11

INTERNATIONAL STANDARDIZED PROFILE

Working Draft 7 of ISO/IEC DISP 11188-3 : 1994

TITLE: Information technology — International Standardized Profile — Common Upper Layer Requirements — Part 3: Minimal OSI upper layer facilities

SOURCE: OIW

STATUS: Working Draft Version 7, 1994-06-30

This version represents changes made at OIW, held in Gaithersburg, MD, 13-17 June 1994. The base text for this version was produced at the April EWOS meeting. The changes made at the June OIW reflect comments from the EWOS TLG, the AOW NMSIG, and other OIW SIGs. Editorial changes were also made.

Endorsement for version 6 of this document was received from EWOS and AOW pending the inclusion of the comments reflected in this version. Endorsement for this version of the document was received by OIW at the June Plenary.

ContentsTOC \oμ

§

AnnexesTOC \fμ

§

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental or non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC1. In addition to developing International Standards, ISO/IEC JTC1 has created a Special Group on Functional Standardization for the elaboration of International Standardized Profiles.

An International Standardized Profile is an internationally agreed, harmonized document which identifies a standard or group of standards, together with options and parameters, necessary to accomplish a function or set of functions.

Draft International Standardized Profiles are circulated to national bodies for voting. Publication as an International Standardized Profile requires approval by at least 75% of the national bodies casting a vote.

This part of ISO/ISP 11188 was prepared with the collaboration of

SYMBOL 151 \f "Times New Roman" \s 10 \h§ Asia-Oceania Workshop (AOW);

SYMBOL 151 \f "Times New Roman" \s 10 \h§ European Workshop for Open Systems (EWOS);

SYMBOL 151 \f "Times New Roman" \s 10 \h§ OSE Implementors Workshop (OIW).

Annexes A , B, C , D, E and F form an integral part of this part of ISO/IEC ISP 11188. Annexes G, H and I are informative.

Introduction

This part of ISO/IEC ISP 11188 is defined within the context of Functional Standardization, in accordance with the principles specified by ISO/IEC TR 10000, "Framework and Taxonomy of International Standardized 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.

A profile (e.g. an ISO/IEC ISP) or the specification of a basic communications application may reference this Profile. In addition, a referencing ISP may specify further requirements on the protocols, provided it does not contradict this ISP. A specification of an implementation may also reference this ISP.

The purpose of this multi-part ISP is to provide common text for ISPs or other referencing specifications which specify A-profiles. In addition to simplifying their drafting, it also facilitates the common implementation of the protocols for use in different A-profile contexts.

This part of ISO/IEC 11188 specifies a profile of the minimal OSI facilities supporting basic connection-oriented communications applications. These facilities are comprised of a subset of the facilities defined by the ACSE, Presentation, and Session service definitions.

Information technologySYMBOL 190 \f "Symbol"§International Standardized ProfileSYMBOL 190 \f "Symbol"§Common upper layer requirements
Part 3:
Minimal OSI upper layers facilities

1 Scope

This part of ISO/IEC ISP 11188 introduces the concept of the minimal set of OSI upper layer facilities for basic communications applications. A basic communications application simply requires the ability to open and close communications with a peer and to send and receive messages with the peer. It is expected that a large portion of potential OSI applications will be basic communications applications.

1.1 General

This Profile specifies the minimal set of upper layer facilities required for the support of basic communications applications. The minimal OSI facilities are referred to as mOSI.

This Profile defines the mOSI facilities in terms of identified features of the upper layer PICS proformas – the ACSE (ISO/IEC 8650-2), the Presentation Layer (ISO/IEC 8823-2), and the Session Layer (ISO/IEC 8327-2). The identified features of these PICS proformas are specified in annexes A, B, and C, respectively.

This Profile complies with the requirements stated in ISO/IEC ISP 11188-1, Basic connection-oriented requirements.

This Profile may be referenced by two classes of entities: upper layer users and upper layer providers.

SYMBOL 190 \f "Symbol"§ mOSI users represent basic communications applications. mOSI users may be profiles (such as A-profiles identified in ISO/IEC TR 10000-2) or specifications of basic communications applications that are not represented by a formal profile. An API specification is a special case of the latter. The term referencing specification is used in this Profile to represent this set of mOSI users.

SYMBOL 190 \f "Symbol"§ mOSI providers represent implementations of the upper layer facilities that provide (at a minimum) the facilities defined in this Profile. The term referencing implementation is used in this profile to represent this set of mOSI providers.

A referencing specification (a mOSI user) may claim compliance to this Profile. It may do so if the OSI upper layer facilities that it requires can be expressed by the facilities of this Profile. Subclause 2.1 summarizes the requirements for making such a statement. Annex D provides a proforma for a profile requirements list (PRL) for a compliant application..

A referencing implementation (a mOSI provider) may claim conformance to this Profile. It may do so if the OSI upper layer facilities that it provides include those expressed in this Profile. That is, an implementation may contain more upper layer facilities than those required to be conformant to this Profile. However, they must contain at least those of this Profile. Subclause 2.2 summarizes the requirements for making a conformance statement. Annex E provides a means for identifying the

features supported for conformance to this Profile.

Annex F assigns object identifier values for specific generic definitions of application context, abstract syntax, and transfer syntax.

1.2 Position within the taxonomy

This Profile does not specify a full A-profile, and therefore is not included in the taxonomy of ISO/IEC TR 10000-2.

2 Compliance and conformance

2.1 Profile or specification of a basic communications application

A specification may reference this Profile to identify its upper layer requirements and may claim compliance to this Profile.

To be compliant, a referencing specification shall do the following:

a) require that all of this Profile's mandatory ("m") features are also mandatory for the referencing specification;

b) require that all of this Profile's out of scope ("i") features are also out of scope for the referencing specification;

c) require that all of this Profile's optional ("o") features are kept as optional, i.e., they are not redefined by the referencing specification;

d) require that all of this Profile's open ("*") features are kept as open or are re-defined as mandatory ("m"), optional ("o") or out of scope ("i"); and

e) comply with the requirements of ISO/IEC ISP 11188-1 and not conflict with the requirements of this Profile.

NOTES

1 The meaning of the status indicators (see 6.1) when used by a referencing specification is given in table 1 column 3.

2 It is recommended that a referencing specification use the tables in annex D to specify its profile requirements list.

A referencing specification may be compliant with this Profile by either of the following ways:

a) It may repeat all of the specifications contained in this Profile. To claim compliance to this Profile, a referencing specification shall assure that its specification of the ACSE, Presentation, and Session features does not violate those in this Profile.

b) It may claim compliance by reference to this Profile instead of repeating the provisions of this

Profile.

2.2 OSI upper layer stack implementation

An implementation may reference this Profile to claim that it supports some or all of the features specified in this Profile. The referencing implementation may in fact support additional upper layer featuresSYMBOL 190 \f "Symbol"§without violating any of those in this Profile.

To be conformant, a referencing implementation shall do the following:

a) support all of this Profile's mandatory ("m") features;

b) follow the guidelines for support of this Profile's out of scope ("i") features outlined in table 1;

c) follow the guidelines for support of this Profile's optional ("o") features outlined in table 1;

d) follow the guidelines for support of this Profile's open ("*") features outlined in table 1; and

e) conform to the requirements of ISO/IEC ISP 11188-1.

NOTE SYMBOL 190 \f "Symbol"§ It is recommended that a referencing implementation use the tables in annex E to specify its profile implementation conformance statement.

2.3 Facilities, roles and options

This Profile defines mOSI compliance and conformance in terms of facilities, roles and options. This Profile has three facilities:

a) association establishment;

b) user data transfer; and

c) association release.

Association establishment includes two optional facilities:

a) authentication; and

b) application context negotiation.

Each facility has roles (Initiator/Responder and Requestor/Acceptor). Within this Profile, each role (or an optional facility) is referenced by a variable (see 6.1). For example, the variable name used to describe the capability to establish an association is establishment-initiator.

The referencing specification assigns each variable one of the following values:

SYMBOL 151 \f "Times New Roman" \s 10 \h§ mandatory ("m")

SYMBOL 151 \f "Times New Roman" \s 10 \h§ optional ("o")

SYMBOL 151 \f "Times New Roman" \s 10 \h§ out of scope ("i")

The meanings of these values are defined in table 1.

mOSI compliance and conformance can be determined by specifying values ("m", "o", or "i") for all of these variables.

2.4 Relationship to base standards

2.4.1 ACSEqq

This Profile specifies the Kernel functional unit. Optionally, the Profile also includes the Authentication functional unit and Application Context Name Negotiation functional unit. The Profile allows the roles for association establishment and release identified in ISO/IEC 8650.

The required facilities of ACSE are specified in annex A. A default value for application context name is defined in annex F. The requirements expressed in ISO/IEC ISP 11188-1 also apply to the ACSE aspects of this Profile.

2.4.2 Presentation Layer

This Profile specifies the Presentation Kernel functional unit.

The required facilities of Presentation Layer are specified in annex B. Default values for user abstract syntax name and user transfer syntax name are defined in annex F. The requirements expressed in ISO/IEC ISP 11188-1 also apply to the Presentation Layer aspects of this Profile.

2.4.3 Session Layer

This Profile specifies the Session Kernel and Duplex functional units.

The required facilities of Session Layer are specified in annex C. The requirements expressed in ISO/IEC ISP 11188-1 also apply to the Session Layer aspects of this Profile.

2.5 Transport-provider

This Profile does not address the lower four OSI layers (Transport, Network, Link, and Physical Layers). They are outside of the scope of this Profile (see also clause 7).

A transport-provider is needed to support the exchange of ACSE, Presentation, and Session PDUs for a conformant mOSI implementation. To meet this requirement, the transport-provider shall supply services equivalent to those defined in the OSI Transport Layer service definition (ITU-T Rec. X.214 | ISO 8072).

3 Normative references

The following ITU-T Recommendations | International Standards contain provisions which, through reference in this text, constitute provisions of this International Standardized Profile. At the time of publication, the editions indicated were valid. All Recommendation and Standards are subject to revision, and parties to agreements based on this International Standardized Profile are encouraged to

investigate the possibility of applying the most recent editions of the ITU-T Recommendations | International Standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards. The ITU-T Secretariat maintains a list of the currently valid ITU-T Recommendations.

3.1 Identical Recommendations | International Standards

SYMBOL 190 \f "Symbol" \s 10 \h§ ITU-T Recommendation X.227 (1994) | ISO/IEC 8650-1 : 1994, Information technology SYMBOL 190 \f "Symbol"§Open Systems InterconnectionSYMBOL 190 \f "Symbol"§Protocol specification for the Association Control Service ElementSYMBOL 190 \f "Symbol"§Part 1 : Protocol Specification.

SYMBOL 190 \f "Symbol" \s 10 \h§ ITU-T Recommendation X.217 (1994) | ISO/IEC 8649 : 1994, Information technology SYMBOL 190 \f "Symbol"§Open Systems InterconnectionSYMBOL 190 \f "Symbol"§Service definition for the Association Control Service Element.

SYMBOL 190 \f "Symbol" \s 10 \h§ ITU-T Recommendation X.200 (1994) | ISO/IEC 7498-1 : 1994, Information technologySYMBOL 190 \f "Symbol"§Open Systems InterconnectionSYMBOL 190 \f "Symbol"§Part 1: Basic Reference Model.

SYMBOL 190 \f "Symbol" \s 10 \h§ ITU-T Recommendation X.225 (1994) | ISO/IEC 8327-1 : 1994, Information technologySYMBOL 190 \f "Symbol"§Open Systems InterconnectionSYMBOL 190 \f "Symbol"§Basic connection oriented session protocolSYMBOL 190 \f "Symbol"§Part 1 : Protocol specification.

SYMBOL 190 \f "Symbol" \s 10 \h§ ITU-T Recommendation X.215 (1994) | ISO/IEC 8326 : 1994, Information technology SYMBOL 190 \f "Symbol"§Open Systems InterconnectionSYMBOL 190 \f "Symbol"§Basic connection-oriented session service definition.

SYMBOL 190 \f "Symbol" \s 10 \h§ ITU-T Recommendation X.226 (1994) | ISO/IEC 8823-1 : 1994, Information technologySYMBOL 190 \f "Symbol"§Open Systems InterconnectionSYMBOL 190 \f "Symbol"§Basic connection oriented presentation protocolSYMBOL 190 \f "Symbol"§Part 1 : Protocol specification.

SYMBOL 190 \f "Symbol" \s 10 \h§ ITU-T Recommendation X.216 (1994) | ISO/IEC 8822 : 1994, Information technologySYMBOL 190 \f "Symbol"§Open Systems InterconnectionSYMBOL 190 \f "Symbol"§Basic connection-oriented presentation service definition.

