF] ISO/IEC CD 10164-13, Information Processing Systems - Open Systems Interconnection - Systems Management - Part 13: Summarization Function, ISO/IEC JTC1/SC21 N5519, December, 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.

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

[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.

3 Status

The following clauses were moved into the Stable Agreements in June 1990:

0 INTRODUCTION

2 NORMATIVE REFERENCES (i.e., only those relevant to the Stable Agreements)

6 MANAGEMENT COMMUNICATIONS

6.2 General Agreements on Users of CMIS

6.3 Specific Agreements on Users of CMIS

6.4 Specific Agreements on CMIP

The following clauses were moved to the Stable Agreements in December 1990:

1 SCOPE

1.1 Phased Approach

1.1.1 Alignment With Evolving Standards

1.1.2 Definition of Phase 1

1.1.3 Future Phases

2 NORMATIVE REFERENCES (i.e., only those relevant to the newly added Stable Agreements)

5 MANAGEMENT FUNCTIONS AND SERVICES

5.1 General Agreements

5.2 Object Management Function Agreements

5.3 State Management Function Agreements

5.4 Attributes For Representing Relationships Agreements

5.5 Alarm Reporting Function Agreements

5.6 Event Report Management Function Agreements

6 MANAGEMENT COMMUNICATIONS

6.1 Association Policies

7 MANAGEMENT INFORMATION

7.1 The Information Model

7.2 Principles of Naming

7.3 Guidelines for the Definition of Management Information

The following clauses are planned to be added to the Stable Agreements in March 1991:

6 MANAGEMENT COMMUNICATIONS

6.5 Services Required by CMIP

ANNEX B.2 Phase 1 Object Identifiers

The following clauses are planned to be added to the Stable Agreements in June 1991:

8 CONFORMANCE

4 Errata

(None as yet)

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 management function [ERMF] 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 [STMF], Attributes For Representing Relationships [ARR], Alarm Reporting [ARF], Event Report Management [ERMF], 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 Summarization [SF].

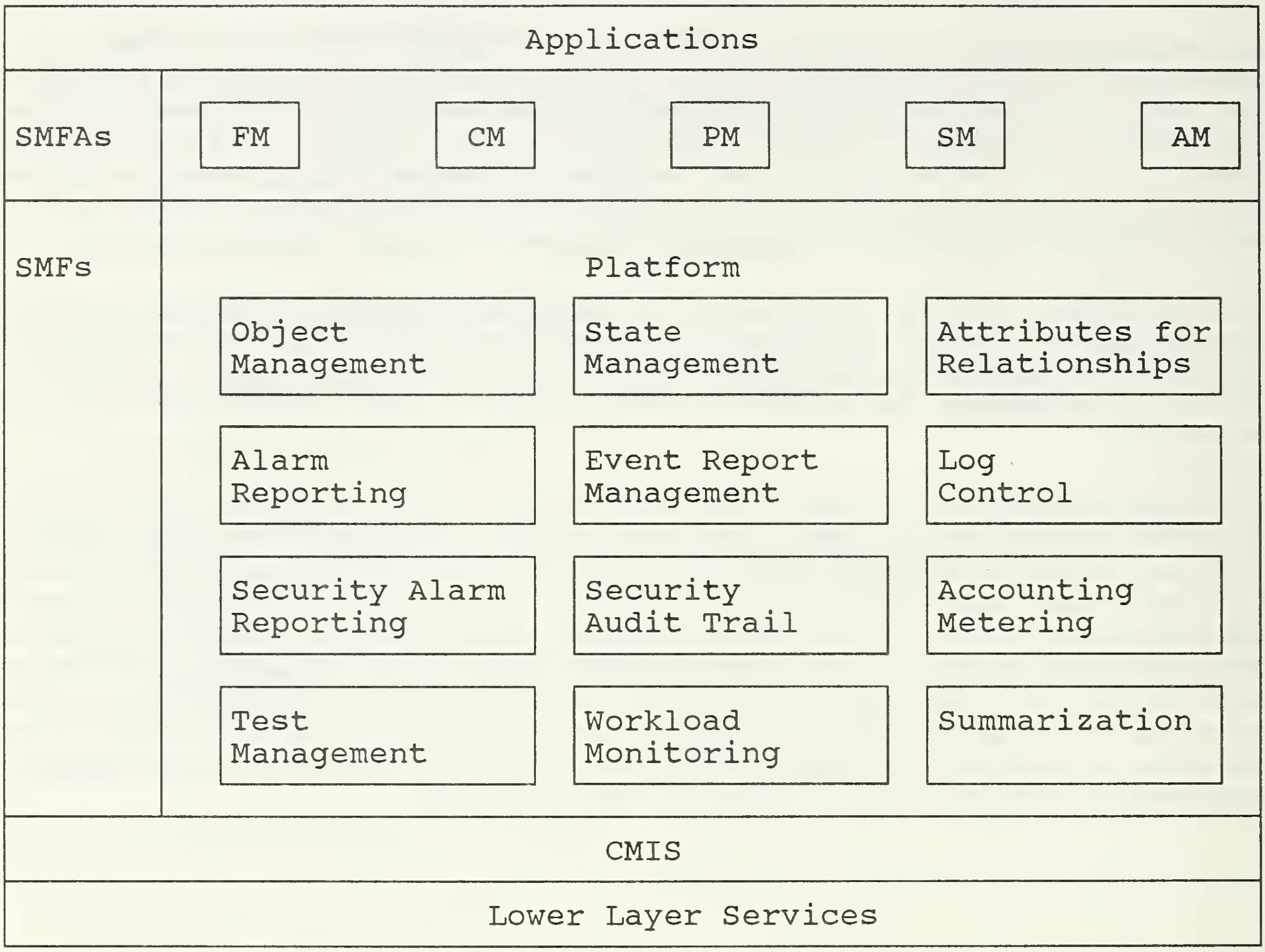


Figure 1: Functional Hierarchy of SMFs and SMFAs.

5.1 General Agreements

(Refer to the Stable Inplementation Agreements Document.)

5.2 Object Management Function Agreements

(Refer to the Stable Inplementation Agreements Document.)

5.3 State Management Function Agreements

(Refer to the Stable Inplementation Agreements Document.)

5.4 Attributes For Representing Relationships Agreements

(Refer to the Stable Inplementation Agreements Document.)

5.5 Alarm Reporting Function Agreements

(Refer to the Stable Inplementation Agreements Document.)

5.6 Event Report Management Function Agreements

(Refer to the Stable Inplementation Agreements Document.)

5.7 Log Control Function Agreements

5.7.1 Introduction

This subclause provides agreements pertinent to the Log Control Function defined by [LCF].

The Log Control Function provides SMF services by which event reports and other PDUs can be selected and stored. Log activity can be scheduled. Events and other PDUs are selected for logging by use of a "Discriminator Construct" attribute within a Log object. Log Control provides the services to initiate, terminate, suspend, or resume the logging activity through the manipulation of a Log object specified in [DMI]. In addition, Log Control can further alter the selection behavior by changing the distribution attributes in a Log object (e.g., Discriminator Construct).

According to the Log Control Model defined by [LCF], the Log object receives event reports, or other PDUs, from various sources, and adds information to their contents to form "potential log records". If the Log object is in a condition that allows it to be active, then it will evaluate the "potential log records" according to matching criteria in the Log objects Discriminator Construct attribute. The result of this sieve process will yield zero, one or more log records to be stored in the Log object for later retrieval.

The Log Control Function uses the State Management Function for the notification of state changes, and the Object Management Function for creating and deleting Log objects, retrieving Log attribute values, and notification of Log attribute value changes, Log record retrieval, and Log record deletion. It also uses the processing alarm notification of the Alarm Reporting Function [ARF].

The Log Control Function makes use of the following management support objects defined in [DMI]:

log, and
logRecord.

The Log Control Function makes use of the following attributes defined in [DMI], in addition to those attributes defined for the object class top:

logID,
discriminatorConstruct,
administrativeState,
operationalState,
usageState,
availabilityStatus,
maxLogSize,
currentLogSize,
numberOfRecords,
capacityAlarmThreshold,
logFullAction,
intervalsOfDay,
startTime,
stopTime,
weekMask, and
schedularName.

The Log Control Function makes use of the following notification types defined in [DMI]:

objectCreation,
objectDeletion,
stateChange,
attributeValueChange, and
processingErrorAlarm.

Editor's Note: [The [LCF] specifies "alarmNotification" which does not exist in [DMI]; the correct notification is "processingErrorAlarm". All other notifications are spelled incorrectly in [LCF]; the [DMI] spellings are used here. [LCF] does not specify "usageState" or "intervalsOfDay", but both are included here and in the [DMI] definition of the "Log" object class.]

5.7.2 General Agreements

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

Table 1: Scope of Agreements Relating to SMF Services Defined by the Log Control Standard [LCF]

Log Control SMF Service	Within Scope Of Agreements	Related Management Support Objects
Initiation of LCF	Yes	Log
Termination of LCF	Yes	Log
Log Modification, Suspension, Resumption	Yes	Log
Retrieving Logging Attributes	Yes	Log
Retrieval of Log Records	Yes	Log, Log Record
Deletion of Log Records	Yes	Log, Log Record

5.7.3 Initiation Of Event Report Logging

5.7.3.1 Introduction

This SMF service allows one open system to request that another open system create a Log object, thereby requesting that new or additional logs be defined.

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

Table 2: Mapping Between LCF Initiation of Logging, OMF PT-Create, and CMIS M-CREATE Service Parameters

SMF Initiation of LCF Parameter	Req	Rsp	OMF PT-Create & CMIS M-CREATE Parameter	Req	Rsp
Invoke Identifier	M	M(=)			
Managed Object Class	M	C			

SMF Initiation of LCF Parameter	Req	Rsp	OMF PT-Create & CMIS M-CREATE Parameter	Req	Rsp
Managed Object Instance	U	C			
Support Object Instance	U	-			
Access Control	U	-			
Reference Object Instance	U	-			
Discriminator Construct	U	C	Attribute List		
Administrative State	U	C	Attribute List		
Operational State	-	C	Attribute List		
Usage State	-	C	Attribute List		
Availability Status	-	C	Attribute List		
Max Log Size	U	C	Attribute List		
Current Log Size	U	C	Attribute List		
Number Of Records	U	C	Attribute List		
Capacity Alarm Threshold	U	C	Attribute List		
Log Full Action	U	C	Attribute List		
Packages	U	C	Attribute List		
Week Mask	U	C	Attribute List		
Intervals Of Day	U	C	Attribute List		
Start Time	U	C	Attribute List		
Stop Time	U	C	Attribute List		
Schedular Name	U	C	Attribute List		
Current Time	-	U			
Errors	-	C			

5.7.3.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the Initiation of Logging SMF service defined by section 9.2 of [LCF]. Relevant CMIS agreements defined in subclause 6.3.5 are repeated here for completeness.

Table 3: Agreements On Parameter Usage Pertinent to the Initiation of Logging SMF Service

SMF Initiation of LCF Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		6.4
Managed Object Class	M	C		6.2.6
Managed Object Instance	U	C		6.2.1, 6.3.5.1
Support Object Instance	U	-		6.2.1
Access Control	U	-		6.2.4
Reference Object Instance	U	-		6.2.1
Discriminator Construct	U	C	[1], 5.1.2.1	6.2.6, 6.3.5.2
Administrative State	U	C		6.2.6, 6.3.5.2
Operational State	-	C		6.2.6, 6.3.5.2
Usage State	-	C		6.2.6, 6.3.5.2
Availability Status	-	C		6.2.6, 6.3.5.2
Max Log Size	U	C		6.2.6, 6.3.5.2
Current Log Size	U	C		6.2.6, 6.3.5.2
Number Of Records	U	C		6.2.6, 6.3.5.2
Capacity Alarm Threshold	U	C		6.2.6, 6.3.5.2
Log Full Action	U	C		6.2.6, 6.3.5.2
Packages	U	C		6.2.6, 6.3.5.2
Week Mask	U	C	[3]	6.2.6, 6.3.5.2
Intervals Of Day	U	C	[2]	6.2.6, 6.3.5.2

SMF Initiation of LCF Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Start Time	U	C	[3]	6.2.6, 6.3.5.2
Stop Time	U	C	[3]	6.2.6, 6.3.5.2
Schedular Name	U>I	C>I	[4]	6.2.6, 6.3.5.2
Current Time	-	U		6.2.3
Errors	-	C		6.4.4, 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.

[2] The Daily Scheduling Package, if supported by an object, shall support at minimum the default 24 hour interval.

[3] The Weekly Scheduling Package, if supported by an object, shall support the default values for Start Time and Stop Time attributes. The Week Mask attribute shall support scheduling for each day of the week, and, at a minimum, the default 24 hour period for intervals of the day.

[4] Support for the External Schedular Package is beyond the scope of these agreements.

Editor's Note: [It is unclear whether "read-only" Log attributes such as LogId, objectClass, nameBindings, allomorphs, and name are permitted in the Attribute List parameter of the PT-CREATE request. This question has been submitted to ANSI X3T5.4. Depending upon the answer, it may be necessary to add an agreement on the initial values of these attributes. For now, the attribute list shown here has been made consistent with the attribute list shown for the corresponding [ERMF] service.]

5.7.4 Termination Of Logging

5.7.4.1 Introduction

This SMF service allows one open system to request that another open system delete one or more logs.

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

Table 4: Mapping Between LCF Termination of Logging, OMF PT-Delete, and CMIS M-DELETE Service Parameters

SMF Termination of LCF 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			

5.7.4.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the Termination of Logging SMF service defined by section 9.3 of [LCF]. Relevant CMIS agreements defined in subclause 6.3.6 are repeated here for completeness.

Table 5: Agreements On Parameter Usage Pertinent to the Termination of Logging SMF Service

SMF Termination of LCF Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		6.4
Linked Id	-	C		6.4
Base Object Class	M	-		6.2.6

SMF Termination of LCF Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Base Object Instance	M	-		6.2.1
Scope	U	-		6.2.2.1
Filter	U	-	5.1.2.1	6.2.2.2
Access Control	U	-		6.2.4
Synchronization	U	-		6.2.2.3
Managed Object Class	-	C		6.2.6
Managed Object Instance	-	C		6.2.1
Current Time	-	U		6.2.3
Errors	-	C		6.3.6.1, 6.4.4

5.7.5 Log Modification, Suspension, and Resumption

5.7.5.1 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 a Log object.

The following informative table defines the mapping between LCF Log 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 [LCF] and section 8.3.2 of [CMIS].

Table 6: Mapping Between LCF Log Modification, Suspension, and Resumption, OMF PT-Set, and CMIS M-SET Service Parameters

SMF LCF 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	-			

SMF LCF Mod/Suspend/Resume Parameter	Req	Rsp	PT-Set & CMIS M-SET Parameter	Req	Rsp
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	
Administrative State	U	C	Mod & Attribute List	M	
Max Log Size	U	C	Mod & Attribute List	M	
Capacity Alarm Threshold	U	C	Mod & Attribute List	M	
Log Full Action	U	C	Mod & Attribute List	M	
Week Mask	U	C	Mod & Attribute List	M	
Intervals Of Day	U	C	Mod & Attribute List	M	
Start Time	U	C	Mod & Attribute List	M	
Stop Time	U	C	Mod & Attribute List	M	
Schedular Name	U	C	Mod & Attribute List	M	
Current Time	-	U			
Errors	-	C			

5.7.5.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the Log Control Modification, Suspension, and Resumption SMF service defined by section 9.4 of [LCF]. Relevant CMIS agreements defined in subclause 6.3.3 are repeated here for completeness.

Table 7: Agreements On Parameter Usage Pertinent to the Log Control Modification, Suspension, and Resumption SMF Service

SMF LCF Mod/Suspend/Resume Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Invoke Identifier	M	M(=)		6.4
Linked Id	-	C		6.4
Mode	M	-	5.1.2.2	
Base Object Class	M	-		6.2.6
Base Object Instance	M	-		6.2.4
Scope	U	-		6.2.2.1
Filter	U	-	5.1.2.1	6.2.2.2
Access Control	U	-		6.2.4
Synchronization	U	-		6.2.2.3
Managed Object Class	-	C		6.2.6
Managed Object Instance	-	C		6.2.4
Discriminator Construct	U	C	[1], 5.1.2.1	6.2.6, 6.3.3.1, 6.3.3.3, 6.3.3.4
Administrative State	U	C		6.2.6, 6.3.3.1, 6.3.3.3, 6.3.3.4
Max Log Size	U	C	[5]	6.2.6, 6.3.3.1, 6.3.3.3, 6.3.3.4
Capacity Alarm Threshold	U	C		6.2.6, 6.3.3.1, 6.3.3.3, 6.3.3.4
Log Full Action	U	C		6.2.6, 6.3.3.1, 6.3.3.3, 6.3.3.4

SMF LCF Mod/Suspend/Resume Parameter	Req	Rsp	SMF Agreements	CMIS Agreements
Week Mask	U	C	[3]	6.2.6, 6.3.3.1, 6.3.3.3, 6.3.3.4
Intervals Of Day	U	C	[2]	6.2.6, 6.3.3.1, 6.3.3.3, 6.3.3.4
Start Time	U	C	[3]	6.2.6, 6.3.3.1, 6.3.3.3, 6.3.3.4
Stop Time	U	C	[3]	6.2.6, 6.3.3.1, 6.3.3.3, 6.3.3.4
Schedular Name	U>I	C>I	[4]	6.2.6, 6.3.3.1, 6.3.3.3, 6.3.3.4
Current Time	-	U		6.2.3
Errors	-	C		6.3.3.2, 6.4.4

[1] As specified in [CMIP], the value "AND {}" shall be used to represent an all-pass Discriminator Construct.

[2] The Daily Scheduling Package, if supported by an object, shall support at minimum the default 24 hour interval.

[3] The Weekly Scheduling Package, if supported by an object, shall support the default values for Start Time and Stop Time attributes. The Week Mask attribute shall support scheduling for each day of the week, and, at a minimum, the default 24 hour period for intervals of the day.

[4] Support for the External Schedular Package is beyond the scope of these agreements.

[5] The appropriate CMIS error (i.e., invalidAttributeValue) shall be returned for any attempt to set Max Log Size less than the value of Current Log Size.

5.7.6 Retrieving Logging Attributes

5.7.6.1 Introduction

This SMF service allows one open system to retrieve any of the readable attributes of the log using the PT-Get SMF service.

5.7.6.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the Log Control Retrieving Logging Attributes SMF service defined by section 9.5 of [LCF]. No agreements have been made beyond those defined for the PT-Get SMF service; refer to subclause 5.2.10 of these agreements.

Editor's Note: [A table will be added to this subclause if any additional LCF agreements are defined.]

5.7.7 Retrieval Of Log Records

5.7.7.1 Introduction

This SMF service allows one open system to retrieve log records from a log using the PT-Get SMF service.

5.7.7.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the Log Control Retrieval Of Log Records SMF service defined by section 9.6 of [LCF]. No agreements have been made beyond those defined for the PT-Get SMF service; refer to subclause 5.2.10 of these agreements.

Editor's Note: [A table will be added to this subclause if any additional LCF agreements are defined.]

5.7.8 Deletion Of Log Records

5.7.8.1 Introduction

This SMF service allows one open system to delete log records from a log using the PT-Delete SMF service.

5.7.8.2 Agreements On Parameter Usage

This subclause provides agreements pertinent to the Log Control Deletion Of Log Records SMF service defined by section 9.7 of [LCF]. No agreements have been made beyond those defined for the PT-Delete SMF service; refer to subclause 5.2.7 of these agreements.

Editor's Note: [A table will be added to this subclause if any additional LCF agreements are defined.]

6 Management Communications

(Refer to the Stable Inplementation Agreements Document.)

6.1 Association Policies

(Refer to the Stable Inplementation Agreements Document.)

6.1.1 Application Context Negotiation

(Refer to the Stable Inplementation Agreements Document.)

6.1.2 Functional Unit Negotiation

(Refer to the Stable Inplementation Agreements Document.)

6.1.3 Security Aspects of Associations

Editor's Note: [The security aspects of management associations are being pursued jointly by the NMSIG and the Security SIG. Any agreements generated as a result of this work will be added to this clause as they become available.]

6.2 General Agreements on Users of CMIS

(Refer to the Stable Inplementation Agreements Document.)

6.3 Specific Agreements on Users of CMIS

(Refer to the Stable Implementation Agreements Document.)

6.4 Specific Agreements on CMIP

(Refer to the Stable Implementation Agreements Document.)

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 part 5, subclause 12.1.7.

6.5.1 P-DATA Encoding

For encoding of each CMIP/ROSE PDU in a P-DATA, implementations shall be able to parse and process a maximum of 10,240 octets as they would be encoded in the Presentation "User-data" type according to the Basic Encoding Rules for ASN.1.

6.6 CMIP PICS

Refer to "Profile AOM12: Full CMIP for Managing and Managed Systems" [SD35].

7 Management Information

(Refer to the Stable Inplementation Agreements Document.)

8 Conformance

Editor's Note: [The editor has taken the liberty of modifying some of the explanatory text in this clause for clarification of the concepts.]

8.1 Introduction

Clause 8 specifies the conformance requirements for the NMSIG Implementation Agreements (IAs). Implementors of products will provide claims of conformance to these requirements. These claims will be in the form of Protocol Implementation Conformance Statements (PICS) and Managed Object Conformance Statements (MOCS). These requirements will also be used to develop test cases which will be used to validate claims of conformance. This clause defines the conformance requirements and criteria which shall be used to test implementations claiming conformance to these agreements.

Note: [Conformance requirements for these IAs, relating to services required of the upper layers and other ASEs, are discussed in clause 6.5.]

8.2 General Requirements of Conformance

Conformance for these agreements is designed to specify a well-defined set of services/functions. In addition, a taxonomy of managed object classes is needed. For the purposes of organization and clarity of these agreements, management has been divided into three classification areas. Clauses 5 (Management Functions and Services), 6 (Management Communications) and 7 (Management Information), state the agreements which comprise the three conformance classification areas, respectively. Within these classification areas, particular conformance classes are specified which delineate conformance requirements for a well-defined and bounded set of services/functions (e.g., within the System Management Functions conformance classification area, a conformance class is specificed which defines conformance to the State Management Function). Once a conformance class is delineated which specifies the set of requirements for that class, tests can be developed to evaluate conformance of products to that conformance class. And finally, for each conformance class, roles (Manager, Agent, or Manager/Agent) are specified. It is required that one or more roles be supported for each conformance class to which an implementation claims conformance. The development of those conformance classes will enable:

1) users to define procurement specifications.