SYMBOL 190 \f "Symbol" \s 10 \h§ ITU-T Recommendation X.690 (1993) | ISO/IEC 8825-1 : 1993, Information technologySYMBOL 190 \f "Symbol"§ASN.1 encoding rulesSYMBOL 190 \f "Symbol"§Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER).

3.2 Paired Recommendations | International Standards equivalent in technical content

SYMBOL 190 \f "Symbol" \s 10 \h§ CCITT Recommendation X.210 (1988), OSI Layer Service Definition Conventions for CCITT applications.

ISO/TR 8509 : 1986, OSI Layer Service Definition Conventions.

SYMBOL 190 \f "Symbol" \s 10 \h§ CCITT Recommendation X.214 (1988), Transport service definition for Open Systems Interconnection for CCITT applications.

ISO 8072 : 1986, Information technologySYMBOL 190 \f "Symbol"§Open Systems InterconnectionSYMBOL 190 \f "Symbol"§Transport service definition.

SYMBOL 190 \f "Symbol" \s 10 \h§ ISO 7498-3 : 1989, Information processing systemsSYMBOL 190 \f "Symbol"§Open Systems InterconnectionSYMBOL 190 \f "Symbol"§Basic Reference ModelSYMBOL 190 \f "Symbol"§Part 3 : Naming and Addressing.

SYMBOL 190 \f "Symbol" \s 10 \h§ CCITT Recommendation X.650 (1993), Open Systems Interconnection (OSI)SYMBOL 190 \f "Symbol"§Basic Reference Model for naming and addressing.

3.3 Additional references

SYMBOL 190 \f "Symbol" \s 10 \h§ ISO/IEC 8327-2 : 1994, Information technology SYMBOL 190 \f "Symbol"§Open Systems InterconnectionSYMBOL 190 \f "Symbol"§Basic connection oriented session protocolSYMBOL 190 \f "Symbol"§Part 2 : Protocol Implementation Conformance Statement (PICS) Proforma.

SYMBOL 190 \f "Symbol" \s 10 \h§ ISO/IEC 8650-2 : 1994, Information technologySYMBOL 190 \f "Symbol"§Open Systems InterconnectionSYMBOL 190 \f "Symbol"§Protocol specification for the Association Control Service ElementSYMBOL 190 \f "Symbol"§Part 2 : Protocol Implementation Conformance Statement (PICS) Proforma .

SYMBOL 190 \f "Symbol" \s 10 \h§ ISO/IEC 8823-2 : 1994, Information technologySYMBOL 190 \f "Symbol"§Open Systems InterconnectionSYMBOL 190 \f "Symbol"§Basic connection-oriented presentation protocolSYMBOL 190 \f "Symbol"§Part 2 : Protocol Implementation Conformance Statement (PICS) Proforma.

SYMBOL 190 \f "Symbol" \s 10 \h§ ISO/IEC 9545 : 1993, Information technologySYMBOL 190 \f "Symbol"§Open Systems InterconnectionSYMBOL 190 \f "Symbol"§Application Layer Structure.

SYMBOL 190 \f "Symbol"§ ISO/IEC 9646-7, Information technologySYMBOL 190 \f "Symbol"§Open Systems InterconnectionSYMBOL 190 \f "Symbol"§Conformance testing methodology and frameworkSYMBOL 190 \f "Symbol"§Part 7 : Implementation conformance statements — Requirements and guidance on ICS and ICS Proformas.

SYMBOL 190 \f "Symbol" \s 10 \h§ ISO/IEC TR 10000-1 : 1992, Information technologySYMBOL 190 \f "Symbol"§Framework and taxonomy of International Standardized ProfilesSYMBOL 190 \f "Symbol"§Part 1 : Framework. .

SYMBOL 190 \f "Symbol" \s 10 \h§ ISO/IEC TR 10000-2 : 1992, Information technologySYMBOL 190 \f "Symbol"§Framework and taxonomy of international standardized profilesSYMBOL 190 \f "Symbol"§Part 2 : Taxonomy of Profiles.

SYMBOL 190 \f "Symbol" \s 10 \h§ ISO/IEC ISP 11188-1 : 1994, Information technologySYMBOL 190 \f "Symbol"§International Standardized ProfileSYMBOL 190 \f "Symbol"§Common upper layer

requirementsSYMBOL 190 \f "Symbol"§Part 1 : Basic connection-oriented requirements.

4 Definitions

This Profile makes use of the following definitions.

4.1 Reference model definitions

4.1.1 Basic Reference Model definitions

This Profile is based on the concepts developed in ITU-T Rec. X.200 | ISO/IEC 7498-1. It makes use of the following terms defined in them:

a) application-entity;

b) Application Layer;

c) application-process;

d) application-protocol-control-information;

e) application-protocol-data-unit;

f) application-service-element;

g) compliance;

h) presentation-connection;

i) Presentation Layer;

j) presentation-service;

k) session-connection;

l) Session Layer;

m) session-protocol;

n) session-service;

o) Transport Layer

4.1.2 Naming and addressing definitions

This Profile makes use of the following terms defined in ISO 7498-3:

a) application-process title;

b) application-entity qualifier;

c) application-entity title;

d) application-process invocation-identifier;

e) application-entity invocation-identifier; and

f) presentation address.

4.2 Service conventions definitions

This Profile makes use of the following terms defined in CCITT Rec. X.210 | ISO/TR 8509:

a) primitive;

b) request (primitive);

c) indication (primitive);

d) response (primitive); and

e) confirm (primitive).

4.3 Presentation definitions

This Profile makes use of the following terms defined in ITU-T Rec. X.216 | ISO/IEC 8822 and ITU-T Rec. X.226 | ISO/IEC 8823-1:

a) abstract syntax;

b) abstract syntax name;

c) default context;

d) defined context set;

e) functional unit [Presentation];

f) normal mode [Presentation];

g) presentation context;

h) presentation data value; and

i) presentation selector

4.4 Session definitions

This Profile makes use of the following terms defined in ITU-T Rec. X.215 | ISO/IEC 8326 and ITU-T Rec. X.225 | ISO/IEC 8327-1:

a) session selector.

4.5 Application Layer Structure definitions

This Profile makes use of the following terms defined in ISO/IEC 9545:

a) application-context;

b) application-entity invocation;

c) control function; and

d) application-service object.

4.6 ACSE service definitions

This Profile makes use of the following terms defined in ISO/IEC 8649:

a) application-association; association;

b) Association Control Service Element;

c) requestor;

d) acceptor;

e) association-initiator; and

f) association-responder.

4.7 Taxonomy of profile definitions

This Profile makes use of the following terms defined in ISO/IEC TR 10000-1:

a) A-Profile;

b) profile requirements list;

c) profile implementation conformance statement; and

d) PICS Proforma.

4.8 Definitions of this Profile

For the purpose of this Profile, the following definitions apply.

basic communications application: An application program that simply requires the ability to open and close communications with a peer and to send and receive messages with that peer.

conformance: The referencing implementation supports an identified and consistent set of dynamic conformance requirements specified in a set of related OSI protocol, profile, abstract syntax, encoding rule and information object specifications.

mOSI specification; mOSI stack specification: This specification that defines the minimal facilities of the Session Layer, Presentation Layer, and ACSE.

mOSI stack: An implementation that supports, at a minimum, the facilities defined in the mOSI stack specification.

mOSI platform specification: The functional specification of a formal programmatic interface and a set of supporting local services for the mOSI stack specification.

mOSI platform: An implementation of the mOSI platform specification.

non-basic communications application: An application program that requires the ability to support functions other than those specified in the definition a basic communications application.

platform: An implementation of an identified platform specification.

platform-based application: An application program that conforms to a platform specification.

PDV-processor: part of an implementation which wraps and unwraps the "PDV envelope" around the syntax sent and received in the identified presentation context.

platform specification: The functional specification of a formal programmatic interface and a set of supporting local services for an identified stack specification.

referencing implementation: A specification of an implementation which references mOSI for defining its upper layer requirements.

referencing specification: A specification of an ISO/IEC ISP or the specification of a basic communications application which references mOSI for defining its upper layer requirements.

specific basic communications application: an application that is not referenced by any ISP.

stack; stack implementation: An implementation of an identified stack specification

stack specification: The functional specification of a set of interrelated standards for the purpose of providing a common service (set of facilities).

standalone application: Any application program which is not a platform-based application.

transport-provider: A provider of those transport services which are defined in ISO 8072.

5 Abbreviations

The following abbreviations are used in this Profile.

ACSE Association Control Service Element

APDU application-protocol-data-unit

API application programmatic interface

ASN.1 Abstract Syntax Notation One

BCA basic communications application

CCITT International Telegraph and Telephone Consultative Committee

CULR Common Upper Layers Requirements

ICS implementation conformance statement

IEC International Electrotechnical Commission

ISO International Organization for Standardization

ISP International Standardized Profile

ITU-T International Telecommunications Union — Telecommunications Systems and Services

mOSI minimal OSI upper layer facilities

OSI Open Systems Interconnection

PDU protocol-data-unit

PDV presentation data value

PICS protocol implementation conformance statement

PPDU presentation-protocol-data-unit

PRL profile requirements list

SPDU session-protocol-data-unit

TSDU transport-service-data-unit

6 Conventions

6.1 Profile status indicators

This Profile states mOSI functionality by defining rules for forming a mOSI Profile Requirements List (PRL) based on the PICS proformas of ACSE, Presentation, and Session. The rules for defining a mOSI PRL are contained in annexes A, B, and C. The rules are specified by a series of tables in each annex. Each table in an annex refers to one identified table in the respective PICS Proforma. Each row in an annex table refers to a row in the corresponding PICS Proforma table.

Annexes A, B, and C consist of rules that determine the status indicator for each entry in the ACSE, Presentation, Session PICS Proformas. A status indicator defines rules that a referencing application or a referencing implementation must follow as it relates to the associated entry in annex A, B, or C. The Profile status indicators used in this Profile are presented in table 1.

The upper layer PICS proforma and consequently an upper layer PRL contain over 120 tables with almost 1000 entries (questions). For BCA applications, a large number of the PICS proforma entries do not apply—they are out of scope (i). Others are obviously required—they are mandatory (m). Some are considered internal and left to the discretion of the implementor—they are optional (o).

The remaining entries of annexes A, B, and C are determined by a referencing specification or a referencing implementation. This is done by assigning status indicators (those defined in table 1) to mOSI variables and open parameters (see 2.3 and 6.2).

An application PRL results when status identifiers are assigned to mOSI variables and open parameters and they are applied to annexes A, B, and C. Table 1 definitions apply to the resultant status identifiers of the application PRLe resultant

Column 2 of table 1 defines rules for a referencing specification. They define how a referencing specification shall view an entry in annexes A, or B, or C. It also defines the meaning of a status indicator when assigned to a mOSI variable or to an open parameter.

Column 3 of table 1 defines rules for a referencing implementation. They define how a referencing implementation shall view an entry in annexes A, or B, or C or that of an applications PRL.

Table 1 – Profile status identifiers

Identifier	Meaning when referenced by a specification	Meaning when referenced by an implementation
1 m	mandatory SYMBOL 190 \f "Symbol"§ The feature shall be required for support. The status of the feature shall remain mandatory in a referencing specification (see note).	mandatory SYMBOL 190 \f "Symbol"§ The implementation shall support the feature, i.e. its syntax and procedures shall be implemented as specified in the base standard or in ISO/IEC ISP 11188-1. However, it is not a requirement that the feature shall be used in all instances of communication unless mandated by the base standard or stated otherwise in ISO/IEC ISP 11188-1. The feature shall be the subject of an ISP conformance test (see note).
2 o	optional — The choice of whether this feature is supported or it is not supported is made by the implementation. The status of the feature shall remain optional in a referencing specification.	optional — The implementation may decide either to support or to not support the feature. SYMBOL 183 \f "Symbol" \s 10 \h§ Supporting the feature means that the feature shall be handled as if it was mandatory. SYMBOL 183 \f "Symbol" \s 10 \h§ Not supporting a feature depends on receiving or sending: For sending, the feature's capability is not used. For receiving an optional parameter, the syntax shall be implemented and the parameter may be ignored). The feature shall be the subject of an ISP conformance test.
3 *	open SYMBOL 190 \f "Symbol"§ The status of this feature shall be decided by the referencing specification. The referencing specification shall indicate that the status of the feature is mandatory, optional, or out of scope. Alternately, the referencing specification may keep the feature open.	open SYMBOL 190 \f "Symbol"§ same as optional
4 x	excluded SYMBOL 190 \f "Symbol"§ The feature shall not be used in a referencing specification. The	

status of the feature shall remain excluded in a referencing specification.
excluded SYMBOL 190 \f "Symbol"§ The implementation shall not support the feature. When completing the associated PICS proforma table, the answer for the support column shall be that the feature has not been implemented. The implementation shall abort if the feature is received. The exclusion should be the subject of an ISP conformance test.

5
i
out of scope SYMBOL 190 \f "Symbol"§ The requirement for the support of this feature is not covered by this Profile. The status of the feature shall remain out of scope in a referencing specification.
out of scope SYMBOL 190 \f "Symbol"§ Support for a feature shall follow the guidelines outlined for optional above with the exception that this feature shall not be the subject of an ISP conformance test.

6
-
not applicable SYMBOL 190 \f "Symbol"§ The feature is not relevant where mentioned in a table. Support for the feature is either meaningless, logically impossible, or physically impossible, after some conditions are evaluated. The status of the feature shall remain not applicable in a referencing specification.
not applicable SYMBOL 190 \f "Symbol"§ The feature is not defined by the base standard in the context where it is mentioned in a table. A support answer is not required from the implementor.

7
c[n]
conditionally supported SYMBOL 190 \f "Symbol"§ Support for the feature is further defined in this Profile by condition ("n") annexed to the table. The value evaluated from the condition evaluates to one of the following values defined in this table: "m"; "o"; "i", or "–".
conditionally supported SYMBOL 190 \f "Symbol"§ Support for the feature is further defined by a condition ("n") which is annexed to the table. Depending on the condition, when completing the associated PICS Proforma table, the answer for the support column shall either be: the feature has been implemented; the feature has not been implemented; or not applicable.

NOTE SYMBOL 190 \f "Symbol"§ The support of a feature can be conditional, depending on the support of a class of features to which it belongs, e.g. a parameter in a PDU, a PDU in a functional unit.

6.2 Definitions of variables

Variables are used in annexes A, B, and C as a method of specifying values for the "Profile status" column of these tables. The variables used are

Establishment-initiator SYMBOL 190 \f "Symbol"§ capability to establish an association or connection

Establishment-responder SYMBOL 190 \f "Symbol"§ capability to respond to an establishment initiation

Establishment-responder-reject SYMBOL 190 \f "Symbol"§ capability to reject an establishment

initiation

Release-requestor SYMBOL 190 \f "Symbol"§ capability to release an association or connection

Release-acceptor SYMBOL 190 \f "Symbol"§ capability to accept a release request

Normal-data-requestor SYMBOL 190 \f "Symbol"§ capability to request normal data transfer

Normal-data-acceptor SYMBOL 190 \f "Symbol"§ capability to accept normal data transfer

Authentication SYMBOL 190 \f "Symbol"§ capability to perform authentication

Application-context-negotiation SYMBOL 190 \f "Symbol"§ capability to perform application context negotiation

Transport-expedited — capability to use transport expedited data for Session PDUs (this does not include the Session ExpeditedData functional unit).

These variables appear in italics. They reflect the roles and options identified in 2.3. The values these variables may take are "m", "o", and "i" as defined in table 1. Values for these variables may be assigned by a referencing specification in annex D and by a referencing implementation in annex E.

7 Model

This clause presents the mOSI model and defines many of the terms used in this Profile. The mOSI model, as shown in figure 1, illustrates the mOSI stack in three different environments which are detailed in 7.2 and 7.3. It can be viewed in two contexts: it can be viewed abstractly – where the various elements represent abstract specifications; or it can be viewed concretely – where the elements represent those of an implementation.

EMBED MSDraw * mergeformatµ §

7.1 Common elements

There are common elements in all three environments shown in figure 1. They are:

SYMBOL 151 \f "Times New Roman" \s 10 \h§ basic communications application

SYMBOL 151 \f "Times New Roman" \s 10 \h§ PDV-processor

SYMBOL 151 \f "Times New Roman" \s 10 \h§ mOSI stack;

SYMBOL 151 \f "Times New Roman" \s 10 \h§ transport services and

SYMBOL 151 \f "Times New Roman" \s 10 \h§ transport provider

A basic communications application (BCA) simply requires the ability to open and close communications with a peer and to send and receive messages with the peer. This Profile addresses the requirements of basic communications applications.

A stack represents a set of layered, interdependent communication standards (in the abstract sense) and their implementation (in the concrete sense). The mOSI stack represents the ACSE, Presentation, and Session standards (protocol specifications) or their implementation with the features specified in this Profile.

NOTE – A stack does not necessarily represent a layered implementation of the layered standards. On the contrary, it is recommended in annex H that the implementation of a mOSI stack is one protocol engine, not three.

From the perspective of the Presentation protocol (ISO/IEC 8823-1), the syntax (encoded data) sent from one application to its peer is a series of one or more presentation-data-values (PDV). The ISO Presentation protocol defines the encoding of the outer envelope of a PDV and the encoding for groups of PDVs (if any). The actual contents of a PDV is a function of the mutually agreed upon abstract and transfer syntax of the PDV – its presentation context. While ASN.1 basic encoding rules can be used for encoding abstract and transfer syntax, it is not the only choice.

The negotiation of the transfer syntax and the encoding/decoding of user information is done by the Presentation Layer. However, the actual encoding/decoding of the transfer syntax sent between connected applications is outside of the scope of the mOSI stack. The PDV-processor represents the wrapping and unwrapping of the PDV envelope around the syntax sent or received in the identified presentation context. As shown in figure 1, the PDV-processor can be located at a number of different

places within the model. The mOSI model assumes that PDV encoding and decoding is done outside of the mOSI stack.

This Profile does not address the four lower OSI layers (Transport, Network, Link, and Physical Layers). They are considered outside of the scope of this Profile. However, a transport-provider is needed to transport the ACSE, Presentation, and session PDUs of a mOSI implementation. As such, the transport-provider supplies transport services equivalent to those defined in the OSI Transport Layer service definition (ISO 8072).

This specification does not place any requirements on the actual transport provider (layer 4 and below) used as long as services equivalent to the OSI transport services are provided.

7.2 Standalone applications

For the purposes of this Profile, a standalone application is one that includes the application PDV-processor and the mOSI stack as a single unit application. For an implementation, the mOSI stack may be a series of separate modules with its own internal programmatic interface or as a single state machine.

7.3 Platform-based applications

A communications platform allows a division between an application program and its communications provider. A platform comprises the communication facilities in one system necessary to support a distributed application. A platform-based application represents the communication aspects of a distributed application in one system. An application programmatic interface (API) is the formal interface between a communication platform and its user [platform-based] applications. It is formal in the sense that the API is specified so as to allow the use of the platform by different types of applications – most often, in parallel. The programmatic interface represents the mapping of the API to the internals of the supporting system.

A mOSI platform consists of a mOSI API, a mOSI stack in conjunction with the normal facilities provided by a platform (e.g. POSIX services in the case of a UNIX based platform).

A mOSI API represents the interface to the mOSI stack. It provides the minimal facilities of the OSI upper layers as defined in this Profile.

This Profile two types of basic communications application: migrant application and kernel application (also see annex G). Depending on the type of application, the PDV-processor could either be a part of the platform or a part of each platform-based application.

7.3.1 Migrant applications

OSI (and mOSI) has two required facilities that are not part of other transport providers:

a) application context; and

b) presentation context – abstract syntax name and transfer syntax name pair.

An OSI upper layer stack requires that names be provided for application context, abstract syntax, and

transfer syntax. These names may be hidden from the API user by having the programmatic interface provide default values (see annex F).

A migrant application running over a stack (see G.2.3.2) is unaware (or at least, not concerned) with formally identifying application context and the presentation context of the data sent and received. Instead, it allows the programmatic interface to provide default values (see annex F). The encoding and decoding of the PDVs are hidden by placing the PDV-processor within the platform.

7.3.2 Kernel application

A kernel application (see G.2.3.1) is an OSI-based application. It is aware of the mandated application context names and presentation context.

Most likely, (but, not necessarily) the application's own protocol will be specified and encoded using ASN.1. For this reason the PDV-processor is shown in figure 1 within the application itself – rather than as part of the platform. It is not expected that a kernel application will use the default values for abstract syntax and transfer syntax defined in annex F.

Annex A

(normative)

tc "A mOSI requirements for ACSE facilities" \l 1§mOSI requirements for ACSE facilities

This annex contains the mOSI specifications for completing the ACSE Profile Requirements List (PRL) for the selected facilities, roles and options (see 2.3).

This annex uses the tables in the ACSE PICS Proforma (ISO/IEC 8650-2). The clause numbers and tables referenced in this annex are those of the PICS Proforma. If the PICs proforma contains tables which are not explicitly outlined in this annex, then the features of those tables are out of scope (i) for this Profile.

The specifications of this annex reference the following variables: Establishment-initiator, Establishment-responder, Normal-data-requestor, Normal-data-acceptor, Release-requestor, and Release-acceptor. The values for these variables may be set by a referencing specification using the proforma provided in table D.1. The values for these variables may be set by a referencing implementation using the proforma provided in table E.1.

This annex contains several open (*) parameters whose requirements are specified by the referencing specification or the referencing implementation. The requirements are expressed in terms of "m", "o", or "i" as defined in table 1. The parameters have the identifier of "*" in the tables of this annex. A referencing specification may set these open parameters by using the proforma provided in table D.2. A referencing implementation may set the open parameters by using the proforma provided in table E.2.

NOTESYMBOL 190 \f "Symbol"§PICS Proforma clauses A.1-A.4 are not covered in this Profile. The questions are answered by an implementor of ACSE.

A.1 Global statement of conformance – [PICS Proforma clause A.5]

Question
Answer
PICS Proforma reference

1
Are all mandatory features supported?
yes
A.5/1

A.2 Supported roles – [PICS Proforma clause A.6]

A.2.1 Association establishment procedure – [PICS Proforma A.6.1]

Role	Profile	PICS Proforma reference	Constraint / [mnemonic]
1	Initiator	c[1]	A.6.1/1 [A-CON-initiator]
2	Responder	c[2]	A.6.1/2 [A-CON-responder]

[1] The value of Establishment-initiator

[2] The value of Establishment-responder

A.2.2 Normal release procedure – [PICS Proforma A.6.2]

Role	Profile	PICS Proforma reference	Constraint / [mnemonic]
1	Requestor	c[1]	A.6.2/1 [A-REL-requestor]
2	Acceptor	c[2]	A.6.2/2 [A-REL-acceptor]

[1] The value of Release-requestor

[2] The value of Release-acceptor

A.2.3 Abnormal release procedure – [PICS Proforma A.6.3]

Role	Profile	PICS Proforma reference	Constraint / [mnemonic]
1 Requestor	m	A.6.3/1	
2 Acceptor	m	A.6.3/2	

A.3 Protocol mechanisms – [PICS Proforma clause A.7]

Protocol mechanism	Profile	PICS Proforma reference	Constraint / [mnemonic]
1 Normal mode	m	A.7/1	
2 X.410-1984 mode	i	A.7/2	Not used by BCA
3 Rules of extensibility	m	A.7/3	
4 Support of session version 2			

m
A.7/4

A.4 Functional units – [PICS Proforma clause A.8]

ACSE functional unit
Profile
PICS Proforma reference
Constraint / [mnemonic]

1
Kernel
m
A.8/1

2
Authentication
c[1]
A.8/2
[A-FU(AU)]

3
AC Name Negotiation
c[2]
not in yet
[A-FU(ACN)]

[1] Value of Authentication

[2] Value of Application-context-negotiation

A.5 Supported APDUs – [PICS Proforma clause A.9]

APDU	Profile: Sender	Profile: Receiver	PICS Proforma reference	Constraint / [mnemonic]
1 AARQ	c[1]	c[2]	A.9/1	
2 AARE	c[2]	c[1]	A.9/2	
3 RLRQ	c[3]	c[4]	A.9/3	
4 RLRE	c[4]	c[3]	A.9/4	
5 ABRT	m	m	A.9/5	

[1] if [A-CON-initiator] then "m" else "-"

[2] if [A-CON-responder] then "m" else "-"

[3] if [A-REL-requestor] then "m" else "-"

[4] if [A-REL-acceptor] then "m" else "-"

A.6 Supporting APDU parameters – [PICS Proforma clause A.10]

A.6.1 A-associate-request (AARQ) – [PICS Proforma A.10.1]

Parameter	Profile: Sender [a]	Profile: Receiver [b]	PICS Proforma reference	Constraint /[mnemonic]
1 Protocol Version	m	m	A.10.1/1	If sent, the value shall be "version 1"; the default value is "version 1"
2 Application Context Name	m	m	A.10.1/2	
3 Calling AP Title	*	*	A.10.1/3	If either is supported, then
4 Calling AE Qualifier	*	*	A.10.1/4	both shall be supported.