2) vendors to define systems capabilities and features.

3) conformance test houses to define test cases.

Implementations claiming conformance to these Implementation Agreements shall comply with the requirements stated in the following clauses.

8.3 Management Roles

During a given association, an implementation shall operate in a manager, agent or manager/agent role as specified in clause 6.

A statement of claim, within each PICS or MOCS, shall be provided stating which role(s) (Manager, Agent or Manager/Agent) an implementation supports for each conformance class.

To claim conformance to the IAs, an implementation shall be conformant to at least one role within at least one of the following areas:

- o Management Communication
- o System Management Functions
- o Managed Object Classes

8.4 Specific Conformance Classifications

8.4.1 Management Communication

To be conformant within the Management Communication classification area, an implementation must conform to agreements in clause 6. Conformance to management communication also requires an implementor to state which optional capabilities (e.g., CMIP functional units) are supported in the implementation. These capabilities shall be stated in a PICS or in a high level statement of claim.

Editor's Note: [If a PICS Proforma (Proposed Clause 6.6 in NMSIG/90-121) is not available, what shall be used for phase 1. Issue: Manager, Agent Roles in CMIP]

No implementation claiming to be conformant to any conformance class of these agreements shall violate the protocol requirements specified in the protocol clause of these agreements. Every implementation must respond appropriately to correct and erroneous PDUs.

Conformance to agreements in clause 6 requires conformance to referenced ISO standards/CCITT Recommendations and to all other clauses referenced in 6, including the underlying services required by CMIP.

8.4.2 System Management Functions

To be conformant within the System Management Functions classification area, an implementation must state which functional unit(s) it supports.

To be conformant within this classification area, an implementation shall support at least one System Management Function in either a manager or agent role.

To be conformant to the Object Management Function [OMF], an implementation must conform to clause 5.2.

Editor's Note: [Is a test managed object needed, and are both functional units required?]

To be conformant to the State Management Function [STMF], an implementation must conform to clause 5.3 and all clauses referenced in 5.3.

Editor's Note: [Is a test object needed?]

To be conformant to the Attributes for Representing Relationships SMF [ARR], an implementation must conform to clause 5.4 and all clauses referenced in 5.4.

Editor's Note: [Is a test object needed?]

To be conformant to the Alarm Reporting Function [ARF], an implementation must conform to clause 5.5 and all clauses referenced in 5.5.

Editor's Note: [Is a test object needed?]

To be conformant to the Event Report Management Function [ERMF], an implementation must conform to clause 5.6 plus the Event Report Record management support object required by the ERMF Function.

Editor's Note: [Is a test object needed?]

Editor's Note: [What are the conformance ramifications for PDUs (information) that are outside the scope of these agreements?]

8.4.3 Managed Object

To claim conformance within the Managed Object classification area, an implementation must implement at least one of the following:

- one or more managed objects from the OIW NMSIG MIL; or
- any managed object not from OIW NMSIG MIL, providing this managed object is defined according to clause 7. Furthermore, this object shall require NMSIG management communication as specified in clause 6 and, as needed, one or more NMSIG SMFs as specified in clause 5. Managed object class definitions must be provided either in full or by reference to publicly available documents. Associated with any such managed object definition must be a registered managed object class identifier. All manadatory abstract syntaxes and semantics associated with that identifier must be used.

An implementation can claim conformance to a managed object if it meets all the criteria for that object class even if the implementation does not claim conformance to any superior object in the containment tree.

Editor's Note: [Name Binding/ Clarification for this requirement]

8.4.3.1 MOCS Proforma

The implementor must provide a statement specifying which managed object classes are supported. A MOCS proforma shall be completed by the implementor for each managed object class supported.

Editor's Note: [The Proforma is possibly a small form to be expanded by implementors for each managed object class.]

For each managed object class supported, the following must be supplied:

- a list of system management functions supported;
- a statement of pragmatic constraints (e.g., attribute values/ranges, initial values) supported, unless such constraints are defined in the managed object class definition;
- a statement of conditional packages supported.

Editor's Note: [Are conditional packages included as an implementor option or an instantiator option?]

ANNEX A -- Management Information Library (MIL)

MANAGEMENT INFORMATION LIBRARY

(MIL)

Version 6.0

December 10, 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.]

Managed objects in the MIL are not normative as far the NMSIG IAs are concerned. Implementors do not have to support any of the MIL managed objects; they may choose to define their own managed objects using the agreements on [GDMO] specified in clause 18.7. However, supporting managed objects from the MIL will increase the potential for interoperability with other network management implementations.

Editor's Note: [Following is proposed text for this section, to replace all the above text:

The Management Information Library 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 are registered in the NMSIG arc of the ISO ASN.1 Object Identifier Tree.

The reasons why the NMSIG has opted to define management information are as follows:

(i) There is an urgent need for network management within the community. Managed objects are critical ingredients of network management; but standards' defined managed objects that represent network/system resources are not available yet. However, there does exist an ISO standard that specifies guidelines for defining managed objects : [GDMO]. Different organizations, including private companies, etc, can use [GDMO] to define their own managed objects. However, two network management implementations can interoperate only if there is a common subset of managed objects supported on both sides. The NMSIG has used the [GDMO] standard to define "public domain" managed objects that meet the needs of the community and foster interoperability.

(ii) Standards' groups are not addressing all the network/system resources that need to be managed; i.e. there is no standards' activity for defining managed objects that represent such resources. The NMSIG has attempted to fill these holes by defining managed objects for these resources, and thus fulfil the needs of the community.

As mentioned earlier, managed objects in the MIL have been provided to foster interoperability. They are not normative as far the NMSIG IAs are concerned. Implementors do not have to support any of the MIL managed objects; they may choose to define their own managed objects using the agreements on [GDMO] specified in Section 18.7. However, supporting managed objects from the MIL will increase the potential for interoperability with other network management implementations.

The NMSIG defined managed objects in the MIL are intended to be implementable but they also serve as a basis from which other implementations may define refinements or alternatives. These definitions do not override or duplicate those provided by standards' groups or other OIW SIGs.

More specifically, the transport and network layer managed objects that have been defined in the MIL are "generally applicable" objects, in that they do not represent any particular transport or network layer protocols, but contain characteristics common across different transport or network layer protocols. These managed objects provide a high level view of the transport and network layers, and are especially useful in managing heterogeneous networks that support various different types of transport and network layer protocols. These managed objects do not override the OSI Transport and Network Layer managed objects that are being defined in ISO. The ISO specified OSI Transport and Network Layer managed objects are "specific" managed objects that represent strictly the OSI Transport and Network protocol layers.]

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 classes 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) An element of management information, once registered, i.e., given an ASN.1 Object Identifier, will never be deleted from the Registration Tree (ASN.1 Object Identifier tree). It may, however, fall into disuse due to lack of requirements for it.

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 MANAGED OBJECT CLASSES

A.4.1 NMSIG Agent

A.4.1.1 NMSIG Agent Definition

```
nmsig-agent    MANAGED OBJECT CLASS
  DERIVED FROM  ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :] top;
  CHARACTERIZED BY  nmsig-agent-Package;

REGISTERED AS  {nmsig-objectClass 1};
```

A.4.1.2 NMSIG Agent Package

```
nmsig-agent-Package   PACKAGE
          BEHAVIOUR DEFINITIONS   agent-behaviour;
    ATTRIBUTES   nmsig-agentId  GET;

REGISTERED AS {nmsig-package 1};
```

A.4.1.3 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.4.2 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)

```
                    computerSystem
                          |
                          |
                          |
 -------------------------------------------------------........
    |        |     |      |      |    |       |
    |        |     |      |      |    |       |
tapeDrive    |   printer  |      |    |   applicationX  ........
         discDrive     processing|   os
                        Entity   |
                                 |
                         coTransportProtocolLayerEntity
                                 |
                            transportConnection
```

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.4.2.1 NMSIG Computer System Definition

```
nmsig-computerSystem    MANAGED OBJECT CLASS
  DERIVED FROM  ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :] top;
  CHARACTERIZED BY  nmsig-computerSystem-Package;
REGISTERED AS  {nmsig-objectClass 2};
```

A.4.2.2 NMSIG Computer System Package

```
nmsig-computerSystem-Package   PACKAGE
        BEHAVIOUR DEFINITIONS  computerSystem-behaviour;
    ATTRIBUTES  nmsig-systemId  GET,
                  ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                administrativeState  GET-REPLACE,
                  ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                operationalState  GET,
                  ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                usageState  GET,
                  ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
```

```
            managementState  GET,
            nmsig-systemTime   GET,                              nmsig-peripheralNames  GET,
            nmsig-userFriendlyLabel  GET-REPLACE;
    NOTIFICATIONS   ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
            objectCreation,
              ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
            objectDeletion,
              ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
            attributeValueChange,
              ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
            stateChange,
              ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
            processingErrorAlarm,
              ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
            environmentalAlarm,
              ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
            equipmentAlarm;

REGISTERED AS {nmsig-package 2};
```

A.4.2.3 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 Package supports all the values of the administrative and operational states. The values supported by the usage state are implementation specific; the 'idle' and 'unknown' values are to be used for computer systems that do not keep track of their usage.

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 additionalCreateInfo field of the objectCreation notification shall contain all the attributes of the created computer sytem instance.

```
        The additionalDeleteInfo field of the objectDeletion notification shall be NULL.

        Attributes that are subject to the attributeValueChange notification are :  nmsig-peripheralNames,
        nmsig-userFriendlyLabel.
        Attributes that are subject to the stateChange notification are :
        administrativeState, operationalState and usageState.
    ;
```

A.4.3 NMSIG Connection Oriented Transport Protocol Layer Entity

A.4.3.1 NMSIG CO Transport Protocol Layer Entity Definition

```
nmsig-coTransportProtocolLayerEntity    MANAGED OBJECT CLASS

   DERIVED FROM       ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
 top;
   CHARACTERIZED BY  nmsig-coTransportProtocolLayerEntity-Package
                 nmsig-productInfo-Package;

REGISTERED AS {nmsig-objectClass 3};
```

A.4.3.2 NMSIG CO Transport Protocol Layer Entity Package

```
nmsig-coTansportProtocolLayerEntity-Package   PACKAGE
            BEHAVIOUR DEFINITIONS  coTransportProtocolLayerEntity-behaviour;
       ATTRIBUTES        nmsig-coTransportProtocolLayerEntityId  GET,
                 ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                      administrativeState  GET-REPLACE,
                 ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                      operationalState  GET,
                 ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                      usageState        GET,
                 ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                      managementState   GET,
                      nmsig-localTransportAddresses  GET,
                      nmsig-maxConnections  GET-REPLACE,
                      nmsig-openConnections  GET,
                 ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                      outgoingConnectionsRequestCounter  GET,
                 ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                      incomingConnectionsRequestCounter  GET,
                 ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                      outgoingConnectionRejectErrorCounter  GET,
                 ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                      incomingConnectionRejectErrorCounter  GET,
                 ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
```

```
                outgoingDisconnectErrorCounter  GET,
            ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                incomingDisconnectErrorCounter  GET,
            ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                incomingDisconnectCounter  GET,
            ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                outgoingDisconnectCounter  GET,
            ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                octetsSentCounter   GET,
            ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                octetsReceivedCounter  GET,
            ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                incomingProtocolErrorCounter  GET,
            ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                outgoingProtocolErrorCounter  GET,
                nmsig-checksumTPDUsDiscardedCounter  GET,
                nmsig-transportEntityType GET,
                nmsig-entityUpTime GET;
  NOTIFICATIONS
            ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                objectCreation,
            ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                objectDeletion,
            ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                attributeValueChange,
            ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                stateChange,
            ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                processingErrorAlarm,
                nmsig-counterWrap,

REGISTERED AS        {nmsig-package 3};
```

A.4.3.3 NMSIG CO Transport Protocol Layer Entity Behaviour

coTransportProtocolLayerEntity-behaviour BEHAVIOUR

DEFINED AS

This is a generally applicable managed object class, in that it does not represent any specific connection-oriented transport protocol - rather it contains characteristics common across various different connection-oriented transport layer protocols. This managed object class is not intended to override any transport layer managed object classes being defined in ISO. It provides a high level view of a connection-oriented transport layer protocol and complements the protocol-specific views being defined in the standards.

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.

The connection-oriented transport protocol layer entity Package 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 usage 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 usage 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 additionalCreateInfo field of the objectCreation notification shall contain all the attributes of the created connection-oriented transport protocol layer entity instance.

The additionalDeleteInfo 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 attributeValueChange notification are : nmsig-localTransportAddresses, nmsig-maxConnections.

Attributes that are subject to the stateChange notification are :
administrativeState, usageState and operationalState.

The counterWrap notification is emitted when any of the counter attributes wrap.

;

A.4.3.4 NMSIG Product Info Package

```
nmsig-productInfo-Package   PACKAGE
          BEHAVIOUR DEFINITIONS  productInfo-behaviour;
      ATTRIBUTES          nmsig-manufacturerInfo  GET,
                  nmsig-productLabel  GET,
                    nmsig-release       GET,
                    nmsig-serialNumber  GET;

REGISTERED AS       {nmsig-package 4};

productInfo-behaviour  BEHAVIOUR

  DEFINED AS

  This package specifies product information of the underlying resource.
  ;
```

A.4.4 NMSIG Connectionless Network Protocol Layer Entity

A.4.4.1 NMSIG Connectionless Network Protocol Layer Entity Definition

```
nmsig-clNetworkProtocolLayerEntity    MANAGED OBJECT CLASS

   DERIVED FROM    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :] top;
   CHARACTERIZED BY  nmsig-clNetworkProtocolLayerEntity-Package,
                nmsig-productInfo-Package;
   CONDITIONAL PACKAGES
             nmsig-clNetworkProtocolLayerEntityRedirection-Package
                 PRESENT IF connectionless network protocol layer entity supports redirection
                      of received PDUS;

REGISTERED AS  {nmsig-objectClass 4}
```

A.4.4.2 NMSIG Connectionless Network Protocol Layer Entity Package

```
nmsig-clNetworkProtocolLayerEntity-Package   PACKAGE
       BEHAVIOUR DEFINITIONS  clNetworkProtocolLayerEntity-behaviour;
    ATTRIBUTES          nmsig-clNetworkProtocolLayerEntityId  GET,
                 ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                    administrativeState  GET-REPLACE,
                 ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                    operationalState  GET,
                 ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                    usageState GET,
                 ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                    managementState  GET,
                    nmsig-localNetworkAddresses  GET,
                    nmsig-nPDUTimeToLive  GET-REPLACE,
                    nmsig-maxPDUSize   GET,
                 ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                    pDUsSentCounter  GET,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                        pDUsReceivedCounter  GET,
                        nmsig-PDUsForwardedCounter  GET,
                        nmsig-PDUsReasmbldOKCounter  GET,
                        nmsig-PDUsReasmblFailCounter  GET,
                        nmsig-PDUsDiscardedCounter   GET,
                        nmsig-networkEntityType GET,
                        nmsig-entityUpTime GET;

        NOTIFICATIONS
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                        objectCreation,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                        objectDeletion,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                        attributeValueChange,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                        processingErrorAlarm,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                        stateChange,
                        nmsig-counterWrap;

REGISTERED AS       {nmsig-package 5};
```

A.4.4.3 NMSIG Connectionless Network Protocol Layer Entity Behaviour

clNetworkProtocolLayerEntity-behaviour BEHAVIOUR

DEFINED AS

This is a generally applicable managed object class, in that it does not represent any specific connectionless network protocol -rather it contains characteristics common across various different connectionless network layer protocols. This managed object class is not intended to override any network layer managed object classes being defined in ISO. It provides a high level view of a connectionless network layer protocol and complements the protocol-specific views being defined in the standards.

This managed object class represents an instantiation of a connectionless network protocol layer. The network protocol layer 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.

The NMSIG connectionless network protocol layer entity Package supports all the values of the administrative and operational state attributes. The values supported by the usage state are implementation specific; the 'idle' and 'unknown' values are to be used for connectionless network protocol layer entities that do not keep track of their usage.

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 additionalCreateInfo field of the objectCreation notification shall contain all the attributes of the created connectionless network protocol layer entity instance.

The additionalDeleteInfo 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 attributeValueChange notification are : nmsig-localNetworkAddresses, nmsig-nPDUTimeToLive.

Attributes that are subject to the stateChange notification are :
administrativeState, usageState and operationalState.

The counterWrap notification is emitted when any of the counter attributes wrap.

;

A.4.4.4 NMSIG CL Network Protocol Layer Entity Redirection Package

```
nmsig-clNetworkProtocolLayerEntityRedirection-Package  PACKAGE
      BEHAVIOUR DEFINITIONS  clNetworkProtocolLayerEntityRedirection-behaviour;
      ATTRIBUTES  nmsig-PDUsRedirected  GET;

REGISTERED AS  {nmsig-package 6};

clNetworkProtocolLayerEntityRedirection-behaviour  BEHAVIOUR

   DEFINED AS
```

This package reflects the redirection capability of the underlying connectionless network protocol layer entity resource.

;

A.4.5 NMSIG Equipment

A.4.5.1 NMSIG Equipment Definition

```
nmsig-equipment   MANAGED OBJECT CLASS
   DERIVED FROM   ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :] top;
```

```
  CHARACTERIZED BY  nmsig-equipment-Package,
                    productInfo-Package;
REGISTERED AS  {nmsig-objectClass 5};
```

A.4.5.2 NMSIG Equipment Package

```
nmsig-equipment-Package  PACKAGE
    BEHAVIOUR DEFINITIONS  equipment-behaviour;
    ATTRIBUTES    nmsig-equipmentId  GET,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                  operationalState  GET,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                  administrativeState  GET-REPLACE,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                  usageState  GET,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                  managementState  GET,
                  nmsig-locationName  GET-REPLACE,
                  nmsig-contactNames  ADD-REMOVE, GET-REPLACE,
                  nmsig-equipmentPurpose      GET-REPLACE,
                  nmsig-vendorName    GET-REPLACE,
                  nmsig-userFriendlyLabel  GET-REPLACE;

  NOTIFICATIONS
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                  environmentalAlarm,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                  equipmentAlarm,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                  objectCreation,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                  objectDeletion,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                  attributeValueChange,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                  stateChange;

REGISTERED AS  {nmsig-package 7};
```

A.4.5.3 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.

Other organizations such as the OSI NM Forum and T1M1.5 have specified definitions for an equipment managed object class. The NMSIG has not adopted those definitions because a) they are not specified in the NMSIG supported version of [GDMO], and b) they do not use generic management information from [DMI].

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 Package supports all permissible values of the administrative, usage and operational states.

The additionalCreateInfo field of the objectCreation notification shall contain all the attributes of the created equipment instance.

The additionalDeleteInfo field of the objectDeletion notification shall contain all the attributes of the deleted equipment instance.

Attributes that are subject to the attributeValueChange notification are : nmsig-locationName, nmsig-contactNames,
nmsig-equipmentPurpose, nmsig-vendorName, nmsig-userFriendlyLabel.

Attributes that are subject to the stateChange notification are :
administrativeState, usageState and operationalState.

;

A.4.6 NMSIG Network

A.4.6.1 NMSIG Network Definition

```
nmsig-network    MANAGED OBJECT CLASS

   DERIVED FROM  ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :] top;
      CHARACTERIZED BY  nmsig-network-Package;
```

```
REGISTERED AS {nmsig-objectClass 6};
```

A.4.6.2 NMSIG Network Package

```
nmsig-network-Package   PACKAGE
      BEHAVIOUR DEFINITIONS  network-behaviour;
      ATTRIBUTES   nmsig-networkId   GET,
                 nmsig-networkPurpose  GET,
                 nmsig-userFriendlyLabel  GET-REPLACE;

      NOTIFICATIONS
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]                    objectCreation,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                   objectDeletion,
                    ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                   attributeValueChange;

REGISTERED AS  {nmsig-package 8};
```

A.4.6.3 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.

Other organizations such as the OSI NM Forum and T1M1.5 have specified definitions for a network managed object class. The NMSIG has not adopted those definitions because a) they are not specified in the NMSIG supported version of [GDMO], and b) they do not use generic management information from [DMI].

The additionalCreateInfo field of the objectCreation notification shall contain all the attributes of the created network instance.

The additionalDeleteInfo field of the objectDeletion notification shall contain all the attributes of the deleted network instance.

Attributes that are subject to the attributeValueChange notification are : nmsig-networkPurpose, nmsig-userFriendlyLabel.

;

A.4.7 NMSIG Processing Entity

A.4.7.1 NMSIG Processing Entity Definition

```
nmsig-processingEntity   MANAGED OBJECT CLASS
    DERIVED FROM   {nmsig-equipment};
       CHARACTERIZED BY  nmsig-processingEntity-Package;
REGISTERED AS {nmsig-objectClass };
```

A.4.7.2 NMSIG Processing Entity Package

```
nmsig-processingEntity-Package   PACKAGE
       BEHAVIOUR DEFINITIONS  processingEntity-behaviour;
       ATTRIBUTES      nmsig-cPU-Type  GET,
                    nmsig-memorySize  GET,
                    nmsig-osInfo   GET,
                    nmsig-entityUpTime  GET;
       NOTIFICATIONS
                   ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                    processingErrorAlarm;

REGISTERED AS  {nmsig-package 9};
```

A.4.7.3 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 additionalCreateInfo field of the objectCreation notification shall contain all the attributes of the created processing entity instance.

The additionalDeleteInfo 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.4.8 NMSIG Transport Connection

A.4.8.1 NMSIG Transport Connection Definition

```
nmsig-transportConnection          MANAGED OBJECT CLASS
        DERIVED FROM  ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :] top;
        CHARACTERIZED BY  nmsig-transportConnection-Package;
        CONDITIONAL PACKAGES
                     nmsig-transportConnectionRetransmission-Package  PRESENT IF  transport protocol supports
                             retransmission;
REGISTERED AS {nmsig-objectClass 8};
```

A.4.8.2 NMSIG Transport Connection Package

```
nmsig-transportConnection-Package   PACKAGE
        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,
                        ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                             pdusSentCounter  GET,
                        ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                             pdusReceivedCounter  GET,
                        ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                             octetsSentCounter  GET,
                        ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                             octetsReceivedCounter  GET,
                        ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                                   peer GET
        NOTIFICATIONS
                        ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                             objectCreation,
                        ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                             objectDeletion,
                        ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                             relationshipChange,
                             nmsig-counterWrap;
```

REGISTERED AS {nmsig-package 10};

A.4.8.3 NMSIG Transport Connection Behaviour

```
transportConnection-behaviour  BEHAVIOUR
  DEFINED AS
```

This is a generally applicable managed object class, in that it does not represent any specific connection-oriented transport protocol; rather it contains characteristics common across various different connection-oriented transport layer protocols. This managed object class is not intended to override any transport layer managed object classes being defined in ISO. It provides a high level view of a connection-oriented transport layer protocol and complements the protocol-specific views being defined in the standards.