5
Calling AP Invocation-identifier
*
*
A.10.1/5
If either is supported then,

6
Calling AE Invocation-identifier
*
*
A.10.1/6
both shall be supported.

7
Called AP Title
*
*
A.10.1/7
If either is supported, then

8
Called AE Qualifier
*
*
A.10.1/8
both shall be supported.

9
Called AP Invocation-identifier
*
*
A.10.1/9
If either is supported, then

10
Called AE Invocation-identifier
*
*
A.10.1/10
both shall be supported.

11
ACSE Requirements
c[1]
c[1]
A.10.1/11

12
Authentication-mechanism Name
c[2]
c[2]
A.10.1/12

13
Authentication-value
c[2]
c[2]
A.10.1/13

14
Application Context List
c[3]
c[3]
not in yet

15
Implementation Information
i
i
A.10.1/14

16
User Information
*
*
A.10.1/15

[a] if NOT [A-CON-initiator] then the entire column has the value of "–" ; else the values are as marked.

[b] if NOT[A-CON-requestor] then the entire column has the value of "–" ; else the values are as marked.

[1] if [A-FU(AU) or A-FU(ACN)] then "m" else "–"

[2] if [A-FU(AU)] then "m" else "-"

[3] if [A-FU(ACN)] then "m" else "-"

A.6.2 A-associate-response (AARE) – [PICS Proforma A.10.2]

Parameter
Profile:
Sender
[a]
Profile:
Receiver
[b]
PICS Proforma
reference
Constraint / [mnemonic]

1
Protocol Version
m
m
A.10.2/1
Value shall be "version 1"; not required to be sent because the default value is "version 1"

2
Application Context Name
m
m
A.10.2/2

3
Responding AP Title
*
*
A.10.2/3
If either is supported, then

4
Responding AE Qualifier
*
*
A.10.2/4
both shall be supported.

5
Responding AP Invocation-identifier
*
*
A.10.2/5
If either is supported, then

6
Responding AE Invocation-identifier

*
*
A.10.2/6
both shall be supported.

7
Result
m
m
A.10.2/7

8
Result Source-diagnostic
m
m
A.10.2/8

9
ACSE Requirements
c[1]
c[1]
A.10.2/9

10
Authentication-mechanism Name
c[2]
c[2]
A.10.2/10

11
Authentication-value
c[2]
c[2]
A.10.2/11

12
Application Context List
c[3]
c[3]

12
Implementation Information

i
i
A.10.2/12

13
User Information
*
*
A.10.2/13

[a] if NOT [A-CON-responder] then the entire column has the value of "–" ; otherwise the values are as marked.

[b] If NOT [A-CON-initiator] then the entire column has the value "–"; otherwise the values are as marked.

[1] if [A-FU(AU) or A-FU(ACN)] then "m" else "–"

[2] if [A-FU(AU)] then "m" else "-"

[3] if [A-FU(ACN)] then "m" else "-"

A.6.3 A-release-request (RLRQ) – [PICS Proforma A.10.3]

Parameter
Profile: Sender
[a]
Profile:
Receiver
[b]
PICS Proforma reference
Constraint / [mnemonic]

1
Reason
*
*
A.10.3/1

2
User Information
*
*
A.10.3/2

[a] if NOT [A-REL-requestor] then the entire column has the value of "–" ; otherwise the values are as marked.

[b] if NOT [A-REL-responder] then the entire column has the value of "–" ; otherwise the values are as marked.

A.6.4 A-release-response (RLRE) – [PICS Proforma A.10.4]

Parameter
Profile:
Sender
[a]
Profile:
Receiver
[b]
PICS Proforma reference
Constraint / [mnemonic]

1
Reason
*
*
A.10.4/1

2
User Information
*
*
A.10.4/2

[a] if NOT [A-REL-acceptor] then the entire column has the value of "–" ; otherwise the values are as marked.

[b] if NOT [A-REL-requestor] then the entire column has the value of "-" ; otherwise the values are as marked.

A.6.5 A-abort (ABRT) – [PICS Proforma A.10.5]

Parameter
Profile:
Sender

Profile:
Receiver
PICS Proforma reference
Constraint / [mnemonic]

1
Abort Source
m
m

A.10.5/1

2
Diagnostic
c[1]
c[1]
A.10.5/2
Only defined for the Authentication functional unit

3
User Information
*
*
A.10.5/3

[1] if [A-FU(AU) or A-FU(ACN)] then "m" else "–"

A.7 Supported parameter forms – [PICS Proforma clause A.11]

A.7.1 AE Title name form – [PICS Proforma A.11.1]

Syntax form
Profile:
Sender

Profile:
Receiver
PICS Proforma
reference
Constraint / [mnemonic]

1
Form 1 (Directory name)
*
m
A.11.1/1

2
Form 2 (Object identifier and integer)
*
m
A.11.1/2

A.7.2 Authentication value form – [PICS Proforma A.11.2]

Syntax form
Profile:
Sender

Profile:
Receiver
PICS Proforma
reference
Constraint / [mnemonic]

1
Graphic string
c[1]
c[2]
A.11.1/1
If the value evaluated for Sender is "o", then at least one syntax form shall be selected for support.

2
BIT STRING
c[1]
c[2]
A.11.1/2

3
EXTERNAL
c[1]
c[2]
A.11.1/3

4
Other
c[1]
c[2]
A.11.1/4

[1] if [A-FU(AU)] then "o" else "-"

[2] if [A-FU(AU)] then "m" else "-"

Annex B

(normative)

tc "B mOSI requirements for Presentation Layer facilities"§mOSI requirements for Presentation Layer facilities

This annex contains the mOSI specifications for completing the Presentation Layer Profile Requirements List (PRL) for the selected facilities, roles and options (see 2.3).

This annex uses the tables in the Presentation Layer PICS Proforma (ISO/IEC 8823-2). The clause numbers and tables referenced in this annex are those of the PICS Proforma. If the PICs proforma contains tables which are not explicitly outlined in this annex, then the features of those tables are out of scope (i) for this Profile.

The specifications of this annex reference the following variables: Establishment-initiator, Establishment-responder, Establishment-responder-reject, Normal-data-requestor, Normal-data-acceptor, Release-requestor, and Release-acceptor. The values for these variables may be set by a referencing specification using the proforma provided in table D.1. The values for these variables may be set by a referencing implementation using the proforma provided in table E.1.

This annex contains several open (*) parameters whose requirements are specified by the referencing specification or the referencing implementation. The requirements are expressed in terms of "m", "o", or "i" as defined in table 1. The parameters have the identifier of "*" in the tables of this annex. A referencing specification may set these open parameters by using the proforma provided in table D.2. A referencing implementation may set the open parameters by using the proforma provided in table E.2.

NOTESYMBOL 190 \f "Symbol"§PICS Proforma clauses A.1-A.4 are not covered in this Profile. The questions are answered by an implementor of the Presentation service.

B.1 Global statement of conformance – [PICS Proforma clause A.5]

Question	Answer	PICS Proforma reference
1 Are all mandatory features supported?	yes	A.5/1

B.2 Protocol mechanisms and functional units – [PICS Proforma clause A.6]

B.2.1 Protocol mechanisms – [PICS Proforma A.6.1]

Protocol mechanism
Profile
PICS Proforma reference
Constraint / [mnemonic]

1
X.410 (1984)
i
A.6.1/1
Not used by BCA

2
Normal mode
m
A.6.1/2

B.2.2 Functional units – [PICS Proforma A.6.2]

Presentation functional units
Profile
PICS Proforma reference
Constraint / [mnemonic]

1
Kernel
m
A.6.2/1

2
Presentation Context management
i
A.6.2/2
Not used by BCA

3
Presentation Context Restoration
i
A.6.2/3
Not used by BCA

Pass-through to Session functional units
Profile
PICS Proforma reference
Constraint / [mnemonic]

4
Negotiated Release
i
A.6.2/4
Not used by BCA

5
Half Duplex
i
A.6.2/5
Not used by BCA

6
Duplex
m
A.6.2/6

7
Expedited Data
i
A.6.2/7
Not used by BCA

8
Typed Data
i
A.6.2/8
Not used by BCA

9
Capability Data Exchange
i
A.6.2/9
Not used by BCA

10
Minor Synchronize
i
A.6.2/10
Not used by BCA

11
Symmetric Synchronize
i
A.6.2/11
Not used by BCA

12
Major Synchronize
i
A.6.2/12
Not used by BCA

13
Resynchronize
i
A.6.2/13
Not used by BCA

14
Exceptions
i
A.6.2/14
Not used by BCA

15
Activity Management
i
A.6.2/15
Not used by BCA

B.3 Elements of procedure related to the PICS Proforma – [PICS Proforma clause A.7]

B.3.1 Kernel functional unit – [PICS Proforma A.7.1]

B.3.1.1 Supported roles – [PICS Proforma A.7.1.1]

B.3.1.1.1 Presentation-connection – [PICS Proforma A.7.1.1.1]

Role	Profile	PICS Proforma reference	Constraint / [mnemonic]
1	Initiator	c[1]	A.7.1.1.1/1
2	Responder	c[2]	A.7.1.1.1/2

[1] The value of Establishment-initiator.

[2] The value of Establishment-responder.

B.3.1.1.2 Normal data – [PICS Proforma A.7.1.1.2]

Role	Profile	PICS Proforma reference	Constraint / [mnemonic]
1	Requestor	c[1]	A.7.1.1.2/1 [P-DATA-requestor]
2	Acceptor	c[2]	A.7.1.1.2/2 [P-DATA-acceptor]

[1] The value of Normal-data-requestor.

[2] The value of Normal-data-acceptor.

B.3.1.1.3 Orderly release – [PICS Proforma A.7.1.1.3]

Role
Profile
PICS Proforma reference
Constraint / [mnemonic]

1
Requestor
c[1]
A.7.1.1.3/1

2
Acceptor
c[2]
A.7.1.1.3/2

[1] The value of Release-requestor.

[2] The value of Release-acceptor.

B.3.1.2 Supported PPDUs associated with the kernel service – [PICS Proforma A.7.1.2]

PPDU
Profile:
Sender

Profile:
Receiver
PICS Proforma reference
Constraint / [mnemonic]

1
CP
c[1]
c[2]
A.7.1.2/1

2
CPA
c[2]
c[1]
A.7.1.2/2

3
CPR
c[3]
c[1]
A.7.1.2/3
[P-REFUSE]

4
ARP
m
m
A.7.1.2/4

5
ARU
m
m
A.7.1.2/5

6
TD
c[4]
c[5]
A.7.1.2/6

[1] if [A-CON-initiator] then "m" else "-".

[2] if [A-CON-responder] then "m" else "-"

[3] The value of Establishment-responder-reject.

[4] if [P-DATA-requestor] then "m" else "-"

[5] if [P-DATA-acceptor] then "m" else "-"

NOTESYMBOL 190 \f "Symbol"§The remainder of the subclauses in A.7 are out of the scope (i) of this Profile.

B.4 Supported PPDU parameters – [PICS Proforma clause A.8]

B.4.1 Connect presentation (CP) parameters – [PICS Proforma A.8.1]

Parameter	Profile: Sender [a]	Profile: Receiver[b]	PICS Proforma reference	Constraint / [mnemonic]
1 Calling presentation selector	o	m	A.8.1/1	Implementation option; only needed if presentation selector is part of the presentation address for the implementation.
2 Called presentation selector	m	o	A.8.1/2	
3 Mode selector	m	m	A.8.1/3	Value shall be "Normal mode"
4 Presentation context definition list	m	m	A.8.1/4	
5 Default context name	*	*	A.8.1/5	May be used for simple encoding

6
Protocol version
m
m
A.8.1/6
If sent, the value shall be "version 1"; the default value is "version 1"

7
Presentation requirements
i
m
A.8.1/7
A BCA only uses the Kernel functional unit. If other functional units are proposed by the initiator, they shall be refused by the acceptor.

8
User session requirements
i
i
A.8.1/8
For a BCA, when sending, the revised session requirements will always be the same as the user session requirements, so this parameter will not be present.

For a BCA, when receiving, the parameter can be ignored because the mOSI session would never be selected.

9
User data
m
m
A.8.1/9
Contains an AARQ APDU

10
CPC Type
*
m
A.8.1/10
Ed Note - get comment from Klaus here.

[a] if NOT [A-CON-initiator] then the entire column has the value of "-"; otherwise the values are as marked.

[b] if NOT [A-CON-responder] then the entire column has the value of "–" ; otherwise the values are as marked.

NOTESYMBOL 190 \f "Symbol"§The X.410 (1984) parameters are out of the scope (i) of this Profile.

B.4.2 Connect presentation accept (CPA) PPDU – [PICS Proforma A.8.2]

Parameter
Profile:
Sender
[a]
Profile:
Receiver [b]
PICS Proforma reference
Constraint / [mnemonic]

1
Responding presentation selector
o
m
A.8.2/1
Implementation option; only needed if presentation selector is part of the presentation address for the implementation.

2
Mode selector
m
m
A.8.2/2
Value shall be "Normal"

3
Presentation context definition
result list
m
m
A.8.2/3

4
Protocol version
m
m
A.8.2/4
Value shall be "version 1"; not required to be sent because the default value is "version 1"

5
Presentation requirements
i
i
A.8.2/5

Not used by BCA

6
User session requirements
i
i
A.8.2/6
For BCA, the revised session requirements will always be the same as the user session requirements, so this parameter will not be present.

7
User data
m
m
A.8.2/7
Contains an AARE + APDU

[a] if NOT [A-CON-responder] then the entire column has the value of "–"; otherwise the values are as marked.

[b] if NOT [A-CON-initiator] then the entire column has the value of "–" ; otherwise the values are as marked.

NOTESYMBOL 190 \f "Symbol"§The X.410 (1984) parameters are out of the scope (i) of this Profile.

B.4.3 Connect presentation reject (CPR) PPDU – [PICS Proforma A.8.3]

Parameter
Profile:
Sender
[a]
Profile:
Receiver
[b]
PICS Proforma reference
Constraint / [mnemonic]

1
Responding presentation selector
o
m
A.8.3/1

2
Presentation context definition
result list
m

m
A.8.3/2

3
Protocol version
m
m
A.8.3/3

4
Default context result
*
*
A.8.3/4

5
Provider reason
m
m
A.8.3/5

6
User data
m
m
A.8.3/6
Contains an AARE- APDU

[a] if NOT [A-CON-responder] then the entire column has the value of "–"; otherwise the values are as marked.

[b] if NOT [A-CON-initiator] then the entire column has the value of "–" ; otherwise the values are as marked.

NOTESYMBOL 190 \f "Symbol"§The X.410 (1984) parameter is out of the scope (i) of this Profile.

B.4.4 Abnormal release user (ARU) PPDU – [PICS Proforma A.8.4]

Parameter
Profile:
Sender
Profile:
Receiver
PICS Proforma reference
Constraint / [mnemonic]

1
Presentation context identifier list
m
m
A.8.4/1

2
User data
m
m
A.8.4/2
Contains an ABRT APDU

NOTESYMBOL 190 \f "Symbol"§The X.410 (1984) parameters are out of the scope (i) of this Profile.

B.4.5 Abnormal release provider (ARP) PPDU – [PICS Proforma A.8.5]

Parameter
Profile:
Sender
Profile:
Receiver
PICS Proforma reference
Constraint / [mnemonic]

1
Provider reason
m
m
A.8.5/1

2
Event identifier
*
*
A.8.5/2

NOTESYMBOL 190 \f "Symbol"§PICS Proforma subclauses A.8.6 and A.8.7 are out of the scope (i) of this Profile.

B.4.6 Presentation data (TD) PPDU – [PICS Proforma A.8.8]

Parameter	Profile: Sender	Profile: Receiver	PICS Proforma reference	Constraint / [mnemonic]
1	User data	c[1]	c[2]	A.8.8/1

[1] The value of Normal-data-requestor.

[2] The value of Normal-data-acceptor.

NOTESYMBOL 190 \f "Symbol"§PICS Proforma subclauses A.8.9 through A.8.14 are out of the scope (i) of this Profile.

B.4.7 Session service primitives not carrying Presentation PCI – [PICS Proforma A.8.15]

Primitive Profile: Sender Profile: Receiver PICS Proforma reference Constraint / [mnemonic]
1 S-REL-req/ind [1] [2] A.8.15/1
2 S-REL-rsp/cnf [2] [1] A.8.15/
3 S-TG-req/ind i i A.8.15/3
4 S-TP-req/ind i i A.8.15/4
5 S-CG-req/ind i i A.8.15/5
6 S-SYNm-req/ind i

i
A.8.15/6

7
S-SYNm-rsp/cnf
i
i
A.8.15/7

8
S-SYNM-req/ind
i
i
A.8.15/8

9
s-SYNM-rsp/cnf
i
i
A.8.15/9

10
S-PER-ind
i
i
A.8.15/10

11
S-UER-req/ind
i
i
A.8.15/11

12
S-ACTS-req/ind
i
i
A.8.15/12

13
S-ACTR-req/ind
i

i
A.8.15/13

14
S-ACTI-req/ind
i
i
A.8.15/14

15
S-ACTI-rsp/cnf
i
i
A.8.15/15

16
S-ACTD-req/ind
i
i
A.8.15/16

17
S-ACTD-rsp/cnf
i
i
A.8.15/17

18
S-ACTR-req/ind
i
i
A.8.15/18

19
S-ACTE-rsp/cnf
i
i
A.8.15/19

[1] if [S-REL-requestor] then "m" else "-"

[2] if [S-REL-acceptor] then "m" else "-"

B.5 Support of syntaxes – [PICS Proforma clause A.9]

B.5.1 Transfer syntaxes supported – [PICS Proforma A.9.1]

Type	Detail	Profile	Reference to definition	Reference to restriction
1	Object identifier {joint-iso-ccitt asn1(1) basic-encoding(1)}	m	ISO/IEC 8825-1	ISO/IEC ISP 11188-1, clause 8
2	Object identifier { joint-iso-ccitt standard(0) culr(11188) mosi(3) default-transfer-syntax(2) version(1)}	*	ISO/IEC ISP 11188-3, clause F.2	none

NOTESYMBOL 190 \f "Symbol"§Other transfer syntaxes may be added to the above table based on the application(s) supported.

B.5.2 Abstract syntaxes supported – [PICS Proforma A.9.2]

Type	Detail	Profile
1	Object identifier {joint-iso-ccitt association-control(2) abstract-syntax(1) apdus(0) version1(1)}	m
2	Object identifier { joint-iso-ccitt standard(0) culr(11188) mosi(3) default-abstract-syntax(1) version(1)}	
(see annex F)		*

NOTESYMBOL 190 \f "Symbol"§Other abstract syntaxes may be added to the above table based on the application(s) supported.

B.5.3 Use of ASN.1 encoding – [PICS Proforma A.9.3]

The following table is used to indicate any coding restrictions for sending all ACSE APDUs, PPDUs and User Information on ACSE APDUs (see PICS Proforma A.9.3).

Restriction	Profile	Constraint / [mnemonic]
1	Only definite form of length encoding used	*
See ISO/IEC ISP 11188-1, 8.1.2		
2	Indefinite form of length encoding used for all constructed types	*
3	Only minimal number of octets used for definite form of length encoding	*
See ISO/IEC ISP 11188-1, 8.1.2		

4	Only primitive encoding used for OCTET STRING	*	See ISO/IEC ISP 11188-1, 8.1.7 (note 2)
5	Only primitive encoding used for BITSTRING	*	See ISO/IEC ISP 11188-1, 8.1.7 (note 2)

NOTES

1 A referencing implementation shall receive all forms of encoding.

2 ISO/IEC ISP 11188-1 restricts encoding of an OCTET STRING or a BIT STRING to only one level of constructed encoding.

NOTESYMBOL 190 \f "Symbol"§ PICS Proforma subclause A.9.4 is out of the scope (i) of this Profile.

Annex C

(normative)

tc "C mOSI requirements for Session Layer facilities"§mOSI requirements for Session Layer facilities

This annex contains the mOSI specifications for completing the Session Layer Profile Requirements List (PRL) for the selected facilities, roles and options (see 2.3).

This annex uses the tables in the Session Layer PICS Proforma (ISO/IEC 8327-2). The clause numbers and tables referenced in this annex are those of the PICS Proforma. If the PICs proforma contains tables which are not explicitly outlined in this annex, then the features of those tables are out of scope (i) for this Profile.

The specifications of this annex reference the following variables: Establishment-initiator, Establishment-responder, Establishment-responder-reject, Normal-data-requestor, Normal-data-acceptor, Release-requestor, and Release-acceptor. The values for these variables may be set by a referencing specification using the proforma provided in table D.1. The values for these variables may be set by a referencing implementation using the proforma provided in table E.1.

The entries in this annexes have not been verified with the latest version of the PICS Proforma. In particular, these tables do not reflect the supported sender and receiver length fields for each feature in clause A.8 of the PICS Proforma. The purpose and requirement for these new fields that almost double the number of entries in the PICS Proforma are an issue. This issue should be raised at the forthcoming SC21 meetings in Southampton, July 1994.

NOTESYMBOL 190 \f "Symbol"§PICS Proforma clauses A.1-A.4.1 are not covered in this Profile. The questions are answered by an implementor of the Session service.

C.1 ITU-T Rec. X.225 | ISO/IEC 8327-1 protocol versions – [PICS Proforma clause A.4.2]

Version	Profile	PICS Proforma reference	Constraint / [mnemonic]
1	Version 1		
		x	

2
Version 2
m
A.4.2/2

NOTESYMBOL 190 \f "Symbol"§PICS Proforma clause A..4.3 is not covered in this Profile. The question is answered by an implementor of the Session service.

C.2 Global statement of conformance – [PICS Proforma clause A.5]

Question
Answer
PICS Proforma reference

1
Are all mandatory features supported?
yes
A.5/1

C.3 Supported functional units and protocol mechanisms – [PICS Proforma clause A.6]

C.3.1 Functional units – [PICS Proforma A.6.1]

Functional unit	Profile	PICS Proforma reference	Constraint / [mnemonic]
1 Kernel	m	A.6.1/1	
2 Negotiated Release	i	A.6.1/2	Not used by BCA
3 Half Duplex	i	A.6.1/3	Not used by BCA
4 Duplex	m	A.6.1/4	
5 Expedited Data	i	A.6.1/5	Not used by BCA
6 Typed Data	i	A.6.1/6	Not used by BCA
7 Capability Data	i		

A.6.1/7
Not used by BCA

8
Minor Synchronize
i
A.6.1/8
Not used by BCA

9
Symmetric Synchronize
i
A.6.1/9
Not used by BCA

10
Data Separation
i
A.6.1/10
Not used by BCA

11
Major Synchronize
i
A.6.1/11
Not used by BCA

12
Resynchronize
i
A.6.1/12
Not used by BCA

13
Exceptions
i
A.6.1/13
Not used by BCA

14
Activity Management
i
A.6.1/14
Not used by BCA

C.3.2 Protocol mechanism – [PICS Proforma A.6.2]

Mechanism

Profile
PICS Proforma reference
Constraint / [mnemonic]

1
Use of transport expedited data
(Extended control Quality Of Service)
c[1]
A.6.2/1
[S-EXP-T]

2
Reuse of transport-connection
i
A.6.2/2

3
Basic concatenation
m
A.6.2/3

4
Extended concatenation (sending)
i
A.6.2/4
Not used by BCA

5
Extended concatenation (receiving)
i
A.6.2/5
Not used by BCA

6
Segmenting (sending)
i
A.6.2/6
Not used by BCA

7
Segmenting (receiving)
i
A.6.2/7
Not used by BCA

8
Max size of SS-user data > 512

(S-CONNECT)
m
A.6.2/8

9
Max size of SS-user data > 10240

(S-CONNECT)
i
A.6.2/9
Allows approximately 10K of user information in the AARQ and AARE APDUs

10
Max size of SS-user data > 9

(S-ABORT)
m
A.6.2/10

[1] The value of Transport-expedited.

C.4 Elements of procedures related to the PICS Proforma – [PICS Proforma clause A.7]

C.4.1 Kernel functional unit – [PICS Proforma A.7.1]

C.4.1.1 Supported roles for the Kernel functional unit services – [PICS Proforma A.7.1.1]

C.4.1.1.1 Session-connection – [PICS Proforma A.7.1.1.1]

Role
Profile
PICS Proforma reference
Constraint / [mnemonic]

1
Initiator
c[1]
A.7.1.1.1/1

2
Responder
c[2]
A.7.1.1.1/2

[1] The value of Establishment-initiator.

[2] The value of Establishment-responder.

C.4.1.1.2 Orderly release – [PICS Proforma A.7.1.1.2]

Role
Profile
PICS Proforma reference
Constraint / [mnemonic]

1
Requestor
c[1]
A.7.1.1.2/1

2
Acceptor
c[2]
A.7.1.1.2/2

[1] The value of Release-requestor.

[2] The value of Release-acceptor.

C.4.1.1.3 Normal data transfer – [PICS Proforma A.7.1.1.3]

Role
Profile
PICS Proforma reference
Constraint / [mnemonic]

1
Requestor
c[1]
A.7.1.1.3/1

2
Acceptor
c[2]
A.7.1.1.3/2

[1] The value of Normal-data-requestor.

[2] The value of Normal-data-acceptor.