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.

The additionalCreateInfo field of the objectCreation notification shall contain all the attributes of the created transport connection instance.

The additionalDeleteInfo field of the objectDeletion notification shall contain all the attributes of the deleted transport connection instance. In addition it shall also contain a 'cause' field and a corresponding 'cause' attribute whose syntax is defined as follows:

```
  Cause  ::=  SEQUENCE {
                INTEGER ( unknown (0),
                          user    (1),
                          provider (2) ),
                INTEGER ( unknown (0),
                          excessiveIdle (1),
                          excessiveRtx  (2) )
                   }
```

[The full definition of this attribute is specified in the Attributes Section - Section 6 of the MIL.]

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.4.8.4 NMSIG Transport Connection Retransmission Package

```
nmsig-transportConnectionRetransmission  PACKAGE
      BEHAVIOUR DEFINITIONS  transportConnectionRetransmission-behaviour;
      ATTRIBUTES  nmsig-maxRetransmissions  GET,
                  nmsig-retransmissionTimerInitialValue GET,
                ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
              pdusRetransmittedErrorCounter  GET,
                ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
              pdusRetransmittedErrorThreshold  GET-REPLACE,
              nmsig-octetsRetransmittedErrorCounter  GET;

     NOTIFICATIONS
                ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                  attributeValueChange,
                ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
                  communicationAlarm;

REGISTERED AS  {nmsig-package 11};

transportConnectionRetransmission-behaviour  BEHAVIOUR

  DEFINED AS

        This package reflects the retransmitting capability of the underlying transport protocol resource.

        Attributes that are subject to the attributeValueChange notification are: pdusRetransmittedErrorThreshold.
    ;
```

A.4.9 NMSIG Transport Connection Profile

A.4.9.1 NMSIG Transport Connection Profile Definition

```
nmsig-transportConnectionProfile   MANAGED OBJECT CLASS
  DERIVED FROM  ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :] top;
    CHARACTERIZED BY nmsig-transportConnectionProfile-Package;
REGISTERED AS {nmsig-objectClass 9};
```

A.4.9.2 NMSIG Transport Connection Profile Package

```
nmsig-transportConnectionProfile-Package   PACKAGE
    BEHAVIOUR DEFINITIONS  transportConnectionProfile-behaviour;
      ATTRIBUTES   nmsig-transportConnectionProfileId  GET,
                nmsig-inactivityTimeout  GET-REPLACE,
              nmsig-maxTPDuSize  GET-REPLACE;
    NOTIFICATIONS
                ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
              objectCreation,
                ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
              objectDeletion,
                ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :]
              attributeValueChange;
REGISTERED AS  {nmsig-package 12};
```

A.4.9.3 NMSIG Transport Connection Profile Behaviour

```
transportConnectionProfile-behaviour  BEHAVIOUR

  DEFINED AS
```

This managed object class is an IVMO (Initial Value Managed Object class). It 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 additionalCreateInfo field of the ObjectCreation notification shall contain all the attributes of the created transport connection profile instance.

The additionalDeleteInfo field of the ObjectDeletion notification shall contain all the attributes of the deleted transport connection profile instance.

Attributes that are subject to the AttributeValueChange notification are : nmsig-inactivityTimeout, nmsig-maxTPDuSize.

;

A.4.10 NMSIG Transport Connection Retransmission Profile

A.4.10.1 NMSIG Transport Connection Retransmission Profile Definition

```
nmsig-transportConnectionRetransmissionProfile   MANAGED OBJECT CLASS

  DERIVED FROM   nmsig-transportConnectionProfile;
     CHARACTERIZED BY   nmsig-transportConnectionProfile-Package;

REGISTERED AS  {nmsig-objectClass 10};
```

A.4.10.2 NMSIG Transport Connection Retransmission Profile Package

```
nmsig-transportConnectionRetransmissionProfile-Package  PACKAGE
      BEHAVIOUR DEFINITIONS  transportConnectionProfile-behaviour;
         ATTRIBUTES  nmsig-maxRetransmissions  GET-REPLACE,
                 nmsig-retransmissionTimerInitialValue GET-REPLACE;

REGISTERED AS  {nmsig-package 13};
```

A.4.10.3 NMSIG Transport Connection Retransmission Profile Behaviour

```
transportConnectionRetransmissionProfile-behaviour  BEHAVIOUR
```

DEFINED AS

This managed object class is an IVMO (Initial Value Managed Object class). It 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 AttributeValueChange notification are : nmsig-maxRetransmissions, nmsig-retransmissionTimerInitialValue.

;

A.5 NAME BINDINGS

This section provides definitions of NAME BINDINGS for the managed object classes defined by the NMSIG.

'Root' is a fictitious object class that represents the root of the containment tree. A name binding with 'root' as the superior object class means that the object class specified as the subordinate object class is effectively the top of the containment subtree within the context of the management entity that supports this name binding.

A.5.1 NMSIG Agent Name Bindings

```
agent-root   NAME BINDING
  SUBORDINATE OBJECT CLASS  nmsig-agent;
    NAMED BY
  SUPERIOR OBJECT CLASS   root;
  WITH ATTRIBUTE nmsig-agentId;
REGISTERED AS  {nmsig-nameBindings 1};
```

A.5.2 NMSIG Computer System Name Bindings

```
computerSystem-network  NAME BINDING
  SUBORDINATE OBJECT CLASS  nmsig-computerSystem;
    NAMED BY
    SUPERIOR OBJECT CLASS nmsig-network;
  WITH ATTRIBUTE nmsig-systemId;
REGISTERED AS  {nmsig-nameBindings 2};

computerSystem-computerSystem  NAME BINDING
  SUBORDINATE OBJECT CLASS  nmsig-computerSystem;
    NAMED BY
    SUPERIOR OBJECT CLASS nmsig-computerSystem;
  WITH ATTRIBUTE nmsig-systemId;
REGISTERED AS  {nmsig-nameBindings 3};

computerSystem-root  NAME BINDING
  SUBORDINATE OBJECT CLASS  nmsig-computerSystem;
    NAMED BY
    SUPERIOR OBJECT CLASS root;
  WITH ATTRIBUTE nmsig-systemId;
REGISTERED AS  {nmsig-nameBindings 4};
```

A.5.3 NMSIG CO Transport Protocol Layer Entity Name Bindings

```
coTransportProtocolLayerEntity-computerSystem  NAME BINDING
  SUBORDINATE OBJECT CLASS  nmsig-coTransportProtocolLayerEntity;
     NAMED BY
     SUPERIOR OBJECT CLASS nmsig-computerSystem;
  WITH ATTRIBUTE nmsig-coTransportEntityId;
REGISTERED AS  {nmsig-nameBindings 5};

coTransportProtocolLayerEntity-system  NAME BINDING
  SUBORDINATE OBJECT CLASS  nmsig-coTransportProtocolLayerEntity;
     NAMED BY
     SUPERIOR OBJECT CLASS
       ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :] system;
  WITH ATTRIBUTE nmsig-coTransportEntityId;
REGISTERED AS  {nmsig-nameBindings 6};

coTransportProtocolLayerEntity-equipment  NAME BINDING
  SUBORDINATE OBJECT CLASS  nmsig-coTransportProtocolLayerEntity;
     NAMED BY
     SUPERIOR OBJECT CLASS nmsig-equipment;
  WITH ATTRIBUTE nmsig-coTransportEntityId;
REGISTERED AS  {nmsig-nameBindings 7};
```

A.5.4 NMSIG CL Network Protocol Layer Entity Name Bindings

```
clNetworkProtocolLayerEntity-computerSystem  NAME BINDING
  SUBORDINATE OBJECT CLASS nmsig-clNetworkProtocolLayerEntity;
     NAMED BY
     SUPERIOR OBJECT CLASS nmsig-computerSystem;
  WITH ATTRIBUTE nmsig-clNetworkEntityId;
REGISTERED AS  {nmsig-nameBindings 8};

clNetworkProtocolLayerEntity-system  NAME BINDING
  SUBORDINATE OBJECT CLASS nmsig-clNetworkProtocolLayerEntity;
     NAMED BY
     SUPERIOR OBJECT CLASS
       ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :] system;
  WITH ATTRIBUTE nmsig-clNetworkEntityId;
REGISTERED AS  {nmsig-nameBindings 9};

clNetworkPrctocoiLayerEntity-equipment  NAME BINDING
  SUBORDINATE OBJECT CLASS nmsig-clNetworkProtocolLayerEntity;
     NAMED BY
     SUPERIOR OBJECT CLASS nmsig-equipment;
  WITH ATTRIBUTE nmsig-clNetworkEntityId;
REGISTERED AS  {nmsig-nameBindings 10};
```

A.5.5 NMSIG Equipment Name Bindings

```
equipment-equipment  NAME BINDING
  SUBORDINATE OBJECT CLASS nmsig-equipment;
     NAMED BY
     SUPERIOR OBJECT CLASS nmsig-equipment;
  WITH ATTRIBUTE nmsig-equipmentId;
REGISTERED AS  {nmsig-nameBindings 11};

equipment-network  NAME BINDING
  SUBORDINATE OBJECT CLASS nmsig-equipment;
     NAMED BY
     SUPERIOR OBJECT CLASS nmsig-network;
  WITH ATTRIBUTE nmsig-equipmentId;
REGISTERED AS  {nmsig-nameBindings 12};

equipment-root  NAME BINDING
  SUBORDINATE OBJECT CLASS nmsig-equipment;
     NAMED BY
     SUPERIOR OBJECT CLASS root;
  WITH ATTRIBUTE nmsig-equipmentId;
REGISTERED AS  {nmsig-nameBindings 13};
```

A.5.6 NMSIG Network Name Bindings

```
network-network  NAME BINDING
  SUBORDINATE OBJECT CLASS nmsig-network;
     NAMED BY
     SUPERIOR OBJECT CLASS nmsig-network;
  WITH ATTRIBUTE nmsig-networkId;
REGISTERED AS  {nmsig-nameBindings 14};

network-root  NAME BINDING
  SUBORDINATE OBJECT CLASS nmsig-network;
     NAMED BY
     SUPERIOR OBJECT CLASS root;
  WITH ATTRIBUTE nmsig-networkId;
REGISTERED AS  {nmsig-nameBindings 15};
```

A.5.7 NMSIG Processing Entity Name Bindings

```
processingEntity-computerSystem  NAME BINDING
  SUBORDINATE OBJECT CLASS nmsig-processingEntity;
     NAMED BY
     SUPERIOR OBJECT CLASS nmsig-computerSystem;
  WITH ATTRIBUTE nmsig-equipmentId;
  DELETE  deletes-contained-objects;
REGISTERED AS  {nmsig-nameBindings 16};
```

A.5.8 NMSIG Transport Connection Name Bindings

```
transportConnection-coTransportProtocolLayerEntity  NAME BINDING
  SUBORDINATE OBJECT CLASS nmsig-transportConnection;
    NAMED BY
    SUPERIOR OBJECT CLASS nmsig-coTransportProtocolLayerEntity;
  WITH ATTRIBUTE nmsig-transportConnectionId;
  BEHAVIOUR  transportConnection-nb-behaviour  DEFINED AS
          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.
    ;
  DELETE  deletes contained objects;
REGISTERED AS  {nmsig-nameBindings 17};
```

A.5.9 NMSIG Transport Connection Profile Name Bindings

```
transportConnectionProfile-coTransportProtocolLayerEntity  NAME BINDING
  SUBORDINATE OBJECT CLASS nmsig-transportConnectionProfile;
    NAMED BY
    SUPERIOR OBJECT CLASS nmsig-coTransportProtocolLayerEntity;
  WITH ATTRIBUTE nmsig-transportConnectionProfileId;
REGISTERED AS  {nmsig-nameBindings 18};
```

A.5.10 NMSIG Transport Connection Retransmission Profile Name Bindings

```
transportConnectionRetransmissionProfile-coTransportProtocolLayerEntity NAME BINDING
 SUBORDINATE OBJECT CLASS nmsig-transportConnectionRetransmissionProfile;
 NAMED BY
 SUPERIOR OBJECT CLASS nmsig-coTransportProtocolLayerEntity;
  WITH ATTRIBUTE nmsig-transportConnectionProfileId;
REGISTERED AS  {nmsig-nameBindings 19};
```

A.6 ATTRIBUTES

This section provides definitions of attributes contained in the managed object classes specified in this document.

All attribute syntaxes have been defined with external type references. External type references take the form:

ModuleName.NamedType

where the ModuleName refers to the name of an ASN.1 module and NamedType referes to a defined type in that module. The ASN.1 module referenced in this section is the NMSIG-SYNTAX-1 module found in Section A.10 of this Annex.

A.6.1 NMSIG Agent Id

```
nmsig-agentId  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX NMSIG-SYNTAX-1.Id   ;
    BEHAVIOUR  agentId-behaviour;
REGISTERED AS     {nmsig-attribute 1} ;

agentId-behaviour  BEHAVIOUR
 DEFINED AS
     This is the distinguishing attribute for the managed object class NMSIG Agent.
 ;
```

A.6.2 NMSIG Cause

```
nmsig-cause   ATTRIBUTE
         WITH ATTRIBUTE SYNTAX  NMSIG-SYNTAX-1.Cause;
    MATCHES FOR  Equality;
    BEHAVIOUR  cause-behaviour;

   REGISTERED AS     {nmsig-attribute 2} ;

 cause-behaviour   BEHAVIOUR
DEFINED AS
     This attribute specifies the reason why a transport connection was deleted.  It is included in the
     additionalDeleteInfo field of the objectDeletion notification.
 ;
```

A.6.3 NMSIG Checksum TPDUs Discarded Counter

```
nmsig-checksumTPDUsDiscardedCounter   ATTRIBUTE
```

```
          DERIVED FROM
          ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :] counter;
         BEHAVIOUR  checksumTPDUsDiscardedCounter-behaviour;

    REGISTERED AS      {nmsig-attribute 3} ;

  checksumTPDUsDiscardedCounter-behaviour   BEHAVIOUR
  DEFINED AS
       This attribute specifies the number of TPDUs discarded due to a bad checksum.
  ;
```

A.6.4 NMSIG CO Transport Protocol Layer Entity Id

```
  nmsig-coTransportProtocolLayerEntityId  ATTRIBUTE
      WITH ATTRIBUTE SYNTAX  NMSIG-SYNTAX-1.Id   ;
      MATCHES FOR  Equality;
          BEHAVIOUR  coTransportProtocolLayerEntityId-behaviour;

  REGISTERED AS      {nmsig-attribute 4} ;

  coTransportProtocolLayerEntityID-behaviour  BEHAVIOUR
  DEFINED AS
       This is the distinguishing attribute for the managed object class connection oriented transport protocol layer
       entity.;
```

A.6.5 NMSIG Connectionless Network Protocol Layer Entity Id

```
  nmsig-clNetworkProtocolLayerEntityId  ATTRIBUTE
      WITH ATTRIBUTE SYNTAX  NMSIG-SYNTAX-1.Id ;
      MATCHES FOR  Equality ;
          BEHAVIOUR  clNetworkProtocolLayerEntityId-behaviour;

  REGISTERED AS       {nmsig-attribute 5};

  clNetworkProtocolLayerEntityId-behaviour   BEHAVIOUR
  DEFINED AS
         This attribute is the distinguishing attribute for the managed object class clNetworkProtocolLayerEntity.;
```

A.6.6 NMSIG Contact Names

```
  nmsig-contactNames   ATTRIBUTE
      WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.AnyName;
      MATCHES FOR  Set Comparison, Set Intersection;
            BEHAVIOUR  contactNames-behaviour   ;
```

```
REGISTERED AS      {nmsig-attribute 6};

contactNames-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies name(s) of one or more contacts.
        ;
```

A.6.7 NMSIG CPU Type

```
 nmsig-cPU-Type   ATTRIBUTE
      WITH ATTRIBUTE SYNTAX  NMSIG-SYNTAX-1.CPU-Type;
  MATCHES FOR  Equality     ;
         BEHAVIOUR  cPU-Type-behaviour;

 REGISTERED AS    {nmsig-attribute 7};

   cPU-Type-behaviour  BEHAVIOUR
DEFINED AS
        This attribute specifies the type of the Central Processor Unit in a processing entity.;
```

A.6.8 NMSIG Entity Up Time

```
 nmsig-entityUpTime    ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.EntityUpTime;
    MATCHES FOR   Equality, Ordering ;
           BEHAVIOUR  entityUpTime-behaviour;

REGISTERED AS      {nmsig-attribute 8} ;

   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.6.9 NMSIG Equipment Id

```
nmsig-equipmentId   ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Id;
    MATCHES FOR  Equality ;
           BEHAVIOUR  equipmentId-behaviour;

REGISTERED AS      {nmsig-attribute 9} ;
```

```
   equipmentId-behaviour  BEHAVIOUR
DEFINED AS
        This is the distinguishing attribute of the NMSIG equipment managed object class.;
```

A.6.10 NMSIG Equipment Purpose

```
nmsig-equipmentPurpose  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX  NMSIG-SYNTAX-1.EquipmentPurpose;
    MATCHES FOR  Equality ;
            BEHAVIOUR  equipmentPurpose-behaviour;

REGISTERED AS       {nmsig-attribute 10};

   equipmentPurpose-behaviour  BEHAVIOUR
DEFINED AS
        This attribute specifies what the equipment is used for ( e.g. switching, processing, etc.).
  ;
```

A.6.11 NMSIG Inactivity Timeout

```
nmsig-inactivityTimeout           ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.InactivityTimeout;
    MATCHES FOR   Equality, Ordering ;
            BEHAVIOUR  inactivityTimeout-behaviour;

REGISTERED AS       {nmsig-attribute 11};

   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.6.12 NMSIG Local Network Address

```
nmsig-localNetworkAddress       ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Address;
    MATCHES FOR   Equality;
            BEHAVIOUR  localNetworkAddress-behaviour;

REGISTERED AS       {nmsig-attribute 12};

   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.6.13 NMSIG Local Network Addresses

```
nmsig-localNetworkAddresses       ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.LocalNetworkAddresses;
    MATCHES FOR  Set Comparison, Set Intersection;
            BEHAVIOUR  localNetworkAddresses-behaviour;

REGISTERED AS      {nmsig-attribute 13};

    localNetworkAddresses-behaviour   BEHAVIOUR
DEFINED AS
```

This attribute specifies a set of local network addresses supported by a network protocol layer entity.;

A.6.14 NMSIG Local Transport Addresses

```
nmsig-localTransportAddresses     ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.TransportAddresses;
    MATCHES FOR  Set Comparison, Set Intersection;
            BEHAVIOUR  localTransportAddresses-behaviour;

REGISTERED AS  {nmsig-attribute 14};

    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.6.15 NMSIG Local Transport Connection Endpoint

```
nmsig-localTransportConnectionEndpoint      ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Address;
    MATCHES FOR  Equality ;
            BEHAVIOUR  localTransportConnectionEndpoint-behaviour;

REGISTERED AS      {nmsig-attribute 15};

    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.6.16 NMSIG Location Name

```
nmsig-locationName   ATTRIBUTE
    WITH ATTRIBUTE SYNTAX    NMSIG-SYNTAX-1.AnyName;
    MATCHES FOR  Equality ;
            BEHAVIOUR  locationName-behaviour;

REGISTERED AS     {nmsig-attribute 16};

    locationName-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies the name of a location (e.g. Hilo Hawaii USA);
```

A.6.17 NMSIG Manufacturer Info

```
nmsig-manufacturerInfo  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX  NMSIG-SYNTAX-1.ManufacturerInfo;
    MATCHES FOR  Equality ;
            BEHAVIOUR  manufacturerInfo-behaviour;

REGISTERED AS      {nmsig-attribute 17};

    manufacturerInfo-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies information about the manufacturer of the product that has implemented the
        underlying resource.;
```

A.6.18 NMSIG Max Connections

```
nmsig-maxConnections          ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.MaxNumber;
    MATCHES FOR   Equality, Ordering;
            BEHAVIOUR  maxConnections-behaviour;

REGISTERED AS     {nmsig-attribute 18} ;

    maxConnections-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies the maximum number of simultaneously open transport connections allowed by
        the transport protocol layer entity.;
```

A.6.19 NMSIG Max PDU Size

```
nmsig-maxPDUSize          ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Length;
    MATCHES FOR   Equality, Ordering;
            BEHAVIOUR  maxPDUSize-behaviour;

REGISTERED AS      {nmsig-attribute 19} ;

    maxPDUSize-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies the maximum length of a PDU that can be supported by the underlying resource
  ;
```

A.6.20 NMSIG Max Retransmissions

```
nmsig-maxRetransmissions  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.MaxNumber;
    MATCHES FOR   Equality, Ordering ;
            BEHAVIOUR  maxRetransmissions-behaviour;

REGISTERED AS      {nmsig-attribute 20} ;

    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.6.21 NMSIG Memory Size

```
     nmsig-memorySize   ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Amount;
    MATCHES FOR   Equality, Ordering ;
            BEHAVIOUR  memorySize-behaviour;

 REGISTERED AS   {nmsig-attribute 21};

    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.6.22 NMSIG Network Entity Type

```
     nmsig-networkEntityType ATTRIBUTE
  WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.NetworkEntityType;
```

```
        MATCHES FOR  Equality          ;
                BEHAVIOUR  networkEntityType-behaviour;

      REGISTERED AS      {nmsig-attribute 22};

           networkEntityType-behaviour   BEHAVIOUR
      DEFINED AS
                This attribute specifies the type of the network protocol layer  entity.;
```

A.6.23 NMSIG Network Id

```
     nmsig-networkId   ATTRIBUTE
          WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Id;
          MATCHES FOR  Equality ;
                   BEHAVIOUR  networkId-behaviour;

     REGISTERED AS      {nmsig-attribute 23} ;

          networkId-behaviour   BEHAVIOUR
     DEFINED AS
               This is the distinguishing attribute of the NMSIG network managed object class.;
```

A.6.24 NMSIG Network Purpose

```
     nmsig-networkPurpose   ATTRIBUTE
          WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.NetworkPurpose;
          MATCHES FOR  Equality ;
                   BEHAVIOUR  networkPurpose-behaviour;

     REGISTERED AS      {nmsig-attribute 24} ;

          networkPurpose-behaviour   BEHAVIOUR
     DEFINED AS
               This attribute specifies what the network is used for ( e.g. manufacturing control,  airline reservation, etc.
               )
          ;
```

A.6.25 NMSIG NPDU Time To Live

```
     nmsig-nPDUTimeToLive          ATTRIBUTE
          WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.MaxNumber;
          MATCHES FOR   Equality, Ordering;
                   BEHAVIOUR  nPDUTimeToLive-behaviour;

     REGISTERED AS       {nmsig-attribute 25};
```

```
    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.6.26 NMSIG Octets Retransmitted Error Counter

```
nmsig-octetsRetransmittedErrorCounter   ATTRIBUTE
          DERIVED FROM ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :] counter;
          BEHAVIOUR  octetsRetransmittedErrorCounter-behaviour;

REGISTERED AS     {nmsig-attribute 26} ;

    octetsRetransmittedErrorCounter-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies the total number of octets that were retransmitted.;
```

A.6.27 NMSIG OS Info

```
 nmsig-osInfo   ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.OsInfo;
    MATCHES FOR Set Comparison, Set Intersection;
          BEHAVIOUR  osInfo-behaviour;

 REGISTERED AS   {nmsig-attribute 27};

    osInfo-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies the names and releases of operating systems supported by the processing entity;
```

A.6.28 NMSIG Open Connections

```
nmsig-openConnections          ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Number;
    MATCHES FOR   Equality, Ordering ;
          BEHAVIOUR  openConnections-behaviour;

REGISTERED AS     {nmsig-attribute 28} ;

    openConnections-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies the number of currently established transport connections.;
```

A.6.29 NMSIG PDUs Discarded Counter

```
nmsig-PDUsDiscardedCounter          ATTRIBUTE
        DERIVED FROM ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :] counter;
         BEHAVIOUR  pDUsDiscardedCounter-behaviour;

REGISTERED AS     {nmsig-attribute 29} ;

   pDUsDiscardedCounter-behaviour   BEHAVIOUR
DEFINED AS
      This attribute specifies the number of PDUs that were discarded by a network protocol layer entity.;
```

A.6.30 NMSIG PDUs Forwarded Counter

```
nmsig-PDUsForwardedCounter          ATTRIBUTE
        DERIVED FROM ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :] counter;
         BEHAVIOUR  pDUsForwardedCounter-behaviour;

REGISTERED AS     {nmsig-attribute 30} ;

   pDUsForwardedCounter-behaviour   BEHAVIOUR
DEFINED AS
      This attribute specifies the number of PDUs forwarded by a network protocol layer entity.;
```

A.6.31 NMSIG PDUs Reassemble Fail Counter

```
nmsig-PDUsReasmblFailCounter          ATTRIBUTE
        DERIVED FROM ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :] counter;
         BEHAVIOUR  pDUsReasmblFailCounter-behaviour;

REGISTERED AS     {nmsig-attribute 31} ;

   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.6.32 NMSIG PDUs Reassembled OK Counter

```
nmsig-PDUsReasmbldOKCounter          ATTRIBUTE
        DERIVED FROM ["Recommendation X.721"|"ISO/IEC DIS 10165-2" :] counter;
         BEHAVIOUR  pDUsReasmbldOKCounter-behaviour;

REGISTERED AS     {nmsig-attribute 32} ;
```

```
    pDUsReasmbldOKCounter-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies the number of PDUs that were reassembled successfully by a network protocol
        layer entity.
        ;
```

A.6.33 NMSIG Peripheral Names

```
nmsig-peripheralNames  ATTRIBUTE
   WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.PeripheralNames;
   MATCHES FOR Set Comparison, Set Intersection  ;
         BEHAVIOUR  peripheralNames-behaviour;

REGISTERED AS     {nmsig-attribute 33};

    peripheralNames-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies the names of auxiliary devices.;
```

A.6.34 NMSIG Product Label

```
nmsig-productLabel  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.ProductLabel;
    MATCHES FOR  Equality ;
           BEHAVIOUR  productLabel-behaviour;

REGISTERED AS      {nmsig-attribute 34};

    productLabel-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies the product label of the product that has implemented the underlying resource.;
```

A.6.35 NMSIG Release

```
nmsig-release  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Release;
    MATCHES FOR  Equality ;
           BEHAVIOUR  release-behaviour;

REGISTERED AS      {nmsig-attribute 35};

    release-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies the release number of the product that has implemented the underlying resource.;
```

A.6.36 NMSIG Remote Network Address

```
nmsig-remoteNetworkAddress        ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Address;
    MATCHES FOR  Equality ;
            BEHAVIOUR  remoteNetworkAddress-behaviour;

REGISTERED AS      {nmsig-attribute 36};

    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.6.37 NMSIG Remote Transport Connection Endpoint

```
nmsig-remoteTransportConnectionEndpoint      ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Address;
    MATCHES FOR  Equality ;
            BEHAVIOUR  remoteTransportConnectionEndpoint-behaviour;

REGISTERED AS      {nmsig-attribute 37};

    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.6.38 NMSIG Retransmission Timer Initial Value

```
nmsig-retransmissionTimerInitialValue  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Number;
    MATCHES FOR   Equality, Ordering;
            BEHAVIOUR  retransmissionTimerInitialValue-behaviour;

REGISTERED AS     {nmsig-attribute 38} ;

    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.6.39 NMSIG Serial Number

```
nmsig-serialNumber  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.SerialNumber;
    MATCHES FOR  Equality ;
            BEHAVIOUR  serialNumber-behaviour;

REGISTERED AS      {nmsig-attribute 39};

    serialNumber-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies the serial number of the product that has implemented the underlying resource.;
```

A.6.40 NMSIG System Id

```
nmsig-systemId  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Id;
    MATCHES FOR  Equality ;
            BEHAVIOUR  systemId-behaviour;

REGISTERED AS      {nmsig-attribute 40};

    systemId-behaviour   BEHAVIOUR
DEFINED AS
        This is the distinguishing attribute of the NMSIG computer system managed object class.;
```

A.6.41 NMSIG System Time

```
nmsig-systemTime  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX GeneralizedTime;
    MATCHES FOR   Equality, Ordering;
            BEHAVIOUR  systemTime-behaviour;

   REGISTERED AS      {nmsig-attribute 41};

    systemTime-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies the current time clocked at the computer system.;
```

A.6.42 NMSIG Transport Connection Id

```
nmsig-transportConnectionId  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Id;
    MATCHES FOR  Equality ;
            BEHAVIOUR  transportConnectionId-behaviour;
```

```
REGISTERED AS      {nmsig-attribute 42};

   transportConnectionId-behaviour   BEHAVIOUR
DEFINED AS
        This attribute is the distinguishing attribute for the managed object class transportConnection.;
```

A.6.43 NMSIG Transport Connection Profile Id

```
nmsig-transportConnectionProfileId  ATTRIBUTE
   WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Id;
   MATCHES FOR  Equality ;
          BEHAVIOUR  transportConnectionProfileId-behaviour;

REGISTERED AS      {nmsig-attribute 43};

   transportConnectionProfileId-behaviour   BEHAVIOUR
DEFINED AS
        This attribute is the distinguishing attribute for the managed object class nmsig-transportConnectionProfile.;
```

A.6.44 NMSIG Transport Connection Reference

```
nmsig-transportConnectionReference      ATTRIBUTE
   WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Address;
   MATCHES FOR  Equality ;
          BEHAVIOUR  transportConnectionReference-behaviour;

REGISTERED AS      {nmsig-attribute 44};

   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.6.45 NMSIG Transport Entity Type

```
    nmsig-transportEntityType ATTRIBUTE
  WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.TransportEntityType;
  MATCHES FOR  Equality ;
        BEHAVIOUR  transportEntityType-behaviour;

 REGISTERED AS      {nmsig-attribute 45};

   transportEntityType-behaviour   BEHAVIOUR
```

```
DEFINED AS
        This attribute specifies the type of the transport protocol layer entity.;
```

A.6.46 NMSIG User Friendly Label

```
nmsig-userFriendlyLabel  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.Label;
    MATCHES FOR  Equality;
            BEHAVIOUR  userFriendlyLabel-behaviour;

REGISTERED AS      {nmsig-attribute 46};

    userFriendlyLabel-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies a user friendly name.;
```

A.6.47 NMSIG Vendor Name

```
nmsig-vendorName   ATTRIBUTE
    WITH ATTRIBUTE SYNTAX   NMSIG-SYNTAX-1.AnyName;
    MATCHES FOR  Equality ;
            BEHAVIOUR  vendorName-behaviour;

REGISTERED AS     {nmsig-attribute 47};

    vendorName-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies the name of a vendor.;
```

A.6.48 NMSIG Wrapped Counter

```
nmsig-wrappedCounter  ATTRIBUTE
    WITH ATTRIBUTE SYNTAX  NMSIG-SYNTAX-1.WrappedCounter;
    MATCHES FOR  Equality;
            BEHAVIOUR  wrappedCounter-behaviour;

REGISTERED AS      {nmsig-attribute 48};

    wrappedCounter-behaviour   BEHAVIOUR
DEFINED AS
        This attribute specifies the attribute Id and value of the counter attribute that wrapped.;
```

A.7 ATTRIBUTE GROUPS

This section provides definitions of attribute groups supported by managed object classes defined by the NMSIG.

A.8 PARAMETERS

This section provides definitions of parameters supported by managed object classes defined by the NMSIG.

A.9 ACTIONS

This section provides definitions of actions supported by managed object classes defined by the NMSIG.

A.10 NOTIFICATIONS

This section provides definitions of notifications emitted by managed object classes defined by the NMSIG.

A.10.1 NMSIG Counter Wrap

```
nmsig-counterWrap   NOTIFICATION
     BEHAVIOUR  counterWrap-behaviour;
     MODE  CONFIRMED AND UNCONFIRMED
     WITH INFORMATION SYNTAX   NMSIG-SYNTAX-1.WrapInfo
       AND ATTRIBUTE IDS  wrappedCounter  wrappedCounter;

REGISTERED AS  {notification};

counterWrap-behaviour   BEHAVIOUR
        DEFINED AS
             This notification indicates that a counter has wrapped.;
```

A.11 REFERENCES

This section lists the names of documents that were referenced in the earlier sections.

[DMI]

A.12 SYNTAX DEFINITIONS

This section contains an ASN.1 module that defines attribute and notification syntaxes referenced by the attribute and notification templates in Sections A.6 and A.8 respectively.

```
NMSIG-SYNTAX-1  {nmsig mil(2) nmsig-modules(0) syntax-1(0)}
        DEFINITIONS IMPLICIT TAGS ::= BEGIN

IMPORTS Attribute FROM CMIP-1 {joint-iso-ccitt ms(9) cmip(1) modules(0) protocol(3) }

       Address ::= OCTET STRING

       Amount ::= INTEGER

    AnyName ::=  SET OF ( CHOICE {  dn   DistinguishedName,
                                ps   PrintableString } )

    Cause  ::=  SEQUENCE {       INTEGER ( unknown (0),
                                      user    (1),
                                      provider (2) ),
                              INTEGER ( unknown (0),
                                      excessiveIdle (1),
                                      excessiveRtx  (2) )
                                }

    CPU-Type ::= PrintableString;

    EntityUpTime ::= INTEGER;

       EquipmentPurpose  ::= PrintableString

    Id ::= PrintableString

       InactivityTimeout ::= INTEGER

    Label  ::= PrintableString

       Length ::= INTEGER

    LocalNetworkAddresses  ::=  SET OF OCTET STRING

       ManufacturerInfo ::= PrintableString
```

```
        MaxNumber ::= INTEGER


    NetworkEntityType ::=   INTEGER { other(0),
                            oSI CLNP (1),
                            internet IP (2) } (0..256)

        NetworkPurpose ::= PrintableString

        Number ::=  INTEGER;

    OsInfo  ::=  SET  OF ( CHOICE { osName  [0] DistingishedName,
                           osSpec  [1] PrintableString } )

    PeripheralNames ::=  SET OF AnyName

        ProductLabel ::= PrintableString

        Release ::= PrintableString

        SerialNumber ::= PrintableString

    TransportAddresses ::=   SET  OF  SEQUENCE {
transportConnectionEndpoint OCTET STRING,
                        networkAddress  OCTET STRING  }

    TransportEntityType ::=  INTEGER { other(0),
                             oSI TP (1),
                             tCP (2),
                             sNA (3) } (0..256)

    WrapInfo  ::=   SEQUENCE {
                  wrappedCounter  Attribute  -- attribute ID and value of counter attribute that wrapped
                    }

    WrappedCounter ::=  Attribute

END -- End of NMSIG-SYNTAX-1 module
```

ANNEX B -- NMSIG Object Identifiers

B. NMSIG Object Identifiers

B.1 Introduction

This Annex (B) specifies object identifier component values which are globally unambiguous. These object identifiers are to be used when referencing NMSIG-specified information objects. As defined in Part 6 of these agreements, the OIW has assigned the following object identifier for use by the NMSIG:

{ iso (1) identified-organization(3) oiw(14) nmsig (2) }

The following object identifers are assigned under the { iso identified-organization oiw nmsig } node, labelled "nmsig".

Table B.1: Object Identifers Assigned Under "nmsig" Node

Identifier	Value	Reference
phase1	1	B.2
mil1	2	B.3

B.2 Phase 1 Object Identifiers

Several of the base standards referenced by Phase I agreements are at Draft International Standard (DIS) level. These draft standards specify a number of information objects, each accompanied by a tentative object identifier. However, actual object identifiers are not assigned and registered by ISO until the base standard reaches International Standard (IS) level.

Implementations require globally unamibiguous object identifiers to interoperate using Phase I agreements. Since these object identifiers are not yet specified by ISO, it is necessary for the NMSIG to assign and register identifiers needed to support Phase I agreements. All such object identifiers are assigned under a single "phase1" node of the NMSIG object identifier tree.

Object identifiers under the phase1 node are assigned such that the phase1 node replaces the corresponding { joint-iso-ccitt ms } node tentatively specified in the Draft International Standards. That is:

```
joint-iso-ccitt───ms─────┬─smo──────┬─application-context──────┬─manager
                         │          │                          ├─agent
                         │          │                          └─manager-agent
                         │          │
                         │          └─negotiationAbstractSyntax──versionl
                         │
                         ├─function─┬─partl───functionalUnitPackage
                         │          ├─part4───functionalUnitPackage
                         │          └─part5───functionalUnitPackage
                         │
                         └─smi────────part2─┬─managedObjectClass
                                            ├─nameBinding
                                            ├─package
                                            ├─attribute
                                            ├─attributeGroup
                                            ├─standardSpecificExtension
                                            └─notification
```

becomes

```
nmsig─────────────phasel─┬─smo──────┬─application-context──────┬─manager
                         │          │                          ├─agent
                         │          │                          └─manager-agent
                         │          │
                         │          └─negotiationAbstractSyntax──versionl
                         │
                         ├─function─┬─partl───functionalUnitPackage
                         │          ├─part4───functionalUnitPackage
                         │          └─part5───functionalUnitPackage
                         │
                         └─smi────────part2─┬─managedObjectClass
                                            ├─nameBinding
                                            ├─package
                                            ├─attribute
                                            ├─attributeGroup
                                            ├─standardSpecificExtension
                                            └─notification
```

Using this methodology, the following object identifiers are assigned under the { iso identified-organization oiw nmsig phase1 } node.

Table B.2: Object Identifers Assigned Under "phase1" Node

Identifier	Value	Reference
smo	0	[GDMO] 8.4
function	2	[GDMO] 8.4
smi	3	[GDMO] 8.4

These object identifiers are to be used when referencing ISO DIS-specified Phase I information objects. During progression from DIS to IS, it is possible that some information object definitions and/or tentatively-specified object identifiers will change. As a result, final IS documents will contain a new set of information objects, each assigned its own ISO object identifer. This shall be the case even if no changes are made to the object definition during progression from DIS to IS. Phase1 object identifiers shall correspond in perpetuity to the referenced DIS text upon which Phase 1 stable agreements are based. However, to encourage timely migration, the use of phase1 object identifiers shall be deprecated 2 years after corresponding International Standard Profiles (ISPs) become available.

B.2.1 SMO Object Identifiers

The following object identifiers are assigned under the { iso identified-organization oiw nmsig phase1 smo } node.

Table B.3: Object Identifers Assigned Under "smo" Node

Identifier	Value	Reference
application-context	0	[GDMO] 8.4
negotiationAbstractSyntax	1	[GDMO] 8.4

The following object identifiers are assigned under the { iso identified-organization oiw nmsig phase1 smo application-context } node.

Table B.4: Object Identifers Assigned Under "application-context" Node

Identifier	Value	Reference
manager	0	[SMO] A.2.3
agent	1	[SMO] A.3.3
manager-agent	2	[SMO] A.4.3

The following object identifiers are assigned under the { iso identified-organization oiw nmsig phase1 smo negotationAbstractSyntax } node.

Table B.5: Object Identifers Assigned Under "negotiationAbstractSyntax" Node

Identifier	Value	Reference
version1	1	[SMO] A.5.4

B.2.2 Function Object Identifiers

The following object identifiers are assigned under the { iso identified-organization oiw nmsig phase1 function } node.

Table B.6: Object Identifers Assigned Under "function" Node

Identifier	Value	Reference
part1	1	[GDMO] 8.4
part4	4	[GDMO] 8.4
part5	5	[GDMO] 8.4

The following object identifiers are assigned under the { iso identified-organization oiw nmsig phase1 function part1 } node.

Table B.7: Object Identifers Assigned Under "part1" Node

Identifier	Value	Reference
functionalUnitPackage	1	[OMF] 10

The following object identifiers are assigned under the { iso identified-organization oiw nmsig phase1 function part4 } node.

Table B.8: Object Identifers Assigned Under "part4" Node

Identifier	Value	Reference
functionalUnitPackage	1	[ARF] 11.3

The following object identifiers are assigned under the { iso identified-organization oiw nmsig phase1 function part5 } node.

Table B.9: Object Identifers Assigned Under "part5" Node

Identifier	Value	Reference
functionalUnitPackage	1	[ERMF] 11.3

B.2.3 SMI Object Identifiers

The following object identifiers are assigned under the { iso identified-organization oiw nmsig phase1 smi } node.

Table B.10: Object Identifers Assigned Under "smi" Node

Identifier	Value	Reference
part2	2	[GDMO] 8.4

The following object identifiers are assigned under the { iso identified-organization oiw nmsig phase1 smi part2 } node.

Table B.11: Object Identifers Assigned Under "part2" Node

Identifier	Value	Reference
managedObjectClass	3	[GDMO] 8.4
nameBinding	6	[GDMO] 8.4
package	4	[GDMO] 8.4
attribute	7	[GDMO] 8.4
attributeGroup	8	[GDMO] 8.4
standardSpecificExtension	0	[GDMO] 8.4
notification	10	[GDMO] 8.4

Object identifiers are assigned under the { iso identified-organization oiw nmsig phase1 smi part2 } node by the following ASN.1 productions.