C.4.1.2 Support for the SPDUs associated with the Kernel services – [PICS Proforma A.7.1.2]

SPDU	Profile: Sender	Profile: Receiver	PICS Proforma reference	Constraint / [mnemonic]
1 Connect (CN)	c[1]	c[2]	A.7.1.2/1	
2 Overflow accept (OA)	i	i	A.7.1.2/2	Not used by BCA
3 Connect Data Overflow (CDO)	i	i	A.7.1.2/3	Not used by BCA. This provides a maximum of 10K of Session user data.
4 Accept (AC)	c[2]	c[1]	A.7.1.2/4	
5 Refuse (RF)	c[3]	c[1]	A.7.1.2/5	
6 Finish (FN)	c[4]			

c[5]
A.7.1.2/6

7
Disconnect (DN)
c[5]
c[4]
A.7.1.2/7

8
Abort (AB)
m
m
A.7.1.2/8

9
Abort Accept (AA)
i
m
A.7.1.2/9
This profile recommends that the response to receiving an Abort (AB) SPDU is to issue a T-DISCONNECT request. The acceptor may not follow this recommendation, so an AA may be received. If received, the requestor shall issue a T-DISCONNECT request.

10
Data Transfer (DT)
c[6]
c[7]
A.7.1.2/10

11
Prepare (PR)
c[8]
c[8]
A.7.1.2/11
Only sent/received if transport expedited is negotiated for the supporting T-connection.

[1] The value of Establishment-initiator.

[2] The value of Establishment-responder.

[3] The value of Establishment-responder-reject.

[4] The value of Release-requestor.

[5] The value of Release-acceptor.

[6] The value of Normal-data-requestor.

[7] The value of Normal-data-acceptor.

[8] The value of Transport-expedited.

C.4.1.3 Support for the SPDUs associated with Token Exchange – [PICS Proforma A.7.1.3]

Profile: Sender	Profile: Receiver	PICS Proforma reference	Constraint / [mnemonic]
1	Give Token (GT)	c[1]	A.7.1.3/1
		c[2]	For the Duplex functional unit, these SPDUs are only used as a null category 0 SPDU in conjunction with the Data (DT) SPDU.
2	Please Token (PT)	i	A.7.1.3/2
		i	This Profile recommends only sending GT as the null category 0 SPDU.

c[1] The value of Normal-data-requestor.

c[2] The value of Normal-data-acceptor.

NOTESYMBOL 190 \f "Symbol"§The remainder of the subclauses in A.7 is out of the scope (i) of this Profile.

C.5 Supported SPDU parameters – [PICS Proforma clause A.8]

C.5.1 Connect (CN) SPDU – [PICS Proforma A.8.1]

C.5.1.1 Connection Identifier – [PICS Proforma A.8.1.1]

PGI "Connection Identifier"
Profile:
Sender

Profile:
Receiver

PICS Proforma reference
Constraint / [mnemonic]

1
Calling SS-user Reference
i
i
A.8.1.1/1
Not used by BCA

2
Common Reference
i
i
A.8.1.1/2
Not used by BCA

3
Additional Reference Information
i
i
A.8.1.1/3
Not used by BCA

C.5.1.2Connect/Accept Item – [PICS Proforma A.8.1.2]

C.5.1.2.1 Connect/Accept Item parameters – [PICS Proforma A.8.1.2.1]

PGI "Connect/Accept
Item"
Profile:
Sender
[a]
Profile:
Receiver
[b]
PICS Proforma reference
Constraint / [mnemonic]

1

Protocol Options
m
m
A.8.1.2.1/1
For BCA, basic concatenation shall be indicated

2
TSDU maximum size
i
m
A.8.1.2.1/2
If received, the connection shall be refused.

3
Version Number
m
m
A.8.1.2.1/3
Value shall be "version 2"

4
Initial Serial Number
i
i
A.8.1.2.1/4
Not used by BCA

5
Token Setting Item
i
i
A.8.1.2.1/5
Not used by BCA

6
Second Initial Serial Number
i
i
A.8.1.2.1/6
Not used by BCA

[a] if NOT [A-CON-initiator] then the entire column has the value of "–" ; otherwise the value is as marked.

[b] if NOT [A-CON-responder] then the entire column has the value of "–" ; otherwise the value is as marked.

C.5.1.2.2 Presence of Connect/Accept Item – [PICS Proforma A.8.1.2.2]

Profile: Sender [a]	Profile: Receiver [b]	PICS Proforma reference	Constraint / [mnemonic]
1			
Sending		m	–
		A.8.1.2.2/1	
2			
Receiving		–	m
		A.8.1.2.2/2	

[a] if NOT [A-CON-initiator] then the entire column has the value of "–" ; otherwise the value is as marked.

[b] if NOT [A-CON-responder] then the entire column has the value of "–" ; otherwise the value is as marked.

C.5.1.3Single Items – [PICS Proforma A.8.1.3]

Single Items
Profile:
Sender
[a]
Profile:
Receiver
[b]
PICS Proforma reference
Constraint / [mnemonic]

1
Session User Requirements
m
m
A.8.1.3/1
Value shall be "duplex". If other requirements (functional units) are received in the CN SPDU, they shall not be included on the AC SPDU.

2
Calling Session Selector
o
m
A.8.1.3/2
Implementation option; only needed if session selector is part of the presentation address for the implementation.

3
Called Session Selector
m
o
A.8.1.3/3

4
Data Overflow
i
i
A.8.1.3/4
Not used by BCA

5
User Data
m
m
A.8.1.3/5
Contains a CP PPDU

6
Extended User Data
m
m
A.8.1.3/6

[a] if NOT [A-CON-initiator] then the entire column has the value of "–" ; otherwise the value is as marked.

[b] if NOT [A-CON-responder] then the entire column has the value of "–" ; otherwise the value is as marked.

C.5.2 Accept (AC) SPDU – [PICS Proforma A.8.4]

C.5.2.1 Connection Identifier – [PICS Proforma A.8.4.1]

PGI "Connection Identifier"
Profile:
Sender
Profile:
Receiver
PICS Proforma reference
Constraint / [mnemonic]

1
Called SS-user Reference
i
i
A.8.4.1/1
Not used by BCA

2
Common Reference
i
i
A.8.4.1/2
Not used by BCA

3
Additional Reference Information
i
i
A.8.4.1/3
Not used by BCA

C.5.2.2 Connect/Accept Item – [PICS Proforma A.8.4.2]

C.5.2.2.1 Connect/Accept Item parameters – [PICS Proforma A.8.4.2.1]

PGI "Connect/Accept Item"	Profile: Sender [a]	Profile: Receiver [b]	PICS Proforma reference	Constraint / [mnemonic]
1 Protocol Options	m	m	A.8.4.2.1/1	Basic concatenation shall be indicated
2 TSDU maximum size	i	m	A.8.4.2.1/2	If received, the connection shall be aborted.
3 Version Number	m	m	A.8.4.2.1/3	Value shall be version 2
4 Initial Serial Number	i	i	A.8.4.2.1/4	Not used by BCA
5 Token Setting Item	i	i	A.8.4.2.1/5	Not used by BCA

6
Second Initial Serial Number
i
i
A.8.4.2.1/6
Not used by BCA

[a] if NOT [A-CON-responder] then the entire column has the value of "–" ; otherwise the value is as marked.

[b] if NOT [A-CON-initiator] then the entire column has the value of "–" ; otherwise the value is as marked.

.

C.5.2.2.2 Presence of Connect/Accept Item – [PICS Proforma A.8.4.2.2]

Profile:
Sender
[a]
Profile:
Receiver
[b]
PICS Proforma reference
Constraint / [mnemonic]

1
Sending
m
–
A.8.4.2.2/1

2
Receiving
–
m
A.8.4.2.2/2

[a] if NOT [A-CON-responder] then the entire column has the value of "–" ; otherwise the value is as marked.

[b] if NOT [A-CON-initiator] then the entire column has the value of "–" ; otherwise the value is as marked.

C.5.2.3Single Items – [PICS Proforma A.8.4.3]

Single Items
Profile:
Sender
[a]
Profile:
Receiver
[b]
PICS Proforma reference
Constraint / [mnemonic]

1
Token Item
i
i
A.8.4.3/1

2
Session User Requirements
m
m
A.8.4.3/2
Value shall be "duplex"

3
Enclosure Item
i
i
A.8.4.3/3

4
Calling Session Selector
o
m
A.8.4.3/4
It is recommended that this parameter not be sent; if sent, it shall be identical to the Calling Session Selector on the CN.

5
Responding Session Selector
o
m
A.8.4.3/5
Implementation option; only needed if session selector is part of the presentation address for the implementation.

6
User Data
m
m
A.8.4.3/6
Contains a CPA PPDU

[a] if NOT [A-CON-responder] then the entire column has the value of "–" ; otherwise the value is as marked.

[b] if NOT [A-CON-initiator] then the entire column has the value of "–" ; otherwise the value is as marked.

C.5.3 Refuse (RF) SPDU – [PICS Proforma A.8.5]

C.5.3.1 Connection Identifier – [PICS Proforma A.8.5.1]

PGI "Connection Identifier"
Profile:
Sender
Profile:
Receiver
PICS Proforma reference
Constraint / [mnemonic]

1
Called SS-user Reference
i
i
A.8.5.1/1
Not used by BCA

2
Common Reference
i
i
A.8.5.1/2
Not used by BCA

3
Additional Reference Information
i
i
A.8.5.1/3
Not used by BCA

C.5.3.2 Single Items – [PICS Proforma A.8.5.2]

Single Items
Profile:
Sender
[a]
Profile:
Receiver
[b]
PICS Proforma reference
Constraint / [mnemonic]

1
Transport Disconnect

i
i
A.8.5.2/1

2
Session User Requirements
m
m
A.8.5.2/2
This parameter indicates the functional units supported by the sender.

3
Version Number
m
m
A.8.5.2/3
This parameter indicates the version(s) supported by the sender.

4
Enclosure Item
i
i
A.8.5.2/4

5
Reason Code
m
m
A.8.5.2/5

[a] if NOT [P-REFUSE] then the entire column has the value of "–" ; otherwise the value is as marked.

[b] if NOT [A-CON-initiator] then the entire column has the value of "–" ; otherwise the value is as marked.

C.5.4 Finish (FN) SPDU – [PICS Proforma A.8.6]

Single Items
Profile:
Sender
[a]
Profile:

Receiver
[b]
PICS Proforma reference
Constraint / [mnemonic]

1
Transport Disconnect
i
i
A.8.6/1

2
Enclosure Item
i
i
A.8.6/2

3
User Data
m
m
A.8.6/3
Contains an RLRQ APDU

[a] if NOT [A-REL-requestor] then the entire column has the value of "–" ; otherwise the value is as marked.

[b] if NOT [A-REL-acceptor] then the entire column has the value of "–" ; otherwise the value is as marked.

C.5.5 Disconnect (DN) SPDU – [PICS Proforma A.8.7]

Single Items
Profile:
Sender
[a]
Profile:
Receiver
[b]
PICS Proforma reference
Constraint / [mnemonic]

1
Enclosure Item
i
i
A.8.7/1

2
User Data
m
m
A.8.7/2
Contains an RLRE APDU

[a] if NOT [A-REL-acceptor] then the entire column has the value of "–" ; otherwise the value is as marked.

[b] if NOT [A-REL-requestor] then the entire column has the value of "–" ; otherwise the value is as marked.

NOTESYMBOL 190 \f "Symbol"§ PICS Proforma subclause A.8.8 is out of scope (i) of this Profile.

C.5.6 Abort (AB) SPDU – [PICS Proforma A.8.9]

Single Items
Profile:
Sender
Profile:
Receiver
PICS Proforma reference
Constraint / [mnemonic]

1

Transport Disconnect
m
m
A.8.9/1

2
Enclosure Item
i
i
A.8.9/2

3
Reflect Parameter Values
m
m
A.8.9/3
If protocol error, contains sender defined value. Processing by receiver is a local matter.

4
User Data
m
m
A.8.9/4
If user abort, contains an ARP or an ARU PPDU

C.5.7 Data Transfer (DT) SPDU – [PICS Proforma A.8.11]

Single Items
Profile:
Sender
[a]
Profile:
Receiver
[b]
PICS Proforma reference
Constraint / [mnemonic]

1
Enclosure Item
i
i
A.8.11/1

2
User Information Field

m
m
A.8.11/2

[a] if NOT [P-DATA-requestor] then the entire column has the value of "–" ; otherwise the value is as marked.

[b] if NOT [P-DATA-acceptor] then the entire column has the value of "–" ; otherwise the value is as marked.

NOTESYMBOL 190 \f "Symbol"§Subclauses A.8.12 through A.8.15 are out of the scope (i) of this Profile.

C.5.8 Give Tokens (GT) SPDU – [PICS Proforma A.8.16]

Single Items
Profile:
Sender
Profile:
Receiver
PICS Proforma reference
Constraint / [mnemonic]

1
Token Item
i
i
A.8.16/1

2
Enclosure Item
i
i
A.8.16/2

3
User Data
i
i
A.8.16/3

NOTE—The remainder of the clauses in the Session PICS Proforma are out of the scope (i) of this

Profile.