```
smi2MObjectClass OBJECT IDENTIFER ::= -- supersedes [DMI] 13.1
    { iso(1) identified-organization(3) oiw(14) nmsig(2)
    phase1(1) smi(3) part2(2) managedObjectClass(3) }
```

```
smi2NameBinding OBJECT IDENTIFER ::= -- supersedes [DMI] 13.1
    { iso(1) identified-organization(3) oiw(14) nmsig(2)
    phase1(1) smi(3) part2(2) nameBinding(6) }

smi2Package OBJECT IDENTIFIER ::= -- supersedes [DMI] 13.1
    { iso(1) identified-organization(3) oiw(14) nmsig(2)
    phase1(1) smi(3) part2(2) package(4) }

smi2AttributeID OBJECT IDENTIFIER ::= -- supersedes [DMI] 13.2
    { iso(1) identified-organization(3) oiw(14) nmsig(2)
    phase1(1) smi(3) part2(2) attribute(7) }

smi2AttributeGroup OBJECT IDENTIFIER ::= -- supersedes [DMI] 13.2
    { iso(1) identified-organization(3) oiw(14) nmsig(2)
    phase1(1) smi(3) part2(2) attributeGroup(8) }

arfProbableCause OBJECT IDENTIFIER ::= -- supersedes [DMI] 13.2
    { iso(1) identified-organization(3) oiw(14) nmsig(2)
    phase1(1) smi(3) part2(2) standardSpecificExtension (0) arf (0) }

smi2Notification OBJECT IDENTIFIER ::= -- supersedes [DMI] 13.3
    { iso(1) identified-organization(3) oiw(14) nmsig(2)
    phase1(1) smi(3) part2(2) notification(10) }
```

These ASN.1 productions effectively register, under the nmsig phase1 smi part2 node, all information objects specified in [DMI]. For example:

```
attributeValueChange NOTIFICATION
REGISTERED AS { smi2Notification 1 }
```

as defined in clause 12.1 of [DMI] is assigned the object identifier:

```
{ iso(1) identified-organization(3) oiw(14) nmsig(2)
phase1(1) smi(3) part2(2) notification(10) attributeValueChange(1) }
```

The majority of these information objects support Phase 1 agreements and shall be used as specified throughout Chapter 18. However, a small number of information objects specified in [DMI] support functionality beyond the scope of Phase 1 agreements. Although such objects are registered under the nmsig phase1 smi part2 node, their use is outside the scope of these agreements.

B.3 MIL Object Identifiers

These are the object identifiers referenced in Annex A -- Management Information Library.

All definitions that are registered under the mil1 node are based on the DIS version of the [GDMO] standard.

MIL Object Identifiers are assigned under the "nmsig" node as follows:

```
nmsig   OBJECT IDENTIFIER  ::=
             { iso identified-organizations(3) oiw(14) 2 }

mil1              OBJECT IDENTIFIER  ::=     { nmsig 2 }

nmsig-modules        OBJECT IDENTIFIER  ::=     { mil1 0 }

nmsig-objectClass    OBJECT IDENTIFIER  ::=     { mil1 1 }

nmsig-package        OBJECT IDENTIFIER  ::=     { mil1 2 }

nmsig-nameBindings   OBJECT IDENTIFIER  ::=     { mil1 3 }

nmsig-attribute      OBJECT IDENTIFIER  ::=     { mil1 4 }

nmsig-attributeGroups  OBJECT IDENTIFIER ::=    { mil1 5 }

nmsig-parameter      OBJECT IDENTIFIER  ::=     { mil1 6 }

nmsig-action         OBJECT IDENTIFIER  ::=     { mil1 7 }

nmsig-notification   OBJECT IDENTIFIER  ::=     { mil1 8 }
```

B.3.1 Object Class Object Identifiers

The following object identifiers are assigned under the { nmsig-objectClass } node:

Table B.12: Object Identifiers Assigned Under "nmsig-objectClass" Node

Identifier	Value	Reference
nmsig-agent	4	A.4.1.1
nmsig-computerSystem	2	A.4.2.1
nmsig-coTransportProtocolLayerEntity	3	A.4.3.1
nmsig-clNetworkProtocolLayerEntity	4	A.4.4.1
nmsig-equipment	5	A.4.5.1
nmsig-network	5	A.4.6.1
nmsig-processingEntity	7	A.4.7.1
nmsig-transportConnection	8	A.4.8.1

Identifier	Value	Reference
nmsig-transportConnectionProfile	9	A.4.9.1
nmsig-transportConnectionRetransmissionProfile	10	A.4.10.1

B.3.2 Package Object Identifiers

The following object identifiers are assigned under the { nmsig-package } node:

Table B.13: Object Identifiers Assigned Under "nmsig-package" Node

Identifier	Value	Reference
nmsig-agent-Package	4	A.4.1.2
nmsig-computerSystem-Package	2	A.4.8.2
nmsig-coTransportProtocolLayerEntity-Package	3	A.4.3.2
nmsig-productInfo-Package	4	A.4.3.4
nmsig-clNetworkProtocolLayerEntity-Package	5	A.4.4.2
nmsig-clNetworkProtocolLayerEntityRedirection-Package	6	A.4.4.4
nmsig-equipment-Package	7	A.4.5.2
nmsig-network-Package	8	A.4.6.2
nmsig-processingEntity-Package	5	A.4.4.2
nmsig-transportConnection-Package	10	A.4.8.2
nmsig-transportConnectionRetransmission-Package	11	A.4.3.4
nmsig-transportConnectionProfile-Package	12	A.4.9.2
nmsig-transportConnectionRetransmissionProfile-Package	13	A.4.10.2

B.3.3 Name Bindings Object Identifiers

The following object identifiers are assigned under the { nmsig-nameBindings } node:

Table B.14: Object Identifiers Assigned Under "nmsig-nameBindings" Node

Identifier	Value	Reference
agent-root	1	A.5.1
computerSystem-network	2	A.5.2
computerSystem-computerSystem	3	A.5.2
computerSystem-root	4	A.5.2
coTransportProtocolLayerEntity-computerSystem	5	A.5.3
coTransportProtocolLayerEntity-system	6	A.5.3
coTransportProtocolLayerEntity-equipment	7	A.5.3
clNetworkProtocolLayerEntity-computerSystem	8	A.5.4
clNetworkProtocolLayerEntity-system	9	A.5.4
clNetworkProtocolLayerEntity-equipment	10	A.5.4
equipment-equipment	11	A.5.5
equipment-network	12	A.5.5
equipment-root	13	A.5.5
network-network	14	A.5.6
network-root	15	A.5.6
processingEntity-computerSystem	16	A.5.4
transportConnection-coTransportProtocolLayerEntity	17	A.5.8
transportConnectionProfile-coTransportProtocolLayerEntity	18	A.5.9
transportConnectionRetransmissionProfile-coTransportProtocolLayerEntity	19	A.5.10

B.3.4 Attribute Object Identifiers

The following object identifiers are assigned under the { nmsig-attribute } node:

Table B.15: Object Identifiers Assigned Under "nmsig-attribute" Node

Identifier	Value	Reference
nmsig-agentId	8	A.6.1
nmsig-cause	8	A.6.8
nmsig-checksumTPDUsDiscardedCounter	8	A.6.3
nmsig-coTransportProtocolLayerEntityId	8	A.6.4
nmsig-clNetworkProtocolLayerEntityId	5	A.6.9
nmsig-contactNames	5	A.6.6
nmsig-cPUType	7	A.6.4
nmsig-entityUpTime	8	A.6.8
nmsig-equipmentId	8	A.6.9
nmsig-equipmentPurpose	16	A.6.12
nmsig-inactivityTimeout	14	A.6.14
nmsig-localNetworkAddress	12	A.6.12
nmsig-localNetworkAddresses	13	A.6.13
nmsig-localTransportAddresses	14	A.6.14
nmsig-localTransportConnectionEndpoint	15	A.6.19
nmsig-locationName	16	A.6.16
nmsig-manufacturerInfo	17	A.6.17
nmsig-maxConnections	16	A.6.18
nmsig-maxPDUSize	16	A.6.19
nmsig-maxRetransmissions	20	A.6.20
nmsig-memorySize	21	A.6.21
nmsig-networkEntityType	20	A.6.22
nmsig-networkId	23	A.6.23
nmsig-networkPurpose	24	A.6.24
nmsig-nPDUTimeToLive	25	A.6.25

Identifier	Value	Reference
nmsig-octetsRetransmittedErrorCounter	26	A.6.26
nmsig-osInfo	28	A.6.28
nmsig-openConnections	28	A.6.28
nmsig-PDUsDiscardedErrorCounter	29	A.6.28
nmsig-PDUsForwardedCounter	30	A.6.30
nmsig-PDUsReasmblFailCounter	38	A.6.31
nmsig-PDUsReasmbldOKCounter	32	A.6.32
nmsig-peripheralNames	33	A.6.33
nmsig-productLabel	34	A.6.34
nmsig-release	35	A.6.35
nmsig-remoteNetworkAddress	36	A.6.36
nmsig-remoteTransportConnectionEndpoint	37	A.6.37
nmsig-retransmissionTimerInitialValue	38	A.6.38
nmsig-serialNumber	39	A.6.38
nmsig-systemId	45	A.6.40
nmsig-systemTime	41	A.6.41
nmsig-transportConnectionId	42	A.6.40
nmsig-transportConnectionProfileId	43	A.6.43
nmsig-transportConnectionReference	44	A.6.41
nmsig-transportEntityType	45	A.6.45
nmsig-userFriendlyLabel	45	A.6.46
nmsig-vendorName	47	A.6.47
nmsig-wrappedCounter	48	A.6.48

B.3.5 Notification Object Identifiers

The following object identifiers are assigned under the { nmsig-notification } node:

Table B.16: Object Identifiers Assigned Under "nmsig-notification" Node

Identifier	Value	Reference
nmsig-counterWrap	1	A.10.1

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 19 - Remote Database Access

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair **Peter Eng**
SIG Editors **Peter Eng**

Table of Contents

List of Figures

Foreword

This part of the Working Implementation Agreements was prepared by the Remote Database Access Special Interest Group (RDASIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop Charter.

Text in this part has been approved by the Plenary of the Workshop. This part replaces the previously existing chapter on this subject.

Part 19 - Remote Database Access

0 Introduction

Remote Database Access (RDA) specifies the communications service and protocol for accessing the capabilities of a database server from a client application. Figure 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 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.

1 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:

a) the RDA generic model, service, and protocol,

b) the RDA SQL Specialization,

c) SQL language restrictions.

2 Normative 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.

3 Definitions

4 Abbreviations

5 RDA Dialogue State Model Agreements

6 Generic RDA Agreements

6.1 Functional Units

6.2 Optional Negotiable Facilities

6.2.1 Open/Close Within Transaction

6.2.2 Other Facilities

6.3 Services

6.3.1 R-BeginDialogue

6.3.1.1 Optional Parameters

6.3.1.2 Parameter Restrictions

6.3.2 R-EndDialogue

6.3.3 R-Open

6.3.4 R-Close

6.3.5 R-Execute

6.3.6 R-Define

6.3.7 R-Invoke

6.3.8 R-Drop

6.3.9 R-BeginTransaction

6.3.10 R-Commit

6.3.11 R-Rollback

6.3.12 R-Status

6.3.13 Cancel

7 SQL Specialization Agreements

7.1 Functional Units

7.2 Optional Facilities

7.3 Services

7.4 SQL Language Agreements

7.4.1 Language/Protocol Mapping

7.4.2 Implementor Defined Items

7.4.3 SQL Functional Restrictions

7.4.4 SQL State and Error Messages

7.5 Conformance Requirements

7.6 Recommended Practices

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 20 - Manufacturing Message Specification (MMS)

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair **Herbert Falk**
SIG Editors **Neal Laurance**

Table of Contents

Foreword

This part of the Working Implementation Agreements was prepared by the Manufacturing Message Specification (MMS) Special Interest Group (MMSSIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for Workshop charter.

Text in this part has been approved by the Plenary of the above-mentioned Workshop. Significant technical change has occurred in this part since it was previously presented.

Part 20 - Manufacturing Message Specification (MMS)

0 Introduction

(Refer to the Stable Agreements, dated December, 1990.)

1 Scope

(Refer to the Stable Agreements, dated December, 1990.)

2 Field of Application

3 Normative References

(Refer to the Stable Agreements, dated December, 1990.)

4 Definitions

(Refer to the Stable Agreements, dated December, 1990.)

5 Corrigenda and Addenda

None at time of publication.

6 Status

Phase 1 is in progress.

7 General Agreements

7.1 Max supported PDU size

(Refer to the Stable Agreements, dated December, 1990.)

7.2 FileName

(Refer to the Stable Agreements, dated December, 1990.)

8 Service-Specific Agreements

8.1 Environment and general management

(Refer to the Stable Agreements, dated December, 1990.)

8.2 VMD Support

8.3 Domain Management

8.3.1 List of capabilities

(Refer to the Stable Agreements, dated December, 1990.)

8.3.2 Initiate Download Sequence service

(Refer to the Stable Agreements, dated December, 1990.)

8.3.3 Download Segment service

(Refer to the Stable Agreements, dated December, 1990.)

8.3.4 Terminate Download Sequence service

(Refer to the Stable Agreements, dated December, 1990.)

8.3.5 Initiate Upload Sequence service

(Refer to the Stable Agreements, dated December, 1990.)

8.3.6 Upload Segment service

(Refer to the Stable Agreements, dated December, 1990.)

8.3.7 Get Domain Attributes service

(Refer to the Stable Agreements, dated December, 1990.)

8.3.8 Get Capability List service

(Refer to the Stable Agreements, dated December, 1990.)

8.4 Program Invocation Management

8.4.1 Start service

(Refer to the Stable Agreements, dated December, 1990.)

8.4.2 Stop service

(Refer to the Stable Agreements, dated December, 1990.)

8.4.3 Resume service

(Refer to the Stable Agreements, dated December, 1990.)

8.4.4 Reset service

(Refer to the Stable Agreements, dated December, 1990.)

8.5 Variable Access

8.5.1 Scattered access

(Refer to the Stable Agreements, dated December, 1990.)

8.5.2 Floating point

(Refer to the Stable Agreements, dated December, 1990.)

8.6 Semaphore Management

Semaphore services are not considered in Phase 1.

8.7 Operator Communication

(Refer to the Stable Agreements, dated December, 1990.)

8.8 Event Management

Event Management services are not considered in Phase 1.

8.9 Journal Management

Journal Management services are not considered in Phase 1.

Annex A (normative)

Backwards compatibility agreements

(Refer to the Stable Agreements, dated December, 1990.)

Annex B (normative)

DIS 9506 modifications required for backwards compatibility

(Refer to the Stable Agreements, dated December, 1990.)

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 21 - Character Set

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Editor Wally Wedel

Foreword

This part of the Working Implementation Agreements was prepared by the Character Set Working Group, formerly affilitated with the Upper Layer Special Interest Group of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). See Procedures Manual for workshop charter. Text inthis part has been approved by the Plenary of the above-named workshop.

Part 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.

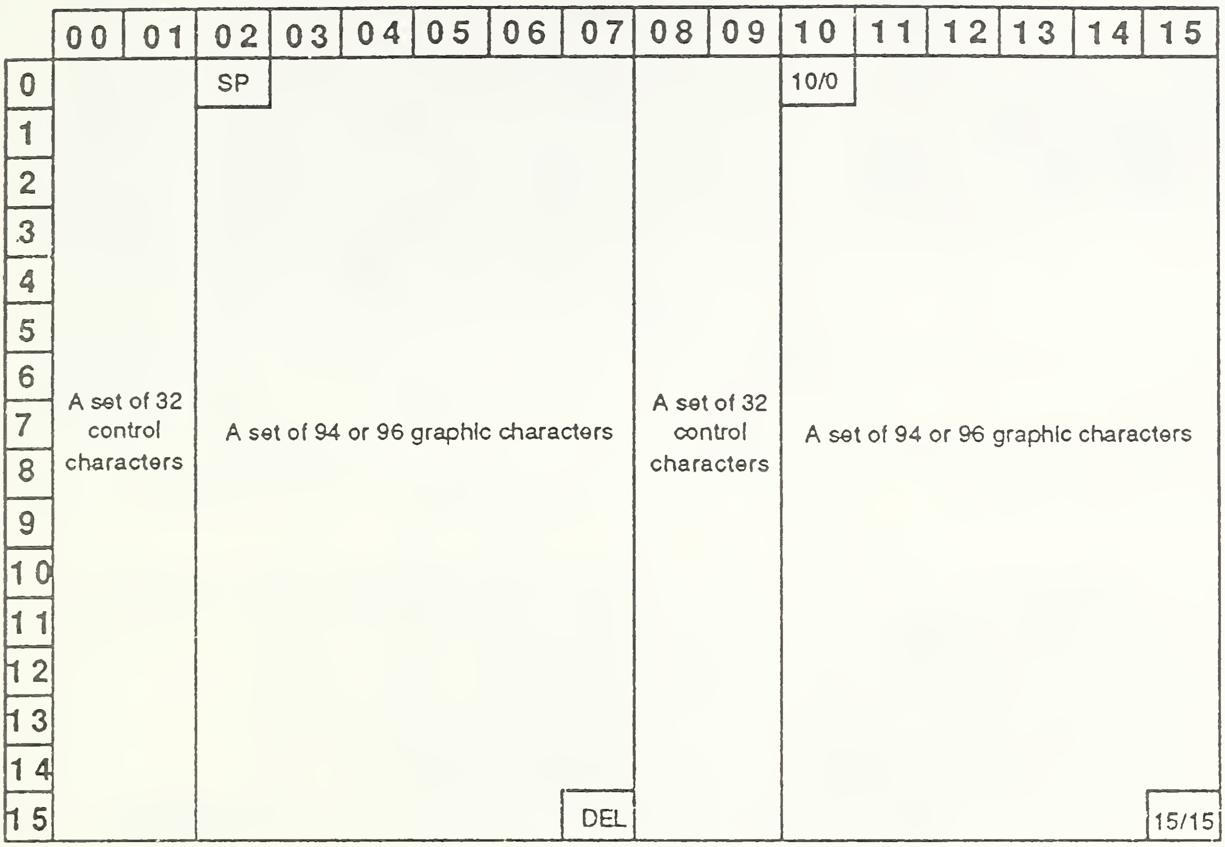
Some underlying assumptions from the standard are recorded here in order to understand the context of these agreements. Clause 2 notes that code extension techniques are designed to be used for data to be processed serially in a forward direction.

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:

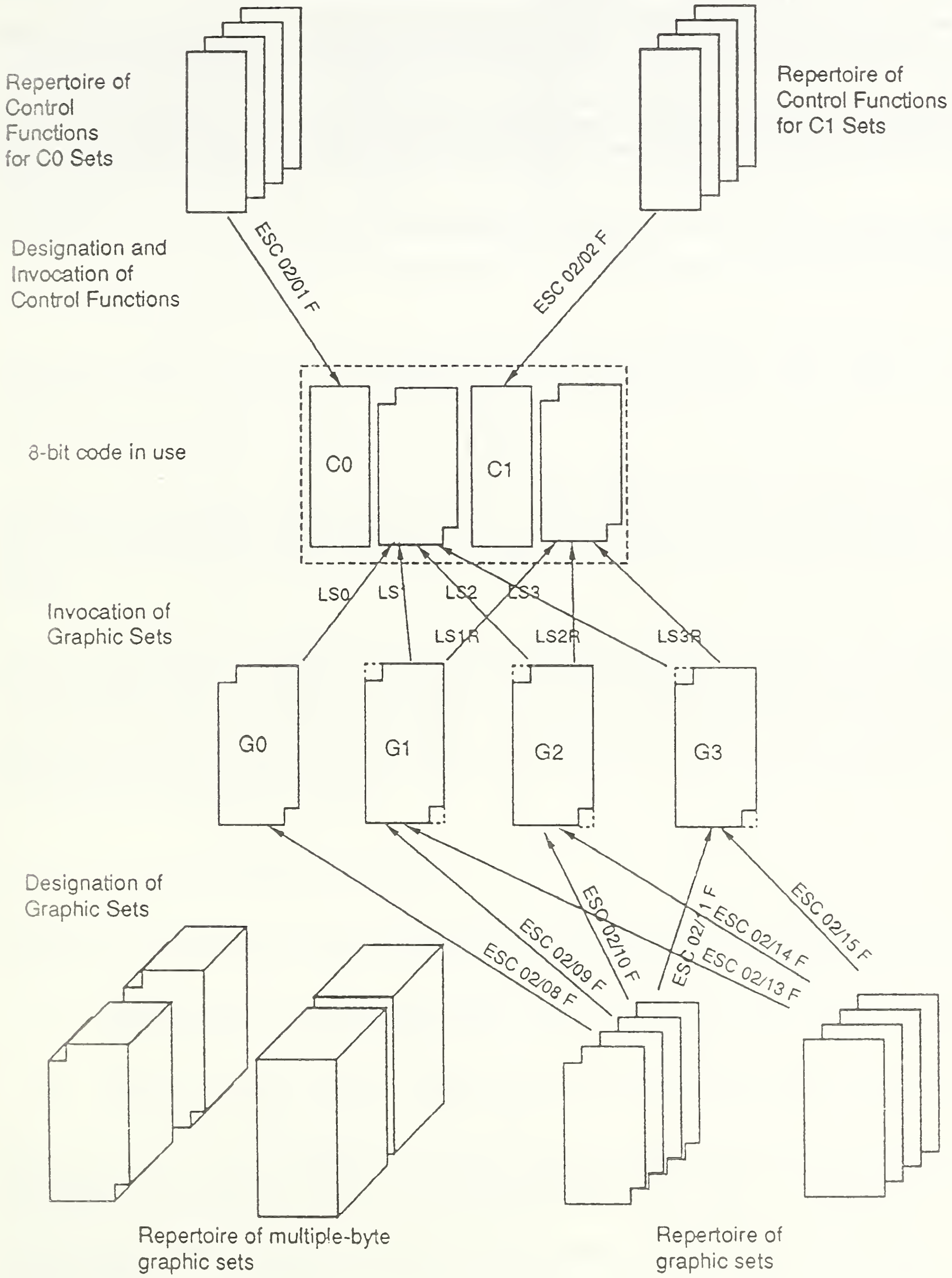
Set	Description	Columns occupied
C0	32 control characters	00 to 01
C1	32 control characters	08 to 09
G0	94 graphic characters	02 to 07
G1	94 or 96 graphic characters	02 to 07 or 10 to 15
G2	94 or 96 graphic characters	02 to 07 or 10 to 15
G3	94 or 96 graphic characters	02 to 07 or 10 to 15

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	\	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	\	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;

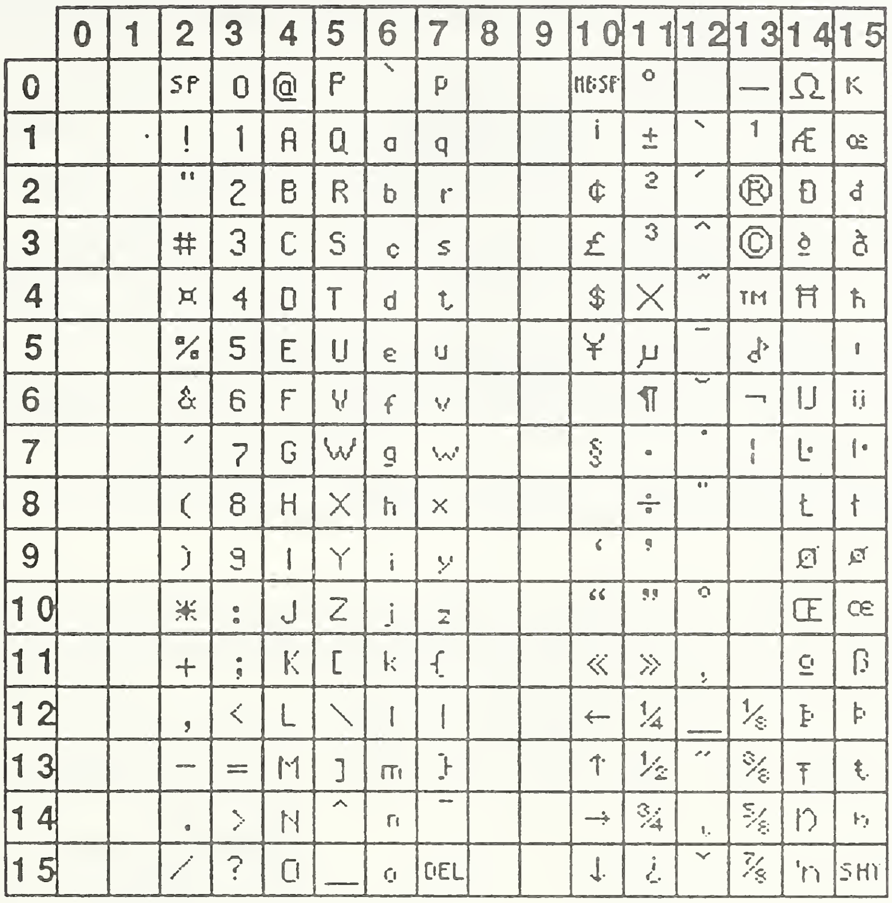
b) specifies coded representations for the graphic characters;

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			$	×	~	TM	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?>>

Working Implementation Agreements for Open Systems Interconnection Protocols: Part 22 - NIST ODA RASTER DAP

Output from the December 1990 NIST Workshop for Implementors of OSI

SIG Chair **Frank Dawson**
Document Editor **Frank Spielman**

Table of Contents

List of Figures

List of Tables

Foreword

This part of the Working Implementation Agreements was prepared by the Office Document Architecture (ODA) Special Interest Group (SIG) of the National Institute of Standards and Technology (NIST) Workshop for Implementors of Open Systems Interconnection (OSI). Development of this document application profile has been done in liaison with several organizations. These include the DoD Computer-aided Acquisition and Logistic Support (CALS) Office, Navy's David Taylor Research Center, and the ad-hoc Tiling Task Group.

This document application profile is intended to be suitable for the interchange of large format raster images.

This part contains four annexes:

o annex A (normative): Addenda and errata;

o annex B (informative): Recommended practices;

o annex C (informative): References to other standards and registers;

o annex D (informative): Supplementary information on attributes.

Part 22 - NIST ODA Raster DAP

0 Introduction

This is the definition of a single specification for two Open Document Architecture (ODA) Document Application Profiles (DAPs) named National Institute of Standards and Technology (NIST) ODA Raster DAP. The two DAPs differ only in the encoding of the data stream. One uses the ASN.1 based ODIF encoding. The other uses the SGML/SDIF based ODIF encoding. When this document refers to *this profile*, it is referring to either of the DAPs defined by this specification.

This DAP is suitable for interchanging documents in formatted form. The documents contain primarily raster graphics images. However, the raster images can be annotated with character, raster graphics or geometric graphics content portions. This DAP has been prepared by the ODA Special Interest Group of the NIST Open Systems Interconnection (OSI) Implementors Workshop. The DAP is defined in accordance with ISO 8613-1 and CCITT T.411 and follows the standardized proforma and notation defined in the proposed Draft Addendum to ISO 8613-1 Annex F (to be published). The DAP is based on ODA as defined in ISO 8613 and the Draft Addendum to ISO 8613, Part 7.

1 Scope

This DAP specifies an interchange format suitable for transfer of structured documents between equipment designed for raster processing. The documents supported by this DAP are based on a paradigm of an electronic engineering drawing or illustration. Such documents contain one or more pages. Each page consists of one or more pages of a base image in the form of a bi-tonal raster graphics content. This base image may be further annotated with character, raster graphics or geometric graphics content. This latter content portions serves to provide revision control for the engineering drawing or illustration. There is no restriction on the minimum size of the base image.

This document defines a DAP 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 DAP, all non-basic features are subject to negotiation.

This DAP 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 exchanged storage media.

The features of a document that can be interchanged using this DAP 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.

2 Normative References

The following references are required in order to implement this DAP:

ISO 8613-1 : 1989, Information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 1: Introduction and General Principles;

ISO 8613-2 : 1989, Information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 2: Document Structures;

ISO 8613-4 : 1989, Information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 4: Document Profile;

ISO 8613-5 : 1989, Information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 5: Open Document Interchange Format;

ISO 8613-6 : 1989, Information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 6: Character Content Architecture;

ISO 8613-7 : 1989, Information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 7: Raster Graphics Content Architectures;

ISO 8613-8 : 1989, Information processing - Text and Office Systems; Open Document Architecture (ODA) and Interchange Format - Part 8: Geometric Graphics Content Architectures;

ISO 8613-1 : (to be published), Information processing - Text and Office Systems; Office Document Architecture (ODA) and Interchange Format - Part 1: DAD - A Document Application Profile Proforma and Notation;

ISO 8613-7 : (to be published), Information processing - Text and Office Systems; Office Document Architecture (ODA) and Interchange Format - Part 7: DAD - Tiled Raster Graphics Addendum to ISO 8613, Part 7;

ISO 646 : 1990, Information processing - ISO 7-bit coded character sets for information interchange;

ISO 8859-1 : 1983, Information processing - 8-bit Single-byte coded graphic character sets - Part 1: Latin alphabet No. 1;

ISO 6937-2 : 1983, Information processing - Coded character sets for text communication - Part 2: Latin alphabet and non-alphabetic characters;

ISO 2022 : 1986, Information processing - ISO 7-bit and 8-bit coded character sets - Code extension techniques;

ISO 7350 : 1984, Text communication - Registration of graphic character subrepertoires;

ISO 8824 : 1987,Information Processing Systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1);

ISO 8825 : 1987,Information Processing Systems - Open Systems Interconnection - Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1);

ISO 8879 : 1986,Information processing - Text and office systems - Standard Generalized Markup Language (SGML);

ISO 9069 : 1988, Information processing - SGML support facilities - SGML Document Interchange Format (SDIF);

CCITT Recommendation T.6 : 1988, Facsimile Coding Schemes and Coding Control Functions for Group 4 Facsimile Apparatus.

3 Definitions and Terminology

3.1 Definitions

The definitions given in ISO 8613-1 are applicable to this document.

3.2 Constituent Names

Each constituent that may be included in a document that conforms to this profile has been given a unique name which serves to identify that constituent throughout this profile.

The convention is that full names are used (i.e., no abbreviations are used), two or more words in a name are concatenated and each word begins with a capital. Examples of constituent names used in this profile are BasicPage, CompositePage and LayoutDocumentRoot.

In clause 6 of this profile, each constituent provided by this profile is underlined once at the point in the text at which the purpose of that constituent is defined. This also serves to identify all the constituents provided by this profile.

The same constituent names are also used in the technical specification in clause 7 of this profile so that there is a one-to-one correspondence between the use of these names in clauses 6 and 7.

Although the constituent names relate to the purpose of the constituents, the semantics of constituents must not be implied from the actual names that are used. Also, these names do not appear in an interchanged document but a mechanism for identifying constituents in an interchange document is provided. Thus in an application using this profile, the constituents may be known to the user by different names.

4 Relationship to other DAPs

Functionally, this DAP is a functional superset of the CCITT Recommendation T.503, A Document Application Profile for the Interchange of Group 4 Facsimile Documents.

5 Conformance

In order to conform to this DAP, a data stream representing a document must meet the requirements specified in 5.1.

The requirements for implementations that originate and/or receive data streams conforming to this DAP are specified in 5.2.

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 or the SGML encoding rules defined in ISO 8879;
- o The data stream shall be structured in accordance with the interchange format defined in clause 8 of this DAP;
- o The document shall be structured in accordance with only the formatted document architecture class specified in clause 7 of this DAP. 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 clause 7;
- o Each constituent shall contain all those attributes specified as required for that constituent in this profile. Other attributes may be specified provided 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 clause 6 of this DAP and shall contain only those features defined in clause 6.

5.2 Implementation Conformance

This clause states the requirements for implementations claiming conformance to this DAP.

A conforming receiving implementation must be capable of receiving *either* any data streams conforming to this profile structured in accordance with ODIF *or* any data streams conforming to this profile structured in accordance with ODL *or* both of the these. Receiving usually, but not always, involves recognizing and further processing the data stream elements.

6 Characteristics Supported by this DAP

This clause describes the characteristics of documents that 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.

6.1 Overview

This DAP 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 DAP describes documents that can be interchanged in the formatted form, which facilitates the reproduction of a document as intended by the originator.

The primary category of content within the document is a raster graphics content in the formatted processable form. This is intended to facilitate the reproduction of the document content as intended by the originator or facilitates the revision of the document content.

Additional content is allowed within the document to annotate revisions to the base raster graphics image(s) in the document. This additional content can be in the form of character, raster graphics or geometric graphics content.

This clause describes the layout features that can be represented in documents conforming to this DAP. The features are described in terms that are typical of the user-perceived capabilities and semantics found in a raster document interchange environment.

For the purpose of interchange, a document is represented as a collection of **constituents**, each of which is represented by a set of attributes. The constituents that make up a formatted document are defined below in this clause and are illustrated in figure 1.

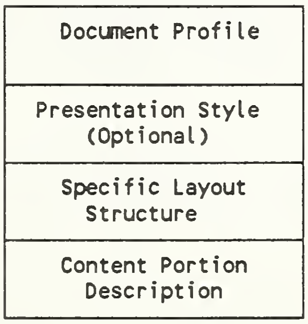

Document Profile
Presentation Style (Optional)
Specific Layout Structure
Content Portion Description

Figure 1 - Constituents

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.

6.2 Logical Constituents

Not applicable.

6.3 Layout Constituents

This clause describes the features of the layout objects that can be represented in documents conforming to this DAP.

6.3.1 Overview of the Layout Characteristics

The document structure allows the document content to presented as having been laid out in one or more pages. Each page in a document may consist of a single raster graphics content. This would be the case for an original image of an engineering drawing, illustration, or other raster scanned image. Optionally, each page in a document may consist of a raster graphics content, representing the original raster scanned image, with additional character, raster graphics or geometric graphics content, representing a set of revision annotation of the original 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 the pages of the document. The document can consist of either all basic pages or all composite pages. The basic page contains the content information, directly. The composite pages consist of frames that contain the content associated with the base image and the revision annotation.

Figure 2 is an illustration of the features of the document layout structure supported by this DAP:

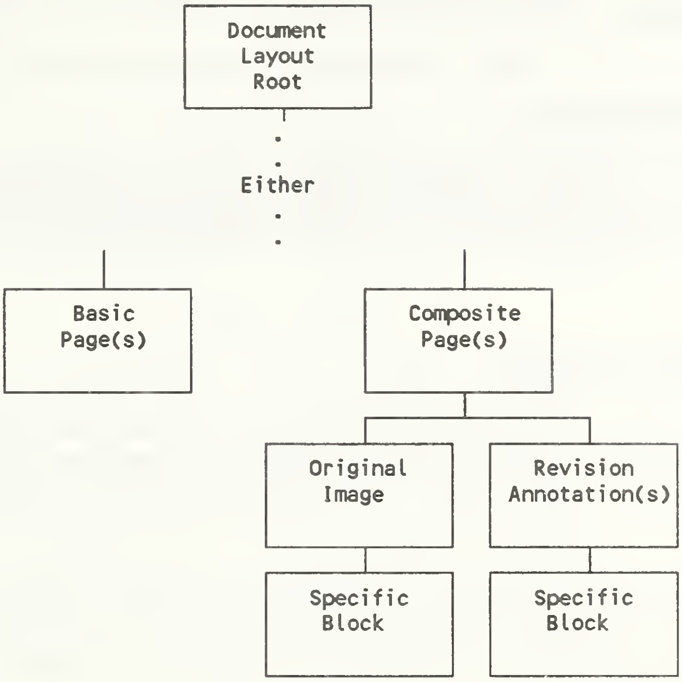


Figure 2 - Document Layout Structure

6.3.2 DocumentLayoutRoot

A DocumentLayoutRoot is the top level in a document layout structure. A DocumentLayoutRoot may consist of a either a sequence of one or more BasicPage constituent constraints or a sequence of one or more CompositePage constituent constraints.

6.3.3 Page Characteristics

Two page constituents are provided to present pages within a document. The use of these constituents is exclusive. That is, either one or the other form will appear in a document.

A document either consists of a sequence of one or more basic pages or a sequence of one or more composite pages. In a document containing basic pages, a single raster graphics content is positioned directly on each page. In a document containing composite pages, two types of frames are used to position content information on the page. One frame type is used to position on a single raster graphics content representing the original image on the page. The second frame type is used to position a character, raster graphics or geometric graphics content representing a revision annotation on the page. In a composite page, there maybe one or more of the frames containing a revision annotation.

6.3.3.1 BasicPage

A BasicPage is a basic layout object that corresponds to the area used for positioning and imaging the raster graphics content of the document. The BasicPage contains only one raster scanned image consisting of a single raster graphics content portion.

6.3.3.2 CompositePage

A CompositePage is a constituent constraint which defines a composite-page that corresponds to the page area used for presenting the sequence of an OriginalImage frame and one or more RevisionAnnotation frames.

6.3.3.3 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 foldout-size images and roll paper. These sizes apply to both portrait and landscape orientations.

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 A1-A4 sizes do fall within the range of the ANSI E sizes and therefore could be considered basic values (See table 2). A0 size is a non-basic value.

The default dimensions are the Common Assured Reproduction Area (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 assured reproduction area (ARA) listed in table 1 be used wherever feasible. See ISO 8613-2, clause 7.3, General rules for positioning pages on presentation surfaces.

6.3.3.4 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 dimensions of the BasicPage exceed the ARA for the specified nominal page size.

Table 1 Dimensions for Various Page Sizes

Page TypeSizeSize (BMU)ARA (BMU)

- Metric (mm)
ISO-A4210X297 9920 x 14030 9240 x 13200
ISO-A3297X42014030 x 1984013200 x 18480
ISO-A2420X59419840 x 2806018898 x 27118
ISO-A1594X84028060 x 3968026173 x 37843
ISO-A0840X118839680 x 5612037843 x 54283

- ANSI, North
American(NA) (inches)
NA-A8.5X1110200 x 13200 9240 x 12400
NA-B11X1713200 x 2040012744 x 19656
NA-C17X2220400 x 2640019500 x 25800
NA-D22X3426400 x 4080025800 x 39600
NA-E34X4440800 x 5280039600 x 52200
NA-F28X4033600 x 4800031400 x 47400
NA-G11X9013200 x 10800012400 x 106800
NA-H28X143 33600 x 17160031400 x 170400
NA-J34X176 40800 x 21120039600 x 210000
NA-K40X143 48000 x 17160047400 x 170400
NA-Legal8.5X1410200 x 16800 9240 x 15480

- Foldouts
Small11X1413200 x 1680012744 x 15480
NA-B11X17(same as NA-B above)

These page sizes are for the portrait orientation.

Table 2 Layout Attributes

DefaultNon-Basic
AttributesBasic ValuesValuesValues

Page Dimensions*CARA NA A-F,CARA NA-AARA NA G-K
CARA NA-LegalISO A0
ISO A4-A111" roll
Small Foldout

Medium-type*NA A-F, NA-ANA G-K
(Nominal pageNA-Legal,ISO A0
size)ISO A4-A111" roll
Small Foldout

* see Table 1

6.3.4 OriginalImage

An OriginalImage is a constituent constraint which defines a lowest level frame used for laying out the original image of an engineering drawing, illustration or other raster scanned image. This frame contains a single SpecificBlock containing a raster graphics content portion.

This type of frame has a fixed position that is equal to the origin of the page. The vertical and horizontal dimensions of this frame are fixed and equal to the maximum size that can be achieved for the position within the area of the page.

6.3.5 RevisionAnnotation

A RevisionAnnotation is a constituent constraint which defines a lowest level frame used for laying out the revision annotation associated with the original image. This frame contains a single SpecificBlock containing either a character content portion, a raster graphics content portion or a geometric graphics content portion.

This type of frame has a fixed position and dimensions. This provision provides for the capability of positioning of revision annotation anywhere on the page. Registration of revision annotation over a portion of the original image, as in revision artwork, is accomplished using this capability.

6.3.6 SpecificBlock

A SpecificBlock is a constituent constraint which defines a basic layout object used to position and image the content portions associated with either an OriginalImage or RevisionAnnotation frame.

The position of the block is fixed and defaults to the origin of the superior frame. The dimensions default to the maximum size that can be achieved for the position within the area of the superior frame.

6.4 Document Layout Characteristics

This DAP provides for only formatted documents. Hence, no provision is made for constraining the document layout process other than as implied in the formatted documents supported by this DAP. In particular, these formatted documents are characterized by the following:

o Documents containing either basic pages or composite pages, exclusively;

o Documents may contain one or more pages;

o Pages may vary by orientation within a document;

o Each page contains a single raster graphics content portion, representing the original image;

o In the case of composite pages, each page may additionally contain one or more character, raster graphics or geometric graphics content portions representing revision annotation;

o In the case of composite pages, content is positioned within fixed position and dimension frames.

6.5 Content Layout and Imaging Control

A document is modelled as an original image with optional revision annotation. The original image is represented by raster graphics content portions, as specified in ISO 8613-7. The revision annotation may be represented by either character, raster graphics or geometric graphics content portions, as specified in ISO 8613-6, ISO 8613-7 and ISO 8613-8, respectively.

The content architectures that may be specified using the attribute *Content architecture class* are formatted character, formatted processable raster graphics and formatted processable geometric graphics. The formatted processable raster graphics is the only content that may be specified as the default in the document profile.

6.5.1 Raster Graphics Content Architecture

The formatted processable raster graphics content is the primary architecture class supported by this DAP. This is the only default architecture class that can be specified in the document profile. Other architecture classes are supported only to assist in revision annotation of the scanned raster image represented by the formatted processable raster graphics content.

When using raster graphics content, only one content portion may be associated with a basic page. In the case of the composite page, only one content portion can be associated with the original image or a given revision annotation.

6.5.2 Raster Graphics Encoding Methods

Three encoding methods, CCITT T.6 (untiled), Tiled, and Bitmap are supported by this profile as basic values. Neither the CCITT T.4 one dimensional method nor the CCITT T.4 two dimensional method is supported.

The CCITT Recommendation T.6 Group 4 compression algorithm shall be used in all cases, tiled and untiled, 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.

When the coding type is specified as CCITT T.6, the encoding must be compressed and the "code extension" technique in T.6 encoding is not used. That is to say that uncompressed data cannot occur within a T.6 encoded data stream. Thus, there is no need for a default value of the Compression attribute and this attribute will not appear in the description of the raster content.

In a content portion, it is required that the Number-of-pels-per-line parameter of the Coding-attributes attribute be specified. The use of the Number-of-lines parameter is optional. 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 ISO 8613-7 Addendum.

See table D.1, Annex D, for a tabulated list of the attributes and their basic, default, and non-basic values.

6.5.3 Raster Presentation

Raster presentation is controlled by the presentation attributes specified in ISO 8613-7. This DAP provides for additional constraints on these presentation attributes as specified below.

The basic Pel-path values supported by this profile are 0 and 90 degrees. The Pel-path values of 180 and 270 degrees are non-basic.

The basic Line-progression value supported by this profile is 270 degrees. The Line-progression value of 90 degrees is non-basic.

The basic Pel-spacing values 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. Values for Pel-spacing other than these ratios are non-basic, i.e., 5, 3, 2, and 1 BMU. These correspond 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 D.2, Annex D, for a tabulated list of the attributes and their basic, default, and non-basic values.

6.5.4 Character Content

The formatted character content is permitted in this DAP only for use in revision annotation of the original raster scanned image.

The specification in a document of a non-basic feature by a presentation attribute or control function must be indicated in the document profile.

6.5.4.1 Character Content Architecture Class

When using character content, only one content portion may be associated with a basic component. The content information in a content portion must be present.

6.5.4.2 Character Repertoires

The basic character set supported by this profile is the primary character set of ISO 8859-1. This must be designated to the G0 set and invoked to the GL. Any other graphic character set which is registered in accordance with ISO 2375 may be designated and invoked at any point in the document provided its use is announced in the document profile as a non-basic value using the character presentation attribute *Graphic character sets*. No locking shift functions are specified in this presentation attribute. The default graphic character sets which apply to the content portions within a document can be specified in the document profile using the presentation attribute *graphic character sets*.

Using code extension techniques, the graphic character sets designated and/or invoked at the beginning of a content portion containing character content are specified using the presentation attribute *graphics character sets*.

If the character set defined in ISO 6937-2 is designated and invoked, then the use of any sub-repertoire registered according to ISO 7350 may be specified. All sub-repertoires are non-basic and their use must be indicated in the document profile.

The code extension techniques specified in ISO 2022 may be used subject to the following restrictions:

o G0 set: only the primary character sets of ISO 6937-2, ISO 8859-X (where ISO 8859-X corresponds to any finalized part of ISO 8859) and a version of ISO 646 may be designated for this set; these character sets may only be invoked in GL;

o G1, G2, G3 sets: no restrictions are placed on the character sets that may be designated for these sets; these sets may only be invoked in GR;

o The locking and single shift functions allowed should be restricted to the following:

LS0 for the G0 set

LS1R for the G1 set

LS2R for the G2 set

LS3R for the G3 set

SS2

SS3;

o When specifying the presentation attribute *Graphic character sets*, it is necessary to invoke character sets for both GL and GR. Thus an allowed character set must be designated into G0, as specified above, and invoked into GR. It is also necessary to invoke a character set into GR which has been designated into G1, G2 or G3 sets.

o The empty set should be designated and invoked in GR if no other specific set is invoked into GR;

The announcement and encoding of these functions are to be as specified in ISO 2022.

6.5.4.3 Line Spacing

Any value of line spacing may be specified. Values of 150, 200, 300 and 400 BMUs are basic; the use of any other value in a document is non-basic and must be indicated in the document profile. The line spacing may be specified at the beginning of the content associated with a basic component using the presentation attribute "Line spacing". The value may be changed anywhere within the content portion using the control functions SVS and SLS.

6.5.4.4 Character Spacing

Any value of character spacing may be specified. Values greater than or equal to 100 are basic; the use of any other value in a document is non-basic and must be indicated in the document profile. The character spacing may be specified at the beginning of the content associated with a basic component using the attribute "Character spacing". The value may be changed anywhere within a content portion using the control functions SHS or SCS.

6.5.4.5 Character Path and Line Progression

Both horizontal and vertical writing directions may be used within a revision annotation. In the case of horizontal writing, the characters progress either from left to right or from right to left across the page and the line progression is from the top of the page to the bottom. In the case of vertical writing, the characters progress from the top of the page to the bottom and the line progression is from the right to the left. The use of these writing directions is used to provide for various revision annotation requirements. The values of character path and line progression may be specified at the beginning of the content associated with a basic component using the presentation attributes *Character path* and *Line progression*, respectively. These values cannot be changed within a content portion.

6.5.4.6 Character Orientation

The character orientation may be specified as 0 or 90 degrees depending on whether vertical or horizontal writing is used. When vertical writing is used, characters are normally orientated at 0 degrees. When horizontal writing is used, characters may be orientated at 0 or 90 degrees. A value of 0 degrees is basic; a value of 90 degrees is non-basic and its use in a document must be indicated in the document profile. The value of the character orientation is specified at the beginning of the content associated with a basic component by the presentation attribute *Character orientation*. This value cannot be changed within the content.

6.5.4.7 Emphasis

The following modes of emphasising graphic characters may be distinguished:

o normal rendition;

o normal intensity;

o increased intensity (bold);

o italicised;

o not italicised;

o underlined;

o doubly underlined;

o not underlined;

o crossed-out;

o not crossed-out.

All the above modes of emphasis are basic. The mode of emphasis may be specified at the beginning of the content associated with a basic component using the presentation attribute *Graphic rendition*. The mode may be changed anywhere within the content using the control function SGR. The mode of emphasis remains in effect within the content associated with a basic component until changed into a mutually exclusive mode or by the specification of *normal rendition*. Mutually exclusive modes are normal/increased intensity, italicized/not italicized, underlined/not underlined and crossed out/not crossed-out. One mode from each mutually exclusive set may be in operation at any point in the document content. Normal rendition cancels the effect of all methods of emphasis that are currently in operation and specifies that the text should be displayed in accordance with the default rendition parameters set for the presentation device. Thus, for example, if it is required to ensure that the content is not underlined, then it is necessary to explicitly specify that underlined is not to be used.

6.5.4.8 Tabulation

Tabulation stop positions may be specified at any character position along the character path. Each stop is specified by means of the following:

o The tabulation position relative to the margin position in the direction opposite to the character path;

o An optional alignment qualifier that specifies the type of alignment to be used at the designated tabulation position. The type may be specified as one of the following:

- start aligned;

- end aligned;

- centered;

- aligned around.

These alignment qualifiers are defined in ISO 8613-6. If the alignment qualifier is not explicitly specified, then it is assumed that start aligned is to be used. Only one set of tabulation stops can be specified to be applicable to the content associated with a basic component. No limit is placed on the number of tabulation stops that can be specified within a given set. The set of tabulation stop positions associated with the content of a basic component are specified using the presentation attribute *Line layout table*. Tabulation stop positions are invoked within the content using the control function STAB.

6.5.4.9 Alignment

This feature is concerned with how the first and last characters on each line of character content is to be laid out during the formatting process. The following values of alignment may be specified:

- start aligned;
- end aligned;
- centred;
- justified.

The semantics of these values are as defined in ISO 8613-6. The presentation attribute *Alignment* is used to specify the alignment that is applicable to the content associated with a basic component. The alignment value cannot be changed within a content portion.

6.5.4.10 Fonts

Any number of fonts may used within a document. The fonts used in a particular document are specified in the document profile using the attribute *Font list*. Further information concerning the specification of font references in the document profile is given in Annex B. The fonts that may be used within the content associated with each basic component are specified by the presentation attribute *Character fonts*. Up to 10 fonts taken from the list specified by the attribute *Font list* may be specified by the attribute *Character fonts*. The font to be used at the start of the content associated with a basic component is specified using the attribute *Graphic rendition*. The fonts used within the content may be changed using the control function *SGR*.

6.5.4.11 Reverse Character Strings

Bi-directional writing is supported by this profile. Hence, a string of characters in a content portion associated with a basic component may be specified to be imaged in the reverse direction of the immediately preceding character string. Such strings can be specified by the control function *SRS* as defined in ISO 8613-6. This control function is provided for cases in which the text belongs to different languages and the character content is written, for example, from left to right or from right to left within the same line of characters, dependent upon the language and/or character set being used.

NOTE - The use of this control function cannot be indicated in the document profile. Thus it is intended that implementations should ignore this control function when reverse character string layout and presentation is not supported.

6.5.4.12 Kerning Offset and Pairwise Kerning

A kerning offset value for the content associated with a basic component may be specified using the presentation attribute *Kerning offset*. It is necessary to specify such a value when certain fonts are invoked to ensure that no part of character images are positioned outside the boundary of the available area. Kerning pairs can be specified with the presentation attribute *Pairwise kerning*.

6.5.4.13 Superscripts and Subscripts

Superscripts and subscripts may be specified anywhere with in the content associated with a basic component by using the control functions *PLU* and *PLD*. The use of these control function shall be in accordance with ISO 8613-6.

6.5.4.14 Substitution of Characters

The control function *SUB* is provided to represent characters produced by a local system that cannot be represented by a character within a character set supported by this profile.

6.5.4.15 Use of Control Functions

The following is a list of all the control functions and parameter values (where applicable) may be specified in character content:

SHS -set horizontal spacing

SCS -set character spacing

S V S
set vertical spacing

SLS -set line spacing

SGR -set graphic rendition

STAB - selective tabulation (allowed parameter values: any)

SRS -start reverse string (allowed parameters: any)

PLD -partial line down

PLU -partial line up

SUB - substitute character

SP -space

CR - carriage return

LF - line feed

-code extension control functions (see 6.5.1.4)

6.5.5 Geometric Graphics Content

The formatted processable graphics content is permitted in this DAP only for use in revision annotation of the original raster scanned image. Such geometric graphics content is encoded as CGM (Computer Graphics Metafile) metafiles in accordance with ISO 8632 and ISO 8613-8. Each CGM figure must consist of a single picture only.

Further information concerning the specification of geometric graphics content information is given in Annex B.

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 sequence of integers is a set of indices representing the octet offsets to the beginning of the respective tiles, 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.

6.7 Document Management Features

Every document interchanged in accordance with this DAP must include a document profile containing information which relates to the document as a whole. The document profile used in this DAP 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 D.3, Annex D.

7 Specification of Constituent Constraints

7.1 Document Profile Constraints

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(FC,"           ASN.1{2 8 2 6 0}") -- Character formatted --
DEFINE(FPR,"          ASN.1{2 8 2 7 2}") -- Raster formatted processable --
DEFINE(FPG,"          ASN.1{2 8 2 8 0}") -- Graphics formatted processable --

-- Macro defining permissible code extension announcers  --

DEFINE(CDEXTEN, "  ESC 02/00 05/00,        -- LS0 --
            [ESC 02/00 05/03],       -- LSR1 --
            [ESC 02/00 05/05],       -- LSR2 --
            [ESC 02/00 05/07],       -- LSR3 --
            [ESC 02/00 05/10],       -- SS2 --
            [ESC 02/00 05/11]        -- SS3 --
")


-- Macro defining permitted graphic renditions  --

DEFINE(GRAPHICRENDITIONS "
           {'cancel' | 'increased-intensity'
            | 'italicised' | 'underlined' | 'crossed-out'
            | 'primary-font' | 'first-alternative-font'
            | 'second-alternative-font' | 'third-alternative-font'
            | 'fourth-alternative-font' | 'fifth-alternative-font'
            | 'sixth-alternative-font' | 'seventh-alternative-font'
            | 'eighth-alternative-font' | 'ninth-alternative-font'
            | 'doubly-underlined' | 'normal-intensity'
            | 'not-italicised' | 'not-underlined' | 'not-crossed-out'}...
")


-- Macros defining final character for designation --

DEFINE(FCORE,  "04/02 -- the 94 characters of the IRV of ISO 646
                (revised 1990) (i.e ASCII) --")

DEFINE(F646,   "-- a final character designating any version of ISO 646
            except 04/02 --")

DEFINE(F94S,   "-- a final character designating any registered 94 single
            byte graphic character set --")

DEFINE(F94M,   "-- a final character designating any registered 94 multi
            byte graphic character set --")

DEFINE(F96S,   "-- a final character designating any registered 96 single
            byte graphic character set --")

DEFINE(F96M,   "-- a final character designating any registered 96 multi
            byte graphic character set --")
```

```
DEFINE(FEMPTY, "07/14  -- the empty set --")


-- Macros defining designation sequences --

DEFINE(DEG-CORE-GO,  "ESC 02/08 $FCORE")
                    -- Designate the 94 characters of the IRV of
                       ISO 646 to G0 --

DEFINE(DEG-646-GO,   "ESC 02/08 $F646")
                    -- Designate any version of ISO 646, except 04/02,
                       to GO --

DEFINE(DEG-ANY-G1,   "{ESC 02/09 $F94S
                      |ESC 02/04 02/09 $F94M
                      |ESC 02/13 $F96S
                      |ESC 02/04 02/13 $F96M}")
                    -- Designate any character set to G1 --

DEFINE(DEG-ANY-G2,   "{ESC 02/10 $F94S
                      |ESC 02/04 02/10 $F94M
                      |ESC 02/14 $F96S
                      |ESC 02/04 02/14 $F96M}")
                    -- Designate any character set to G2 --

DEFINE(DEG-ANY-G3,   "{ESC 02/11 $F94S
                      |ESC 02/04 02/11 $F94M
                      |ESC 02/15 $F96S
                      |ESC 02/04 02/15 $F96M}")
                    -- Designate any character set to G3 --

DEFINE(DEG-EMPTY-G1, "ESC 02/09 $FEMPTY")
                    -- Designate the empty set to G1 --


-- Macros defining shift functions --

DEFINE(LSO,    "00/15")        -- locking shift invoking G0 to GL --

DEFINE(LS1R,   "ESC 07/14")    -- locking shift invoking G1 to GR --

DEFINE(LS2R,   "ESC 07/13")    -- locking shift invoking G2 to GR --

DEFINE(LS3R,   "ESC 07/14")    -- locking shift invoking G3 to GR --

DEFINE(SS2,    "08/14")        -- single shift invoking G2 to GL --

DEFINE(SS3,    "08/15")        -- single shift invoking G3 to GL --
```