Annex D

(normative)

tc "D Profile requirements list proforma"§Profile requirements list proforma

D.1 Use of tables D.1 and D.2 by a referencing specification

The tables in annexes A, B, and C define the requirements for the mOSI Profile as a function of a set of variables and a set of open parameters (see 2.3). The list of the mOSI variables and their allowed (compliant) values may be found in table D.1, rows 1–9. The list of the open (*) parameters and their allowed (compliant) values may be found in table D.2.

When "Specification's choice" status values are selected for tables D.1 and D.2, "Profile status" values are determined for features in the tables of annex A (ACSE requirements), annex B (Presentation requirements), and annex C (Session requirements). For example, in annex A the mOSI variable Establishment-initiator is used in table A.2.1. In line 1 of table A.2.1, the [ACSE] Initiator role will be assigned the Profile status value ("m", "o", or "i") of the variable Establishment-initiator. This variable is used several other places in Annexes A, B, and C.

A referencing specification may use Tables D.1 and D.2 as a basis of its upper layer Profile Requirements List (PRL). Tables D.1 and D.2 provide a proforma for specifying the mOSI variables and open parameters. A referencing specification can use these tables and any supplemental information (see clause D.2) to define concisely its upper layer requirements based on the mOSI profile. It may do this by completing the "Specification's choice" columns in tables D.1 and D.2.

If the upper layer requirements of the referencing specification are defined by completing tables D.1 and D.2 and if the values selected are compliant, the referencing specification may claim mOSI compliance. Such a claim indicates that upper layer requirements of the referencing specification are identified by the features specified in this Profile as indicated by tables D.1 and D.2 plus any supplemental information mentioned in clause D.2.

Table D.2 may be used by a referencing specification to specify the open (*) parameters in annexes A, B, and C. If the open parameters are either all "m" (and "–" for non-supported roles) or all "o" (and "–" for non-supported roles) or all "i" (and "–" for non-supported roles) , this table need not be completed. In this case, line 13 of table D.1 would have the answer "m", or "o", or "i", respectively.

D.2 Completing a PRL based on tables D.1 and D.2

Several additional pieces of information are needed to complete the definition of the PRL of a referencing specification after completing tables D.1 and D.2:

a) List of transfer syntaxes supported;

b) Abstract syntaxes supported; and

c) Use of ASN.1 encoding.

Each is discussed below.

Table D.1 – Profile requirements list proforma

Item / variable	Compliant choice	Specification's choice	Constraint / value
1 Establishment-initiator	m; o; i; -		Both shall not be "i".
2 Establishment-responder	m; o; i; -		
3 Establishment-responder-reject	m; o; i; -		The value shall be "i" if Establishment-responder has the value "i"
4 Normal-data-requestor	m; o; i; -		Both may be "i".
5 Normal-data-acceptor	m; o; i; -		
6 Release-requestor	m; o; i; -		Both may be "i".
7			

Release-acceptor
m; o; i; -

8
Authentication
m; o; i; -

9
Application-context-negotiation
m; o; i; -

10
Transport-expedited
m; o; i; -

11
Number of presentation-contexts required
1 or more

The value chosen includes the presentation-context used for ACSE PDUs.

12
ISO/IEC ISP 11188-1 compliance?
yes

If the answer is not "yes", the referencing specification may not claim mOSI compliance

13
Status values for all open (*) parameters (see table D.2)
all "m"; all "o"; all "i"; or "mixed"

If the answer is "mixed" (i.e. not all "m" and "–", or not all "o" and "–", or not all "i" and "–"), details shall be given in table D.2

Table D.2 – Open parameters (*)

Referenced table (in annexes A, B and C)	Parameter	Specification's statement — Sender [a]	Specification's statement — Receiver [a]	Constraint / value
1	A.6.1 Calling AE title			Includes both the AP title and AE
2	[AARQ] Called AE title			qualifier for each
3	Calling invocation ids			Includes both the AP invocation identifier and the AE invocation
4	Called invocation ids			identifier for each
5	User Information			

6
A.6.2
Responding AE title

Includes both the AP title and AE qualifier

7
[AARE]
Responding Invocation Identifiers

Includes both the AP invocation identifier and the AE invocation identifier

8

User Information

9
A.6.3
Reason

10
[RLRQ]
User Information

11
A.6.4
Reason

12
[RLRE]
User Information

13
A.6.5

[ABRT]
User Information

14
A.7.1

[AARQ and
Form 1 (Directory name)

For Receiver, compliant answer is "m" or "–". That is, if AE titles

15
AARE]
Form 2 (Object id+integer)

are supported for receiving, both forms are mandatory.

16
B.4.1

[CP]
Default context name

May be used for simple encoding of ACSE and user PCI

17

CPC Type

m
Shall be mandatory for receipt

18
B.4.3

[CPR]
Default context result

19
B.4.5

[ARP]
Event Identifier

[a] Compliant answer for each row is "m", "o", "i", "x" or "–", unless indicated otherwise.

D.2.1 List of transfer syntaxes supported

Table B.5.1 (Presentation Layer PICS Proforma table A.10.1) lists the object identifiers for the ASN.1 Basic Encoding Rules (BER) and for the Default Abstract Syntax for Minimal OSI defined in F.1.

It is suggested that a referencing specification make a list (based on table B.5.1) of the transfer syntaxes required for its application. This list shall include the object identifier for BER that is used for Presentation and ACSE PCI.

D.2.2 Abstract syntaxes supported

Table B.5.2 (Presentation Layer PICS Proforma table A.10.2) lists the object identifiers for ASN.1 and for the Default Transfer Syntax for Minimal OSI defined in F.2.

It is suggested that a referencing specification make a list (based on table B.5.2) of the abstract syntaxes required for its application. This list shall include the object identifier for ASN.1 that is used for ACSE PCI.

D.2.3 Use of ASN.1 encoding.

Table B.5.3 (Presentation Layer PICS Proforma table A.10.3) contains a list of questions used to indicate any coding restrictions for sending all ACSE APDUs, PPDUs and User Information on ACSE APDUs.

It is suggested that a referencing specification make a table based on table B.5.3 and add their status values.

D.3 A referencing specification's upper layer PRL

As discussed in 2.3, this mOSI Profile defines its upper layer requirements in terms of facilities, roles and options. It does this by defining variables and open parameters whose values are assigned by a referencing specification.

The referencing specification, itself, may elect to define its upper layer requirements in terms of its own facilities, roles and options. It could do this by defining its own variables and open parameters that map to mOSI variables and open parameters. A specification that references it (in a recursive sense) would then select values for the referencing specification's variables and open parameters. These, in turn, would define values for mOSI variables and open parameters producing a completed upper layer requirements list as defined by this Profile.

For example, consider a referencing specification that defined roles for Client and Server to be selected by its referencing specification. The mOSI variables for Establishment-initiator and Release-initiator would map to the referencing specification's Client variable (role); the mOSI variables Establishment-responder and Release-acceptor would map to the referencing specification's Server variable (role). The appropriate answer in table D.1 would be "*" for Establishment-initiator under the "Specification's statement" column.

Annex E

(normative)

tc "E Implementation conformance statement proforma"§Implementation conformance statement proforma

E.1 Use of tables E.1 and E.2 by a referencing implementation

The tables in annexes A, B, and C define the requirements for the mOSI Profile as a function of a set of variables and a set of open parameters (see 2.3). A summary of the mOSI variables and their allowed (conformant) support answers may be found in table E.1, rows 1–9. A summary of the open (*) parameters and their allowed (conformant) support answers may be found in table E.2.

The answers ("yes", "no", "–") given by a referencing implementation implicitly assign status values ("m", "o", and "–") for the mOSI variables used in the tables of annexes A, B, and C. The implicit assignment is:

SYMBOL 151 \f "Times New Roman" \s 10 \h§ a support answer of "yes" translates to a status value of "m";

SYMBOL 151 \f "Times New Roman" \s 10 \h§ a support answer of "no" translates to a status value of "i"; and

SYMBOL 151 \f "Times New Roman" \s 10 \h§ a support answer of "–" translates to a status value of "–".

When Implementation support answers are selected for tables E.1 and E.2, values for the associated mOSI variables and open parameters are determined as described above. These values, in turn, determine Profile status values for all the tables in annex A (ACSE requirements), annex B (Presentation requirements), and annex C (Session requirements). For example, the mOSI variable Establishment-initiator is used in table A.2.1. In line 1 of table A.2.1, the [ACSE] Initiator role is assigned the value ("m", "o", or "i") of the variable Establishment-initiator. This variable is used several other places in Annexes A, B, and C. In this example the Establishment-initiator variable (and the [ACSE] Initiator role) has the value:

"m" — if the implementation's statement is "yes";

"i" — if the implementation's statement is "no".

An implementation of the upper layers may use tables E.1 and E.2 as a basis to do the following:

a) summarize the upper layer facilities it supports

b) complete the ACSE, Presentation, and Session PICS proformas.

It may do this by completing the "Implementation's statement" columns in tables E.1 and E.2.

If the upper layer facilities (or a subset thereof) of the implementation are defined by completing tables E.1 and E.2 and if the choices made are conformant, the referencing implementation may claim mOSI conformance. Such a claim indicates that the implementation supports some or all of the features specified in this Profile. The implementation may in fact support more of the upper layer facilities of this ProfileSYMBOL 190 \f "Symbol"§without violating any of the features of this Profile.

Table E.2 may be used by a referencing implementation to specify its support answers for the open (*) parameters in annexes A, B, and C. If the open parameters are either all "yes" (and "–" for non-supported roles) or all "no" (and "–" for non-supported roles), this table need not be completed. In this case, line 11 of table E.1 would have the answer "yes" or "no", respectively.

Table E.1 – Implementation support questionnaire

Item / variable	Conformant answer	Implementation's answer	Constraint / value
1 Establishment-initiator	yes; no		Both shall not be "no".
2 Establishment-responder	yes; no		If "yes", the associated variable is implicitly assigned the value of "m"; otherwise the variable is "i"
3 Establishment-responder-reject	yes; no; –		Answer shall be "–" if Establishment-responder is "no". If "yes", Establishment-responder-reject is implicitly assigned the value of "m"; otherwise the value is "i"
4 Normal-data-requestor	yes; no		Both may be "no".
5 Normal-data-acceptor	yes; no		If "yes", the associated variable is implicitly assigned the value of "m"; otherwise the value is "i"
6 Release-requestor	yes; no		Both may be "no".
7 Release-acceptor	yes; no		

If "yes", the associated variable is implicitly assigned the value of "m"; otherwise the value is "i"

8
Authentication
yes; no

If "yes", the Authentication is implicitly assigned the value of "m"; otherwise the value is "i"

9
Application-context-negotiation
yes; no

If "yes", the Application-context-negotiation is implicitly assigned the value of "m"; otherwise the value is "i"

10
Transport-expedited
yes; no

11
Number of presentation-contexts supported per association
1 or more

The value includes the presentation-context used for ACSE PDUs.

12
ISO/IEC ISP 11188-1 conformance?
yes

If the answer is not "yes", the referencing implementation may not claim mOSI conformance

13
Support for all "*" parameters
yes; no; mixed

If the answer is "mixed" (i.e., not all "yes" and "–", or not all "no" and "–"), details shall be given in table E.2

Table E.2 – Open parameters

Referenced table (in annexes A, B and C) Parameter	Implemen-tation's statement — Sender [a]	Implemen-tation's statement — Receiver [a]	Constraint / value
1 A.6.1 Calling AE title			Includes both AP title and AE
2 [AARQ] Called AE title			qualifier for each
3 Calling invocation identifiers			Includes both AP invocation identifier and AE invocation
4 Called invocation identifiers			identifier for each
5 User Information			
6			

A.6.2
Responding AE title

Includes both AP title and AE invocation identifier

7
[AARE]
Responding Invocation Identifiers

Includes both AP invocation identifier and AE invocation identifier

8

User Information

9
A.6.3
Reason

10
[RLRQ]
User Information

11
A.6.4
Reason

12
[RLRE]
User Information

13

A.6.5
[ABRT]
User Information

14
A.7.1

[AARQ and
Form 1 (Directory name)

For Receiver, conformant answer is "yes" or "–"

15
AARE]
Form 2 (Object id+integer)

16
B.4.1
[CP]
Default Context Name

May be used for simple encoding of ACSE and user PCI.

17

CPC Type

yes
Shall be supported for receipt

18
B.4.3

[CPR]
Default context result

19
B.4.5

[ARP]
Event Identifier

[a] Conformant answer for each row is "yes", "no" or "–".