-- Macro defining permissible graphic character sets. --

```
DEFINE(PERMIT-GRCHAR,  "  {$DEG-CORE-GO $LS0
                          |$DEG-646-G0 $LS0},
                          {$DEG-ANY-G1 $LS1R
                           |$DEG-ANY-G2 $LS2R
                           |$DEG-ANY-G3 $LS3R}...
                          |{$DEG-EMPTY-G1 $LS1R}  ")
```

-- Macro defining default graphic character sets --

```
DEFINE(DAP-DEFAULT-GRCHAR, "$PERMIT-GRCHAR")
```

-- Macro defining basic character sets. Note that this macro is defined for clarification of the specification and is not to be used in any other part of this DAP specification. --

```
DEFINE(BASIC-GRCHAR, "  $DEG-CORE-G0 $LSO,
                        $DEG-EMPTY-G1 $LS1R  ")
```

-- Macro defining non-basic character sets --

```
DEFINE(NON-BASIC-GRCHAR,  "  {$DEG-646-G0
                              |$DEG-ANY-G1
                              |$DEG-ANY-G2
                              |$DEG-ANY-G3}... ")
```

-- Macro defining character sets used in document profile attributes --

```
DEFINE(PROFCHAR, "  {$DEG-CORE-G0 $LS0,
                    |$DEG-646-G0 $LS0},
                   {$DEG-ANY-G1 $LS1R
                    |$DEG-ANY-G2 $LS2R
                    |$DEG-ANY-G3 $LS3R
                    |$DEG-EMPTY-G1 $LS1R}  ")
```

-- Macro defining comments character sets --

```
DEFINE(COMCHAR, " {ESC 02/00 05/00,          -- LS0 --
              [ESC 02/00 05/03],       -- LSR1 --
              [ESC 02/00 05/05],       -- LSR2 --
              [ESC 02/00 05/07],       -- LSR3 --
              [ESC 02/00 05/10],       -- SS2 --
              [ESC 02/00 05/11]},       -- SS3 --
             {$DEG-CORE-G0 [LS0]
```

```
           |$DEG-646-G0 [LS0]},
          {{$DEG-ANY-G1 [$LS1R]
           |$DEG-ANY-G2 [$LS2R]
           |$DEG-ANY-G3 [$LS3R]}...
           |$DEG-EMPTY-G1 $LS1R}}  ")
```

```
-- Macro defining character sets used for alternative representation --

DEFINE(ALTCHAR, "$PROFCHAR")
```

7.1.2 Constituent Constraints

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            {-- See clause 8 for a definition of the permitted values for
                                               this attribute. --},

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}},

--  Any permitted medium type.  Both landscape and portrait may be specified.  --

       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                 {-- See clause 8 for a definition of the permitted values for
                                                 this attribute. --},
REQ     ODA-version

        {REQ #standard-or-recommendation         {'ISO 8613'},
        REQ #publication-date                    {"1989-07-04"}},

-- Non-basic document characteristics --

PERM    #Profile-character-sets                  {$PROFCHAR},
PERM    #Comments-character-sets                 {$COMCHAR},
PERM    #Alternative-representation-character-sets       {$ALTCHAR},
PERM    #Page-dimensions                         {PMUL {$NonBasicPageDimensions}},
PERM    #Medium-types                            {PMUL{
        REQ     #nominal-page-size               {$NominalPageSizes},
        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},


PERM    Presentation-features    {
        PERM    #character-presentation-features         {
          PERM #character-orientation            {'90-degrees'},
          PERM #character-path                   {'90-degrees', '180-degrees', '270-degrees'},
          PERM #graphic-character-sets           {ANY_EXCEPT $BASIC-GRCHAR},
          PERM #graphic-character-subrepertoire {ANY_VALUE},
          PERM #Line-spacing                     {ANY_EXCEPT 150,200,300,400},
          PERM #Line-progression                 {'90-degrees'}},


-- Additional document characteristics --

PERM    Fonts-list          {PMUL  {REQ #font-identifier {ANY_VALUE},
                                   REQ #font-reference {ANY_VALUE}}},
-- The format of the parameter "font-reference" is defined in annex B --


-- Document management attributes --

REQ  Document-reference                          {ANY_VALUE}};
```

7.2 Logical Constituent Constraints

No logical constituents applicable in this clause.

7.3 Layout Constituent Constraints

7.3.1 Macro Definitions

```
DEFINE(CHAR,"          CONTENT_ID_OF(CHARACTER)")
DEFINE(RAST," CONTENT_ID_OF(RASTER)")
DEFINE(GEOM,"          CONTENT_ID_OF(GEOMETRIC)")
```

7.3.2 Factor Constraints

```
FACTOR:        ANY-LAYOUT             {

SPECIFIC:
PERM   Object-type                    {VIRTUAL},
PERM   Object-identifier              {ANY_VALUE},
PERM   Subordinates                   {VIRTUAL},
PERM   User-visible-name              {ANY_VALUE},
PERM   User-readable-comment          {ANY_VALUE},
}
```

7.3.3 Constituent Constraints

7.3.3.1 LayoutDocumentRoot

```
LayoutDocumentRoot           : ANY-LAYOUT          {

SPECIFIC:
REQ    Object-type                          { 'document-layout-root'},
REQ    Subordinates                         { {SUB_ID_OF(BasicPage)+}    |
                                            {SUB_ID_OF(CompositePage)+}
}
```

7.3.3.2 BasicPage

```
BasicPage                     : ANY-LAYOUT          {

SPECIFIC:
REQ    Object-type                          { 'basic-page'},

PERM   Page-position                        {ANY_VALUE},
PERM   Dimensions                           {REQ #horizontal-dimension
                                                 {REQ #fixed-dimension {$BasicPageDimensions
                                                 |  $NonBasicPageDimensions}},
                                            REQ #vertical-dimension
                                                 {REQ #fixed-dimension {$BasicPageDimensions
                                                 |  $NonBasicPageDimensions}}},
PERM   Medium-type                          {REQ #nominal-page-size {NON_BASIC},
                                            REQ #side-of-sheet {ANY_VALUE}},
PERM   Application-comments                 {PMUL {INTEGERS}},
                                   -- See clause 8.2 --
PERM   Content-portions                     { CONTENT_ID_OF(RASTER)},

PERM   Presentation-style                   {STYLE_ID_OF(PStyle},
PERM   Presentation-attributes              {
       PERM   #raster-attributes            {
              PERM  #Pel-path               {ANY_VALUE},
              PERM  #Line-progression       {ANY_VALUE},
              PERM  #Pel-spacing            {ANY_VALUE},
              DIS   #Compression            {ANY_VALUE},
              PERM  #Clipping               {ANY_VALUE}}},
}
```

7.3.3.3 CompositePage

```
CompositePage                 : ANY-LAYOUT          {

SPECIFIC:
REQ    Object-type                          {'composite-page'},
PERM   Dimensions                           {REQ #horizontal-dimension
                                                 {REQ #fixed-dimension {$BasicPageDimensions
                                                 |  $NonBasicPageDimensions}},
                                            REQ #vertical-dimension
                                                 {REQ #fixed-dimension {$BasicPageDimensions
                                                 | $NonBasicPageDimensions}}},
PERM   Page-position                        {ANY_VALUE},
PERM   Medium-type                          {REQ #nominal-page-size {NON_BASIC},
                                            REQ #side-of-sheet {ANY_VALUE}},
PERM   Imaging-order                        {ANY_VALUE},
PERM   Application-comments                 {ANY_VALUE},
                                   -- See clause 8.2 --
}
```

7.3.3.4 OriginalImage

```
OriginalImage                 : ANY-LAYOUT        {

SPECIFIC:
REQ    Object-type                           {'frame'},
REQ    Subordinates                     {SUB_ID_OF(SpecificBlock)},
PERM   Application-comments                  {ANY_VALUE},
                                        -- See clause 8.2 --
PERM   Presentation-style                    {STYLE_ID_OF(PStyle}
}
```

7.3.3.5 RevisionAnnotation

```
RevisionAnnotation            : ANY-LAYOUT        {

SPECIFIC:
REQ    Object-type                           {'frame'},
REQ    Subordinates                     {SUB_ID_OF(SpecificBlock)},
PERM   Position                              {REQ #fixed-position   {
                                                   REQ #horizontal{ANY_INTEGER}
                                                   REQ #vertical  {ANY_INTEGER}}},
PERM   Dimensions                            {REQ #horizontal       {
                                                   REQ #fixed     {ANY_INTEGER}},
                                             REQ #vertical    {
                                                   REQ #fixed     {ANY_INTEGER}}}
                                             },
PERM   Application-comments                  {ANY_VALUE},
                                        -- See clause 8.2 --
PERM   Presentation-style                    {STYLE_ID_OF(PStyle1)   |   STYLE_ID_OF(PStyle2)   |
                                             STYLE_ID_OF(PStyle},

}
```

7.3.3.6 SpecificBlock

```
SpecificBlock                 {

SPECIFIC:
REQ    Object-type                           {'block'},
REQ    Object-identifier                     {ANY_VALUE},
REQ    Content-portions                      {$CHAR | $RAST | $GEOM},
PERM   Content-architecture-class            {$FC | $FPR | $FPG},
PERM   Transparency                          {'transparent' | 'opaque'},
PERM   Colour                                {'colourless' | 'white'},
PERM   User-readable-comments               {ANY_STRING},
PERM   User-visible-name                    {ANY-STRING}
PERM   Application-comments                 {ANY_VALUE},
                                    -- See clause 8.2 --
PERM   Presentation-attributes              {

  CASE Content-portions OF {

$CHAR:
       PERM  #character-attributes          {
             PERM  #alignment               {ANY_VALUE},
             PERM  #character-spacing       {ANY_VALUE},
             PERM  #character-fonts         {ANY_VALUE},
             PERM  #character-orientation   {'0-degrees' | '90-degrees'},
             PERM  #character-path          {'0-degrees'    |    '90-degrees'    |    '180-degrees'    |
                                            '270-degrees'},
             PERM  #code-extension-announcers {$CDEXTAN},
             PERM  #graphic-character-sets {$PERMIT-GRCHAR},
             PERM  #graphic-character-subrepertoire {$GRAPHICRENDITIONS},
             PERM  #graphic-rendition       {$GRAPHICRENDITIONS},
             PERM  #kerning-offset          {ANY_VALUE},
             PERM  #pairwise-kerning        {ANY_VALUE},
             PERM  #line-progression        {'90-degrees' | '270-degrees'},
             PERM  #line-spacing            {ANY_VALUE},
             PERM  #line-layout-table       {ANY_VALUE},
},

$RAST:
       PERM  #raster-graphics-attributes              {
             PERM  #Pel-path                {ANY_VALUE},
             PERM  #Line-progression        {ANY_VALUE},
             PERM  #Pel-spacing             {ANY_VALUE},
             DIS   #Compression             {ANY_VALUE},
             PERM  #Clipping                {ANY_VALUE}}},

$GEOM:
       PERM  #geometric-graphics-attributes {
             PERM  #picture-dimensions      {ANY_VALUE},
             PERM  #picture-orientation     {ANY_VALUE},
             PERM  #text-rendition          {PERM #fonts-list {ANY_VALUE},
                                            PERM #character-set-lists {ANY_VALUE}}}
}
```

7.4 Layout Style Constraints

No layout style constraints applicable in this clause.

7.5 Presentation Style Constraints

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},
 ")
```

7.5.2 Factor Constraints

No factor constraints applicable to this clause.

7.5.3 Constituent Constraints

7.5.3.1 PStyle

```
PStyle  :ANY-PRESENTATION-STYLE    {


REQ    Presentation-style-identifier     {ANY_VALUE},
PERM   User-readable-comments            {ANY_VALUE},
PERM   User-visible-name                 {ANY_VALUE},
PERM   Presentation-attributes           {$R-Pres-Attr},
}
```

7.6 Content Portion Constraints

7.6.1 Macro Definitions

No macro definitions are applicable to this clause.

7.6.2 Factor Constraints

No factor constraints are applicable to this clause.

7.6.3 Content Portion Constraints

7.6.3.1 Character Content Portion

```
{
PERM  Content-identifier-layout          {CONTENT_ID_OF(CHARACTER)},
PERM  Type-of-coding                     {ASN.1{2 8 2 6 0}},
PERM  Alternative-representation         {ANY_STRING},
PERM  Content-information
          {CHARACTER, {#STAB  {ANY_VALUE}
                      |#SHS   {0,1,2,3,4}
                      |#SGR   {$GRAPHICRENDITIONS}
                      |#SVS   {0 1 2 4}
                      |#SLS   {ANY_VALUE}
                      |#SCS   {ANY_VALUE}
                      |#SRS   {ANY_VALUE}
                      |#VPR   {ANY_VALUE}
                      |#VPB   {ANY_VALUE}
                      |#CR
                      |#LF
                      |#PLD
                      |#PLU
                      |#SP
                      |#SUB
                      |#$LS0
                      |#$LS1R
                      |#$LS2R
                      |#$LS3R
                      |#$SS2
                      |#$SS3
                      |#$DEG-CORE-G0
                      |#$DEG-646-G0
                      |#$DEG-ANY-G1
                      |#$DEG-ANY-G2
                      |#$DEG-ANY-G3
                      |#$DEG-EMPTY-G1
                      }...}
}
```

7.6.3.2 Raster Graphics Content Portion

```
{
PERM  Content-identifier-layout          {CONTENT_ID_OF(RASTER)},
PERM  Coding-attributes                  {
      PERM  Number-of-lines                        {>0},
      REQ   Number-of-pels-per-line      {>=0},
      PERM  Type-of-coding               {ASN.1{2 8 3 7 0}  -- T.6 encoding --
```

```
                                                              |ASN.1{2 8 3 7 3} -- bitmap encoding --
                                                              | ASN.1{2 8 3 7 5} -- tiled encoding --},
        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   Alternative-representation                {ANY_STRING},
PERM   Content-information                       {RASTER}
}
```

7.6.3.3 Geometric Graphics Content Portion

```
{
PERM   Content-identifier-layout                 {CONTENT_ID_OF(GEOMETRIC)},
PERM   Type-of-coding                            {ASN.1{2 8 3 8 0}},
PERM   Alternative-representation                {ANY_VALUE},
PERM   Content-information                       {GEOMETRIC}
}
```

7.7 Additional Usage Constraints

No other usage constraints are currently defined.