Table E.3 – Presentation and Session selectors

Referenced table(in annexes A, B and C)	Parameter	Conformant support answer— Sender	Conformant support answer— Receiver	Implementa-tion's statement—Sender	Implementa-tion's statement—Receiver
1	B.4.1/1	[CP] Calling presentation selector	yes; no; —	yes; —	
2		Called presentation selector	yes; —	yes; ignore; —	
3	B.4.2	[CPA] Responding presentation selector	yes; no; —	yes; —	
4	B.4.3	[CPR] Responding presentation selector	yes; no; —	yes; —	
5					

C.4.1.3

[CN]
Calling session selector
yes; no; —
yes; —

6

Called session selector
yes; —
yes; ignore; —

7
C.4.2.3

[AC]
Calling session selector
yes; no; —
yes

8

Responding session selector
yes; no; —
yes; —

E.2 Completing a PICS based on tables E.1 and E.2

Several additional pieces of information are needed to complete the PICs of an implementation after completing tables E.1 and E.2:

a) List of transfer syntaxes supported;

b) Abstract syntaxes supported;

c) Use of ASN.1 encoding; and

d) Presentation and Session selectors supported

Each is discussed below.

E.2.1 List of transfer syntaxes supported

Table B.5.1 (Presentation Layer PICS Proforma table A.10.1) lists the object identifiers for the ASN.1 Basic Encoding Rules (BER) and for the Default Abstract Syntax for Minimal OSI defined in F.1.

A referencing implementation should make a list (based on table B.5.1) of the supported transfer syntaxes. This list shall include the object identifier for BER that is used for Presentation and ACSE PCI.

E.2.2 Abstract syntaxes supported

Table B.5.2 (Presentation Layer PICS Proforma table A.10.2) lists the object identifiers for ASN.1 and for the Default Transfer Syntax for Minimal OSI defined in F.2.

A referencing implementation should make a list (based on table B.5.2) of the supported abstract syntaxes. This list shall include the object identifier for ASN.1 that is used for ACSE PCI.

E.2.3 Use of ASN.1 encoding.

Table B.5.3 (Presentation Layer PICS Proforma table A.10.3) contains a list of questions used to indicate any coding restrictions for sending all ACSE APDUs, PPDUs and User Information on ACSE APDUs.

A referencing implementation should make a table based on table B.5.3 and add their status values.

E.2.4 Presentation and Session selectors supported

The approach taken by the mOSI Profile concerning presentation and session selectors is as follows. The implementation may decide whether it sends presentation and session selectors that identify itself. However, it shall be capable of receiving selectors from its peer.

Table E.3 list the selector parameters. It is suggested that a referencing implementation make a table based on table E.3 and add its statements for the support answers.

Annex F

(normative)

tc "F Minimal OSI object identifiers"§Minimal OSI object identifiers

The following are the object identifiers for the default abstract syntax, default transfer syntax and default application context for use with Minimal OSI. These object identifiers are registered with this Profile.

F.1 Default abstract syntax for minimal OSI

This object identifier can be used as the abstract syntax name when the application protocol (above ACSE) can be treated as single presentation data values (PDVs). Each PDV is a sequence of consecutive octets without regard for semantic or other boundaries. The object identifier may also be used when, for pragmatic reasons, the actual abstract syntax of the application is not identified in Presentation Layer negotiation.

The OBJECT IDENTIFIER for the default abstract syntax is:

{ joint-iso-ccitt standard(0) culr(11188) mosi(3) default-abstract-syntax(1) version(1)}

NOTES

1. Applications specified using ASN.1 should not use the default abstract syntax.

2. As this OBJECT IDENTIFIER is used by all applications using the default abstract syntax for mOSI, it cannot be used to differentiate between applications. One of the ACSE parameters, e.g. AE-Title, may be used to differentiate between applications.

F.2 Default transfer syntax for minimal OSI

A transfer syntax is the representation of the abstract syntax during data transfer. If an application doesn't make a distinction between the abstract and transfer syntax, the same object identifier should be used to denote both syntaxes.

In the case where:

a) the abstract and transfer syntax are not the same; and

b) the default abstract syntax object identifier has been used (see F.1 above)

the following default transfer syntax object identifier may be used:

{ joint-iso-ccitt standard(0) culr(11188) mosi(3) default-transfer-syntax(2) version(1)}

F.3 Default application context for minimal OSI

The default application context for mOSI is the application context used to denote the application's universe of discourse.

The OBJECT IDENTIFIER for the mOSI default application context is

{ joint-iso-ccitt standard(0) culr(11188) mosi(3) default-application-context(3)}

This application context supports the execution of any application using the default abstract syntax defined in F.1.

Annex G

(informative)

tc "G Minimal OSI concepts"§Minimal OSI concepts

This annex defines concepts used in the Minimal OSI upper layer facilities.

G.1 Definitions of minimal OSI upper layer facilities

The minimal OSI Profile (mOSI) specifies a minimal set of OSI upper layer facilities that support basic communications applications. A basic communications application (BCA) simply requires the ability to open and close communications with a peer and to send and receive messages with the peer. The OSI upper layer facilities are defined by the ACSE, Presentation and Session protocol specifications. When these facilities are specified as a set of interrelated standards for the purpose of providing a common service that functional specification is a Stack specification. An implementation of an identified stack specification is a Stack or a stack implementation. The specification that defines the minimal facilities of the Session Layer, Presentation Layer, and ACSE is the mOSI specification or the mOSI stack specification – this Profile.

A functional specification of a formal programmatic interface and a set of supporting local services for an identified stack specification is a Platform specification. A Platform is an implementation of an identified platform specification. The functional specification of a formal programmatic interface and a set of supporting local services for the mOSI stack specification is the mOSI platform specification and an implementation of the mOSI platform specification is a mOSI platform.

A functional specification of the formal programmatic interface to an identified stack specification is an API specification. An API is an implementation of an identified API specification. Likewise, a mOSI API specification is a functional specification of the formal programmatic interface to the mOSI stack specification.

G.2 Use of the mOSI stack specification

The mOSI stack specification is intended for connection-oriented applications that do not require all of the services of the upper layers, but only the basic communications services. Figure G.1 is a Venn Diagram which represents a classification of the universe of all possible connection-oriented applications. Each subset of the universe applicable to this discussion is supplied below.

EMBED MSDraw * mergeformatµ §

G.2.1 Connection-oriented applications

The set of connection-oriented applications consists of

a) applications needing only basic communications services. A basic communications application requires the ability to open and close communications and to send and receive messages.

b) applications needing more that basic communications services

G.2.2 Non-basic connection-oriented applications needing more than basic services

The set of connection-oriented applications needing more than basic communications facilities, from an OSI perspective, includes those applications which use one or more of the following facilities: Session major or minor synchronize; resynchronize; activity management. Some examples are

a) RTSE-based applications, e.g. MTA-MTA transfer of X.400

b) FTAM with optional recovery

c) TP with optional two-phase commitment (i.e. CCR)

d) some optional aspects of VTP

G.2.3 Connection-oriented applications needing only basic services

The set of connection-oriented applications needing only basic communications services include those applications which require the ability to open and close communications and to send and receive messages. Some examples are

a) Kernel applications, which are written specifically for OSI services.

b) Migrant applications, which are "non-OSI applications" such as those currently supported by TCP programmatic interfaces. The X Window system (X) and IPS applications are examples of potential migrant applications.

G.2.3.1 Kernel applications

The set of all possible Kernel applications include those which are written specifically for OSI services

that only require basic communications services. Some examples are:

a) ISO and ITU-T defined applications which do not use the RTSE, e.g. UA-MS transfer of X.400;

b) FTAM implementations which do not use recovery or restart data transfer;

c) TP implementations which do not use two-phase commitment;

d) VTP without destructive interrupt facility; and

e) all ROSE based applications.

These applications may access the mOSI stack by using either an API or by mapping directly onto mOSI. Applications using an API are considered to be platform applications. Those applications mapping directly onto mOSI are considered to be stand-alone applications.

G.2.3.2 Migrant applications

The set of all possible Migrant applications include those that are either to be migrated from TCP to OSI or they are applications that require the least common denominator communication facility – because they must operate over several "transport" mechanisms. Some examples are:

a) TCP/IP applications;

b) Connection-Oriented "user written" applications; and

c) X-windows and IPS applications.

These applications may access the mOSI stack by using either a migrant API or by mapping directly onto mOSI. A migrant API is considered to be a platform for accessing the mOSI stack. Therefore those applications which use a migrant API are considered to be platform applications. Those applications mapping directly onto mOSI are considered to be stand-alone applications.

G.3 Users of the mOSI stack specification

Out of the set of all connection-oriented applications, those applications using only basic communications services are possible users of the mOSI stack specification. This specification is intended to address implementors of migrant and kernel applications using either a platform or accessing the mOSI stack directly (stand-alone users). This specification is also intended to be referenced by ISP designers.

G.4 OSI misconceptions

Today's APIs do not efficiently support basic communications applications with a conformant seven-layer stack. However basic communications applications (both migrant and kernel) represent the overwhelming majority of potential OSI applications.

OSI is most often viewed as only being fat and slow. This is because OSI is perceived as the implementation of the full function upper layer stack with a corresponding complex, mostly unneeded,

and difficult to use full API. The facilities required by migrant and kernel applications represent approximately 5% of the overall functionality provided by a full stack.

A full function stack and its API are intended for those applications needing more than the basic communications services, such as MTA to MTA transfer of X.400 and TP with CCR. The facilities of the full upper layers simplify the design of sophisticated distributed applications that require check pointing with recovery - applications such as RTSE and CCR.

For a basic communications mapping onto a full stack, the unused 95% of functionality intended for a fuller application, could potentially impact performance. Having an API which only provides mapping onto basic services would make OSI less intimidating.

Another misconception about OSI deals with the use of ASN.1 Some believe that the use of OSI predicates the use of ASN.1 for application semantics. This is not the case. ASN.1 abstract definitions and the use of ASN.1 Basic Encoding Rules is just one of the options available to an applications designer.

Annex H

(informative)

tc "H Implementation considerations"§Implementation considerations

This Internationally Standardized Profile is not an implementation specification. However, the size and efficiency of an implementation of OSI (any OSI implementation, not just of mOSI) is significantly affected by the implementation design. The OSI 7-layer model is a protocol specification model and in many cases may not be the best way to implement OSI.

This informative annex is concerned only with the implementation of the upper three layers of OSI. In particular, this annex is concerned with an implementation of the mOSI specified subset of the OSI upper layers facilities. This annex makes several implementation suggestions. Experience has shown that these implementation approaches yield small and fast implementations—especially when compared to some of the well known OSI proof-of-concept implementations.

H.1 Layering for mOSI implementations

The services and protocols for the upper three layers of OSI are specified separately. However, considerable efficiency can be gained if all three protocol machines are combined as one module and not as three separate modules requiring the definition of formal interfaces between them. This is especially true if operating system context switches occur when transferring between different modules.

NOTE—Currently no formally defined interfaces exist for Session and Presentation.

Combining layer protocol machines applies not only to mOSI implementations, but also to all OSI upper layer implementations. However, combining layers precludes testing them individually. Individual layer testing is not recommended. The three upper layer protocols provide an integrated set of services; these services are not useful individually.

H.2 PDU generation for mOSI implementations

mOSI supports uncomplicated application protocols, i.e., byte stream or simple record oriented data transfer (which probably constitutes 95 to 99% of user application protocols). Therefore, the encoding of embedded PDU headers (PCI) for all three protocols is uniform. Using pre-defined protocol headers allows significant gains in protocol machine efficiency. This is especially true for data PDUs— the PDUs that generally constitute the majority of PDUs exchanged.

The THINOSI Internet Draft RFC is an example of explicit pre-defined PDU encodings for mOSI implementations. The RFC contains a full description of this technique including BER encodings for all upper layer PDUs.

H.3 Parsing incoming protocol for mOSI

An upper layer protocol machine that only supports the facilities defined in mOSI need only recognize

a very limited subset of all potential OSI PDU sequences. In particular, they need not recognize any non-mOSI protocol sequences.

A direct consequence of implementing only the mOSI defined facilities is that non-mOSI sequences are treated as unrecognized PDUs. The receipt of an unrecognized PDU results in a protocol error and in the release of the association. This does not lessen the utility of the implementation but it does reduce the amount of code needed for error handling.

NOTE—It is always legal for an OSI protocol machine to abort an association.

H.4 Interfaces for mOSI implementations

ISPs currently do not include the selection APIs. However, portability for a mOSI implementation can be significantly enhanced by the use of X/Open's XTI interface. XTI provides a generic interface for connect/disconnect and send/receive. The XTI interface is explicitly for basic communications applications. Appendix H of X/Open's XTI CAE defines the mapping of the generic procedure call interface to OSI upper layer service primitives. The use of XTI provides a common interface for networked applications. It simplifies migration of networked applications from one open networking environment to another.

a mOSI implementation can also use the XTI interface to access transport services. This makes a mOSI implementation easily portabl