8 Interchange Format

Two interchange formats are supported by this profile. The Interchange Format Class B can be used by applications requiring a binary encoding based on ASN.1. The Interchange Format SDIF can be used by applications requiring a SGML based clear text encoding. This latter interchange format is an SGML application, called Office Document Language (ODL). For the purposes of interchange, the ODL ENTITIES are placed in an ASN.1 wrapper, as defined by SDIF. Each encoding form has inherent advantages. Conversion of document encoded in one interchange format into the other should not produce the loss of semantic document information.

8.1 Interchange Format Class B

8.1.1 Interchange Format

The value of the document profile attribute "interchange format" for this interchange format is "if-b". This form of ODIF is 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.

8.1.2 DAP Identifier

The value for the document profile attribute "Document application profile" for this interchange format is represented by the following object identifier.

Editor's Note - To be supplied.

8.1.3 Encoding of Application Comments

ISO 8613-5 define the encoding of the attribute Application Comments as an octet string. This DAP 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
```

8.2 Interchange Format SDIF

8.2.1 Interchange Format

The document profile attribute "Interchange format" does not apply for this interchange format. This form of ODIF is defined in Annex E of ISO 8613-5. In addition, ISO 8613-6, -7, and -8 contain additional specifications for this form of ODIF.

8.2.2 DAP Identifier

The value for this attribute "Document application profile" for this interchange format is represented by the following object identifier.

Editor's Note - To be supplied.

8.2.3 Encoding of Application Comments

The encoding of the attribute "Application comments" is defined in a data stream conforming to this profile with the following DTD definition:

```
<!DOCTYPE odaac [
<!--
<!DOCTYPE doc PUBLIC "-//USA-OIW//SGML ENCODED ODA APPLICATION
COMMENTS//EN"> -->

<!ELEMENT objappc   - O (#PCDATA)>
        <!-- Object application comment -->
]>
```

Editor's Note - The above DTD definitions must be verified by a SGML expert and modified as required.

8.3 Encoding of Raster Content Information

The encoding of raster content information in the bitmap encoding scheme is that specified in clause 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.

Annex A (normative)

Amendments and Corrigenda

A.1 Amendments

A.1.1 Amendments to the base standard

The amendments applicable to this DAP includes the ISO 8613 - Amendment 1: 1990. This amendment includes text to be included in ISO 8613-1 as the following annexes:

- o Annex E: Use of ISO/IEC 10021 (MOTIS) to interchange documents conforming to ISO 8613;
- o Annex F: Document application profile proforma and notation;
- o Annex G: Conformance testing methodology;
- o Annex H: Recording of documents conforming to ISO 8613 on flexible disk cartridges conforming to ISO 9293.

In addition, this amendment addresses the inclusion of the ISO 8613 Technical Corrigenda 1.

This DAP does not include the following features of the amendment:

- o Addendum on security;
- o Addendum on styles;
- o Addendum on alternative representation.

Additionally, this DAP includes features from the Draft Addendum (DAD) to ISO 8613-7, Tiled Raster Graphics Addendum, dated January 1990. The DAD dated has been balloted and the disposition of all comments has been processed by ISO/IEC JTC1/SC 18/WG5. The document was distributed as a CCITT Study Group VIII document (CCITT/SGVIII/Q.27). A new ISO 8613-7 will be issued after the Colour Addendum is incorporated which is anticipated to be in March 1991.

A.2 Corrigenda

A.2.1 Corrigenda to this DAP

The previous version of this document (September 1990) incorporated all the changes approved at the September 1990 ODA SIG meeting. A summary of these changes are listed below:

- o A technical change to add an option for using SGML/SDIF based data stream encoding. This required changes to the following clauses: 0, 2, 5.1, 6.7 (table 5), 7.1.2.1 and 8;

o Editorial changes to clauses 0-6 resulting from a preliminary DOD review and comment period;

o Editorial changes to clauses 7 and 8 resulting from review and comments by the ODA SIG members at the September 1990 meeting.

This version of the document (December 1990) incorporates all the changes approved at the December 1990 ODA SIG meeting. A summary of these changes are listed below:

o Editorial changes to align with the approved proforma and notation for ODA DAPs;

o Editorial changes to align with the format for ODA DAP ISPs;

o Technical changes to provide support for revision annotation on the base, raster scanned image.

Annex B (informative)

Recommended Practices

B.1 Transfer methods for ODA

B.1.1 Conveyance of ODA over CCITT X.400-1984

This recommendation describes how ODA body parts are to be encoded for transmission over a CCITT X.400-1984 service.

An ODA body part is encoded as OdaBodyPart in the definition given below:

```
OdaBodyPart ::= SEQUENCE { OdaBodyPartParameters, OdaData }
OdaBodyPartParameters ::= SET {
        document-application-profile
                [0] IMPLICIT OBJECT IDENTIFER,
        document-architecture-class
                [1] IMPLICIT INTEGER {
                        formatted (0),
                        processable (1),
                        formatted-processable (2) }
OdaData ::=     SEQUENCE OF Interchange-Data-Element
```

NOTE - It is recommended to transfer an ODA document as a single body part with tag 12:

Oda [12] IMPLICIT OCTETSTRING

The content of the octet string is encoded as OdaBodyPart, defined above. However, this is out of the scope of this profile.

B.1.2 Conveyance of ODA over FTAM

This recommendation describes the FTAM Document Type to be used for minimal storage and transfer capabilities of ODA data streams. It is recognized that enhanced capabilities may at some point be added.

When using FTAM to transfer an ODA file, the FTAM-3, "ISO FTAM Unstructured Binary", document type should be specified. However, since files that do not contain ODA data streams can have the same document type, it is left up to the user of application programs that remotely access files using FTAM to know that a given file contains an ODA data stream.

B.1.3 Conveyance of ODA over DTAM

This recommendation provides for information concerning the interchange of ODA based documents with

DTAM (Document Transfer and Manipulation) protocols.

DTAM is defined in the T.430-Series of recommendations and is, like ODA, an integral part of the T.400-Series of CCITT Recommendations named *Open Document Architecture, Transfer and Manipulation.*

The T.520-Series of recommendations contain *Communication Application Profiles (CAP)*. Recommendation T.522 describes the Communication Application Profile BT1 for document bulk transfer. Recommendation T.522 is applicable for the Office Document Format Profile (FOD) published in this ISP.

> **NOTE** - The use of BT1 within the end-to-end oriented Telematic Services Telefax 4 and Teletex is described in Recommendation T.561, clause 7.1 and Recommendation T.562, clause 7.1.

B.1.4 Conveyance of ODA over flexible disks

The recommended method for interchanging ODA documents between systems by the exchange of magnetically recorded Flexible Disk Cartridges is by the use of an annex to ISO 8613-1 (to be published), *Recoding of Documents Conforming to ISO 8613 on Flexible Cartridges Conforming to ISO 9293.* This annex provides for recording each ODA document as a separate file as defined by ISO 9293, *Volume and File Structure of Flexible Disk Cartridges for Information Interchange.*

> **NOTE** - Document encoded in ODL can be stored such that each SGML ENTITY is recorded in a separate file or in the case of an SDIF encoding, the file can be stored in a single file.

B.2 Font reference

The recommended method for specifying a font reference is to be based on ISO 9541. Such a reference is to be specified by the following ASN.1 encoding.

```
Fonts-Reference     ::=     SET {

user-visible-name           (0) IMPLICIT Comment-String OPTIONAL,
user-readable-comment       (1) IMPLICIT Comment-String OPTIONAL,
reference-attributes        (2) IMPLICIT SEQUENCE OF SET {
        precedence-number           (0) IMPLICIT INTEGER OPTIONAL,
        attributes                  (1) IMPLICIT Font-Attribute-Set,
        user-readable-comment       (2) IMPLICIT Comment-String OPTIONAL }
}
```

Font sizes from 6 to 72 points (100 to 1200 BMU) are intended to be supported by implementation conforming to this informative recommendation. All other values of font sizes may additionally be supported, but implementations may also support using some form of "fallback".

The minimum font properties and values from ISO 9541 that are to be specified in a Font-Attribute-Set be those specified by the following document application profile notation.

```
Font-Attribute-Set   {

PERM   Fontname                    {ANY_VALUE},
PERM   Standardversion             {-- To be supplied --},
PERM   Dsnsource                   {ANY_VALUE},
PERM   Fontfamily                  {ANY_VALUE},
PERM   Posture                     {'upright' | 'italic-forward'},
PERM    Weight                              {'light' | 'medium' | 'bold'},
PERM    Propwidth                           {ANY_VALUE},
PERM    Glyphcomp                           {
        PERM #inclgyphcols                          {ANY_VALUE},
        PERM #exclgyphcols                          {ANY_VALUE},
        PERM #inclgyphs                             {ANY_VALUE},
        PERM #exclgyphs                             {ANY_VALUE} },
PERM    Dsnsize                             {ANY_VALUE},
PERM    Minsize                             {
        PERM #numerator                                       {100 .. 1200},
        PERM #denominator                           {1} },
PERM    Maxsize                             {
        PERM #numerator                                       {100 .. 1200},
        PERM #denominator                           {1} },
        -- BMU Units equivalent to range of 6..72 point sizes --
PERM    Dsngroup                                    {
        PERM #group-code                            {ANY_VALUE},
        PERM #subgroup-code                         {ANY_VALUE},
        PERM #specific-group-code                   {ANY_VALUE} },
PERM    Structure                           {ANY_VALUE},
PERM    Wrmodes                             {
        PERM #wrmodename                            {ANY_VALUE},
        PERM #nomescdir                                       {'0-degrees' | '90-degrees' | '180-degrees' | '270-degrees'},
        PERM #esclass                               {ANY_VALUE},
        PERM #avgescx                               {ANY_VALUE},
        PERM #avgescy                               {ANY_VALUE} }
}
```

B.3 ISO 8632 (CGM) constraints for this DAP

It is recommended that geometric graphics content information contain only those elements listed in this portion of the document, in addition to the constraints imposed by ISO 8613-8. It is believed that this subset of the CGM is sufficiently implemented to enable interworking of geometric graphics for application conforming this document application profile.

Where an element has parameters, recommended constraints on the values are given. The "--" symbol indicates that there is no recommended constraint.

Requirements in ISO 8632 and ISO 8613-8 concerning mandatory elements, parameters must be fulfilled.

B.3.1 Delimeter elements

No-Op	See Note 1
Begin Metafile	See Note 2
End Metafile	--
Begin Picture	See Note 2
Begin Picture Body	--
End Picture	--

B.3.2 Metafile descriptor elements

Metafile Version	1

Metafile Description	See Notes 2, 3
VDC Type	--
Integer Precision	8, 16
Real Precision	(0,9,23), (1,16,16)
Index Precision	16
Colour Precision	8, 16
Colour Index Precision	8, 16
Maximum Colour Index	--
Colour Value Extent	--
Metafile Element List	--
Metafile Defaults Replacement	See Note 4
Font List	--
Character Set List	See Note 5
Character Coding Announcer	basic-7-bit, basic-8-bit

B.3.3 Picture descriptor elements

Scaling Mode	See Note 6
Colour Selection Mode	--
Line Width Specification Mode	--
Marker Size Specification Mode	--
Edge Width Specification Mode	--
VDC Extent	--
Background Colour	--

B.3.4 Control elements

VDC Integer Precision	16, 32
VDC Real Precision	(0,9,23), (1,16,16)
Auxiliary Colour	--
Transparency	--
Clip Rectangle	--
Clip Indicator	--

B.3.5 Graphical primitive elements

Polyline	See Note 7
Disjoint Polyline	See Note 7
Polymarker	See Note 7
Text	See Note 2
Restricted Text	See Notes 2, 8
Append Text	See Notes 2, 8
Polygon	See Note 7
Polygon Set	See Note 7
Cell Array	See Note 9
Rectangle	--

Circle	--
Circular Arc 3 Point	--
Circular Arc 3 Point Close	--
Circular Arc Centre	--
1Circular Arc Centre Close	--
Ellipse	--
Elliptical Arc	--
Elliptical Arc Close	--

B.3.6 Attribute elements

Line Bundle Index	--
Line Type	--
Line Width	--
Line Colour	--
Marker Bundle Index	--
Marker Type	--
Marker Size	--
Marker Colour	--
Text Bundle Index	--
Text Font Index	--
Text Precision	--
Character Expansion Factor	--
Character Spacing	--
Text Colour	--
Character Height	--
Character Orientation	--
Text Path	--
Text Alignment	--
Character Set Index	--
Alternate Character Set Index	--
Fill Bundle Index	--
Interior Style	--
Fill Colour	--
Hatch Index	--
Pattern Index	1 .. 8
Edge Bundle Index	--
Edge Type	--
Edge Width	--
Edge Colour	--
Edge Visibility	--
Fill Reference Point	--
Pattern Table	See Notes 10, 11
Pattern Size	--
Colour Table Specification	See Notes 12, 13
Aspect Source Flags	--

B.3.7 External elements

Message	No action
Application Data	See Note 2

NOTE -

1. An arbitrary sequence of n octets. Where n=0, 1, .., 32767. The sequence of zero or more octets is for padding purposes.

2. Support will be provided for strings with a length up to 256 octets, except for data records which will support strings with a length up to 32767 octets.

3. The METAFILE DESCRIPTION string parameter will be used to include the sub-string "ISO FCG13" to label the content information as conforming to this agreement. In addition, generator of content are encouraged to append a sub-string that identifies the company and product that produced the CGM.

4. The METAFILE DEFAULTS REPLACEMENT element shall not be partitioned. No part of the element will be partitioned. Multiple occurrences of the MDR element may be used to avoid the need for partitioning. The MDR element must appear in the CGM to establish the defaults for TEXT PRECISION and any other elements whose defaults are different than those specified in ISO 8632-1 and -3.

5. The only character sets that may be specified are those specified for character content portions. Refer to Section 16.7.1, Document profile, for further detail on which character sets are supported by this document application profile. The default character set for geometric graphics content is the same as the default character set for character content architecture.

6. The Scale Factor parameter of SCALING MODE element is always a 32-bit floating point value, even when the REAL PRECISION has selected fixed point for other real numbers. It is not apparent in ISO 8632 what the precision of this floating point value is when fixed point has been selected. Its precision shall be (0,9,23).

7. The minimum support for the length of point lists is 1024 elements.

8. The complete restricted text string, including appended text, shall be included in a metafile conforming to this agreement. The complete restricted text string shall be scaled isotropically such that the specified aspect ratio for the text is not distorted and the string fits into the text extent parallelogram.

9. The minimum support for the length of colour lists parameter for the CELL ARRAY element is 1,048,576. This supports a 1024x1024 image.

10. The PATTERN TABLE element has an unspecified effect when it appears in a picture subsequent to any graphical primitives. The PATTERN TABLE element shall appear prior to any graphical primitive element to assure that interpreting systems without dynamic pattern update can render the intended effect.

11. The minimum support for the length of the Colour Array parameter for the PATTERN TABLE element is 2048. This will support 8 patterns of 16x16.

12. The COLOUR TABLE element has an unspecified effect when it appears in a picture subsequent to any graphical primitives. The COLOUR TABLE element shall appear prior to any graphical primitive elements to assure that interpreting systems without dynamic colour update can render the intended effect.

13. The minimum support for the length of the Colour List parameter in the COLOUR TABLE element is 61. This will support a 63 entry colour table.

B.4 Interoperability with SGML applications

The recommended method for the exchange of documents between Standard Generalized Markup

Language (ISO 8879, SGML) based systems and systems based on this ODA document application profile is by means of exchanging a document representation conforming to these agreements in an encoded form of the SGML language known as the Office Document Language (ODL). ODL is a standardized SGML application for representing documents conforming to the ODA base standard. Such a representation can be converted into the Office Document Interchange Format (ODIF) supported by this document application profile.

Annex C (informative)

References to Other Standards and Registers

CCITT Recommendation T.400 : 1988, Introduction to Document Architecture, Transfer and Manipulation;

CCITT Recommendation T.411 : 1988, Open Document Architecture (ODA) and Interchange Format: Introduction and General Principles;

CCITT Recommendation T.412 : 1988, Open Document Architecture (ODA) and Interchange Format: Document Structures;

CCITT Recommendation T.414 : 1988, Open Document Architecture (ODA) and Interchange Format: Document Profile;

CCITT Recommendation T.415 : 1988, Open Document Architecture (ODA) and Interchange Format: Open Document Interchange Format;

CCITT Recommendation T.416 : 1988, Open Document Architecture (ODA) and Interchange Format: Character Content Architecture;

CCITT Recommendation T.417 : 1988, Open Document Architecture (ODA) and Interchange Format: Raster Graphics Content Architecture;

CCITT Recommendation T.418 : 1988, Open Document Architecture (ODA) and Interchange Format: Geometric Graphics Content Architecture;

CCITT Recommendation T.502 : 1990, Document Application Profile PM-11 for the Interchange of Character Content Documents in Processable and Formatted Forms;

CCITT Recommendation T.503 : 1984, Document Application Profile for the Interchange of Group 4 Facsimile Documents;

CCITT Recommendation T.505 : 1990, Document Application Profile PM-26 for the Interchange of Enhanced Mixed Content Documents in Processable and Formatted Forms;

ISO 8571 : 1988, Information processing systems - Open Systems Interconnection - File transfer, access and management;

ISO 9070 : 1990, Information processing - SGML support facilities - Registration procedures for public owner identifiers;

ISO/TR 9573 : 1988, Information processing - SGML technical report - Techniques for using SGML;

ISO 10021 : (to be published), Information processing systems - Text communication - Message Oriented Text Interchange System;

ISP FOD11 : (to be published), Office document format profile for the interchange of basic function character content document in processable and formatted forms;

ISP FOD26 : (to be published), Office document format profile for the interchange of enhanced function mixed content documents in processable and formatted forms;

ISP FOD36 : (to be published), Office document format profile for the interchange of extended function mixed content documents in processable and formatted forms;

MIL-R-28002A : 1990, MILITARY SPECIFICATION, RASTER GRAPHICS REPRESENTATION IN BINARY FORMAT, REQUIREMENTS FOR.

Annex D (informative)

Supplementary Information on Attributes

Table D.1 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 D.2 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 BMU (200, 300)	4 BMU (300)	5,3,2,1 BMU
Clipping	Two Coord. Pairs (any non-negative integer, any non-negative integer)	(0,0), (N-1, L-1)	None

Table D.3 Document Profile Attributes

AttributeClassPermissible Values

Specific-layout-structurempresent

Presentation-stylesnmpresent

Document-characteristicsM

Document-architecture-classmformatted

Document-application-profilem{-- See clause 8 for a definition of the permitted values for this attribute. --}

Content-architecture-classesm{2 8 2 7 2}

Interchange-format-classmB

ODA-versionmISO 8613, 1989-07-04

Document-architecture-defaultsM

Content-architecture-classmformatted processable

Type-of-codingnmT.6 Encoding (default)
Tiled Encoding

Page-dimensionsnmSee list in table 1,
(Default value is NA-A,
9240 x 13200 BMU)

Medium-typesnmSee list in table 1,
(Default value is NA-A,
9240 x 13200 BMU)

Page-positionnmany coordinate pair
within page

Raster-gr-content-defaultsNM

Pel-pathnm0, 90, 180, 270 degrees
(0 is normal default)

Line-progressionnm90, 270 degrees
(270 is normal default)

Clippingnmany coordinate pair
within page

Pel-spacingnm6 BMU (200 pels/in.)
5 BMU (240 pels/in.)
4 BMU (300 pels/in.)
3 BMU (400 pels/in.)
2 BMU (600 pels/in.)
1 BMU (1200 pels/in.)
(Normal default is 4 BMU)

Non-basic-doc-characteristicsNM

Page-dimensionsnmSee table 1,
NA-F through NA-K,
roll paper

Medium-typesnmSee table 1,
NA-F through NA-K,
roll paper

Raster-gr-presentation-
featuresNM

Pel-pathnm180, 270 degrees

Line-progressionnm90 degrees

Pel-spacingnm5 BMU (240 pels/in.)
3 BMU (400 pels/in.)
2 BMU (600 pels/in.)
1 BMU (1200 pels/in.)

Document-management-attributesM

Document ReferencemAny string of characters

The following notation is used in the class column of this table:

- o m mandatory attribute

- o nm non-mandatory attribute

- o d defaultable attribute

Capital letters (M, NM, and D) are used for groups of attributes.

READER RESPONSE FORM

Please retain my name for the next mailing of the NIST/OSI Implementors Workshop.

NAME: ______________________________

ADDRESS: ______________________________

____

____

PHONE NO.: ______________________________

EMAIL ADDRESS: ______________________________

Mail this page to:
National Institute of Standards and Technology
NIST Workshop for Implementors of OSI
Brenda Gray, Workshop Coordinator
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 4507

2. PERFORMING ORGANIZATION REPORT NUMBER

3. PUBLICATION DATE
MARCH 1991

4. TITLE AND SUBTITLE

WORKING IMPLEMENTATION AGREEMENTS FOR OPEN SYSTEMS INTERCONNECTION PROTOCOLS -- DECEMBER 1990

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

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (STREET, CITY, STATE, ZIP)

10. SUPPLEMENTARY NOTES

□ DOCUMENT DESCRIBES A COMPUTER PROGRAM; SF-185, FIPS SOFTWARE SUMMARY, IS ATTACHED.

11. ABSTRACT (A 200-WORD OR LESS FACTUAL SUMMARY OF MOST SIGNIFICANT INFORMATION. IF DOCUMENT INCLUDES A SIGNIFICANT BIBLIOGRAPHY OR LITERATURE SURVEY, MENTION IT HERE.)

This document records Working Implementation Specification Agreements 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 assist 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)

NIST/OSI WORKSHOP, LOCAL AREA NETWORKS: NETWORK PROTOCOLS: 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
417

15. PRICE
A20

ELECTRONIC FORM